

## CAM を用いた分岐予測器の提案 The suggestion of branch predictor based CAM

大下 尊晃<sup>†</sup>      永松 優児<sup>‡</sup>      孟 林<sup>‡</sup>      小柳 滋<sup>†</sup>  
 Takaaki Oshita      Yuji Nagamatsu      Lin Meng      Shigeru Oyanagi

### 1. はじめに

スーパースカラプロセッサの性能向上のためには、命令レベルの並列性を最大限利用するために大きな命令ウィンドウが必要である。しかし、大きな命令ウィンドウが逆に原因となって分岐予測ミスペナルティを増大させてしまうという問題点がある。現在までに分岐予測ミスペナルティを減少させるために様々な分岐予測器が提案されているが、依然として予測ミスが存在している。

予測ミスが発生する原因として、異なる分岐命令が同じ Pattern History Table (PHT) の INDEX に分岐結果を登録しているために破壊的競合が発生しているという点が挙げられる。我々は従来の分岐予測器を分析することで、予測ミスが一部の分岐命令で集中的に発生していること、つまり予測ミスには偏りがあるということを発見した。この特性を利用し、予測ミスに偏りのある分岐命令を例外的に処理することで、分岐予測の精度向上を目指す。

### 2. 予備実験

従来手法の分岐予測器で、予測ミスがどの程度偏っているのかを調査するため予備実験を実施した。本実験では分岐予測器は Gshare 予測器を使用し、実験環境は SimpleScalar を用いて、ベンチマークは SPECint2000 の bzip, gcc, gzip, mcf, parser, twolf, vortex, vpr を 1 億命令 (2000 万命令 × 5) 実行して評価した。

図 1 が 2000 万命令毎に予測ミスの多い上位 16 個の分岐命令についてミス率の合計をまとめたグラフである。この図からいかに特定の分岐命令で予測ミスが起きているか見て取れる。命令によって 90% を超えているものさえある。

### 3. 提案手法

予備実験により、破壊的競合が一部の分岐命令で集中的に発生していることがわかった。

そこで我々は、従来の分岐予測器に CAM (連想メモリ: Content Addressable Memory) を用いた分岐予測器 (CBT: CAM based Table) を付け加えることで、破壊的競合を緩和させる手法を提案する。

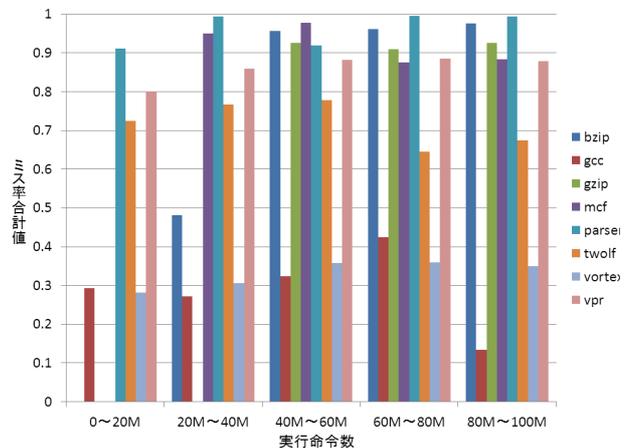


図 1: 各ベンチマークの予測ミスの偏る分岐命令の予測ミス率合計 (上位 16 個まで)

図 2 が、その提案手法を図で表したものである。なお、点線で囲っている部分が従来予測器 (Gshare 予測器) となる。CBT は、分岐ミスが集中的に発生する一部の分岐命令を格納して、それぞれのエンタリは、Addr (アドレス), NT (NotTaken と Taken により増減する 2 bit 飽和型カウンタ), Conf (信頼性評価機構 2bit), FR (Failure Rate) となっている。また信頼性とは、分岐予測に対してその予測がどれくらい信頼できるかという値である。今回の提案手法では、分岐ミスが集中的に発生する一部の分岐命令アドレスを CAM based Table に手動で追加し、分岐結果のローカル履歴を用いて 2 bit カウンタで分岐予測を行う。そして、その予測に対しての信頼性評価機構は MissResettingCounter [1] で信頼性を評価するものとする。

MissResettingCounter とは、最後に予測を失敗してからの予測成功数を数えるカウンタである。予測失敗した時、カウンタ値を 0 にリセットし、再びカウントを始める。カウンタ値が予め設定した閾値より小さければ信頼性が低いと判断する。

本実験では閾値は 3 とする。FR は、CBT の予測ミスが多発したことを検出するものである。ここで、CBT の予測結果が失敗、かつ従来予測器の予測結果が成功するときに FR のインクリメントを行う。閾値になると、CBT の予測ではミスが多発すると判断し、このエンタリを使用しないものとする。本実験の閾値は 255

<sup>†</sup>立命館大学情報理工学部

<sup>‡</sup>立命館大学大学院理工学研究科

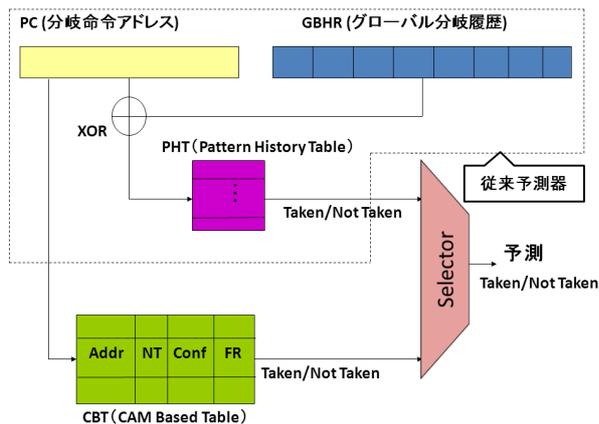


図 2: 提案手法のモジュール図

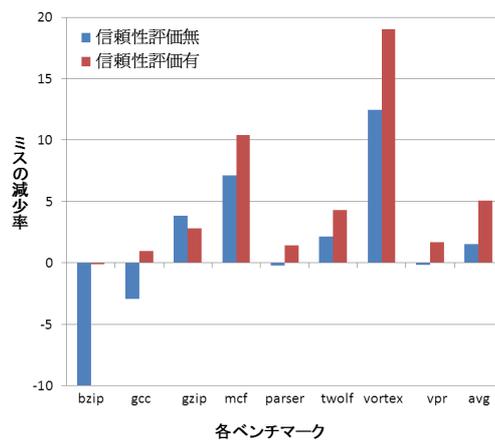


図 3: 1 億命令全体での各ベンチマークのミス減少率

表 1: 従来手法と比較した場合の提案手法のミス増減数 (vortex:2000 万命令 ~ 4000 万命令)

	アドレス	信頼性評価無	信頼性評価有
01	477E28	+54992	0
02	504B50	+8882	0
03	5065B8	+196419	+71
04	47EB48	-1826	-1825
05	4FB990	+2313	-8
06	477C38	-1599	-1596
07	490658	-1191	-1190
08	477BF0	-1183	-1183

とする。

#### 4. 実験結果と考察

ここでは一例として、従来手法と提案手法での予測ミス増減数の比較 (ベンチマーク: vortex の 2000 万命令 ~ 4000 万命令間) を表 1 に、1 億命令全体で各ベンチマーク (bzip, gcc, gzip, mcf, parser, twolf, vortex, vpr) でどの程度の割合でミスが減少したのかを図 3 に記す。

提案手法は信頼性評価を行ったもので行わなかったものの 2 つで評価した。また表 1 の「信頼性評価無」と「信頼性評価有」の列は、従来手法に比べて提案手法のミス数が増加している場合には + を、減少している場合には - をつけている。表 1、図 3 と共に信頼性評価機構を付加したものがミスがより減っている。具体的には、信頼性評価機構有りとした時、vortex で最大で約 19.0% 減り、平均では約 5.0% ミスを減少させることができた。しかし、信頼性評価の有無に関わらず、CAM を使うことで逆に予測ミスが増大してしまっている部分があることが問題である。

#### 5. おわりに

本稿では、破壊的競合を緩和するためにローカル履歴により予測する手法を提案した。今後の課題としては以下のような点が挙げられる。

1. 分岐ミスが集中的に発生する一部の分岐命令アドレスを、手動ではなく動的に CAM based Table に追加できるようにする。
2. CAM based Table の予測器部分を 3bit カウンタでも試してみる。
3. 本実験で使用した信頼性評価機構である MissResettingCounter でも良い実験結果を得ることができたが、性能が良いと言われる他の信頼性評価機構を試してみる予定である。具体的には Updown Counter を使用する。

#### 参考文献

- [1] 西岡 拓生, 孟 林, 小柳 滋, "分岐予測の成否パターンを用いた信頼性評価手法", SACSIS 2010, pp.87-88, May. 2010.
- [2] 仲沢由香里, 山名早人, "予測カウンタの偏向を利用したハイブリッド分岐予測器", 早稲田大学院修士論文, Feb. 2005.
- [3] ニノ宮康之, 阿部公輝, "パーセプトロン分岐予測器を用いた予測信頼性の動的判定に基づく電力削減", SACSIS 2009, pp.327-334, May. 2009.