

マルチレベル分岐予測の検討と評価

3F-6

辻 秀典, 中村 友洋, 吉瀬 謙二, 安島 雄一郎, 田中 英彦
東京大学大学院 工学系研究科

1 はじめに

命令レベル並列性を利用したマイクロプロセッサが一般的となった現在、さらなる性能向上を求めて次世代のプロセッサに関する研究が盛んに行われている。それらは、より一層の命令レベル並列性の抽出、大規模な投機的処理などを目指しているものが多い。本稿では、そのような処理を行うにあたって性能向上が必要不可欠な部分である分岐予測について検討、評価を行う。

2 効率的分岐制御の必要性

現在のマイクロプロセッサは、スーパースカラ構造や VLIW 構造等の命令レベル並列性を利用するものである。このようなプロセッサは、命令実行のスループットを稼ぐことにより高い性能を得ることを目的としている。しかし、スループットの向上とともに相対的に分岐命令の出現頻度が増加し性能向上を妨げる要因となっているため、分岐命令の効率的制御は必要不可欠である。

プログラムコードの 2 割以上を占める分岐命令 (制御転送命令) には、表 1 のようなものが存在する。なお、本稿で記述されるプロセッサ命令は SPARC アーキテクチャのものである。

種類	命令
無条件分岐	CALL
条件分岐	Bicc, FBfcc, CBccc
レジスタ間接分岐 (復帰)	JMPL, RETT
トラップ	Ticc

表 1: 分岐命令の種類

命令	分岐先
CALL	taken only
Bicc, FBfcc, CBccc	taken or not taken
JMPL, RETT	複数
Ticc	複数

表 2: 各分岐命令の分岐先

分岐命令のうち 7~8 割が XXXcc 形式の命令である。そのためこの形式の分岐命令に対しては高い予測性能が要求される。さらに、表 2 から分かるように、JMPL, RETT は複数の分岐先を予測する必要のあるレジスタ間接分岐命令であるため分岐予測が難しい。XXXcc に

比べれば出現頻度は低いものの効率的な分岐制御の必要性が高い。なお、本稿には XXXcc 形式の条件分岐命令に視点を置く。

これまで、分岐命令を効率的に処理するためのさまざまな機構が研究されてきた。おもに分岐履歴情報を用いた手法が提案・実装されているが、いまだに完全な分岐予測機構はなく、現在、静的な分岐予測機構では 65~70% 程度、動的な分岐予測機構でも 80~90% 程度の予測成功率しか得られていない。動的な分岐予測においては分岐履歴情報を BHT (Branch History Table) と呼ばれる表に保存し、これを参照することによって分岐予測を行う。具体的な分岐予測成功率については図 1 を参照。これは BHT を用いた動的な分岐予測を実装し SPEC92 ベンチマークを実行した結果である。履歴ビット数は 2bit とした。

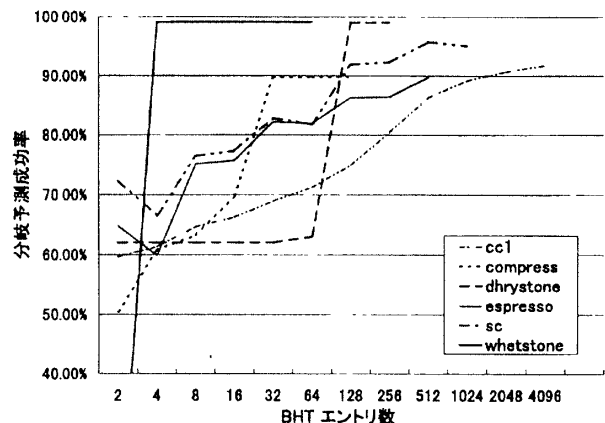


図 1: BHT を用いた動的な分岐予測の予測成功率

プロセッサの性能向上のためには分岐命令を越えた命令の解析が必要である。分岐命令の出現頻度は高く、単体の分岐命令の予測成功率向上だけでは性能向上につながらない。そのために、複数の連続する分岐予測が高い確率で成功する必要がある。従来の分岐命令毎の分岐予測だけでなく、複数の分岐命令に対する分岐予測を統合的に行うマルチレベル分岐予測機構が必要である。本稿では、単体の分岐命令を連続して予測する手法と複数の分岐命令をまとめて処理する手法について検討する。

2.1 マルチレベル分岐予測の検討と評価

ここでは、条件分岐命令 XXXcc (Bicc, FBfcc, CBccc) に限定し、連続する n 段のパスの分岐予測を以下の (a),

(b) に示す方法で行う。条件分岐命令に限定するが、無条件分岐命令である CALL が予測パス上に存在した場合は削除する。

(a) 単体の分岐命令に対する予測と同様に、BHT による分岐予測を個々の分岐命令に対して多段にわたって行う。この方法は図2に示すように、跨ぐ分岐命令に対してのみ分岐予測を適用するため分岐予測段数の増加と計算量の増加は等しい。

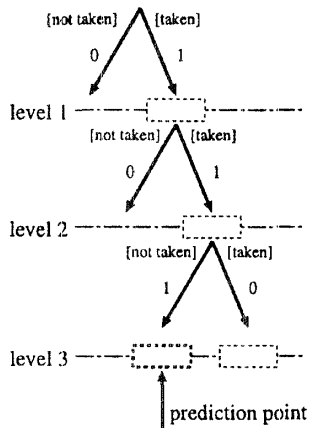


図2: 多段の分岐命令を跨いだ分岐予測 (a)

(b) 通常 BHT の分岐履歴ビットは2ビット以上であるためこれを数値として利用する。具体的には図3に示すように、考えられる全てのパスに対して予測値を求める。この値は通過する分岐命令の BHT の履歴ビット値の積であり、この値の一番大きなパスを予測結果とする。この手法は、跨ぐ分岐命令の数が増加すると指数的に計算量が増加する。

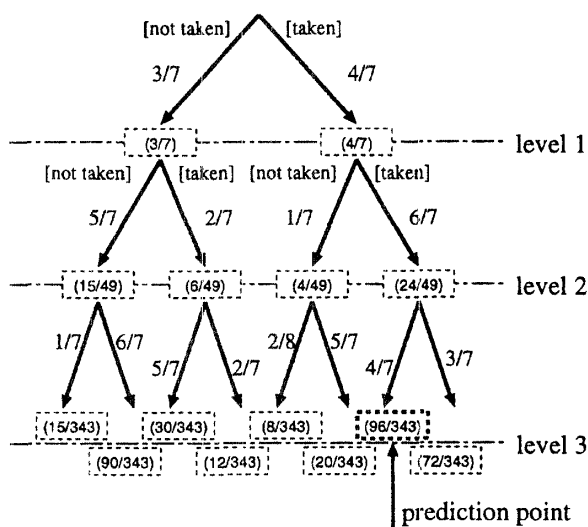


図3: 多段の分岐命令を跨いだ分岐予測 (b)

この2つの手法について SPEC92 を用いたシミュレーションを行った。1000 万命令実行時の結果を図4に示す。履歴ビットは3bit (1~7 を利用)、BHT のエントリ数は制限無しとした。

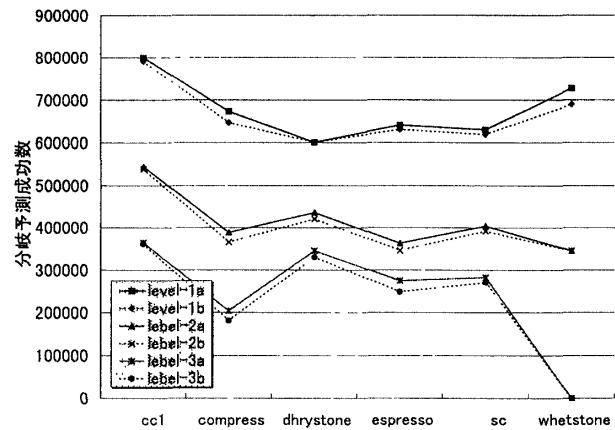


図4: マルチレベル分岐予測予測成功率

条件分岐命令 (XXXcc) に限定してシミュレーションを行ったため、多段の分岐予測を行う際、予測パス上にレジスタ間接分岐命令 (JMPL, RETT) 及びトラップ命令 (Ticc) が存在した場合、予測を行わない。そのため分岐予測成功率については言及しない。

履歴情報の利用方法の違う2者の間の分岐予測成功数はほぼ同じである。(a) と比較して (b) の方が成功数が少なめであり、段数が増えると計算量が指数的に増加することから、この仮定での (b) の手法は適当なものではないと思われる。

3 おわりに

多数の分岐命令を跨いだ分岐予測の成功率を向上させることは、マイクロプロセッサの性能向上のためには必要不可欠である。今回の検討と評価は初期的なものであるが、今後、履歴情報の利用方法の再検討、レジスタ間接命令の分岐先予測も含めた分岐予測の検討と評価、さらに、マルチレベル分岐予測に加えフェッチコストを考慮に入れたフェッチシステムとしての評価を行っていく。

謝辞

本研究の遂行にあたり、文部省科学研究費（一般研究 (B) 課題番号 07458052 「大規模データバスプロセッサの研究」）のご支援を頂きました。ここに感謝の意を表します。

参考文献

- [1] 田中英彦, “ここいらで、計算機アーキテクチャを再考しよう”, 情処研究会 ARCH 108-6, vol.94, no.91, pp.30-40, Oct 1994.
- [2] 中村友洋, 吉瀬謙二, 辻秀典, 安島雄一郎, 田中英彦, “大規模データバスプロセッサの構想”, 情処研究会 ARCH 124-3, vol. 97, no. 61, pp.13-18, Jun 1997.