

分岐命令に着目した投機的実行支援情報収集機構の設計とFPGAへの実装

蛭田智則† 小池汎平‡§ 佐谷野健二‡ 山名早人†§

† 早稲田大学理工学部情報学科 ‡ 産業技術総合研究所 § アドバンスト並列化コンパイラ研究体

1 はじめに

投機的実行を支援するための情報として分岐命令に着目し、トレース情報を収集するホットトレースディテクタの設計を行った。また、FPGA上で動作するMIPS互換プロセッサコアにホットトレースディテクタを実装し、SPEC95ベンチマークの129.compressを用いて実験を行った。

2 投機実行のための情報収集

投機的実行はプログラムの統計的性質を積極的に利用してプログラム実行の高速化を図るテクニックである。実行の高速化に利用できるプログラムの統計的性質として最も基本となるのは、連続した複数の分岐命令の振る舞いを把握し、実行頻度の高いプログラム中の一連のコントロールフロー、すなわちトレース [1] を見極めることである。この情報はVLIWプロセッサの投機的命令スケジューリングをはじめとした様々な投機的最適化 [2] に利用することができる。

3 ハードウェアによる情報収集

一般に、プログラムの統計的性質を知るためには、対象となるプログラムを実際に実行して振る舞いを観測する必要がある。このためにプログラム中に統計測定用の命令列を挿入して試験実行させ、ソフトウェア的に測定することが一般的であるが、この方法は、オーバーヘッドを伴うこと、キャッシュなどの動作に干渉を及ぼす恐れがあること、プログラムの最適化と実行を同時に行う実行時最適化に向かないこと、などの問題点がある。最適化に有用な情報を抽出する支援ハードウェアを効果的に付加することによりこれらの問題点は解決される。

このような試みとしてイリノイ大のホットスポットディテクタがある [3]。これは分岐命令の実行頻度のヒストグラムを求めることにより、プログラム中の実行頻度の高い部分 (ホットスポット) を特定するものである。しかし、ホットスポット

ディテクタでは単一の分岐命令の振る舞いのみに注目しているため、トレースを正しく抽出することはできない。そこで今回、複数の分岐命令に渡った実行フローの頻度を測定することのできるホットトレースディテクタを設計し試作を行った。

4 ホットトレースディテクタの構成

図1にホットトレースディテクタの構成を示す。ホットトレースディテクタはトレース抽出部とヒストグラム作成部からなる。

トレース抽出部はCPUのパイプラインより分岐命令実行のたびに分岐IDを受け取り、トレースIDを作成してヒストグラム作成部に渡す。分岐IDはPC値と分岐信号からなり、トレース抽出部では連続した3つの分岐IDをトレースID列に変換する。

ヒストグラム作成部は、トレース抽出部より受け取ったトレースIDのヒストグラムを作成する。ヒストグラムテーブルが溢れるとCPUに割り込みをかけ、中間結果の情報を読み出させる。読み出された中間結果は主記憶に保存されプログラム終了時にまとめられて最終的なヒストグラムが作成される。

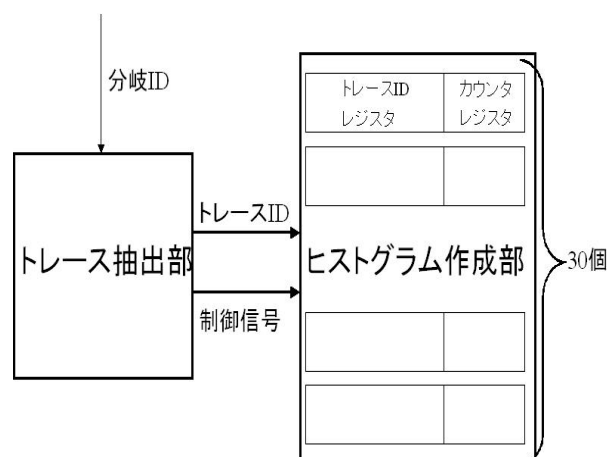


図1: ホットトレースディテクタの構成図

Design of a Mechanism Gathering Speculative Execution Support Information and Implementation to FPGA

Tomonori Hiruta†, Hanpei Koike‡§, Kenji Sayano‡, Hayato Yamana†§

†Department of Information and Computer Science, School of Science and Engineering, Waseda University

‡National Institute of Advanced Industrial Science and Technology

§Advanced Parallelizing Compiler Project

5 ホットトレースディテクタの試作

ホットトレースディテクタの動作を確認するために、産業技術総合研究所で開発された FPGA を利用したリコンフィギュラブル実験装置 REX[4] を用いてハードウェアの試作を行った。REX 上で動作する MIPS 互換プロセッサのパイプラインから必要な信号線を取り出し、新たに設計したホットトレースディテクタと接続した。

ヒストグラムテーブルの大きさ等の仕様を表 1 にまとめる。ホットトレースディテクタ部は Verilog HDL で記述されており 1247 行である。

6 結果

ホットトレースディテクタの付加された CPU で、SPEC95 ベンチマークの 129.compress(入力データ test) を実行させ、実行統計情報を取得させる実験を行った。結果を表 2 に示す。CPU の動作周波数は 25MHz である。この情報から、compress 本体における最も実行頻度の高いトレースは図 2 の 1 に示すものであることがわかった。また、表 2 から図 2 における while ループのトレースは 2 パターンのみであることがわかった。この実験においてヒストグラムテーブル溢れによる割り込みは 380 回発生した。なお、REX 上での compress の動作速度は SimpleScalar 上での動作速度の 8 倍である

7 おわりに

今回行った実験では、ホットトレースディテクタ内部のヒストグラムテーブル溢れによる CPU への割り込みにより若干の実行速度の低下が見られた。今後はホットトレースディテクタに CPU への割り込みを減らすための改良を施す予定である。

参考文献

- [1] J.A.Fisher, "Trace Scheduling: A Technique for Global Microcode Compaction", IEEE Trans, Vol.30, pp.478-490, 1981
- [2] 石川隼輔, 山名早人, "ループへの効果的な投機的実行適用手法", 情報処理学会研究報告, 2002-HPC-89, pp.121-126, 2002
- [3] Thomas Ball, Jamaes R. Laurs, "Optimally Profiling and Tracing Programs", Computer Sciences Department University of Wisconsin-Madison Technical Report #1031, 1991
- [4] 佐谷野健二, 片下敏宏, 小池汎平, 児玉祐悦, 坂根広史, 甲村康人, "大容量 FPGA の応用によるマルチプロセッサエミュレーションシステムの評価", 情報処理学会研究報告, 2001-ARC-144, pp.25-30, 2001

表 1: ヒストグラムテーブルの仕様

	サイズ	個数
トレース記憶レジスタ	48bit	30
カウンタレジスタ	32bit	30

表 2: トレースと実行回数

トレース番号	トレース			実行回数
	10ac T	10ac T	10ac T	
	10ac T	10ac T	10ac T	107725
	1458 T	1458 T	1458 T	16128
	14ec T	14ec T	14ec T	16128
	15f8 T	15f8 T	15f8 T	16000
	a04 T	a04 T	a04 T	6350
1	1110 N	428 T	47c N	2912
	428 T	47c N	1110 N	2900
	47c N	1110 N	428 T	2900
	428 T	47c T	4c8 N	2813
2-2	47c T	4c8 N	6f8 N	2813
2-1	1110 N	428 N	47c T	2813
2-3	848 N	858 N	5a0 N	2813
	c00 N	c30 N	cc4 N	2813
	5a0 N	1110 N	428 T	2800
	858 N	5a0 N	1110 N	2800

T ... taken N ... not taken

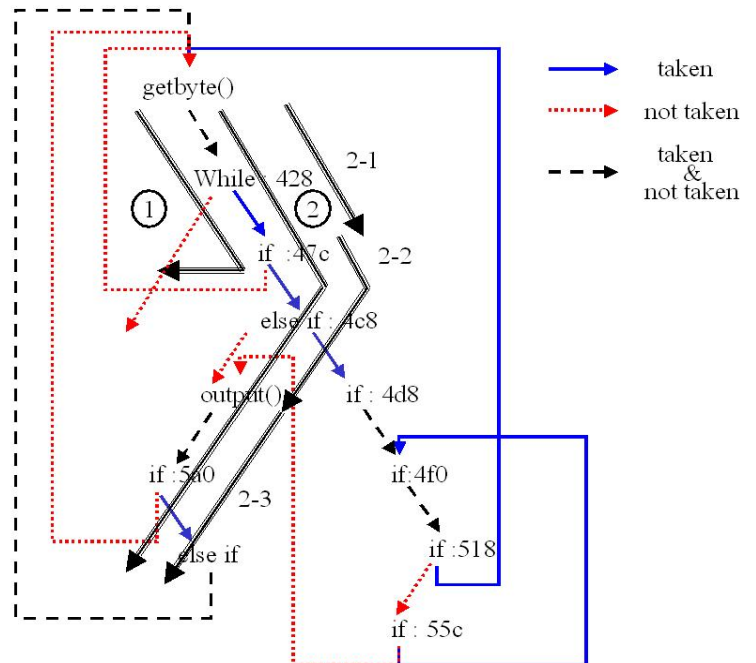


図 2: トレース図