

らの情報を用いて、データが消滅する際に、そのデータがショート・リブド・データであるかを検出できる。

4 ショート・リブド・データの動的な予測

ある命令の生成したデータがショート・リブド・データだった場合に、その命令の次回の実行結果もショート・リブド・データである確率が高いといった偏りがあれば、ショート・リブド・データの動的な予測の可能性が生まれる。

命令アドレス (プログラム・カウンタ) をインデックスとして前回の実行の結果がショート・リブド・データであったかを保存するテーブルを用意し、前回のデータがショート・リブド・データだった場合に、今回の結果もショート・リブド・データであると予測した場合の予測成功率を測定する。ここで、ショート・リブド・データの検出には3章で定義した機構を用いる。

5 評価環境

トレース・ドリブンのシミュレータを用いて、ショート・リブド・データの割合と予測成功率を評価する。評価に用いるベンチマークを表1にまとめる。プログラムの入力セットは実行命令数が1億程度に収まるように調整し、コンパイルには Gnu C compiler (最適化オプション-O、ターゲット SPARC version 7) を用いた。

Program	Description	Input Set
cc1	From GCC 2.7.1	genrecog.i
compress	File compression	small.in
go	Plays the game of Go	9 9
m88ksim	Moto88K simulator	ctl.raw
perl	Manipulates strings	"admits" in 1/8 input
xlisp	LISP interpreter	6 queens

表 1: ベンチマークプログラム

評価には図1に示す標準的なスーパースカラのモデルを用いる。プロセッサの主なパラメータを表2にまとめる。

Parameter	Value
Bandwidth of decode, dispatch, commit, retire	8 instr. per cycle
Reorder buffer	32 entries
Instruction Window	8 entries
Memory latency (no cache miss)	2 cycle
The number of Functional Unit	No limitation

表 2: 評価に用いたプロセッサ・モデルのパラメータ

6 評価結果

ショート・リブド・データの検出結果を表3に示す。それぞれのプログラムについて、表3の左の列から実行命令数、生成されたデータの量、ショート・リブド・データの量 (生成されたデータに対するショート・リブド・データの割合) を示した。表中の M は million を意味している。表3の結果より、生成されたデータの72%から82%がショート・リブド・データであることがわかる。もし、これらのデータをリネーム・レジスタやレジ

スタ・ファイルに書き戻す必要がないとすると、データ転送量の7割以上を削減できることになる。

Program	Executed Inst.	Created Data	Short-lived Data (ratio)
cc1	120 M	60 M	49 M (82.1%)
compress	54 M	27 M	22 M (81.3%)
go	136 M	78 M	62 M (79.2%)
m88ksim	124 M	66 M	54 M (82.8%)
perl	101 M	52 M	40 M (78.3%)
xlisp	42 M	19 M	13 M (72.4%)

表 3: ショート・リブド・データの割合

ショート・リブド・データの予測について、予測成功の回数 (Hit)、失敗の回数 (Miss)、予測成功率 (Hit / (Hit + Miss)) を表4にまとめる。ここでいう予測とは、4章で説明した、前回の実行により生成されたデータがショート・リブド・データだったときに今回の実行結果もショート・リブド・データであると予測する単純なものである。

Program	Hit	Miss	Hit Ratio
cc1	46,669,409	2,243,563	95.4%
compress	21,432,337	844,565	96.2%
go	58,699,172	3,052,470	95.1%
m88ksi	53,769,345	667,142	98.8%
perl	38,778,014	1,567,074	96.1%
xlisp	12,690,702	719,340	94.6%

表 4: ショート・リブド・データの予測成功率

評価結果 (表4) よりショート・リブド・データの予測成功率は94%以上に達することがわかった。ショート・リブド・データである確率は82%以下だが、前回の履歴を用いることで94%以上の確率で予測できるという結果から、前回のデータがショート・リブド・データだった場合には今回もショート・リブド・データであるという偏りがあることがわかる。

7 まとめと今後の課題

フォワードイング・パスのみにより供給されるデータをショート・リブド・データと定義し、その検出機構と動的な予測の可能性を検討した。標準的なスーパースカラ・プロセッサを想定して評価した結果、生成されるデータの7割以上のデータがショート・リブド・データであることを確認した。また、前回の履歴を用いることにより、94%以上という高い確率でショート・リブド・データを動的に予測できることを示した。ただし、プロセッサ内でショート・リブド・データを予測する場合には、予測に失敗した場合の回復機構が必要となる。回復機構を含む予測機構の検討は今後の課題である。

参考文献

- [1] Luis A. Lozano C. and Guang R. Gao. Exploiting Short-Lived Variables in Superscalar Processors. In Proc. of MICRO-28, pages 292-302, 1995.