

Tri-Mode 分岐予測器の提案

齋藤 史子[†] 山名 早人^{††}

命令間の制御依存によってパイプライン処理を滞らせないために、近年のプロセッサでは分岐予測が採用されている。分岐予測は、未解決の分岐命令を超えた実行（投機的実行）を可能とする。一方、近年の命令パイプライン長の深化により、分岐ミスペナルティが増大している。そのため、分岐予測ミス率の低減はプロセッサの性能向上において不可避の課題となっている。

現在までさまざまな分岐予測器が提案されてきた。なかでも、複数の予測表で構成されたハイブリッド分岐予測器は高精度な予測器として知られている。代表的なハイブリッド予測器には、分岐命令の偏向に応じて予測表を使い分ける Bi-Mode 分岐予測器がある。Bi-Mode 分岐予測器では、Choice PHT が分岐命令の偏向を判断し、偏向に応じて採用する Direction PHT(Taken/NotTaken PHT) を決定する。本稿では、Choice PHT が Weakly 分岐 (Weakly Taken/NotTaken) と判断した場合に、分岐命令に偏向がないことに着目した。そこで、Weakly 分岐を予測するための予測表 (Weakly PHT) を追加し、Direction PHT では Choice PHT が Strongly 分岐 (Strongly Taken/NotTaken) と判断する分岐のみを扱うことを提案する。本稿では、Weakly PHT を追加した Bi-Mode 予測器を Tri-Mode 分岐予測器と呼ぶ。SPECint95(ref) ベンチマークの実験において、12KB 容量の分岐予測器では、BTB hit/miss 参照手法を適用した Tri-Mode 予測器が、従来の Bi-Mode 予測器と比較して平均 2.78% の予測ミス削減率を示した。

The Proposal of Tri-Mode Branch Predictor

FUMIKO SAITO[†] and HAYATO YAMANA^{††}

The branch prediction is installed on the recent processor to avoid stalling pipeline. Branch prediction is a kind of speculative execution for control dependence. In the recent year, the deeper pipeline gets, the higher branch miss prediction penalty reaches. Thus, branch miss prediction rate must lower to rise processor performance.

Recently, many various branch predictors have been proposed. Hybrid branch predictors composed of multiple pattern history tables (PHT) show the highest accuracy among them. The Bi-Mode branch predictor is the most famous of the hybrid branch predictors. On the Bi-Mode predictor, the Choice PHT judges the branch bias and selects the Direction PHT(Taken or NotTaken PHT). This paper focuses on the Weakly Branches which the Choice PHT judges Weakly Taken or NotTaken don't have the branch bias. In order to avoid the Weakly branch influence on the Direction PHTs, we propose "the Tri-Mode brach predictor" added the Weakly PHT predicting the Weakly branches. On the 12KB Tri-Mode predictor, the branch miss reduction rate from the Bi-Mode predictor shows average 2.78% in the SPECint95(ref inputs) benchmark simulation.

1. はじめに

命令間の制御依存によってパイプライン処理を滞らせないために、近年のプロセッサでは分岐予測が採用されている。分岐予測は、未解決の分岐命令を超えた実行（投機的実行）を可能とする。しかし、分岐予測ミス率が等しくても、命令パイプライン長の深化によ

り分岐ミスペナルティが増大する。すなわち、近年の 30～40 段、さらには、今後の命令パイプライン長の深化したプロセッサを性能向上させるためには、分岐予測ミス率の低減が必要不可欠となっている。

現在までさまざまな分岐予測器が提案されてきた。なかでも、複数の予測表で構成されたハイブリッド分岐予測器は高精度な予測器として知られている。代表的なハイブリッド予測器には、分岐命令の偏向に応じて予測表を使い分ける Bi-Mode 分岐予測器がある。Bi-Mode 予測器は、分岐偏向ごとの予測表 (Taken PHT, NotTaken PHT) とどちらの予測表を採用するかを決定する Choice PHT から構成される。Choice

[†] 早稲田大学大学院理工学研究科
Graduate School of Science and Engineering,
Waseda University
^{††} 早稲田大学理工学術院
Science and Engineering, Waseda University

PHTは、分岐命令ごとの成立・不成立に応じてカウンタ状態を遷移させる。そして、分岐命令の分岐傾向(Taken/NotTaken)に応じて予測表を選択する。Bi-Mode 予測器では、分岐傾向ごとに予測表を分けることによって、予測表競合を減少させ、予測精度を向上させた。予測表競合とは、異なる分岐が同一の予測表エントリに割振られる現象である。

Bi-Mode 予測器では、Choice PHT が分岐命令の偏向を判断し、偏向に応じて採用する Direction PHT(Taken/NotTaken PHT) を決定する。本稿では、Choice PHT が Weakly 分岐 (Weakly Taken/NotTaken) と判断した場合に、分岐命令に偏向がないことに着目した。そこで、Weakly 分岐を予測するための予測表 (Weakly PHT) を追加し、Direction PHT では Choice PHT が Strongly 分岐 (Strongly Taken/NotTaken) と判断する分岐のみを扱うことを提案する。これによって、Direction PHT における偏向のある Strongly 分岐と偏向のない Weakly 分岐との競合を回避し、予測精度向上を図る。本稿では、Weakly PHT を追加した Bi-Mode 予測器を Tri-Mode 分岐予測器と呼ぶ。

本稿の構成を次に示す。2 節は、基本的な分岐方向予測器の構成と関連研究、3 節は、Choice PHT の分岐偏向判断の信頼度、4 節は、Tri-Mode 分岐予測器の提案、5 節は、実験結果について述べる。6 節はまとめである。

2. 分岐方向予測器

分岐方向予測器は、単一の予測表で構成される単体予測器と複数の予測表で構成されるハイブリッド予測器に分類できる。予測表は、分岐命令アドレスや分岐履歴に関連付けられた飽和カウンタの表である。

以降、代表的な分岐方向予測器の構成について説明する。

2.1 単体予測器

単体予測器は、ハイブリッド予測器の構成要素になる。ここでは、基本的な予測器である Bimodal 予測器¹⁾ と Gshare 予測器²⁾ について説明する。Bimodal 予測器は、分岐命令アドレスごとに飽和カウンタを用意する。Gshare 予測器は、(Global 分岐履歴 xor 分岐命令アドレス) ごとに飽和カウンタを用意する。

2.2 ハイブリッド予測器

ハイブリッド予測器は、複数の予測表から予測を採用するひとつの予測表を選択する。これによって、異なる分岐命令同士に予測表のエントリが共有される競

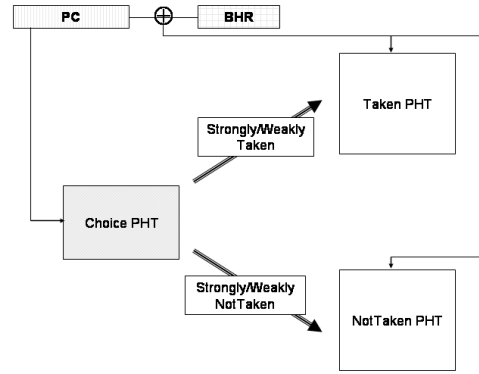


図 1 Bi-Mode 分岐予測器³⁾

合という現象を緩和できる。ここでは、分岐偏向に着目したハイブリッド予測器である Bi-Mode 予測器³⁾ を紹介する。

Bi-Mode 予測器

Bi-Mode 予測器は、3 つの予測表 (PHT: Pattern History Table) で構成される (図 1)。1 つめの予測表は分岐偏向を判断し、Choice PHT と呼ばれる。残る 2 つの予測表は、Direction PHT と呼ばれ、各々、異なる偏向の分岐を対象に予測する。Taken 偏向分岐を対象とする予測表は、Taken PHT、Not Taken 偏向分岐を対象とする予測表は、NotTaken PHT と呼ばれる。Bi-Mode 予測器では、Choice PHT の予測結果に基づいて、Taken/NotTaken PHT のどちらを採用するか決定する。基本的な Bi-Mode 予測器は、Choice PHT に分岐命令ごとに過去数回の分岐傾向が判る Bimodal 予測器、Direction PHT に履歴を利用した高精度な Gshare 予測器で構成される。

吉瀬らは、Bi-Mode 予測器にさらに改良を加えた⁴⁾。分岐命令ごとに NotTaken 強偏向を判断するカウンタを用意し、NotTaken 偏向分岐を予測器の予測対象外とした。さらに、Choice PHT で弱偏向 (Weakly 分岐) と判断される分岐に対して、Choice PHT を含む全 PHT の予測の多数決に基づいて予測する。これによって、8KB 容量において、従来の Bi-Mode 予測器と比較して予測ミス率は平均 0.63% 低減した。これは、Bi-Mode 予測器の予測ミス率のうち 13.7% を削減したことになる。

3. Choice PHT の偏向判断信頼度

Bi-Mode 分岐予測器では、Taken 偏向分岐、Not-Taken 偏向分岐ごとに予測表を用意し、すべての条件分岐を Taken 偏向、もしくは、NotTaken 偏向に振り

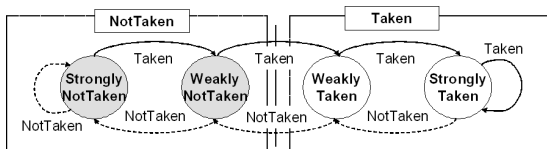


図 2 2bit 飽和カウンタの状態遷移図

分ける．しかし，すべての条件分岐に偏向があるわけではない⁵⁾．そのため，偏向のない分岐が偏向のある分岐と同一の予測表に振り分けられ，競争を起こすことによって悪影響を与える可能性がある．

以降，Choice PHT における分岐偏向の判断が正確であるかを確認する．

3.1 予測カウンタ状態の説明

予測表は，複数の飽和カウンタ（予測カウンタ）の集まりであり，予測カウンタの状態に応じて予測する．図 2 に 2bit 飽和カウンタの状態遷移を示す．2bit 飽和カウンタは，Taken と予測される 2 状態 (Strongly/Weakly Taken) と NotTaken と予測される 2 状態 (Strongly/Weakly NotTaken) からなる．

また，予測表エントリに割振られる分岐パターンに対して，Strongly 状態 (Strongly Taken/NotTaken) の場合偏向のあるパターン，Weakly 状態 (Weakly Taken/NotTaken) の場合偏向のないパターンとも判断できる．さらに，分岐ミスすると予測カウンタ状態は，Weakly 状態に遷移することから，Weakly 状態の分岐は前回の予測が失敗している．本稿では，予測カウンタが Strongly 状態である動的分岐を Strongly 分岐，予測カウンタが Weakly 状態である動的分岐を Weakly 分岐と呼ぶことにする．

次節で，各予測 (Taken/NotTaken) 状態の Strongly/Weakly 分岐の傾向を調べる．

3.2 予測カウンタ状態ごとの傾向

Bi-Mode 予測器は，各偏向 (Taken/NotTaken) に応じた予測表を採用する．Choice PHT の予測カウンタ状態が Strongly/Weakly Taken の場合は Taken 偏向，Strongly/Weakly NotTaken の場合は NotTaken 偏向と判断される．ここでは，各々の状態に実際に偏向があるのかを確かめる．以降，BTB hit/miss 参照手法⁶⁾を適用した 4KB Choice PHT (Bimodal 予測器) の結果を示すが，0.5KB, 8KB でも同様の傾向を示した．

本稿では，偏向のあるパターンが割振られていると判断できる Strongly 分岐と偏向のないパターンが割振られていると判断できる Weakly 分岐に着目した．図 3 に Choice PHT における予測カウンタ状態の割合と偏向判断一致率を示す．図 3 に示される数字は，偏向判

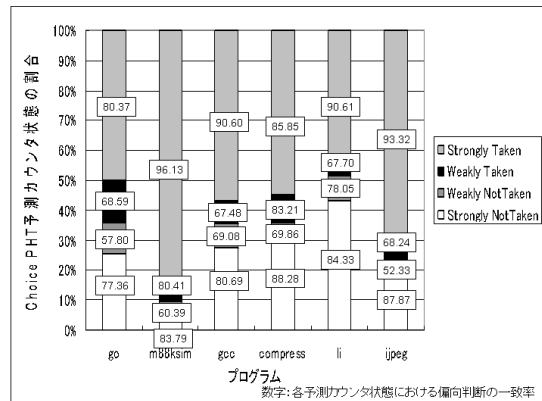


図 3 Choice PHT における予測カウンタ状態の割合と偏向判断一致率 (%)

断一致率である．偏向判断一致率は，Choice PHT の偏向判断と実際の分岐結果が一致した割合 (/ 各状態の分岐数) を表す．つまり，Strongly/Weakly Taken の場合は分岐が Taken である割合，Strongly/Weakly NotTaken の場合は分岐が NotTaken である割合を示している．

Strongly 分岐

Strongly 分岐の占める割合は 75.59 ~ 93.80% であり (表 1)，大多数の分岐は Strongly 分岐である．また，偏向判断一致率は，Strongly Taken 状態で 80.37 ~ 96.13%，Strongly NotTaken 状態で 77.36 ~ 88.28% であり (図 3)，Choice PHT の偏向判断は正しいと言える．Strongly NotTaken 状態の偏向判断一致率が Strongly Taken 状態よりも小さいのは，BTB hit/miss 参照手法⁶⁾によって，常に NotTaken である分岐が除外されているためである．

Weakly 分岐

Weakly 分岐の占める割合は 6.20 ~ 24.41% であり (表 1)，Weakly 分岐は比較的少ない．また，偏向判断一致率は，Weakly Taken 状態で 67.48 ~ 83.21%，Weakly NotTaken 状態で 52.33 ~ 78.05% である (図 3)．プログラムによっては Taken に分岐する場合と NotTaken に分岐する場合が半々であるため，片方向に偏向していると判断しがたい状況になっている．

以上から，Strongly 分岐は強偏向分岐，Weakly 分岐は弱偏向分岐であることが判った．

4. Tri-Mode 分岐予測器の提案

前節の各予測カウンタ状態の分岐傾向の解析から，Choice PHT の偏向判定のうち Weakly 分岐には偏向

表 1 Strongly/Weakly 分岐の占める割合 (%)

分岐の種類	099.go	124.m88ksim	126.gcc	129.compress	130.li	132.jpeg
4KB Choice PHT (Bimodal 予測器)						
Strongly 分岐	75.59	93.80	84.28	85.53	85.47	88.71
Weakly 分岐	24.41	6.20	15.72	14.47	14.53	11.29

BTB hit/miss 参照手法⁶⁾ を適用 .

があると言えないことが判った . そこで , Direction PHT において , 偏向のある Strongly 分岐と偏向のない Weakly 分岐の競合を回避した Tri-Mode 予測器を提案する .

Tri-Mode 分岐予測器

偏向のない Weakly 分岐が Direction PHT に登録されることで , Strongly 分岐の予測に悪影響を与える可能性がある . そこで , Bi-Mode 分岐予測器に Weakly 分岐を対象とする予測表 (Weakly PHT) を新たに追加する (図 4) . また , Weakly PHT で扱う分岐は弱偏向のため , 従来の 2bit 飽和カウンタよりも Local な分岐パターンを細かく解析する 3bit 飽和カウンタで予測できる可能性がある .

次に , Tri-Mode 予測器の予測方式について説明する . Tri-Mode 予測器では , Bi-Mode 予測器と同様に , Choice PHT の予測カウンタ状態に応じて採用する予測表を決定する . 以下に , Choice PHT の予測カウンタ状態と採用する予測表を示す .

Choice PHT の予測カウンタが Strongly Taken 状態を示す場合

Taken PHT を採用する .

Choice PHT の予測カウンタが Weakly Taken/NotTaken 状態を示す場合

Weakly PHT を採用する .

Choice PHT の予測カウンタが Strongly Not-Taken 状態を示す場合

NotTaken PHT を採用する .

予測器の更新についても , Bi-Mode 予測器と同様に , 分岐履歴は常時更新し , PHT は予測を採用した場合のみ更新する .

5. 分岐予測器の性能比較

本節では , シミュレーション環境と提案した分岐予測器の分岐予測精度を示す .

5.1 実験環境

シミュレータは , SimpleScalar 3.0d/PISA simbpred⁷⁾ , ベンチマークは , SPECint95 (ref 入力) (表 2) から分岐予測ミス率の高いプログラムを採用した . コンパイラは , PISA 用の gcc 2.7.2.3 を利用

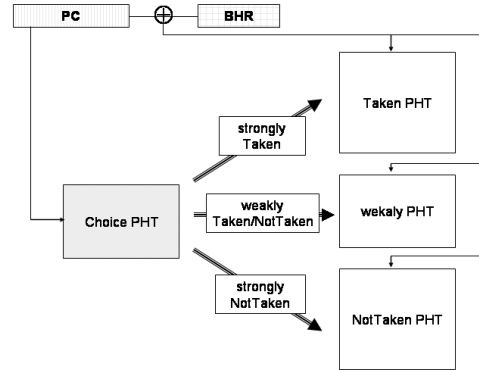


図 4 Tri-Mode 分岐予測器

表 2 ベンチマーク [100 万命令]

プログラム	入力	全命令	条件分岐
099.go	50 21 null.in	39,414	41,97
124.m88ksim	ctl.in	79,706	12,385
126.gcc	varasm.i	260	39
129.compress	bigtest.in	45,636	5,972
130.li	*.lsp	56,381	8,574
132.jpeg	vigo.ppm	30,708	1,840

した .

シミュレーション対象とした予測器の構成は , BTB は , 4 way set-associative 512 エントリ , 分岐方向予測器については , 表 3 に示す . 表 3 中の BTB hit/miss 参照手法⁶⁾ とは , BTB hit の場合のみ分岐方向予測器を利用し , BTB miss の場合は NotTaken と予測する方法である . 分岐方向予測器のハードウェア容量は , 1.5KB, 12KB, 24KB を実験対象にした .

5.2 実験結果

提案手法適用が予測精度に与える効果を示す .

5.2.1 予測ミス率

BTB hit/miss 参照手法⁶⁾ の適用によって , 従来の Bi-Mode 予測器に対する平均予測ミス削減率は , 1.5KB 容量で 3.94% , 12KB 容量で 1.22% , 24KB 容量で 0.86% を示す (表 4) . Bi-Mode 予測器に対する予測ミス削減率とは , Bi-Mode 予測器の予測ミス率のうち改善できた割合 (低下予測ミス率 / Bi-Mode 予

表 3 分岐方向予測器の構成

予測器の種類	構成
Bi-Mode	Choice PHT:Taken PHT: Not-Taken PHT=1:1:1 の容量比 .
Tri-Mode2	Choice PHT:Weakly PHT:Taken PHT:NotTaken PHT=1:1:2:2 の容量比 .
Tri-Mode3	Tri-Mode2 の Weakly PHT を 3bit 飽和カウンタに変更 .
+BTB	BTB hit/miss 参照手法 ⁹⁾ を適用 .

測器の予測ミス率)である .

さらに, Weakly PHT に 3bit 飽和カウンタを採用した Tri-Mode 予測器は, BTB hit/miss 参照手法⁶⁾ と比較して, 12KB 容量では, gcc で 0.12% 予測ミス率が増加してしまっただが, gcc 以外のプログラムでは, 0.03 ~ 0.55% 予測ミス率が低下した . 従来の Bi-Mode 予測器に対する平均予測ミス率削減率は, 1.5KB 容量で 3.83%, 12KB 容量で 2.78%, 24KB 容量で 2.32% を示した .

以降, 各予測器に対して, 提案手法適用による予測精度変動の要因を解析する .

5.2.2 予測精度に与える影響の解析

本手法を適用した場合の予測精度向上 / 低下の要因を適用した予測器ごとに調べる . 以降, すべてのデータは, BTB hit/miss 参照手法⁶⁾ を適用している . また, 多くの文献で実験されている 12KB 容量のハイブリッド予測器のデータを示す .

Tri-Mode 予測器では, 偏向のある Strongly 分岐が偏向のない Weakly 分岐の影響を受けないように, 予測表を分割した . そこで, Strongly 分岐と Weakly 分岐の予測精度から予測精度向上 / 低下の要因を考察する . また, Weakly PHT に 3bit 飽和カウンタを採用した場合の効果についても解析する .

Strongly 分岐の予測精度

提案手法適用の効果を調べるために, Bi-Mode と Tri-Mode の Strongly 分岐の予測ミス率 (/ Strongly 分岐数) を示す (図 5) . Tri-Mode2, Tri-Mode3 では Direction PHT の構成が共通のため, Tri-Mode2, Tri-Mode3 のデータは Tri-Mode としてまとめて表す .

比較の結果, m88ksim では 0.04% 予測ミス率が増加したが, go, gcc, jpeg では予測ミス率が 0.07 ~ 1.26% 低下した . compress, li の予測ミス率は殆ど変動しなかった . このことから, Weakly 分岐が Strongly 分岐と同じ予測表を利用することによって, Strongly 分岐の予測精度を低下させていたことが判る .

Weakly 分岐の予測精度

提案手法の適用が Weakly 分岐に及ぼした効果を調

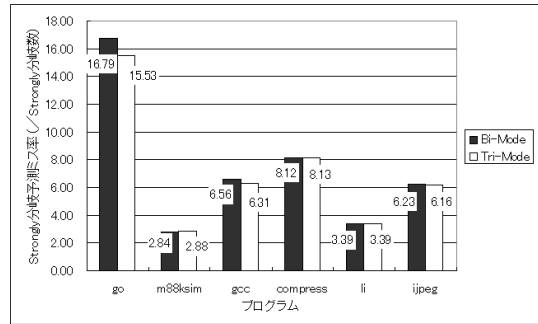


図 5 Strongly 分岐の予測ミス率 (%)

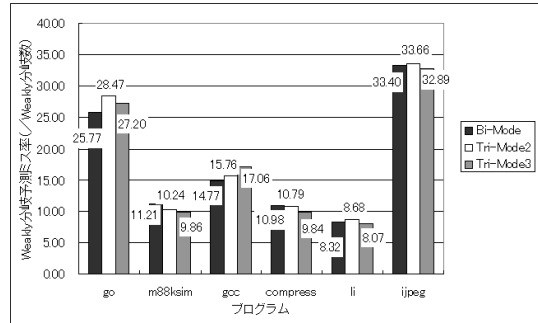


図 6 Weakly 分岐の予測ミス率 (%)

べるために, Bi-Mode と Tri-Mode2 の Weakly 分岐予測ミス率 (/ Weakly 分岐数) を比較する (図 6) . 比較の結果, m88ksim, compress では予測ミス率が 0.19 ~ 0.97% 低下し, それら以外のプログラムでは予測ミス率が 0.25 ~ 2.70% 増加した . m88ksim, compress の予測ミス率低下は, 強偏向の分岐の影響を受けなくなったためと考えられる . それら以外のプログラムの予測ミス率増加は, 弱偏向分岐は予測カウンタの Weakly 状態を頻繁に遷移するため, 2 つの Weakly 状態 (Weakly Taken/NotTaken) しかもたない 2bit 飽和カウンタでは十分な学習ができなかったためと考えられる .

そこで, Weakly PHT に 3bit 飽和カウンタを採用した効果を調べるために, Tri-Mode2 と Tri-Mode3 の Weakly 分岐予測ミス率を比較する (図 6) . 比較の結果, gcc 以外のプログラムで予測ミス率が 0.38 ~ 1.27% 低下し, gcc では予測ミス率が 1.30% 増加した . さらに, Bi-Mode と Tri-Mode3 の比較の結果, Tri-Mode2 では予測ミス率が増加していた li, jpeg で 0.25 ~ 0.52% 予測ミス率が低下した . このことから, Weakly PHT に 3bit 飽和カウンタを採用することによって, Weakly 状態を頻繁に遷移する弱偏向分岐が予測可能になったことが判る .

表 4 分岐方向予測ミス率 (%)

予測器 (表 3 参照)	099.go	124.m88ksim	126.gcc	129.compress	130.li	132.jpeg	平均	削減率 [平均]
1.5KB 容量								
Bi-Mode	23.43	3.38	10.49	9.11	4.56	9.50	10.08	—
+BTB								
Bi-Mode	<u>22.70</u>	3.28	9.04	9.11	4.51	9.45	<u>9.68</u>	<u>3.94</u>
Tri-Mode2	22.98	3.26	<u>9.03</u>	9.07	4.50	9.44	9.71	3.64
Tri-Mode3	23.06	<u>3.24</u>	9.15	<u>8.92</u>	<u>4.43</u>	<u>9.35</u>	9.69	3.83
12KB 容量								
Bi-Mode	18.50	3.00	6.84	8.46	3.51	8.94	8.21	—
+BTB								
Bi-Mode	18.06	2.96	6.77	8.46	3.47	8.93	8.11	1.22
Tri-Mode2	17.79	2.94	<u>6.74</u>	8.44	3.52	8.90	8.05	1.88
Tri-Mode3	<u>17.51</u>	<u>2.92</u>	6.89	<u>8.30</u>	<u>3.44</u>	<u>8.82</u>	<u>7.98</u>	<u>2.78</u>
24KB 容量								
Bi-Mode	15.80	2.96	6.15	8.06	3.43	8.85	7.54	—
+BTB								
Bi-Mode	15.51	2.93	6.23	8.06	3.39	<u>8.74</u>	7.48	0.86
Tri-Mode2	15.40	2.91	<u>6.12</u>	8.05	3.41	8.83	7.47	0.99
Tri-Mode3	<u>15.10</u>	<u>2.88</u>	6.23	<u>7.92</u>	<u>3.33</u>	<u>8.74</u>	<u>7.37</u>	<u>2.32</u>

Tri-Mode3 は 1.5KB では 1.625KB, 12KB では 13KB, 24KB では .26KB .
数字は, 各構成で最も低い予測ミス率, もしくは, 最も高い予測ミス削減率を示す.

以上, Weakly PHT の追加による影響を確かめた . Direction PHT で Strongly 分岐のみを扱うことによる予測精度低下は m88ksim 以外のプログラムでは認められず, 殆どのプログラムで予測精度が向上した . また, Weakly 分岐のみを対象とする PHT (3bit 飽和カウンタ) を用意することによって, 多くのプログラムで予測精度が向上した . ただし, gcc では, Strongly 分岐の予測精度は向上したが, Weakly 分岐の予測精度が大幅に低下したため, Tri-Mode2 の予測精度が最も高くなっている .

6. おわりに

Bi-Mode 分岐予測器の Choice PHT が Weakly 分岐と判断した場合に分岐偏向がないことに着目し, Bi-Mode 予測器に Weakly 分岐用の予測表 (Weakly PHT) を追加した Tri-Mode 分岐予測器提案した . Weakly PHT の追加によって, Direction PHT において, 偏向のある Strongly 分岐と偏向のない Weakly 分岐の競合を回避できる .

SPECint95(ref) ベンチマークの実験において, 12KB 容量の分岐予測器では, BTB hit/miss 参照手法⁶⁾ を適用した Tri-Mode 予測器 (Tri-Mode3) が, 従来の Bi-Mode 予測器と比較して平均 2.78% の予測ミス削減率を示した . 殆どのプログラムで 3bit 飽和カウンタを Weakly PHT に採用した Tri-Mode 予測器 (Tri-Mode3) が最も高い予測精度を示したが, gcc では 2bit 飽和カウンタを Weakly PHT に採用した Tri-

Mode 予測器 (Tri-Mode2) が最も高い予測精度を示した . これは, gcc では, Direction PHT の予測精度向上よりも, Weakly PHT の予測精度低下による影響が大きかったためである .

謝辞 本研究の一部は, 21世紀 COE プログラム「プロダクティブ ICT アカデミア」によるものである .

参 考 文 献

- 1) Smith, J. E.: A Study of Branch Prediction Strategies, *Proc. of 8th ISCA*, pp. 135-148 (1981).
- 2) McFarling, S.: Combining branch predictors, Technical Report TN-36, Digital Western Research Laboratory (1993).
- 3) Lee, C. C., Chen, I. K. and Mudge, T. N.: The Bi-Mode Branch Predictor, *Proc. of MICRO-30*, pp. 4-13 (1997).
- 4) 吉瀬, 片桐ほか: 極端な偏りを利用する Bi-Mode++分岐予測器の提案, 情処研報 (2005-ARC-161), pp. 57-62 (2005).
- 5) Haungs, M., Sallee, P. and Farrens, M.: Branch Transition Rate: A New Metric for Improved Branch Classification Analysis, *Proceedings of 6th HPCA*, pp. 241-250 (2000).
- 6) 斎藤, 山名: BTB のエントリ有無を参照した分岐予測器, 情処 ACS 論文誌, Vol. 45, No. SIG 11(ACS 7), pp. 71-79 (2004).
- 7) Burger, D. and Austin, T. M.: The SimpleScalar Tool Set, Version 2.0, Technical report (1997).