

# Bloom フィルタを用いた自動メモ化プロセッサの省電力化

藤井 政圭\*, 佐藤 裕貴, 津邑 公暁 (名古屋工業大学)

A Low-Energy Implementation of Auto-Memoization Processor by utilizing Bloom Filters

Masayoshi Fujii, Yuki Sato, Tomoaki Tsumura (Nagoya Institute of Technology)

## 1. はじめに

プロセッサの高速化手法として、プログラムの並列性に着目した手法が目されている。一方、これとは全く異なる概念である計算再利用と呼ばれる高速化手法がある。本研究では、この計算再利用の実行モデルの1つである自動メモ化プロセッサ<sup>(1)</sup>において、**Bloom** フィルタ<sup>(2)</sup>と RAM を用いることで、性能を維持しつつ、消費エネルギーの削減を図る。

## 2. 自動メモ化プロセッサ

自動メモ化プロセッサは、関数およびループを計算再利用の対象区間と見なし、実行時にその入力と出力の組を再利用表と呼ばれる表へ登録する。そして、再び同一命令区間を同一入力を用いて実行する際に、再利用を適用できるかどうか調べるために現在の入力と過去の入力を比較する。これを、再利用テストと呼ぶ。この再利用テストに成功した場合、自動メモ化プロセッサは命令区間の実行を省略する。

なお、計算再利用を適用する際、過去の入力を再利用表から検索し、現在の入力との一致比較を行うための、オーバーヘッドが発生する。このオーバーヘッドが大きい場合、命令区間の実行を省略することによるサイクル数削減の効果を十分に得ることができない。そのため自動メモ化プロセッサでは、全てのエンタリに対して同時に一致比較を行うことが可能なメモリである CAM を再利用表に用いることでこのオーバーヘッドを抑えている。

## 3. 提案手法

CAM は高性能である反面、消費電力、面積、製造コストが非常に大きいため、実用化を考えると CAM のサイズは極力小さく抑えることが望ましい。しかし、自動メモ化プロセッサがこれまで想定してきた CAM のサイズはオンチップに実装するには大きく、ハードウェアコストの面で自動メモ化プロセッサ実用化における障害の一つとなっている。

そこで本論文では、RAM を用いて CAM の一部を代替することで、自動メモ化プロセッサのハードウェアコストを削減する手法を提案する。なお、RAM は低コストである一方、RAM 上での連想検索は CAM に比べ低速であるため、CAM を用いた場合より性能が低下することが考えられる。そのため、提案手法では、Bloom フィルタを用いて連想検索が失敗するかどうかを事前に判定することで、RAM 上での連想検索によるオーバーヘッドを抑制する。また、連想検索に適した構成で RAM を用いることで、RAM の読み出し回数を削減し、オーバーヘッドを抑制する。このようにして、ハードウェアコストを削減しつつ、既存

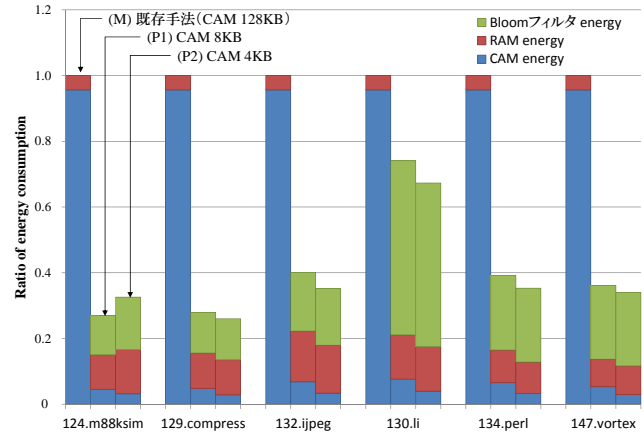


Fig.1 Evaluation Results

の自動メモ化プロセッサと同等な性能の維持を目指す。

## 4. 評価

評価対象のプログラムには、汎用ベンチマークである SPEC CPU95 を用い、自動メモ化プロセッサにおける、既存手法および提案手法を用いた場合の再利用表の消費エネルギーを算出した。評価結果を Fig.1 に示す。結果は、(M) 既存の自動メモ化プロセッサ (CAM 128KBytes), (P1) 提案モデル 1 (CAM 8KBytes), (P2) 提案モデル 2 (CAM 4KBytes) を示している。提案モデルでは、(M) と比べ、再利用表の消費エネルギーが削減でき、(P2) の消費エネルギー削減率は、平均 50.4 %, 最大 67.5 % となった。また、(P2) の実行サイクル数削減率を算出したところ、(M) と比べ、平均 2.5 %, 最大 6.4 % となり、性能がむしろ向上することを確認した。

## 5. おわりに

本研究では、自動メモ化プロセッサのハードウェアコストを削減する手法として、Bloom フィルタと RAM を用いて、CAM の一部を代替する手法を提案した。これにより、既存の自動メモ化プロセッサと同等以上の性能を維持しつつ、ハードウェアコストを削減した。今後の課題としては、自動メモ化プロセッサに適した Bloom フィルタの構成の考案があげられる。

### 文献

(1) Tsumura, T., et.al.: Design and Evaluation of an Auto-Memoization Processor, *Proc. Parallel and Distributed Computing and Networks*, pp. 245–250 (2007)

(2) Bloom, B. H.: Space/Time Trade-offs in Hash Coding with Allowable Errors, *Commun. ACM*, Vol. 13, No. 7, pp. 422–426 (1970)