

演算器アレイ型アクセラレータのための命令変換手法

森 浩 大^{†1} 大 上 俊^{†1} 下 岡 俊 介^{†1}
吉 村 和 浩^{†1} 中 田 尚^{†1} 中 島 康 彦^{†1}

1. はじめに

画像処理や科学技術計算といった膨大な計算能力が要求される場面では、性能向上だけでなく低消費電力化が求められている。専用ハードウェアは、特定の問題に対して最適化されたハードウェアにより安定した性能を実現し、かつ、低消費電力を実現している。しかし、専用ハードウェアでは、日々進歩する様々な規格への対応や、膨大な開発コストの問題から採用が困難となっており、柔軟性と高効率を両立するアーキテクチャが求められている。

本稿では、演算器アレイで既存プログラムを高速実行するために求められる制約を明らかにし、GCC が生成した命令列からアレイ実行可能な命令列への命令変換手法を提案する。これにより高いプログラマビリティをプログラムに提供する。

2. 関連研究

ソフトウェアとの親和性、および、リコンフィギャラブルデータパスによる高速性の両立を目指す仕組みに、ADRES¹⁾、TRIPS²⁾ や PPA³⁾ などの粗粒度リコンフィギャラブルアレイ (CGRA) がある。ADRES は、VLIW エンジンにて既存 VLIW 命令を実行し、アクセラレータエンジンにて専用コンパイラで生成されたループカーネルを高速実行する。4×4 の機能ユニット構成の ADRES は、4 命令発行の一般的な VLIW 構成のバックエンドとして、各演算ユニットにローカルレジスタファイルを備えた 4×3 程度の CGRA を接続する。専用コンパイラが、制限付きの C 言語により記述されたプログラムから、4×3 命令発行に対応する CGRA 命令を生成する。TRIPS はデータフローとアレイ構造を直接記述できる EDGE 命令セット⁴⁾ を用いることで、多数の演算器を制御することを目指している。PPA は、2×2 の CGRA を 8 コア接続し、

ループカーネルに応じてアレイサイズの構成を変更できる。

しかし、既存のいずれのアクセラレータも、構成情報を出力可能な新たなコンパイラの開発や専用命令セットを使用しなければならないために、既存プログラムとの互換性が低いという問題がある。特に、ループカーネルの命令数が機能ユニットの数を超える場合には、1 つの機能ユニットが複数の命令を切り替えながら実行しなければならず、時間軸方向の割り当ても考慮しなければならない。

3. プログラマビリティを備えた演算器アレイ

既存プログラムを直接実行できるプログラマビリティをもつ CGRA は以下の特徴を備えていることが必要である。

- (1) 演算器は幅 $W \times$ 段数 D の 2 次元アレイ。
 - (2) 同一段に対するバイパスと次段 (D の正方向) への演算器間ネットワークを持つ。次段へのネットワークは $W \times W$ の完全結合である。
 - (3) 命令幅 W の VLIW 命令セットを用いる。
- この中で (3) は必須事項ではないが、幅方向の並列性抽出を VLIW 対応の既存コンパイラで行うことで、実行時のハードウェアの負担を大幅に軽減できる。
- 以上を満たす CGRA であれば、以下の制約条件を満たすループが直接実行の候補となる。
- (4) ループ中の分岐は先頭に戻る後方分岐と、ループを抜ける前方分岐のみである。
 - (5) イタレーションを超える依存関係が無い。
 - (6) アレイ対象ループの開始位置、最大ループ回数、入出力データの範囲が明らか。

これらの条件を満たすループの検出とヒント命令の挿入を自動変換機構で実現することにより、既存プログラムを CGRA で直接実行することが出来る。

4. 命令変換

すべての最内ループに対して、前述の制約条件を満たすループを検出し、必要ならばプログラムの変形を

^{†1} 奈良先端科学技術大学院大学
Nara Institute of Science and Technology

行った後に、ヒント命令の挿入を行うプログラムを作成した。このプログラムはGCCの出力するアセンブリコードに対するトランスレータとして実装した。すべての制約を満たすことができる最内ループのみを変換対象とし、それ以外のループの変換は行わない。

現在の実装では、アセンブリコードからループ回数が明示的に判断可能なループのみを対象としているほか、リンクリストのようなアクセス範囲の解析が困難なデータ構造を検出すると変換対象外とする。

5. 評価

自動変換プログラムによって変換されたコードをRTLシミュレータを用いて評価した。評価に用いる演算器アレイは幅 6×32 段の構成とした。ロードおよびストアのレイテンシは2である。また、現在は評価を簡単化するため、すべての浮動小数点演算が1サイクルで完了すると仮定している。浮動小数点演算のレイテンシが延びると、より大きなアレイが必要となるが、現在検討中の時分割多重による並列化機構⁵⁾を応用することにより、解決可能であると考えられる。

評価プログラムにはSPEC95のtomcatv, swim, mgridの3つを用いた。ただし、現在のシミュレータは単精度浮動小数点のみをサポートするため、倍精度を用いている部分は単精度へと変更した。ベンチマークの入力データセットにはtomcatvとswimにはtrainを、mgridにはrefを用いた。評価は各ベンチマークのデータ初期化完了後、40億命令または実行終了までを対象とした。

提案手法を適用した結果、すべての評価プログラムを正しく自動変換できることが確認できた。

各ベンチマークの実行サイクルの内訳を図1に示す。結果は変換前の各ベンチマークの実行サイクル数で正規化した。ここで、変換対象外は変換対象とならなかった部分、変換対象は変換対象となった部分を通

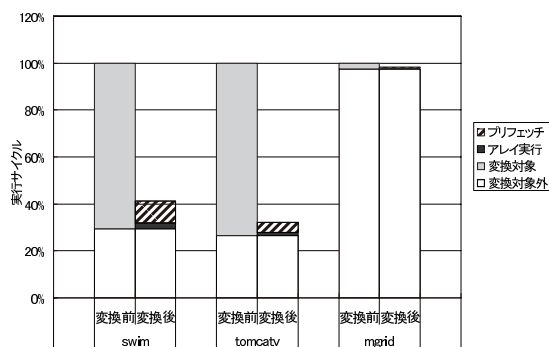


図1 実行サイクル

常実行した場合のサイクル数、アレイ実行は変換対象ループの実行サイクル数、プリフェッチはアレイ実行のために必要になったプリフェッチのサイクル数を示す。この結果から、tomcatv, swimは70%以上の部分がアレイ実行出来ており、それぞれ3.12倍、2.42倍の性能向上を達成している。また、変換対象区間のみで比較するとtomcatvは約13倍、swimは約6倍の性能向上がみられる。一方mgridは、最も大きいループを自動変換できなかったため、図1の変換対象の占める割合が他の2つのベンチマークに比べて非常に小さくなっている。そのため性能向上は約1%に留まったと考えられる。

自動変換できなかったmgridの最内ループはループが大きすぎるのが原因であり、これはループを分割する等の方法により解決できる。

6. むすび

本稿では、電力効率に優れたリコンフィギュラブルプロセッサの互換性問題を解決するための命令変換手法を提案した。提案手法をコンパイラ拡張として実装し、3種の科学技術計算ベンチマークの自動変換を確認した。評価の結果、変換前と比較しRTLシミュレータ上で最大3.12倍の性能向上を確認した。

今後は、変換プログラムを改良し変換可能なループの割合を増やすとともに、より多くのベンチマークによる評価を行う。

謝辞

本研究の一部は科学研究費補助金(若手研究(B)課題番号22700053)による。

参考文献

- 1) Bouwens, F.J. et al.: Architecture Enhancements for the ADRES Coarse-Grained Reconfigurable Array, *HiPEAC'08*, pp.66-81 (2008).
- 2) Sankaralingam, K. et al.: Distributed Microarchitectural Protocols in the TRIPS Prototype Processor, *MICRO 39*, pp.480-491 (2006).
- 3) Park, H. et al.: Polymorphic pipeline array: a flexible multicore accelerator with virtualized execution for mobile multimedia applications, *MICRO-42*, pp.370-380 (2009).
- 4) Burger, D. et al.: Scaling to the End of Silicon with EDGE Architectures, *Computer*, Vol.37, No.7, pp.44-55 (2004).
- 5) 岩上拓矢, 吉村和浩, 中田 尚, 中島康彦: 仮想化機構による演算器アレイ型アクセラレータの効率化, 先進的計算基盤システムシンポジウム SACSIS2011 (2011).