様式Ｃ－１９

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成 25 年 5 月 28 日現在

機関番号：12601
研究種目：挑戦的萌芽研究
研究期間：2010～2012
課題番号：22650008
研究課題名（和文）データレジデントコンピューティングの研究

研究課題名（英文）　Research on Data Resident Computing

研究代表者
中村　宏（NAKAMURA　HIROSHI）
東京大学・大学院情報理工学系研究科・教授
研究者番号：20212102

研究成果の概要（和文）：
今日のコンピュテーションにおいて高性能化と低消費電力化を妨げている大きな壁は、演算や処理を行う部分ではなく、演算処理を行う部分に対するデータ転送にこそ存在している。そこで、データ転送を直接最適化することが可能な新しい実行パラダイムであるデータレジデントコンピューティングを提案する。メニーコアプロセッサを対象に、このパラダイムに基づく２つの新しい処理方式として、転送路の物理的な場所を意識したデータ圧縮方式と、転送路における競合を考慮したプロセススケジューリング方式の２つを提案し、その有効性を示した。

研究成果の概要（英文）：

Currently, inefficiency of data transfers between computing components prevents computer systems from further performance improvement or power reduction, rather than computing components themselves. To overcome this problem, this research proposes a new computing paradigm called Data Resident Computing, which can directly optimize data transfers. This paradigm is applied to many-core processors, and two new computing methods are proposed. One is data compression by making use of locative information of paths of data transfers, and the other is novel process scheduling by considering the interferences between data transfers of different cores. Experimental results reveal the effectiveness of the proposed techniques.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2010 年度	1,100,000	0	1,100,000
2011 年度	1,200,000	360,000	1,560,000
2012 年度	600,000	180,000	780,000
総　計	2,900,000	540,000	3,440,000

研究分野：総合領域
科研費の分科・細目：情報学、計算機システム・ネットワーク
キーワード：①コンピューティング②実行モデル③アーキテクチャ④高性能化⑤低消費電力化

１．研究開始当初の背景

情報システム機器は日本全体の電力消費の約 5%を消費し、2025 年にはその消費電力は 5 倍になると予想されている。したがって、情報化社会のさらなる高度化を実現するためにはコンピュテーションの高性能化と低消費電力化が重要かつ緊急の課題となってい

る。
VLSI 内部での実行から広域分散環境での実行に至るあらゆるコンピュテーションにおいてその高性能化と低消費電力化を妨げている大きな壁は、演算や処理を行う部分ではなくこの演算処理部とのデータ転送にこそ存在している。しかし、現在のコンピュテーションは本質的には処理の種類と処理に用いるデータの論理的な場所のみを指定する実行モデルに基づいているため、上記の壁を解決する直接的な手段を持ち合わせていない。
例えば、VLSI 内部では演算処理部へデータを供給するためのメモリアクセスが性能面でも消費電力面でもボトルネックである。遅くて遠いメモリからデータを供給するのは長い時間と大きな電力を必要とするため小容量高性能なキャッシュメモリを搭載するが、プログラムからはデータの「論理的な場所」しか指定できないため、必要なデータがキャッシュメモリに存在することを直接的には保証できない。広域分散環境ではサービスを処理するサーバの物理的な場所を意識することなくユーザにシームレスなサービスを提供するクラウドコンピューティングが注目されている。しかし遠くのサーバとのデータ転送に長い時間と大きな電力を要するのは自明である。いつどこでどの程度の転送が生じるかをプログラムあるいはソフトウェアから制御できない点はシステムを設計する上でも大きな問題であり、転送路の性能仕様を過剰にせざるを得ずコストの増加につながる。さらにネットワークの消費電力は実際の転送量ではなく性能仕様に依存するため、消費電力面でも大きな問題となっている。

２．研究の目的

本研究では、上記問題を解決する新しい実行モデル「データレジデントコンピューティング」を提案する。これは演算処理を行う部分、データ、およびデータ転送路の物理的な場所情報を陽に指定できそれらの情報を用いたデータ転送の最適化が可能な、新しい実行モデルに基づくコンピュテーションの実現を目指すものである。本研究の目的は、この新しい実行パラダイムによって上記の壁を打破し VLSI 内部から広域分散環境下に至るまで、社会が必要としているあらゆるコンピュテーションの高性能化と低消費電力化を実現することである。研究期間内に、この実行モデルの実現可能性を明らかにし、実際のコンピュテーションに適用した場合の性能面と消費電力面での効果を評価し有効性を示す。本研究により、情報化社会のさらなる高度化を支える新しい高性能・低消費電力コンピュテーションへの道が切り拓かれることが期待される。

３．研究の方法

データレジデントコンピューティングの有効性を実証するプラットフォームとして、メニーコアプロセッサを取り上げることとした。メニーコアプロセッサは、VLSI 内部で多数のプロセッサコアがパケットスイッチングネットワークで接続される。メニーコアプロセッサは現在すでに実用化されており、半導体集積度の向上により搭載されるコア数は増加する傾向にある。そのため、性能と消費電力のボトルネックが、演算器などの計算資源だけではなく、計算資源に対するデータ転送路にも存在すること、また一つの VLSI 内部に搭載されるコア数が増大する傾向にあるため、今日のコンピュータシステムのボトルネックがよりデータ転送に移っていくという状況を端的に表す、わかりやすく現実的なハードウェアプラットフォームだからである。
研究方法としては、メニーコアプロセッサを対象に、以下の２点を中心に、提案する実行パラダイムであるデータレジデントコンピューティングの有効性を検討した。

(1) データ転送最適化：データ転送路の混雑が性能に与える影響を評価する評価環境を構築した。次にデータ転送路の混雑を緩和する方式としてデータ圧縮に着目し、データ圧縮と性能向上の間の関係を検討し、データレジデント方式の具体例として、転送路の物理的な場所を意識したデータ圧縮方式を新規に提案し、その有効性の評価、を行った。

(2) プロセススケジューリング方式：メニーコアプロセッサでは、一つのプロセスが複数のコアを占有しつつ、同時には、複数のプロセスが異なるコアの上で独立に実行される、という処理形態となる。この場合、コアという計算資源を有効に活用することが性能向上のために必須である一方、複数の独立した依存関係のないプロセスがデータ転送路を共有することによる性能低下を抑えることも重要となる。この観点から、データレジデント方式に基づく新しいプロセススケジューリング手法を提案し、その有効性を検討した。

４．研究成果

前項で述べた２つの研究項目に関して以下の成果を得ることができた。

(1) データ転送最適化：複数のコアがメッシュネットワークで接続される 3 次元積層のメ

ニーコアプロセッサにおいて、コアの動作とネットワーク上のデータ転送をサイクルレベルで詳細に評価できるシミュレーション環境を構築した。3 次元積層のメニーコアプロセッサでは、実装上の制約により 2 次元平面（レイヤー）内の転送路はビット幅が広くバンド幅は大きいが、レイヤー間の転送路は物理構成上ビット幅が狭く、バンド幅は小さい。このような状況も反映できるようにシミュレーション環境を構成した。

メニーコアプロセッサでは、データ転送路の混雑が性能低下の原因となる場合がある。そこで、データ転送路の混雑緩和とデータ転送時間の短縮を実現する手法としてデータ圧縮に着目した。データ圧縮には種々の手法があるが、以下が一般に成立する。

- データ圧縮率はデータの中身に依存する。圧縮できず却ってデータ量が増える場合もある。
- データ圧縮でき転送量が減れば、データ転送に要する時間は短縮される。
- データ圧縮・復元自体にも時間を要するので、データ圧縮に成功しデータ転送時間が短縮しても必ずしも性能は向上しない。

そこで、ネットワークのバンド幅が一様ではない 3 次元積層のメニーコアプロセッサを対象に以下の 4 つの手法を提案し、データ圧縮を行わない場合との比較評価を行った。

- SC（Static Compression）:データの特性、送受信の位置関係に依らず常にデータ転送時にデータ圧縮を行う方法。
- AC1:データ圧縮可能かどうかを確認し圧縮に成功する場合だけ圧縮を行う
- AC2:データの送受信の位置関係を考慮し、3 次元実装のレイヤー間の転送の時のみ圧縮をする。
- AC1+2: データの送受信の位置関係を考慮し、3 次元実装のレイヤー間の転送の時で、しかも圧縮に成功するときだけ圧縮を行う。

AC2 および AC1+2 が、データ転送路の物理的な位置を考慮しており、提案するデータレジデントコンピューティングに基づく手法となる。

評価においては以下の仮定を置いた。

- 圧縮手法として、Wisconsin 大学の A. R. Alameldeen らによる ‘Frequent Pattern Compression’ を用いる。
- 圧縮・復元には 3 サイクルを要する
- 圧縮可能か否かのチェックには 1 サイクルのみ要する。
- 送受信の位置関係のチェックはヘッダの数ビットを比較するだけなので、時間オーバヘッドはない。
- レイヤー内（2 次元平面内）の転送路は 128bit、レイヤー間の転送路は 16bit
- コア数は 16 に固定するが、レイヤー数は 2, 4, 8 の 3 通りを想定する。

これらの条件下で評価を行った結果を図 1 に示す。この図は、圧縮を行わない場合に対する相対実行時間を表し、グラフは小さい方が高性能となる。

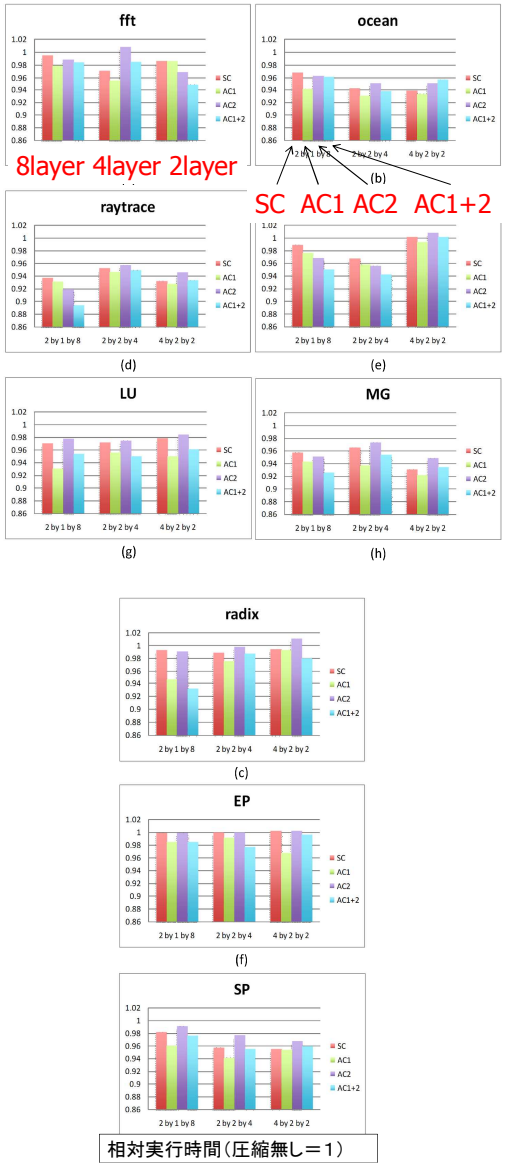

図 1　データ圧縮の評価結果

図 1 よりわかるように、どの手法が良いかはプログラムに依存する。しかし、レイヤー数が多くなると AC1+2 が最もよくなるプログラムが多くなり、全プログラムの平均を取ると非圧縮時と比べて AC1+2 は約 1 割の性能向上を達成できた。これは、狭い転送路がある場合にはその転送路を経由するか否かを考慮したうえでデータ圧縮を行うことが最も良いことを表し、データレジデントコンピューティングの有効性が示された。今後、コアの能力は向上するがコア間の転送路を増強す

ることは技術的に難しいため、提案手法の有効性はさらに高まるものと考えられる。

(2) プロセススケジューリング方式：メニーコアプロセッサでは、一つのプロセスが複数のコアを占有しつつ、同時には、複数のプロセスが異なる複数のコア上で独立に実行される、という処理形態となる。プロセスにはできるだけ多くのコアを割り当てたほうが一般に性能は向上するため、どれだけのコアをどのプロセスに割り当てるかというプロセススケジューリングが重要となる。このプロセススケジューリングにおいては以下の 2 点を考慮する必要がある。第 1 に、利用可能なコア数を増やすことが性能向上へどの程度寄与するかはプロセスの特性によるため、限られたコアという計算資源をどのプロセスにいくつ割り当てるのが良いかは、各プロセスの特性による。第 2 に、複数の独立したプロセスがデータ転送路（正確にはキャッシュメモリを含むデータ供給部、以下データ供給部と表す）を共有することになるため必然的にデータ供給部における競合が発生する。したがって、どのプロセスにいくつのコアを割り当てるのが良いかはこの競合状況にも依存する。そのためデータ供給部の使用を最適化する必要があるが、独立なプロセス間の競合はあらかじめ予測することのできない点が問題を難しくしている。そこで、この第 2 の点を、データレジデントコンピューティングに基づいて解決することを目指す。

提案する手法はデータ供給部の競合を考慮した実行モデルとこのモデルに基づくプロセススケジューリングである。提案する実行モデルは、性能向上率と割り当てるコア数の関係を扱う統一的なモデルであり、あらかじめ予測できない実行時に発生する競合を、モデルの係数として実行中に取得・学習しモデルに反映させることを可能とする。そしてこのモデルに基づいてプロセススケジューリングを最適化する。

モデル式は以下で与えられる。

$$\text{性能向上率} = \frac{1}{\alpha + \left(\beta / \text{コア数}\right) + \left(\gamma \times \text{コア数}\right)}$$

ただし、$\alpha + \beta + \gamma = 1$

分母の第 2 項は、並列化に伴いコア数を増加した時の実行時間の短縮を表し、分母の第 3 項はデータ競合によりコア数を増加した時の実行時間の伸びを表す。分母の第 1 項と第 2 項は並列化による性能向上を示すアムダールの法則であるが、この第 3 項はこれまで考えられたことがなく、データレジデントコンピューティングに基づく新しい着想である。

このモデルの正しさを確認するために、実際に 12 コアを搭載する AMD Opteron 6172 microprocessors を 4 ソケット実装する IBM Systemx3755 M3 server（合計で 48 コアを搭載）上で、専有するコア数を変化させて並列アプリケーションを実行したときの性能を測定した。図 2 にその結果を示す。図 2 は横軸がコア数、縦軸が相対性能である。図中の赤字は、コア数を増やすとむしろ性能が低下する現象を観測している部分であり、この現象は、提案モデルの分母の第 3 項でしか説明できない。このことから提案するモデルの正しさと、データ供給路の競合を考慮することの妥当性を示すことができた。

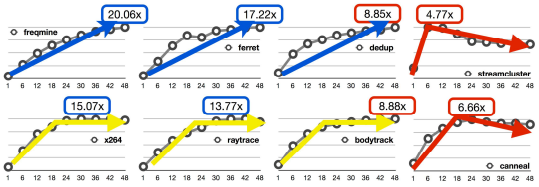

図 2：コア数と相対性能の関係

このモデルに基づくスケジューリング方式として以下の 3 つを提案し、既存の OS である Linux と比較してその有効性を調べた。

- SBMP-base:提案するモデルで性能向上率を予測するスケジューリング
- SBMP-PP: base に対し、実行状況が変化することを予測(PP: Phase Prediction)し、実行状況ごとにモデルに基づいて適応的にスケジューリングを行う方式
- SBMP-CD: SBMP-PP に対し、プロセスに割り当てるコア数を実行中に融通(CD: Core Donation)する方式

評価結果を図 3 に示す。横軸は同時に実行するプロセス組の ID を表す。縦軸は ANTT (Average Normalized Turnaround Time)を表し、値が小さい方が性能は良い。図 3 を見るとわかるようにどの方式が良いかはプロセスの組み合わせに依存するが、すべてのプロセス組に対する平均を取ると ANTT の値は SBMP-base で 1.99, Linux で 1.83, SBMP-PP で 1.70, SBMP-CD で 1.65 となり、提案する SBMP-PP と SBMP-CD の有効性が確認できた。また、ほとんど全てのプロセス組に対して、提案手法の SMBP-PP と SMBP-CD は既存の OS である Linux よりも ANTT が小さく高性能化を達成することができる。このことから、実行時にデータ供給部の競合を検出し、その競合が性能に与える影響をモデル化することで、この競合状況を考慮してメニーコアプロセッサ上でのプロセススケジューリングを最適化する、というデータレジデントコンピ

ューティングに基づく手法の有効性が確認できた。

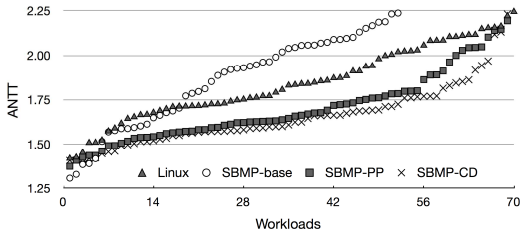

図 3：スケジューリング方式と ANTT の関係

以上の(1)と(2)の成果から、データレジデントコンピューティングの有効性を示すことができた。

５．主な発表論文等
（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計　１件）

Yuan He, Hiroki Matsutani, Hiroshi Sasaki, and Hiroshi Nakamura, “Adaptive Data Compression on 3D Network-on-Chips”, IPSJ Transactions on Computing Systems, Vol.5 No.1, 2012, 80-87

〔学会発表〕（計　２件）

武安聡, 今井雅, 中村宏, ”パケット転送経路の偏りに着目した高性能非同期式ネットワークオンチップの検討”, 電子情報通信学会技術研究報告, VLD2010-66, pp. 66-72, 2010

Hiroshi Sasaki, Teruo Tanimito, Koji Inoue, and Hiroshi Nakamura, “Scalability-Based Manycore Partitioning,”, IEEE International Conference on Parallel Architecture and Compilation Technology, 2012.9.20-2012.9.22, Minneapolis, USA

６．研究組織
(1)研究代表者
中村　宏（NAKAMURA　HIROSHI）
東京大学・大学院情報理工学系研究科・教授
研究者番号：20212102

(2)研究分担者：なし

(3)連携研究者：なし