

科学研究費助成事業 研究成果報告書

平成 26 年 6 月 13 日現在

機関番号：12601

研究種目：挑戦的萌芽研究

研究期間：2012～2013

課題番号：24650019

研究課題名(和文) ストリーム処理技術を応用したシステム自動修復技術に関する研究

研究課題名(英文) Automatic correction of hardware systems based on stream processing

研究代表者

藤田 昌宏 (Fujita, Masahiro)

東京大学・学内共同利用施設等・教授

研究者番号：70323524

交付決定額(研究期間全体)：(直接経費) 3,000,000円、(間接経費) 900,000円

研究成果の概要(和文)：画像、ウェブやセンサネットワークからのデータ(一般のビッグデータ)を処理する際には、同じ計算を個々のデータに適用するという処理が繰り返される。それらのデータをストリームと呼ばれるデータの流れて表現し、個々の計算は表を多重に引くことで処理する新規手法を考案した。表を多数用意しておくことで、異なるデータに対する異なる処理を超並列に実行できる(パイプライン処理と呼ばれる)。表はメモリで構成されるため、従来のプロセッサベースの計算に対して、メモリベースコンピューティングとも呼ばれる。特に、一定範囲で表に不具合が発生した場合に、周りの表を修正することで、自動修復する技術を中心に研究開発した。

研究成果の概要(英文)：In big data analysis, such as analyzing data on images, web manipulations, and sensor networks, same operations are repeated on each data item. Such big data are arranged as very long flows of data, called data stream, so that operations on different data items can be processed in parallel as highly pipelined operations. We have developed a new methodology where such data streams are processed by multiple retrievals of relatively small tables. As all tables are retrieved in parallel, we can expect huge parallelisms. It is also called memory-based computing in contrast to processor-based processing which is widely used today. We have developed a set of techniques by which the processing system can be automatically corrected even if some of the tables are disorder by adjusting other tables appropriately.

研究分野：情報学

科研費の分科・細目：計算システム

キーワード：ストリーム処理 設計自動修復 高性能計算 多重表引 メモリベース計算

1. 研究開始当初の背景

(1) 半導体微細化技術の進歩や実アプリケーションを考慮したアーキテクチャの進歩により、マルチコア・メニーコアなどの技術が浸透してきており、計算機側にデータさえ揃えば、高速計算処理が可能な状況になっている。一方、アプリケーション側では、ウェブアプリやセンサーネットワーク、さらに各種イーコマース処理などから膨大なデータが抽出され、その効率的な利用が期待されている(ビッグデータ処理)。このようなデータ処理をすべてプロセッサ側で処理しようとする、データを格納してあるストレージやデータベース側からプロセッサ側に大量のデータを転送する必要があるが、両者の間のデータ転送速度には限界があり、結果的にプロセッサがデータ待ちの状態で待機している時間が少なくなく、処理効率が悪くなっている。これは、処理速度の面だけでなく、処理に対するエネルギー効率の面でも効率が悪い。

(2) また、社会の隅々まで浸透している組込みシステムでも大量のデータ処理、特に画像や信号処理が要求されている。これらの大量のデータ処理をすべて汎用組込み用プロセッサで実行するには、電力消費の大きい高速プロセッサを多数用意する必要があり、高エネルギー効率での実行が最重要課題である組込みシステムでは望ましくない。画像や信号処理における大量データに対する処理を全てプロセッサで実行することはエネルギー効率の面と処理速度の両面から難しくなっている。

2. 研究の目的

(1) 現状、ほぼすべての処理がストレージ側からプロセッサ側に一度データを転送してから実行されているが、それではデータを転送するための時間がボトルネックとなってしまう場合が少なくない。そこで、プロセッサ側の処理の一定量をストレージ側で、ストレージ側からプロセッサ側へデータを送る前に実行できる処理手法を研究開発する。

(2) 従来のハードウェアでの計算はその演算専用の回路を作成して実行している。現状ではその内、一部がFPGA(Field Programmable Gate Array)を利用して表引きとプログラマブル配線により柔軟に実行されている。FPGAによる実行は、ソフトウェアによる実行に比べれば高エネルギー効率で処理を実行できるが、プログラマブル配線によりその効率には限界がある。そこで、プログラマブル配線を極力少なくした新しいメモリベースコンピューティングアーキテクチャを新規に考案し、解析する。

(3) メモリベースコンピューティングでは、種々の計算を多数の比較的小さなメモリを

多重に検索することで実現する。このため、1つのメモリが故障か何かの理由により利用できなくなる、あるいは表引き結果が一部誤るようになっても、周りのメモリの内容を適切に修正することにより、トータルとしての計算を正しく実行できる可能性がある。この修復が自動的にできれば、信頼性の高いシステムを構築できると考えられ、効率のよい自動修復技術を研究開発する。

(3) メモリベースコンピューティング技術は新しいものであり、その動作の正しさの確認手法(設計検証や故障の有無の検査)も効率的なものを新規に考案する必要がある。例えば、故障モデルとして、従来の縮退故障モデルは適切であるとは言えない。そこで、メモリベースコンピューティングに合わせた新規の故障モデルと、そのモデルに対する効率的なテストパターン生成技術について、研究開発する。

3. 研究の方法

(1) 現状、ほぼすべてのデータ処理は、ストレージ側からプロセッサ側に図1のようにデータを一度転送して実行されている。これでは、ストレージ側とプロセッサ側とのデータ転送がボトルネックとなるため、プロセッサ側の処理の多くをストレージ側に移動して実行できる、メモリベースコンピューティングを実現する。

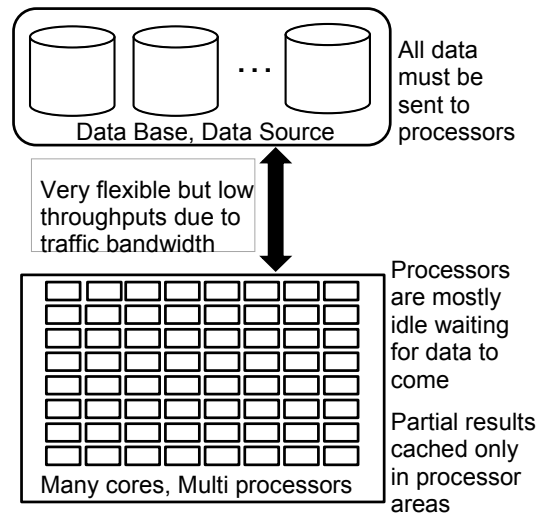


図1 従来の処理の流れ

(2) メモリベースコンピューティングは新規技術であり、それを利用するためのコンパイラや支援ソフトウェアも新規に研究開発する。特に、処理が比較的小さいメモリを多重に参照して実行されるため、与えられた任意のプログラムや順序回路から自動的にそのような多重メモリ参照による実行を自動生成する技術について研究開発する。

(3) メモリベースコンピューティング向けの故障モデルは、従来の縮退故障などでは

まくモデル化できないため、新規に考案する。特に、メモリに故障や不具合があるとメモリの内容が任意に変化すると考えられるため、それに合わせた故障モデルを新規に考案するとともに、それに対する効率的なテストパターン生成技術を研究開発する。

(4) メモリベースコンピューティングのためのデバッグ技術を新規に研究開発する。メモリベースコンピューティングでは、個々の計算ステップはメモリ参照であるため、デバッグもそれらのメモリの内容を更新することで実現できる可能性がある。設計中のある部分集合を表で表現し、それらの値を変化させることにより、設計を正しく修正する技術について、新規に研究開発する。

4. 研究成果

(1) メモリベースコンピューティングシステムとして、図2に示す構成のアーキテクチャを新規に考案した。従来プロセッサ側で処理していたものの一定量をストレージ側にメモリベースコンピューティングとして移動しており、プロセッサ側へ転送すべきデータ量を大幅に削減できる。メモリベースコンピューティングシステムはストレージに並列に直結され、それらの間のデータ転送速度は十分高くできる。また、プロセッサ側にもメモリベースコンピューティングを導入することもできる。組込みシステムなどでは、画像・信号処理のほとんどをメモリベースコンピューティングで実現でき、結果的に組込みプロセッサの負荷を大幅に削減できるとともに、結果的に高エネルギー効率で実行できる。

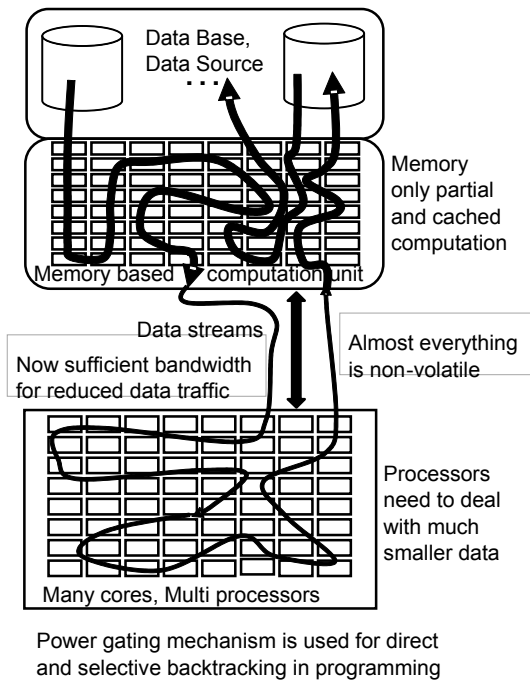


図2 提案しているメモリベースコンピューティングシステム

(2) 上の計算例として、津波伝搬シミュレーションとその計算結果を統計的に評価する処理を取り上げた。メモリベースコンピューティング用のコンパイラはまだ完成していないため、仮の実装として FPGA 上に実装した。評価した結果、統計処理も部分的にメモリ側に移動でき、プロセッサの負荷が大幅に縮小され、結果的に 50 倍程度の速度向上と、3 倍以上の実行エネルギー効率を実現できた。

(3) 任意の与えられた順序回路を 12 から 16 入出力程度（これは仮の値で、より大きくすることも可能）の部分回路の集合に分割し、個々の部分回路を多重表引きで実現する手法を新規に考案した。1つの部分回路を1つの大きなメモリで実現すると、例えば、16入力、16出力の場合、1Mビット必要となる。しかし、4入力、4出力の小さいメモリを16個使って図3のような構成にすると、全体としては16入力、16出力であるが、必要なメモリ量は1Kビットで済む。実現できる論理関数の種類が1,000分の1になる代わりに、必要なメモリビット量も1,000分の1になる。通常のハードウェア設計で見れる論理回路を16入力の部分回路に分けて解析してみると、実現しようとする論理関数の種類は、入出力の否定の有無と入力の順番の変更をすれば非常に少なく、12入力では数万程度に抑えられることがわかっている。このため、図3のような回路を一定数用意しておくことで、必要となる論理関数のほとんどを実現できると考えられる。

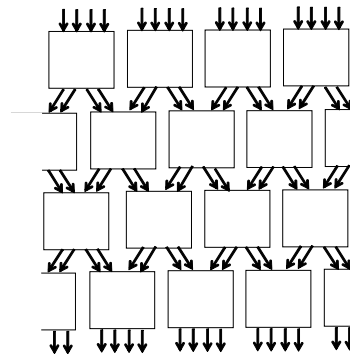


図3 多重表引きによる計算

(4) 図3のような構成の回路に対する一般的な故障もでるとして、FOF (Functionally Observable Fault) を新規に考案し、FOFのための効率の良いテストパターン生成手法も新規に開発した。FOFは故障している部分の論理関数が同じ入力変数に対する任意の論理関数になってしまう可能性があるという故障モデルで、故障による機能の変化という意味では最も一般的なモデルであり、従来はそのような機能故障に対するテストパターン生成は非常に困難であると考えられてきた。しかし、我々の提案手法は、そのような機能故障は一定数多重に存在しても効率よく検査できるパターンを生成できるものであ

る。この技術は従来の故障モデルに対する多重故障のテストパターン生成にも応用でき、今後も期待できる。

(5) 図3のような回路構成に対する、設計デバッグ技術を上記のテストパターン生成手法を発展させることで実現した。回路中の一部のメモリに不具合がある場合に、周りのメモリの内容を変更することでその不具合を回避する手法としても、通常の論理設計のバグを自動修復する手法としても利用できる。通常の設計デバッグ技術として評価したところ、マイクロプロセッサのパイプライン処理のバグなどをうまく修復できることを実証した。

(6) 提案して技術の応用として、与えられた設計から自動的にアサーション(仕様)を抽出する技術も新規に開発した。設計を自動的に解析し、アサーションの集合が生成される。これにより、もし設計が間違っている場合には、生成されるアサーションも間違っているため、それを設計者が確認することで、設計バグを早期に検出できる可能性がある。また、生成された正しいアサーションは、設計の動作に関する一種のコメントであるとも言え、設計ドキュメントして利用できる。

(7) メモリベースコンピューティングを意識しながら、ハードウェアのデータパスを効率よく生成する技術についても新規に考案した。これは、回路レベルの解析ではなく、演算器などをマクロとして、それらのマクロ間の接続を自動生成するもので、全体の処理アーキテクチャを効率よく生成できる。上記で説明した回路レベルの生成手法と組み合わせることで、大規模設計も処理可能となる。

5. 主な発表論文等

〔雑誌論文〕(計 1件)

S. Jo, T. Matsumoto, and M. Fujita, "SAT-based automatic rectification and debugging of combinational circuits with LUT insertions," *IPSJ Transactions on System LSI Design Methodology*, Vol.7, February Issue, pp. 46-55, Feb. 2014.

〔学会発表〕(計 12件)

S. Ono, T. Matsumoto, and M. Fujita, "Automatic Assertion Extraction in Gate-Level Simulation Using GPGPUs," *Proc. of IEEE 30th International Conference on Computer Design*, pp.522-523, Oct. 2012.
S. Jo, T. Matsumoto, and M. Fujita,

"SAT-based automatic rectification and debugging of combinational circuits with LUT insertions," *Proc. of IEEE 21st Asia Test Symposium*, pp.19-24, Nov. 2012.

M. Fujita, "Diagnosis and correction of buggy hardware/software with formal approaches," *17th International Symposium on VLSI Design and Test*, July 2013.

M. Fujita, "High Performance and Flexible Computing Systems with Electronic Design Automation," *2013 IEEE International Conference on Electronic Measurement & Instruments*, Aug. 2013

M. Fujita, "Diagnosis and Correction of Buggy Hardware/Software with Formal Approaches," *International Workshop on Design and Implementation of Formal Tools and Systems*, Portland, OR October 19, 2013.

K. Oshima, T. Matsumoto, and M. Fujita, "A Debugging Method for Gate Level Circuit Designs by Introducing Programmability," *Proc. of 21st IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC '13)*, pp. 78-83, Oct. 2013.

K. Oshima, T. Matsumoto, and M. Fujita, "Hardware Implementation of BLTL Property Checkers for Acceleration of Statistical Model Checking," *Proc. of the International Conference on Computer-Aided Design (ICCAD '13)*, pp. 670-676, Nov. 2013.

M. Fujita, S. Jo, S. Ono, and T. Matsumoto, "Partial synthesis through sampling with and without specification," *Proc. of the International Conference on Computer-Aided Design (ICCAD '13)*, pp. 787-794, Nov. 2013.

S. Jo, A.M. Gharehbaghi, T. Matsumoto, and M. Fujita, "Debugging Processors with Advanced Features by Reprogramming LUTs on FPGA," *Proc. of the International Conference on Field-Programmable Technology (ICFPT)*, pp. 50-57, Dec. 2013.

S. Ghandali, B. Alizadeh, M. Fujita, Z. Navabi, "RTL datapath optimization using system-level transformations," *Proc. of International Symposium on Quality Electronic Design (ISQED '14)*, pp.309-316, Mar 2014.

藤田昌宏, 城怜史, 松本剛史, "アサーション自動生成とそのシミュレーションによる完全検証," *組込技術とネット*

ワークに関するワークショップ
(ETNET2014), 2014年3月.
松本剛史, 城怜史, 藤田昌宏, “プログラム可能データベースとSMTソルバーを利用した高位設計デバッグ手法,” 組込技術とネットワークに関するワークショップ(ETNET2014), 2014年3月.

〔図書〕(計 0件)
(執筆中)

〔産業財産権〕
出願状況(計 0件)

取得状況(計 0件)

〔その他〕
ホームページ等
www.cad.t.u-tokyo.ac.jp

6. 研究組織

(1) 研究代表者

藤田 昌宏 (FUJITA, Masahiro)
東京大学大規模集積システム設計教育研究センター 教授
研究者番号: 70323524

(2) 研究分担者

松本 剛史 (MATSUMOTO, Takeshi)
東京大学大規模集積システム設計教育研究センター 助教
研究者番号: 40536140

(3) 連携研究者