

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 24 年 3 月 31 日現在

機関番号：12608

研究種目：若手研究(B)

研究期間：2010～2011

課題番号：22700047

研究課題名(和文) アクセラレータスーパーコンピュータ向けスケラブルかつ高速なチェックポイント技術

研究課題名(英文) Scalable, Fast Checkpointing for Heterogeneous Supercomputers

研究代表者

丸山 直也 (MARUYAMA NAOYA)

東京工業大学・学術国際情報センター・助教

研究者番号：60532801

研究成果の概要(和文)：本研究課題ではスーパーコンピュータなどの大規模並列計算環境においてアプリケーションを長時間実行可能にするためのソフトウェア技術を開発した。大規模なスーパーコンピュータではハードウェア等の故障による障害の発生は避けられず、システムが部分的に利用不可になる場合がある。我々はその問題に対してアプリケーション実行中に途中の状態を高速に保存し、障害が発生した場合に保存した状態から高速に復帰するためのアルゴリズムおよびその実装を行い、実際の大規模アクセラレータスーパーコンピュータ環境において実証した。

研究成果の概要(英文)： We developed a fault-tolerant software technique that allows for reliable executions of long-running applications on large-scale supercomputers, where component failures are not exceptions but norm. Such failures can be a huge problem when obtaining scientific simulation results that can be only obtainable with days of weeks of executions. Our algorithm for fault tolerant application executions in the presence of component failures allows for very fast saving of application runtime states so that they can be restarted from the saved states upon failures. We have developed a prototype implementation of the proposed algorithm and demonstrated its highly scalable performance on a large-scale heterogeneous supercomputer.

交付決定額

(金額単位：円)

	直接経費	間接経費	合計
2010年度	1,600,000	480,000	2,080,000
2011年度	1,100,000	330,000	1,430,000
年度			
年度			
年度			
総計	2,700,000	810,000	3,510,000

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：計算機システム、高性能計算、耐故障性、GPU 計算

1. 研究開始当初の背景

本研究を開始した 2010 年度は GPU 等のアクセラレータがそれまでのデスクトップコンピュータ等の小規模な環境における限られた利用から、スーパーコンピュータ等の大規

模 HPC 環境における利用へと普及し始めた時期であり、同時にアクセラレータの信頼性が重要視されるようになった時期である。例えば、東京工業大学の TSUBAME2.0 スーパーコンピュータは GPU アクセラレータをノードあ

たり3基搭載し、合計4200基のGPUを持った大規模GPUスーパーコンピュータである。このような大規模計算機資源においてはシステムの障害に対する対策が重要であり、障害の防止、障害の検知、障害に対するリカバリーがそれぞれ研究開発されてきた。しかしながら、従来の技術はその適用可能な規模に限界があり、今後のペタスケールおよびポストペタスケールスーパーコンピューティングにおける耐故障性を実現するためには不十分であった。また、アクセラレータのような新アーキテクチャに対応した耐故障技術も解決されておらず、大規模アクセラレータ環境における耐故障性達成が課題であった。

2. 研究の目的

本研究課題では近年のHPC技術の傾向と今後の発展を踏まえ、チェックポイント技術を大規模スーパーコンピュータへ適用する際の以下の問題の解決を目的とした。

(1) システムの規模に対するスケーラビリティ

今後のさらなるトランジスタ集積度の加速度的な増加により、2015年頃の100テラフロップス級システムでは数万ノード数、100万コア数程度が予測されており、さらに2018年頃のエクサフロップス級システムでは10万ノード数、2000万コア数程度が予測されている。平均故障間隔は現状のペタフロップス級システムで1日程度だが、100ペタフロップス級システムで一時間、さらにエクサフロップス級システムでは数十分となる可能性がある。主流な並列プログラミングモデルであるMPIではシステム全体の正常動作を必要とするため、このような平均故障間隔の減少によりアプリケーションのチェックポイント間隔を短くせざるを得ない。しかし、従来のチェックポイント技術ではリモートストレージへのチェックポイントの書き込みがボトルネックとなり、チェックポイント時間が平均故障間隔を上回ることが予測される。

(2) GPU等のアクセラレータへの対応

近年のHPC技術の重要な傾向の一つとしてGPUなどのアクセラレータ技術の活用が挙げられる。GPUは現在のマルチコアCPUと比較してその計算性能かつメモリバンド幅が数倍におよび、特にHPCにおける有効性が広範囲のアプリケーション分野において急速に認知されつつある。それに伴いGPUを各計算ノードに備えたGPUクラスタの導入が進んでいるが、このようなアクセラレータの急速なHPC分野への浸透に対して、その耐故障技術は十分な研究開発がされておらずGPUにおけるチェックポイントの透過

的な取得については未だ技術的な問題が残されている。例えば、通常のCPU上のシステムではOSによるメモリ管理機構を応用することでアプリケーションの利用メモリを透過的かつ効率的にチェックポイントする技術が確立されている。しかし、GPUではそもそもメモリ管理機構はファームウェアに隠蔽されており、チェックポイント等に用いることは事実上不可能である。従って、従来のCPUにおける手法とは異なるGPUに適した手法を明らかにする必要がある。

3. 研究の方法

アクセラレータスーパーコンピュータに対応したスケーラブルかつ高速なチェックポイントを実現するために、まずチェックポイントのスケーラビリティ、高速化、GPUへの対応を進めた。スケーラビリティに関しては計算ノードを一時的なチェックポイントストレージとして活用することを検討した。計算ノードを一時的なストレージとして用いるためには計算ノード自体の障害への対応が必要であり、本研究では符号化理論を応用した信頼性確保手法を検討した。高速化については、SSD等の高バンド幅デバイスの活用、マルチコアCPUを用いたチェックポイントイメージの圧縮等によるサイズの削減を検討した。GPUへの対応については、まずGPUアーキテクチャに適した手法を調査し、既存のCPU向けチェックポイントを拡張としてGPUアプリケーションのチェックポイントの実現を検討した。これらの研究を推進するために東京工業大学TSUBAME2.0スーパーコンピュータを活用した。

4. 研究成果

(1) 符号化によるスケーラブルチェックポイント

研究成果としてまず符号理論を応用することで計算ノードのローカルストレージを活用したチェックポイント手法を提案、開発し、実際に1000並列GPUアプリケーションなどの大規模実験においてその有効性を確認した。従来の共有ストレージにチェックポイントを保存する方式ではそのI/Oコストが問題であり、特に今後のスーパーコンピュータが大規模化されるにつれてより高頻度かつ高速なチェックポイント手法が必要とされる。我々の方式では各計算ノードのローカルストレージを活用することでスケーラビリティを達成し、かつ符号化により冗長性を持たせることで計算ノードの障害発生時にも再実行可能にした。また本方式ではチェックポイントの符号化コストが大きくなりうるが、我々は計算ノードを適切にグループ化することでそのコストを大幅に抑えることが可能であることを示した。またさらに符号化に

CPU・GPU を適応的に選択して用いることでそのコストを隠蔽できることを示した。

(2) 汎用的な GPU チェックポイント

GPU プログラムのチェックポイントを実現するために CUDA プログラムについて調査を行い、そのチェックポイント方式について検討を行った。CUDA プログラムではランタイムライブラリおよびGPUドライバがGPUの状態を管理しており、適切にチェックポイント・再開するためにはこれらのライブラリが保持する状態を制御する必要がある。我々は CUDA プログラムの状態の一つである GPU メモリに関して、ランタイムライブラリを拡張として独自メモリ管理機構を実現し、これによりメモリのチェックポイント・再開を可能にした。同手法を MPI を併用した用いた並列 GPU プログラムへ適用し、チェックポイントの保存および再開が正しく実行可能であることを確認した。

(3) フレームワークによる透過的なチェックポイント

並列 GPU プログラムを高速かつ透過的にチェックポイントすることを目的に上述のチェックポイント技法をもとにした研究開発を行った。具体的には透過的なチェックポイントを実現するためにアプリケーションフレームワークを構成し、フレームワークによって透過的なチェックポイントを実現した。GPU プログラミングにおいて主流である CUDA を用いたアプリケーションでは一般的には種々の技術的制約により透過性を達成することは不可能である。我々は透過的にチェックポイントが可能なアプリケーションフレームワークとして、ステンシル計算を対象としたドメイン特化型フレームワークを構築した。アプリケーションは同フレームワークのドメイン特化型言語を用いて機種独立に記述され、実際の実行コードはドメイン特化型言語から特定の機種向けソースコードを生成しそれを通常のプログラムとしてコンパイルすることで生成する。GPU 向けには CUDA コードを生成するが、チェックポイント可能なタイミングにおいて定期的に昨年度までに開発した CUDA チェックポイント技法を応用することでアプリケーションコンテキストをチェックポイントする。GPU 向けシステムレベルチェックポイントでは Pinned メモリなどの利用が必ずしもできないが、本フレームワークでは適切にリソースの破棄・復元を行うことで通常の CUDA アプリケーションと同様に Pinned メモリを使った非同期転送などの最適化を実現している。我々は本フレームワークのプロトタイプとして、チェックポイントおよびリスタート機能の試験実装を行い、GPU クラ

スタ環境において適切に動作することを確認し、Pinned メモリを用いることなどによる効率改善を確認した。

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)

Naoya Maruyama and Satoshi Matsuoka, "Model-based Fault Localization: Finding Behavioral Outliers in Large-scale Computing Systems," *New Generation Computing*, Vol 28, No. 3, pp. 237--255, 2010. 査読有り

[学会発表](計12件)

Kento Satou, Adam Moody, Kathryn Mohror, Todd Gamblin, Bronis R. De Supinski, Naoya Maruyama, Satoshi Matsuoka, "Towards an Asynchronous Checkpointing System," *IPSJ SIG Technical Reports* 2011-ARC-197 2011-HPC-132 (HOKKE-19), Sapporo, Nov 28, 2011.

Leonardo Bautista, Naoya Maruyama, Dimitri Komatitsch, Tsuboi Seiji, Franck Cappello, Satoshi Matsuoka, and Nakamura Takeshi, "FTI: High performance Fault Tolerance Interface for hybrid systems," *ACM/IEEE Supercomputing (SC '11)*, pp. 32:1--32:32, Seattle, USA, Nov 16, 2011.

Naoya Maruyama, Tatsuo Nomura, Kento Sato, and Satoshi Matsuoka, "Physis: An Implicitly Parallel Programming Model for Stencil Computations on Large-Scale GPU-Accelerated Supercomputers," *ACM/IEEE Supercomputing (SC '11)*, pp. 11:1--11:12, Seattle, USA, Nov 15, 2011.

Naoya Maruyama, "Accelerating the TSUBAME Supercomputer with Graphics Processing Units and its Implications for Systems Research," *Workshop on Large-Scale Parallel Processing (LSPP'11) in conjunction with IEEE International Parallel and Distributed Processing Symposium (IPDPS'11)*, Anchorage, USA, May 20, 2011. 招待講演

Tatsuo Nomura, Naoya Maruyama, Toshio Endo, Satoshi Matsuoka, "A Sequential Programming Framework for Large-Scale GPU-Accelerated Structured Grids," SIAM Conference on Computational Science and Engineering, Reno, USA, March 3, 2011.

Leonardo Bautista, Akira Nukada, Naoya Maruyama, Franck Cappello, Satoshi Matsuoka, "Low-overhead checkpoint for large-scale GPU-accelerated systems," High Performance Computing Conference (HiPC), Goa, India, Dec 20, 2010.

Takashi Shimokawabe, Takayuki Aoki, Chiashi Muroi, Junichi Ishida, Kohei Kawano, Toshio Endo, Akira Nukada, Naoya Maruyama, Satoshi Matsuoka, "An 80-Fold Speedup, 15.0 TFlops, Full GPU Acceleration of Non-Hydrostatic Weather Model ASUCA Production Code," ACM/IEEE Supercomputing (SC'10), pp. 1--11, New Orleans, USA, Nov 16, 2010.

Toan Nguyen, Hideyuki Jitsumoto, Naoya Maruyama, Tatsuo Nomura, Toshio Endo, Satoshi Matsuoka, "MPI-CUDA Applications Checkpointing," 並列/分散/協調処理に関するサマー・ワークショップ, 金沢, 2010年8月4日.

野村達雄, 丸山直也, 遠藤敏夫, 松岡聡, "GPU クラスタを対象にした並列ステンシル計算の自動コード生成フレームワーク," 並列/分散/協調処理に関するサマー・ワークショップ, 金沢, 2010年8月3日

Leonardo Bautista, Naoya Maruyama, Franck Cappello, Satoshi Matsuoka, "Distributed Diskless Checkpoint for Large Scale Systems," IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid'10), Melbourne, Australia, May 18, 2010.

Toshio Endo, Akira Nukada, Satoshi Matsuoka, and Naoya Maruyama, "Linpack Evaluation on a Supercomputer with Heterogeneous Accelerators," IEEE International Parallel & Distributed Processing Symposium (IPDPS 2010), Atlanta, USA, Apr 21, 2010.

Naoya Maruyama, Akira Nukada, and Satoshi Matsuoka, "A High-Performance Fault-Tolerant Software Framework for Memory on Commodity GPUs," IEEE International Parallel & Distributed Processing Symposium (IPDPS 2010), Atlanta, USA, Apr 20, 2010.

〔図書〕(計1件)

松岡聡, 青木尊之, 遠藤敏夫, 丸山直也, 佐藤仁, 滝澤真一朗, 實本英之, 「TSUBAMEの造り方から探るPCクラスターと『スパコン』のあいだ」, アスキー・メディアワークス, pp.48--55, 2010.

〔産業財産権〕

出願状況(計0件)
取得状況(計0件)

6. 研究組織

(1) 研究代表者

丸山直也 (MARUYAMA NAOYA)

東京工業大学・学術国際情報センター・助教
研究者番号: 60532801