「ポストペタスケール高性能計算に資する
システムソフトウェア技術の創出」

H23 年度
実績報告

平成22年度採択研究代表者

堀 敦史

(独)理化学研究所計算科学研究機構 研究員

メニーコア混在型並列計算機用基盤ソフトウェア

§1. 研究実施体制

(1)堀グループ

① 研究代表者:堀 敦史 (理化学研究所計算科学研究センター、研究員)

② 研究項目

・メニーコア用 OS カーネルの開発

・スケーラブル並列ファイルシステム

・超軽量マルチスレッド機構

(2)並木グループ

① 主たる共同研究者:並木 美太郎 (東京農工大学 大学院工学研究院、教授)

② 研究項目

・メニーコア用 OS における資源管理と仮想化方式

(3)辻田グループ

① 主たる共同研究者:辻田 祐一 (近畿大学工学部、准教授)

② 研究項目

・高スケーラブルな通信と I/O の実現

(4)Graham グループ

① 主たる共同研究者:Richard Graham (米 Oakridge National Laboratory, Group Leader)

② 研究項目

・耐故障性の研究

※ 今年度は契約までには至らなかった。

§2. 研究実施内容

(文中に番号がある場合は(3－1)に対応する)

本研究では、ポストペタスケールにおいて主流のひとつになると思われるメニーコアアーキテクチャをターゲットとし、その上で各種 HPC アプリケーションをスケーラブルに動作させるべく、システムソフトウェアを研究開発しようとするものである。このため研究範囲は広範であり、OS カーネル、メモリや通信の遅延を隠蔽するための軽量スレッド、低レベル通信機構と標準メッセージ通信ライブラリ MPI、ファイル IO などがその範疇となっている。今年度は、設計の基礎となるべく各種基礎パラメータの計測、アイデアやモデルの検証、ソフトウェアの試作などを主におこなった。

(1) OS カーネルの開発(堀グループ、並木グループ)

ここではマルチコアとメニーコアそれぞれの CPU が IO バス(PCI)によりつながっている構成を想定し、マルチコア側には通常の Linux を、メニーコア側には軽量カーネルを新たに開発し、ぞれぞれの特性を活かし、マルチコアとメニーコアのそれぞれの OS が協調するような方式を検討する。

(1-1)資源管理方式の検討

資源管理方式の検討では、メニーコアに軽量カーネルを実装することで、通常のカーネルの場合に比べ、マルチスレッドや IO のオーバヘッドを低減できることが実験で確認された。

(1-2)メニーコア抽象化ソフトウェアレイヤー

メニーコアは開発途上の技術であり、今後技術の進歩によりそのアーキテクチャは大幅に変化することが予想されている。しかしながら、OS の開発においてアーキテクチャの変化に追随することは容易ではない。このため、本研究の最初のステップとし、メニーコアアーキテクチャを抽象化するソフトウェアレイヤー、Accelerator Abstraction Layer (AAL) を開発した。これにより、今後予想されるアーキテクチャの変化に追随することが容易になるものと期待される。

(1-3)メニーコアに適したプロセス／スレッドのモデルの検討

メニーコア上では、その名の通り数十から 100 のオーダーのプロセスやスレッドが同時に走ることが可能である。本研究では、メニーコアアーキテクチャの特性を活かし、かつ並列処理におけるプロセス間通信あるいはスレッド間の排他制御のオーバヘッドを低減する新たなプロセスモデルを検討している。

本研究においては、並木グループがコンセプトの実証を、堀グループが並木グループの成果を受けて OS の開発を進める方式となっている。今後としては、まずはアプリケーションが動く程度のプロトタイプ開発を急ぎ、以降はプロトタイプを段階的に改良し、最終的に実使用に耐える OS の開発を目指す。

(2) 軽量スレッド(堀グループ)

ポストペタスケールにおいてはメモリや通信の相対的な遅延はより大きくなると予想されている。本研究では、特にメモリのアクセス遅延の隠蔽を目的に、ハードウェアが提供する Simultaneous Multi-Threading (SMT) 機能を活用し、増大するメモリ遅延を隠蔽可能とするマルチスレッドライブラリの研究開発が目的である。実験的な実装による検証では、pThread に比べ高速なスレッド生成と同期が実現できることを確認した。今後としては、メニーコアアーキテクチャ上での検証と、実アプリケーションを用いた評価をおこない、改良していくつもりである。

(3) ファイル IO(堀グループ、辻田グループ)

(3-1)スケーラブルな並列 IO

ポストペタスケールにおけるスケーラブルな並列 IO を目指した研究では、メタデータのボトルネックを回避する方式について 2 つの方式を検討し評価した。ひとつは、細粒度で沢山の数のファイルに対する IO を、ひとつの大きなファイルに集約することで、メタデータサーバへの負担を軽減する方式、もうひとつはハッシュを用いて複数のメタデータサーバに対し効率的に負荷分散をおこなう方式の検討である。今年度では双方のプロトタイプを実装し、スケーラビリティの確認をおこなった。

(3-2)MPI-IO の効率化

現在の並列ファイルアクセスでは MPI-IO が多く用いられている。しかしながら、現在の MPI-IO の実装には非効率な部分がある。本研究では、この非効率な部分を改良することで、MPI-IO の性能を向上させることが可能であることを実証した。

スケーラブルな並列 IO の研究は、今後プロトタイプの開発を進め実用的なファイルシステムの開発まで進める予定である。MPI-IO の改良も同時に進め、最終的には両者を統一される予定である。

(4) 通信機構(辻田グループ、堀グループ)

(4-1) MPI 実装方式の改良

上記 (2) の研究の副産物として生まれた効率的な同期方式アイデアを、MPI の同期方式に応用することで、低消費電力な MPI を実装できる可能性がある。今後はこの方式の検証を進めるつもりである。

(4-2) メニーコアにおける MPI 実現方式の検討

メニーコアアーキテクチャの各コアは、マルチコアのコアに比べ非力である。このため MPI の全ての処理をメニーコアでおこなうよりも、一部をマルチコア側で処理した場合があると想定される。これはマルチコアに比べコア当たりのメモリ量が少ないという観点からも、メニーコアとマルチコアで処理を分担すべきという根拠になると考える。本研究では、処理の分担方式を検討した。今後は、定量的な評価をベースに、メニーコアとマルチコアで処理を分担する MPI の実装を進める予定である。

(5) 耐故障レジリエンス基盤

残念ながら本年度中に共同研究先の米 Oak Ridge National Lab. と契約に至ることができなかった。来年度中に契約を結び、今年度の遅れを取り戻す予定である。

次年度以降においては、本年度の成果をベースに、プロトタイプの開発を進め、それを段階的に改良する事で全体を統合し、評価改良を経て、ポストペタスケールのソフトウェア基盤となるべくソフトウェアをオープンソースとして公開する予定である。

§3. 成果発表等

（3－1） 原著論文発表

（3－2） 知財出願

① 平成 23 年度特許出願件数（国内 0 件）

② CREST 研究期間累積件数（国内 0 件）