

課題名 (タイトル) :

高並列アプリケーションプログラムの研究開発

利用者氏名 : ○寺井 優晃

所属 : 計算科学研究機構 運用技術部門 ソフトウェア技術チーム

<p>1. 本課題の研究の背景、目的、関係するプロジェクトとの関係</p> <p>運用技術部門ソフトウェア技術チームでは、京の運用業務においてコンパイラ、プロファイラ、各種ツールの評価や改善を行っている。これら評価では、マイクロベンチマーク等のトイプログラムによる確認の他に、実際のアプリケーションを用いることもあり、その中で大規模並列実行が必要な場合も多い。本課題は、京とは異なるアーキテクチャでの、アプリケーションのコンパイル、単体性能および大規模並列実行の比較評価を行うことで、京の運用業務で発生した課題に資することを目的としている。</p> <p>2. 具体的な利用内容、計算方法</p> <p>チューニング作業において、アプリケーションの性能を最大限引き出す為には、単体実行時と高並列実行時の特性を両面から把握する必要がある。</p> <p>高並列時の評価では、演算量の偏り、経過時間に占める通信時間の割合及び通信の種類、メモリ量の増加等を定量的に評価することで大規模並列実行した際の特性を早期に明らかにし、問題があれば修正を行う。</p> <p>一方、プロセッサをターゲットとした単体性能評価では、マイクロベンチマーク、カーネル等を用いた性能プロファイリングを行う。プロファイラの出力情報に依存するが、京では、この評価によりコンパイラが生成した命令群と実際にプロセッサで消費されたサイクル数が得られ、ボトルネックが判明する。その結果に基づいて、改善のためのチューニングを実施する。その後、改善したか再度プロファイルを取得する。ただし、昨今のコンパイラやプロセッサは高度化しており、教科書にあるような単純なソースコードの書換えでは期待する結果にならないことがある。そのため、システムとチューニングの特性を理解する</p>	<p>ための基礎性能をデータとして蓄積することが必要となる。</p> <p>3. まとめ</p> <p>RICC は IA ベースの一般的なアーキテクチャを採用しており、SPARC64VIIIfx を採用している京と比較する上で都合が良かったため、これまで京の比較検証環境として利用している。</p> <p>昨年度までに、重点化アプリケーション等を用いた RICC での性能評価は京の高度化にフィードバックされた。アプリケーションの高並列・高性能化手法の中で一般化できるものはコンパイラに組み込まれ、今後の計算科学分野における経験や知見となって蓄積されていくことが期待される。</p> <p>その一方、本年度は京の重点化アプリケーションを用いた大規模並列実行時の性能評価はほぼ完了しており、業務としては基礎性能を取得するためのマイクロベンチマーク測定が大部分を占めたものの、より新しいアーキテクチャを採用したシステムでの性能評価が優先された。</p> <p>4. 利用がなかった場合の理由</p> <p>本年度は、京開発プロジェクトで重点課題とされたアプリケーション開発と評価が終了し、大規模並列実行の必要性がほぼなくなったこと、また、単体性能評価で重視されるプロセッサのアーキテクチャにおいて、より新しいものが他のシステムに搭載されるようになっており、RICC を用いた性能評価の優位性が低下したことがある。</p>
---	---