

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 : 長谷川 幸弘

所属 : 本所 次世代スーパーコンピュータ開発実施本部
開発グループ/アプリケーション開発チーム

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

現在開発中の次世代スーパーコンピュータに向けて、様々な重要アプリケーションの開発を行っている。次世代スーパーコンピュータ向けのアプリケーション開発においては、数万のオーダーの並列数でのスケーラビリティの確保と高性能 CPU の持つ性能を極限まで引き出すチューニングが必須となる。数千並列で実行可能な RICC をテスト環境として利用し、各種チューニングを試行・検証する。

2. 具体的な利用内容、計算方法

実時間 DFT コードの MPI プロセスにおける通信性能向上を目的として、アルゴリズムの改良を行った。

3. 結果

実時間 DFT コードでは空間軸とバンド軸の2つの並列軸を使って高並列を実現しているが、主ルーチンとなる正規直交化ルーチンではバンド軸方向の並列では同期が必要なアルゴリズムであるため、同期待ちによる性能劣化が大きい。今回は同期待ちによる性能劣化を改善するために、従来は他のプロセッサの処理待ちとなる部分に本来後で行われる処理をはめ込むことで処理をオーバーラップさせる方法を実装した(図1参照)。これにより、シリコン 4096 原子モデルのデータでバンド軸の並列数4の場合は8%、並列数8の場合は12%の性能改善が見られた。バンド並列数が大きくなるほどに改善効果が大きくなることが確認できた。

4. まとめ

高並列において問題になる通信を洗い出し、これを改善するアルゴリズムを実装した。

5. 今後の計画・展望

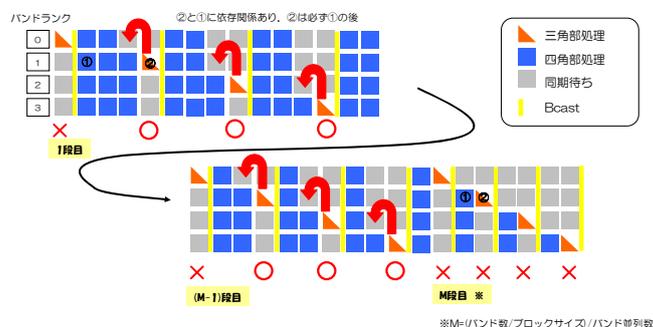
今年度に引き続き、高並列アプリケーションプロ

グラムの性能評価および性能改善を実施する。

6. 利用した状況、継続して利用する際に行う具体的な内容

これまで、次世代スーパーコンピュータをターゲットに高並列アプリケーションプログラムの性能評価、特に通信性能の評価を行ってきた。高並列になってくると通信性能の増加が予想されるため、RICC を利用した高並列での通信性能の評価は非常に有用である。今後も継続して、高並列環境での通信性能の評価を実施するために利用する。

例) 4バンド並列の場合



○の▲部のみ■部とオーバーラップ可能、×の▲部はオーバーラップできない。
上記の例の場合、ランク0の処理、M段目の処理、
6/12の▲部がオーバーラップ可能。

図1. オーバーラップアルゴリズムのイメージ