

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 コードのウィークスケールでのスケーラビリティを測定した。また、MPI とコンパイラ自動並列によるスレッドを組み合わせるハイブリッド並列を試行した。

3. 結果

図 1 に並列数を変化させた時の共役勾配法ルーチンの実行時間の変化を示す。"S:隣接"は空間方向の隣接通信, "S:大域"は空間方向の大域通信, "B:大域"はバンド方向の大域通信である。演算部の性能はプロセス数が増加した時でもほぼ一定であることから、演算部には非並列部がないことが確認できた。一方、通信部についてはプロセス数の増加に伴い隣接通信, 大域通信どちらも増加傾向であることがわかる。図 2 に 512 コアの Flat MPI と 128MPI×4 スレッド並列での測定結果を示す。スレッド並列と組み合わせた場合、通信時間の削減効果が大きいこと、特にグローバル通信の削減効果が大きいことがわかる。

4. まとめ

プログラムのスケーラビリティを調査し、RICC における挙動を確認した。これにより高並列において問題になる通信を洗い出すことができた。また、MPI+スレッドのハイブリッド並列が有効で

あることを確認した。

5. 今後の計画・展望

MPI プロセスの高並列対応チューニングおよび CPU 単体性能の向上を目指す。

6. これまで利用した状況、継続して利用する際に行う具体的な内容

今年度は高並列で問題となる MPI 通信の情報を取得した。また、スレッド並列の有効性を自動並列で確認した。今後は大域通信のコストを低減するチューニングや OpenMP によるスレッド並列を実装する。また、単体性能の向上を目指してキャッシュチューニングを実施する。

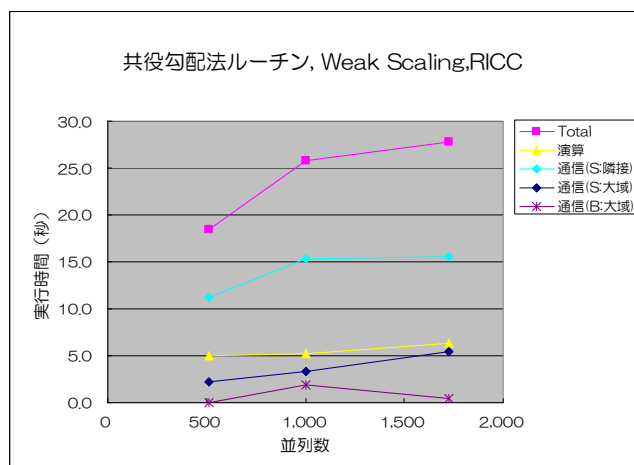


図 1 共役勾配法ルーチンのスケーラビリティ

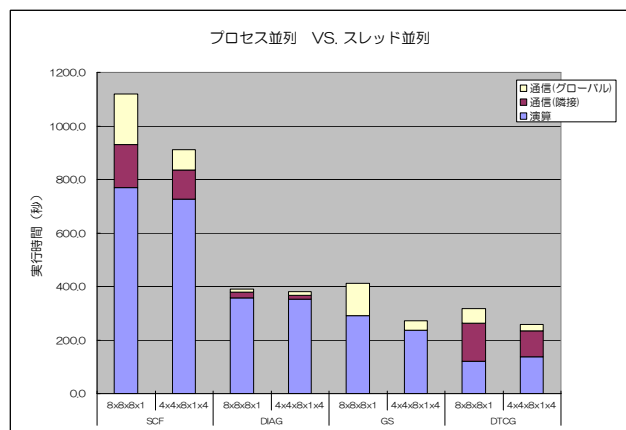


図 2 自動並列によるスレッド並列の効果

