

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 : 黒田 明義

理研での所属研究室名 :

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

報告内容

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, 次世代スーパーコンピュータ開発実施本部では, ナノ分野のシミュレーション科学を担うターゲットアプリケーションである PHASE, CREST との連携開発を行っている固有値ソルバ, またグランドチャレンジ実証研究として扱われるアプリケーション群について, 京速コンピュータ「京」(以下「京」と略す)に向けた高並列化・高性能化の作業を行っている. 本年度は試作結果に対する評価を中心に実施しており, 本課題では RICC システムを用いて, 特に並列性能に着目して「京」での可用性について評価を実施した.

2. 具体的な利用内容, 計算方法

本プロジェクト開発実施本部では, 重点的に高速化に取り込むとしたアプリケーション選定した 6 本のうち, ナノ分野での計算科学において重要とされるアプリケーション PHASE を, 「京」での高速化に向けた試作並びに評価を行った. PHASE とは, 擬ポテンシャルと密度汎関数法によるナノ材料第一原理分子動力学プログラムであり, 局在基底ではなく平面波基底を用いることにより, 分子から固体まで多くの物質に対して高精度な電子状態計算が可能となるアプリケーションである.

PHASE の主要コストを分析すると, 主に行列-行列積の形に書き換え可能な区間, FFT を含む区間, 対角化を含む区間の 3 種類に分類が可能である. 単体性能の面から性能阻害要因を見積もると, 行列-ベクトル積で記述されているため, 性能がメモリバンド幅律速のものが挙げられ, 並列性能の面からは, 並列粒度が細くなることによる性能低下, FFT の並列性能, 対角化の並列性能の 3 点が考えられる. 以上の 4 つの観点に着目

して, それぞれ性能最適化手法を検討し, 試作, 評価を実施した.

一般に, 密度汎関数法は, ハミルトニアン行列に対する固有値問題であるため, 系の規模 N に比べて演算量が大きく $O(N^3)$ である. これは, 計算に必要なデータ量に比べて演算量が多く, 計算の工夫により, メモリ帯域を圧迫しない実装が可能である. 単体性能向上策として, 演算密度が高いカーネルの演算効率を上げる工夫として, BLAS Level3 の適用について説明する. 行列-行列積の形で記述されたカーネルでは, 行列積計算の要素数 $2N^2$ 個のデータを用いて, $2N^2$ 回の演算を行う. 1 回の参照で多数演算を行うため, メモリへ何度もデータを読み込む必要がなく, 演算密度が飛躍的に向上する可能性がある.

Gram-Schmidt 直交化では, 新しい直交ベクトルを得るために, オリジナルベクトルと, 直交ベクトル束を用いるため, それまでに生成した全直交ベクトル束が必要となり, 依存関係のため直交ベクトル束を一度に計算することが出来ない. しかしある程度の世代まで直交ベクトルが生成できていれば, 依存関係のある部分とない部分を分離することが可能である.

多軸並列化については, 一般に密度汎関数法では, 演算量が系の規模 N に比べて $O(N^3)$ であり, SCF による反復計算も加わるため, 演算量が非常に多い手法である. このため超並列システムでは, メモリ量や演算時間の制限から, N の大きな計算が困難となる. N を大きく出来ない, 並列粒度が細くなるなどの問題が新たに生じ, 極端なケースでは N が並列数を下回り, それ以上の分割が出来ない事態も生じ得る. しかし, コーディングの観点から見ると, プログラム中のループ構造は N のループだけではなく, 他の変数のループを含む多重ループ構造となっている. これは N 以外のループでの並列化も可能であることを意味し, これを多軸並列化と呼ぶ.

PHASE での並列方法はカーネルごとに異なるが、多くはエネルギーバンドのループを用いて並列化(1 軸並列)されている。しかし、各エネルギーバンドは、波動関数のフーリエ級数展開により生成した波数方向の自由度を持ち、その方向に並列軸を拡大(2 軸並列)することで、並列効率の向上が見込まれる。

バンド並列と波数並列が混在していると、並列軸の入れ替えに伴いデータの持ち替えが必要となり、全体通信が発生する。しかし 2 軸並列化により、これらの軸の入れ替えに伴う全体通信を局所化することが可能となり、通信の削減効果も見込まれる。

Gram-Schmidt 直交化の 2 軸並列化については、その他の区間と異なり、並列数の制約から波数方向に並列化を行っていたが、超並列時には、バンド方向に大量のデータ縮約通信が多数回発生する。2 軸並列化を行うことで、バンド方向へ並列軸の拡大により、この縮約通信の並列効果が期待できる。しかし、BLAS Level3 適用に伴う、バンド方向のブロック化と干渉し、ブロックサイズを大きくするのが難しくなることによる性能への影響や、依存関係のある箇所により引き起こされるロードインバランスなどが問題となる。本手法では、計算順序を工夫することで、バンド方向の分割による性能への影響を波数方向の分割でカバーし、高い並列性能を実現している。

次に FFT を含む箇所の並列化について。本アプリケーションはフーリエ級数を基底関数として用いているためフーリエ変換を多用する。フーリエ変換は高速フーリエ変換(FFT)を採用することで、超並列時の実行時間は行列-行列積の箇所に比べて負荷割合は増大しにくい。しかし並列 FFT の転置計算の際に全体通信が発生し、これが並列数と共に増大する傾向にある。ここで、2 軸並列を考慮すると、並列 FFT は全ノード間全体通信の必要がなく、ある並列軸方向の通信を行うことで十分であることが分かる。これは通信相手が減るだけでなく、通信マッピングの最適化により経路競合も防げるため、1 軸並列を用いた並列手法に比べて大幅に通信時間を削減が可能となる。「京」全体で FFT を行うと、システム全体で 80,000 程度のノード間の全体通信が発生するが、2 軸並列により、300 程度のノード間の全体通信に縮小が可能となる意義は大きい。またこれらの FFT はバッファリングによりある程度まとめて処理することも可能であり、メモリ使用量に余裕があ

る場合、通信回数を削減することも可能である。

最後に対角化に関する並列化について、波動関数の収束計算にて、精度を保ちつつ高速化するために、直交化や対角化が使用され、対角化には Householder 法など精度を要求するアルゴリズムが用いられている。直交化は BLAS level3 適用で高速化できたが、これらの対角化手法は Householder 変換として行列ベクトル積の部分が残存し、加えて並列効率が低いという問題が残る。一般に、対角化の対象となる行列の規模は、原子数の数倍程度と比較的小規模なものである。アプリケーションを実行する際に用いられる実問題では、全空間対角化を必要とはしておらず、部分空間での対角化を行っているため、小規模環境での計算では非並列の LAPACK を用いてきた。しかし「京」上での計算では問題規模の増大を想定し、ScaLAPACK を導入して、対角化箇所の並列化を行った。しかし ScaLAPACK は、並列性能に限界があることが知られており、より並列性能の良い対角化手法が必要とされる。「京」では、高並列対角化ライブラリの開発を共同で進めており、現在そのライブラリの導入も試みている。

3. 結果

Gram-Schmidt 直交化における BLAS Level3 適用結果により、ブロック化済みの箇所は、演算効率 40~70%となっている。それに対して、適用以前の箇所は 4%~5%の演算効率である。Gram-Schmidt 直交化は、もともと依存関係のある計算のため一部消去できないオリジナルプログラムの依存関係が行列-ベクトル積の形で残っており、その箇所の効率が 4%~5%程度であるということは、ブロック化する前の効率も同等の 4%~5%とみなせるので、BLAS Level3適用箇所は 8倍~15倍程度性能が向上しているのが分かる。全体で見ても効率が 4~5%から 28%へと 5倍以上性能が向上したことになる。

2 軸並列化の効果について、非局所ポテンシャル計算カーネルの 2 軸並列化の効果について、1 軸並列時は、100 並列で既に並列効率が悪化しており、経過時間の下げ止まりが見られるが、2 軸並列版では、分割軸が増えたことに伴う新規通信が多少発生しているが、128 並列でも良好な並列効率を保っている。

Vlocal の逆 FFT に関する並列性能について、比較

的小規模での FFT 並列特性の評価結果について、2 軸並列化に伴い 16 並列の低並列時に性能が 24[s]→46[s]と劣化しているものの、128 並列では 11[s]→9[s]と並列性能は比較的良好である。低並列時の性能劣化は、2 軸並列化によりオリジナルの分割方法であるバンド方向の分割数が減少したことによる影響と考えられる。高並列時の並列性能の改善は、2 軸並列化に伴い並列粒度が増大したこと。更に 2 軸並列により全ノード間の全体通信が必要なくなったことで説明が可能である。

ScaLAPACK を用いた対角化を含む区間の実行時間である。BLAS は並列性能が良いことが分かるが、対角化部分は既に並列性能が限界に達しており、並列数が増大するに従って性能が悪化している。これは、測定した原子数 1,500 の系では行列サイズ 5,000 程度と小さいことが影響している。この行列サイズ規模の ScaLAPACK の並列特性として 1,000 並列程度の並列性能限界が知られている。しかし 1,000 並列程度まで並列性能が得られれば、アプリケーション全体に占める割合は 30%程度までには抑えられ、影響は比較的小さいと予測できる。

4. まとめ

「京」2,048 並列で実測した実行時間と実行効率は、BLAS 化した箇所は、40%以上の実行効率を達成し、SCF ループ全体を通じて 20%を越える実行効率を達成できた。

同じ系の各並列数でのカーネル種別毎の実行時間の内訳を見ると、HfSiO₂ 1,536 原子の場合 2,000 並列程度まで並列性が保たれるのを確認した。この計算では、既に原子数が並列数を下回っており、今までの並列手法では分割粒度が細かく計算不可能な問題である。仮にオリジナルプログラムを用いても計算できたとして最適化前後の性能を比較すると、BLAS を導入されていない箇所の効率は 5%程度、FFT を含む箇所は、性能最適化前後で同程度の 5%以下の効率が想定されるので、全体通信を含む本アプリケーションにて、実行効率で 20%を越えるのは画期的と言える。

5. 今後の計画・展望

密度汎関数法の場合、演算密度が高いため、中心となるカーネルの大部分は演算順序などの工夫により BLAS Level3 の適用が可能となり、適用箇所の実行効率が数十%程度まで達成可能である。

また昨今のシステムの超並列化に伴い、FFT などの全体通信を含む通信アルゴリズムの性能低下が危惧されていたが、並列軸の追加により、通信をグループ化でき、超並列にもある程度対応可能であることが示唆された。FFT に関しては、今後「京」では、多次元並列版 FFT が整備されるなど、更なる高速化が期待される。更に現在 FFT を行うためにパッキングなどの前処理を行っているが、この箇所のデータを連続化することで、更に並列効率の向上が見込まれる。

対角化については、汎用ライブラリでは、それほど並列性能が見込まれないが、必要とする計算が全空間での対角化ではなく、部分空間という比較的小規模な対角化であるため、1,000 並列程度とある程度並列性能が保証されていれば、全体の計算時間への影響は低く抑えられると考えている。今後、高並列版対角化ライブラリを導入することで、より高並列にも対応可能である。

本アプリケーションは「京」共用開始時に、SCF 収束回数を考慮しても、理論上は 100,000 原子程度の電子状態計算が日単位で達成できる見込みである。しかし様々なデバイス特性を調べるために、分子動力学計算や反応中間体の計算などを実施する場合、収束時間を短くしなければならないという要請から、10,000 原子程度の問題規模を全 80,000 プロセスを用いて計算する必要がある。この問題規模は、今までの手法では並列制約から実現不可能とされたが、今回の性能最適化により、初めてその目標達成の目処がたったといえる。

6. RICC の継続利用を希望の場合は、これまで利用した状況や、継続して利用する際に行う具体的な内容

本年度、PHASE の「京」向けの評価に道筋をつけた。本年度は、実際のサイエンス問題を適用し、更なる性能向上を目指す。

7. 一般利用で CPU 時間を使い切れなかった理由

なし.

8. 利用研究成果が無かった場合の理由

本年度は次世代スーパーコンピュータにて性能を高めるためのアプリケーションの試作並びに、性能調査並びに評価を目的に利用を行った。

次年度以降、本年度の試作評価結果をもとにしたアプリケーション開発・高速化の実装へのフェーズへとシフトしていくため、そこでの成果を発表していく機会が出てくる予定である。

また、今後は、新規アルゴリズムを用いた新規アプリケーションの開発にも手がけるフェーズへ展開していく可能性もあり、今後 RICC を用いた比較成果を発表していく機会が増えていくと思われる。