

4P124 SH4 マルチプロセッサシステムに対応した 分子軌道計算プログラムの開発と並列性能評価

(CREST JST¹, 産総研², 富士総研³, 九大院シ情⁴)

梅田宏明^{1,2}, 稲富雄一^{1,2}, 本田宏明^{3,4}, 長嶋雲兵^{1,2}

序

機能性材料の設計や医薬品開発のような分子レベルのシミュレーションから、車や飛行機などの設計、天気予報や彗星の軌道の予測など、幅広い分野において科学技術計算が利用されている。しかしながら大規模科学技術計算に要する計算機資源は非常に大きく、大学の研究室などで容易に利用できる安価な PC クラスタシステムでは実現が大変困難であった。

研究室内外の既存の多くの計算機を有効に活用することのできるグリッド技術はこの問題に対する一つの解決策である。グリッド技術を用いることにより、セキュリティポリシーの異なる組織間で計算機を共有することができる。また PC クラスタで利用される MPI では得られないレベルの耐障害性の高いアプリケーションを作成することもできるため、長時間に及ぶ計算も実行することも可能となると考えられる。

計算用途を限定した専用計算機の利用も、大規模科学技術計算を実現するための解決策の一つである。ある種の科学技術計算においては特定の方程式を解くためだけに非常に大きな計算量を要しており、この方程式を高速に計算することに特化した専用のアーキテクチャを持つ計算機システムが作成できれば、一般の PC よりもはるかに性能の高い計算機を比較的安価に利用することが可能になる。このタイプの専用計算機の代表としては GRAPE や MD-Engine などがあげられる。

計算化学の分野においては、例えば生体分子のような大規模系の分子軌道計算ではフォック行列の生成に全計算時間の 9 割以上の時間を費やすことが知られている。専用計算機によってフォック行列を高速に生成することができれば、比較的手軽に大規模分子軌道計算が実行できる環境が実現できると思われる。

EHPC (Embedded High Performance Computing) Project では、フォック行列生成において中心となる二電子積分計算専用の専用ロジック LSI の開発を進めており、これを多数搭載できるプラットフォームアーキテクチャ上へのアプリケーションの移植が必要とされていた。特に EHPC プロジェクトでは、専用 LSI 単体の速度をある程度犠牲にして電源消費量を削減し、搭載可能な LSI の数を増加させることによって全体の性能を出すアーキテクチャを採用しているため、プラットフォームアーキテクチャ上でのアプリケーションの並列性能を上げることが本質的な意味を



図 1 プラットフォームアーキテクチャ
評価システム

持っている。本研究では現在開発中の専用 LSI の代わりに汎用の CPU を多数搭載したプラットフォームアーキテクチャ評価システム(図 1)上にフォック行列の生成ルーチンを移植し、その性能を評価した。

プラットフォームアーキテクチャ評価システム

プラットフォームアーキテクチャ評価システムでは専用 LSI の代用として汎用 CPU(Hitachi SH4, 200MHz)を搭載したシステムである。一つの筐体は、SH4 が 4 個搭載したボード(図 1 左)が 7 枚と Pentium3 CPU を搭載した PC からなっており、これが 3 台 LAN で接続されている。ボード上にある 4 個の SH4 のうちの 1 つは制御用 CPU であり、二電子積分計算は残りの 3 個の SH4 で実行される。PC 制御用 SH4 計算用 SH4 の通信は階層構造(図 2)をとっており、T-COM (Tree-Communication) API を用いて行なわれる。具体的には計算用 SH4 同士の直接通信はできず、そのボード上の制御用 SH4 に対してのみ通信が可能である。制御用 SH4 もそれ同士の通信はできず、その筐体の PC 又は自らの管理する計算用 SH4 とのみ通信が可能となっている。T-COM API では、これらの通信は同期的な send/recv および wait_any によって実現される。PC 間は通常のネットワークであるため MPI 等を用いた自由な通信が可能である。

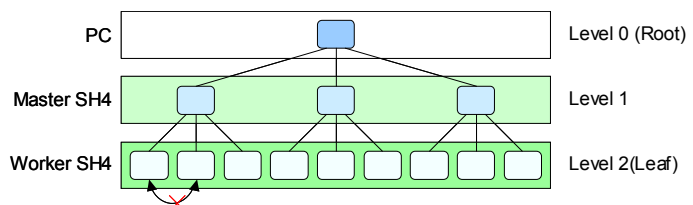


図 2 T-COM API の階層構造

性能評価

ターゲットとした分子軌道計算アプリケーションは GAMESS であり、その RHF 計算についてのフォック行列生成部分を今回作成したものと置き換えて使用した。二電子積分は専用ロジック LSI にあわせて小原法を用いて計算している。並列化は計算用 SH4 に対する縮約シェルセットの振り分け法に応じて静的負荷分散(IJ)、静的負荷分散(IJKL)および動的負荷分散の 3 つのアルゴリズムを採用して、実装を行なった。

対象とした計算はグリシンの 5 量体の RHF/STO-3G 計算である。これは最近の PC で実行した場合には 5 分程度で終了してしまうような比較的小さな計算であり、積分計算を省略することによるジョブ粒度のばらつきが比較的強調されやすい。図 3 に実際にプラットフォームアーキテクチャ評価システム上で測定した並列化による速度向上比を示した。静的負荷分散(IJ)では粒度のばらつきの影響で性能の向上が抑えられてしまっている。一方で動的負荷分散においては 63CPU の場合でも非常に良い性能を示しており、専用ロジック LSI を用いた場合に高い性能を発揮することが予想できる。

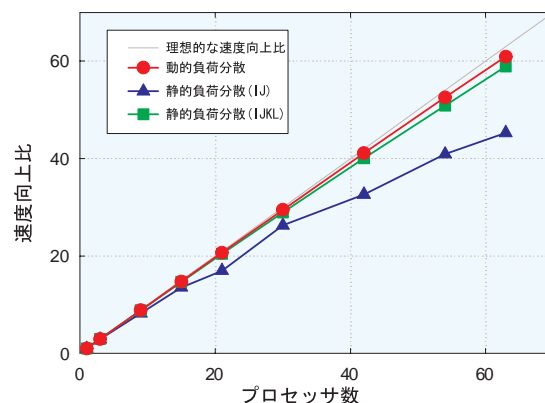


図 3 並列性能(速度向上比)