

GRAPE 型専用計算機での Direct SCF 計算アルゴリズム

(名大院情報^a 理研^b 東大院理^c) ○安田耕二^a 山木大輔^a 松原裕樹^b 戎崎俊一^b 牧野淳一郎^c

【目的】我々は、Gaussian を現在の 100 倍以上の速さで実行できる、安価な専用計算機を、2 年以内に開発する事を目指している。この専用計算機は、PCI バスに挿す基盤として供給され、これを研究室にある普通のパソコンに挿すだけで、実効性能が数テラ FLOPS のスーパーコンピューターに変身する[1]。実験化学者も手軽に使えるシステムとし、計算化学の能力を飛躍的に高める事を目指している。

【背景】ab initio 法の発展に加え、計算機の高速化により、生体分子のような大分子まで、全電子計算が行われるようになった。ここ 20 年の計算機の高速化は、Reduced Instruction Set Computer (RISC)アーキテクチャ、多段パイプラインを採用し、CPU 内部クロックを高速化できた事が主な要因である。しかし現在では CPU 上のかんりのトランジスタが、高速メモリー（キャッシュ）や制御機構に使われている。なぜなら (i)パイプライン段数が増加したため、より正確な分岐予測が必要となり (ii)メモリーの読み書き速度は CPU よりずっと遅いため、高速な CPU にデータを供給する大容量のキャッシュが必要となったためである[2]。高い浮動小数処理能力が必要な科学技術計算にとっては、計算をしないキャッシュ等が多くの特ランジスタを占有するのは望ましくない。また近年では CPU 発熱量の増大により、内部クロック高速化は困難になり、複数の CPU を単一チップに集積して高速化を図っている。

【GRAPE DR】浮動小数の加算乗算器を備えた単純なCPUは、 10^5 個程度のトランジスタで作れるので、キャッシュや複雑な制御機構を省略すれば、1 チップ上に 1000 個程度のCPU (Processor element, PE) を集積し、1GHz程度の周波数で動作させる事は可能である。GRAPE DRプロジェクト[3]では、多数の演算器を集積する事で、科学技術計算に特化した専用計算機を作る事、それらをネットワークで結合し、ペタFLOPSコンピューターを作る事を目的としている。全てのPEに同じ命令を実行させる、いわゆる単一命令流、複数データ流 (SIMD)型アーキテクチャとするが、一般には通信速度がボトルネックになり得る。他方我々の興味ある量子系、古典系の問題では、対ポテンシャルの計算が殆どの時間を占める。この種の問題では細粒度の並列性があり、適切なアルゴリズムでは通信量を劇的に減られるため、GRAPE DRで高速に計算できる。例えばi番目のPEにi番目の粒子を担当させ、全PEにj番目の粒子位置を同時に放送する事で、通信量を抑えて粒子間力を並列計算できる。

GRAPE DR のプログラム開発は、普通の processor 向けのプログラム開発/移植とは次の点が異なる。

- (1) GRAPE DR はその設計思想から、通信速度を必要最小限に抑えており、また各 PE はレジスタ以外の記憶域を持たない。従って送るべきデータ量を減らし、計算の中間結果がレジスタ上に載るよう、アルゴリズムを再考する必要がある。
- (2) コンパイラーが開発中なのでプログラムは最終的に手でアセンブラか機械語に直す必要がある。また除算、開平、指数関数などの数学関数も現段階で開発されていないため、プログラムする必要がある。この点は今後改善される予定である。
- (3) GRAPE DR チップへのコード/データ供給は、C の API 関数を通じて行う。Gaussian 等の主プログラムはこの関数を呼び、GRAPE DR での計算を行い、結果を回収する。

【SCF アルゴリズム】当面の目標を、Gaussian03 で、6-31G**基底を用いて、密度汎関数法による 1 点エネルギー計算、構造最適化ができる事とする。この場合 Fock 行列中の Coulomb、交換、相関項の計算と、Fock 行列対角化が、計算時間の殆どを占める。以下それぞれの計算法を記す。

Coulomb 項 Tree code[4]や Fast multipole method (FMM)[5]を使えば、系の基底数 N に比例する計算量 $O(N \log N)$ や $O(N)$ にできる。Tree code は複雑な分岐を含み、SIMD processor 上での動作が難しい事、FMM が Gaussian03 に既に実装されている事から、FMM を GRAPE DR 加速ボードで行う事とした。

まず近接場積分つまり近接 Gauss 関数間の通常の 2 電子積分計算法を述べる。各 PE には別々の shell pair P を担当させ、全 PE に shell pair Q と密度行列要素を同時に放送する。通常の 2 電子積分ではなく、Challacombe らによるトリック、Fast J 法[4]を用いた。この方法では、密度行列の基底を Hermite Gaussian に変換する事で、Coulomb 項を高速で計算する。また必要な 2 電子積分も Hermite Gaussian を基底とした単純なものになる。この Fast J 法は Gaussian03 に既に実装されている。

2 電子積分の代表的な計算法には、McMarchie-Davidson (MD), Obara-Saika法がある[6]。d型Gauss関数の作る 2 電子積分(dd|dd)計算を例にGRAPE DRへの実装法を考えると、これらの方法では必要な中間結果を各PEのレジスタ上に保存できない点が問題になる。そこで記憶域が少なく済むRys法を実装した。

この方法ではn次のRys多項式の零点 r_k と重み w_k を計算する必要がある。これら 2n個の量はshellの 4 つ組で値が変わる $\alpha' = \alpha R^2$ の関数で、普通 α' の区間を 10 通り程度に細分して、多項式近似で求める。SIMD processorでは各PEの α' は違う値なので、異なる多項式係数を使う必要があるが、これはレジスタの間接参照で実現できる。多項式係数はPEのレジスタに保存できないので、必要時に全係数をPEに放送する必要がある。通信量を最小にするため、区間の分割数や多項式次数を最小とする、零点と重みの補間表を作った。積分値は次式で与えられる。

$$(\text{積分値}) = (pq)^{-3/2} R^{-t+\dots-v-1} \sum_k w_k r_k^{t+\dots+v} H_{t+t'}(n_x r_k) H_{u+u'}(n_y r_k) H_{v+v'}(n_z r_k) \times (\text{規格化}) \times (\text{MD係数})$$

積分の計算精度を与えると、零点や重みに必要な精度が決まるが、それらは次の理由から異なる。

(1) 基底関数の Gauss 指数の違い。HCNO 等の典型元素の 6-31G**基底では、d 関数の指数はほぼ一定だが、s, p 関数ではより広範囲にわたる。5 次の Rys 多項式は(dd)積分のみに使われるが、1 次のそれは全ての積分計算に必要とされる。つまり低次では高精度が必要である。

(2) 被積分式の差。積分値には零点が多項式として含まれ、その最高次数は shell 4 つ組の各運動量の和から決まる。つまり大きな n や大きな零点に対応する重みには、高精度が必要である。

また零点に必要な精度は、上式の零点に関する 1 次変分量から分かる。そこで我々は、計 $5^3=125$ 個の原子、H, C, N, O, Na, P, S, Cl をほぼ正方格子 (核間距離 0.8~1.7 Å の乱数) に並べ、6-31G**基底を使い、零点と重みにかかる係数を α' の関数として調べた。これから積分精度 10^{-10} に必要な、零点と重みの精度を α' の各区間で決めた。一例を図に示す。

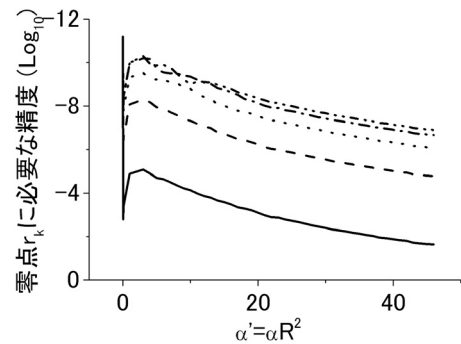


図: 2電子積分精度 10^{-10} を達成するのに必要な、5次のRys多項式の零点の精度

次にこの精度で零点と重みを再現する、できるだけ小さな補間表を作った。1, 2 次の零点と重みは誤差関数から簡単に計算できる。3 次以上の零点と重みの α' 依存性を知るために、複素 α' に対してそれらの値を調べると、虚軸方向には振動的な振る舞いが見られた。これは虚軸付近でもはっきり見られたので、零点と重みを新しい変数 $\alpha'' = \exp(-\alpha'/m)$ の関数として近似する事にした (m は 4~8 程度の整数)。区間を 4 分割し、漸近部分を除き、4~10 次のチェビシェフ補間により、補間係数を決めた。なお $\alpha'=0$ は高精度が必要なので、この点を補間選点にとった。

除算、開平等はニュートン法、指数関数は STL 法で計算した。以上の結果をまとめ、近接場の Coulomb 項を計算するアセンブラコードを書いた。また shell-pair や密度行列を送り、計算した Coulomb 項を回収する C 関数を書き、Gaussian03 とリンクした。現在残りの積分の計算法を検討している。

交換、相関項 純粋な密度汎関数法(DFT)を用いると、これらの項は (i)3 次元グリッド上で電子密度、密度勾配を求め (ii)交換相関汎関数、基底関数の値を計算し (iii)重みを掛けて求和する事で得られる。普通使われる交換相関汎関数は、初等関数の組み合わせからなり、細粒度の並列化は容易で、GRAPE DR 上への実装も問題なく行える。Hartree-Fock 交換項を一部使う hybrid functional では、2 電子積分から交換項を求める必要がある。GRAPE DR 上での実現方法として、Rys 法で Hermite Gaussian 間の 2 電子積分を計算し、McMarchie-Davidson 変換係数と縮約する方法を検討している。

Fock 行列対角化 対角化を $O(N)$ で行う標準的な方法は、McWeeny purified density matrix を使ったエネルギー期待値を、共役勾配法等で最小化する方法[7]である。この方法は Gaussian03 に既に実装されている事から、これを GRAPE DR 上で行う事を検討した。ここで最も時間が必要なのは勾配の計算や $N*N$ 行列の積の計算である。これを計算するアセンブラコード、インターフェースの C 関数を書き Gaussian03 とリンクした。

専用計算機は 2 年後に完成予定である。そこで発表当日は Gaussian03 をチップシミュレータ上で実行した結果を示す。

- [1] 地球シミュレータの理論ピーク性能は 40 テラ FLOPS である。
- [2] ヘネシー, パターソン「コンピュータ・アーキテクチャ」(日経 BP 社)
- [3] GRAPE DR プロジェクトホームページ <http://grape-dr.adm.s.u-tokyo.ac.jp/>
- [4] M. Challacombe, E. Schwegler, J. Chem. Phys. 106, 5526 (1997)
- [5] C. A. White, H. Head-Gordon, J. Chem. Phys. 101, 6593 (1994)
- [6] T. A. Helgaker, P. Jorgensen, J. Olsen, Molecular Electronic-structure Theory (Wiley, England, 2000)
- [7] X. P. Li, R. W. Nunes, D. Vanderbilt, Phys. Rev. B 47, 10891 (1993)