

FMO-MO 法による大規模分子軌道計算

稲 富 雄^{†,††} 梅 田 宏 明^{†,††} 渡 邊 寿 雄^{†,††}
櫻 井 鉄 也^{††,†††} 長 嶋 雲 兵^{†,†}

本論文では、フラグメント分子軌道 (FMO) 法に基づいた分子軌道 (FMO-MO) 計算についての概要と、実際のタンパク分子への適用結果を示す。FMO-MO 法では、巨大な分子全体に対する反復解法を行わずにすむため、従来の分子軌道法では非常に困難であった 1,000 原子を超える大規模分子の分子軌道計算を、比較的短時間で行うことができる。大規模クラスター計算機を用いた場合には、卵白リゾチーム分子に対する FMO-MO 計算を、4 時間半程度で行うことができた。また、小さなサイズの分子軌道計算では見えなかった一般化固有値問題を解く時間が、全計算時間の 34% となり比較的大きなウェイトを占めることが明らかとなった。

Large-scale Molecular Orbital Calculation by FMO-MO Method

YUICHI INADOMI,^{†,††} HIROAKI UMEDA,^{†,††} TOSHIO WATANABE,^{†,††}
TETSUYA SAKURAI^{††,†††} and UMPEI NAGASHIMA^{†,†}

We describe brief explanation of molecular orbitals based on fragment molecular orbital method (FMO-MO), and demonstrate its efficiency showing the result of the example FMO-MO calculation for a protein. Using FMO-MO method, molecular orbital calculations for molecules containing over 1,000 atoms can be achieved in a short time because large-scale iterative procedure for entire molecule is not required in FMO-MO calculation. FMO-MO calculation for Lysozyme molecule has been finished in almost 4.5 hours using large-scale PC cluster. The generalized eigenvalue problem is 34% of total wall clock time in our calculation and needs significant computational time for large-scale molecular orbital calculation with FMO-MO.

1. はじめに

量子力学に基づいた理論計算の一種である分子軌道 (Molecular Orbital, MO) 法¹⁾ では分子中の電子がどのように分布しているかを表す多くの分子軌道が得られる。そのうちフロンティア軌道²⁾ と呼ばれる 2 つの MO と、そのエネルギー (MO エネルギー) の解析が、さまざまな化学反応のメカニズムを理論的に解明する際に重要となる。近年、DNA の塩基配列や、多くのタンパクの立体構造が明らかになってきたが、タンパクなどの巨大生体分子の反応機構の解明は、その

分子の大きさゆえに、反応する部位を特定することすらままならないことが多い。したがって、DNA やタンパクなどの巨大分子のフロンティア軌道を求めることは、反応部位、および反応機構の理論的な解明に非常に重要であり、創薬などの分野で有用であると考えられる。

従来の MO 法による計算では、分子のエネルギーや電子密度分布とともに、MO や MO エネルギーも得られる。しかし、計算量が分子サイズの 4 乗に比例し、かつ大規模な繰返し計算が必要であることなどから、1,000 原子を超えるような大規模分子に対して従来の MO 法を適用すると、少し大きめの PC クラスターなどを用いた場合でも、10 日～数週間といった長い時間が必要になる。いろいろな有機化合物から有効な薬品になる物質の候補を絞るスクリーニングなどに MO 法を利用しようと考えた場合などでは、この計算時間が大きな障害となる。Kitaura らによって提案されたフラグメント分子軌道 (FMO) 法^{3)~5)} は、このような従来の MO 法の欠点を克服するための計算

† 産業技術総合研究所グリッド研究センター
Grid Technology Research Center, National Institute of
Advanced Industrial Science and Technology

†† 科学技術振興機構戦略的創造研究推進事業
Core Research for Evolutional Science and Technology,
Japan Science and Technology Agency

††† 筑波大学大学院システム情報工学研究科
Department of Computer Science, University of
Tsukuba

手法の1つとして、最近注目されている．FMO法では、大規模MO計算をいくつかの小規模MO計算に分割して行い、その結果を用いて巨大分子全体のエネルギーや電子密度分布を求める．このFMO法は並列化が非常に容易であるため、すでに複数のグループが並列FMO計算プログラムの開発・性能評価を行っており、非常に高い並列性能が出ることが示されている．ただ、FMO法による計算だけでは分子全体のフロンティア軌道を求めることはできない．しかし、FMO法で得られた電子密度行列を使った追加の計算を行うことで、フロンティア軌道をはじめとするMOやMOエネルギーが精度良く求められることが分かった⁶⁾．この追加の計算も並列化が非常に容易であるため、計算量は多いものの、高速化が図りやすい．本論文では、大規模分子に対して、このFMO法における分子軌道(FMO-MO)計算を規模の大きなクラスター型計算機で実行して、その性能を評価した．2章では簡単な説明を行う．3章では、今回の計算に用いた計算機環境の概要を示す．また、4章で卵白リゾチーム分子への計算結果と、それに対する考察を行う．

2. 理 論

ここでは、FMO法を含めたMO法の概要を述べる．この章では、MOを求める際によく用いられている非経験的MO法の1つであるHartree-Fock(HF)法、ならびにHF法を基にしたFMO法、FMO-MO法の説明を行う．

2.1 従来MO法(Hartree-Fock分子軌道法)

まず従来MO法で一般的に用いられている、Hartree-Fock(HF)法について、その概略を述べる．MO法では、分子軌道 $\{\psi_a(\mathbf{r})\}$ を「基底関数」と呼ばれる既知の関数 $\{\varphi_i(\mathbf{r})\}$ の線形結合で表す．

$$\psi_a(\mathbf{r}) = \sum_i^N (c^a)_i \varphi_i(\mathbf{r}) \quad (1)$$

ここで、 N は計算に用いられる基底関数の数である．MOの線形結合定数(MO係数ベクトル、 $\{c^a\}$)を求めるために、HF法では、以下に示すHartree-Fock-Roothaan(HFR)方程式という行列方程式を解く必要がある．

$$\mathbf{FC} = \mathbf{SC}\epsilon \quad (2)$$

$$\mathbf{F} = \mathbf{H}^{\text{core}} + \mathbf{G} \quad (3)$$

$$H_{ij}^{\text{core}} = \int d\mathbf{r} \varphi_i(\mathbf{r}) \hat{h} \varphi_j(\mathbf{r}) \quad (4)$$

$$G_{ij} = \sum_{k,l}^N D_{kl} \{2(ij|kl) - (il|kj)\} \quad (5)$$

$$S_{ij} = \int d\mathbf{r} \varphi_i(\mathbf{r}) \varphi_j(\mathbf{r}) \quad (6)$$

$$D_{ij} = \sum_a^{n/2} C_{ia} C_{ja} \quad (7)$$

$$(ij|kl) = \int d\mathbf{r}_1 \int d\mathbf{r}_2 \varphi_i(\mathbf{r}_1) \varphi_j(\mathbf{r}_1) \times \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} \varphi_k(\mathbf{r}_2) \varphi_l(\mathbf{r}_2) \quad (8)$$

ここで、行列 \mathbf{F} 、 \mathbf{H}^{core} 、 \mathbf{G} 、 \mathbf{S} 、 \mathbf{D} は、それぞれFock行列、核-1電子ハミルトニアン行列、1電子ハミルトニアン行列、重なり行列、および(電子)密度行列と呼ばれる行列であり、 $N \times N$ のサイズを持つ．このうち、重なり行列 \mathbf{S} は正定値対称行列であり、他は対称行列である． $\mathbf{C} \equiv (\mathbf{c}^m, \mathbf{c}^{m+1}, \dots, \mathbf{c}^p)$ ($1 \leq m \leq p \leq N$)はMO係数ベクトルの組みで構成されるMO係数行列である．また、 \hat{h} は1電子演算子で、 n は電子数を表している．HFR方程式ではFock行列と重なり行列を与えて、一般化固有値問題を解くことになる．しかし、この方程式を解くには、以下のような困難がある．

- (1) HFR方程式で与えるべきFock行列が、密度行列を通して、方程式の解であるMO係数行列に依存している．したがってHFR方程式は非線形方程式であり、反復解法(Self-Consistent Field(SCF)法)で解く必要がある．その繰返し回数は分子の大きさや種類にも依存するが、通常10~100回ほどである．
- (2) Fock行列作成の際には、2電子積分($(ij|kl)$)を使って求める2電子ハミルトニアン行列の計算が必要であるが、この積分の計算量は $O(N^4)$ となる．また、2電子積分の数が外部記憶に保存することも困難なほど多いため、通常、反復のたびに計算する必要がある．
- (3) 最終的にフロンティア軌道(に対応する固有値、固有ベクトル)しか必要でない場合でも、反復計算中は密度行列計算のために、少なくとも電子数の半分の固有ベクトルを求めなくてはならない．そのベクトルの数は、タンパクなどの巨大分子では、数千~数万となる．

以上のことから、タンパクなどの大規模分子のフロンティア軌道を、従来MO法を用いて求めることは、計算コストの点から見ると非常に困難である．

2.2 フラグメント分子軌道(FMO)法

FMO法では、分子を20~30個程度の原子で構成される小さなフラグメントに分割して、このフラグメン

トとフラグメントペアに対して、従来 MO 法を適用する。その結果得られる、フラグメント、ならびにフラグメントペアの密度行列（それぞれ $\{\mathbf{D}_{IJ}\}, \{\mathbf{D}_{IJ}\}$ ）。 N_f はフラグメント数）を用いて、FMO 法における分子全体の密度行列 \mathbf{D}^{FMO} を計算する。

$$\mathbf{D}^{\text{FMO}} = \sum_{I>J}^{N_f} \mathbf{D}_{IJ} - (N_f - 2) \sum_I^{N_f} \mathbf{D}_I \quad (9)$$

FMO 法では、各フラグメントやフラグメントペアに対する MO 計算を独立に行うことができるため、並列処理による高速化が効率良く行えることがすでに示されている⁵⁾。また、分子全体を一度に扱うことがないため、計算量は $O(N^2)$ になり、従来 MO 法の $O(N^4)$ に比べて大幅に減少する。さらにフラグメント間の距離に応じて、フラグメント間相互作用項やフラグメントペアの MO 計算に対して、精度を維持したままさまざまな近似を適用することが可能である。したがって、FMO 法では、巨大分子のエネルギー、ならびに電子密度分布を、従来 MO 法に比べて非常に短時間に、しかも精度良く求めることが可能である^{5),8)}。その一方で、FMO 法では、分子全体の MO を求めることができないため、そのままでは、タンパクなどの巨大分子のフロンティア軌道を求める目的には使えない。

2.3 FMO-MO 法

FMO 法では、分子全体の電子密度行列を得ることができるが、MO を求めることができなかった。FMO 法で計算できる分子のエネルギーや電荷分布なども反応メカニズムを理論的に解明するための有効な情報ではあるが、活性部位などの議論のためには、やはり分子全体の MO を求めることが望まれる。フラグメントの MO で分子全体の MO の代用をすることも提案されているが、比較的小さな分子の計算結果でも、MO の準位が従来 MO 法と違うことがあり、精度の面で若干の不安がある⁷⁾。そこで著者らは、FMO 法で得られる分子全体の密度行列を基に、分子全体に対する Fock 行列を一度だけ計算して、その Fock 行列（と、重なり行列）を用いて一般化固有値問題を解くことで、分子全体における MO、ならびに MO エネルギー（の近似値）を求める、という MO の計算方法を提案した。こうして得られた MO を FMO 法における MO (FMO-MO) と呼び、この FMO-MO が従来 MO 法で得られる MO とほぼ一致することがすでに示されている⁶⁾。従来 MO 法のような反復処理 (SCF 計算) が不要なために、分子全体に対する Fock 行列をただ一度だけ作成するだけでよいこと、またフロンティア

```

int nao2 = N*(N+1)/2;
double D[nao2], Gpartial[nao2], G[nao2];
MPI_Comm_size(MPI_COMM_WORLD, &nproc);
MPI_Comm_rank(MPI_COMM_WORLD, &myid);
if (myid == 0) { read density matrix D;
MPI_Bcast(D, nao2, MPI_DOUBLE, 0, MPI_COMM_WORLD);
Gpartial = 0.0;
ijcs = 0;
for (i=0; i<ncs; i++) {
    for (j=0; j<=i; j++) {
        ijcs++;
        if ( (ijcs%nproc) != myid) continue;
        for (k=0; k<=i; k++) {
            lmax = (k==i ? (j-1) : k);
            for (l=0; l<=lmax; l++) {
                calculate integral (ij|kl);
                addition to Gpartial using
                    i, j, k, l, D and (ij|kl);
            }
        }
    }
}
MPI_Barrier(MPI_COMM_WORLD);
MPI_Allreduce(Gpartial, G, nao2,
    MPI_DOUBLE, MPI_SUM, MPI_COMM_WORLD);

```

*: N = number of basis functions.
ncs = number of contracted shells.

図 1 並列 Fock 行列作成ルーチンの擬似コード

Fig.1 Pseudo-code for parallel Fock matrix generation.

軌道に対応するごく一部の固有値、固有ベクトルしか計算する必要がない、などといった利点が FMO-MO 計算には存在する。

FMO-MO を求めるためには、(1) FMO 計算、(2) 分子全体の Fock 行列作成、および (3) Fock 行列の対角化、の 3 つのステップが必要である。このうち、最初のステップである FMO 計算は、先に述べたとおり、計算量が $O(N^2)$ と、従来 MO 法に比べて計算コストが少ないうえに、並列処理による高速化が期待される部分である。次の Fock 行列作成部分については、図 1 に擬似コードを示す。コード中の、nao2 は対称行列の要素数 ($= N(N+1)/2$) を、また ncs は縮約シェル数である。縮約シェル数は基底関数の数に比例する量であり、基底関数の数 (N) が 10,000 くらいであれば縮約シェル数 (ncs) は 6,000 ~ 7,000 である。先に示した Fock 行列中の 2 電子ハミルトニアン行列の式や、この擬似コードの 4 重ループ構造が示すとおり、Fock 行列作成で必要となる 2 電子積分計算の計算量は $O(N^4)$ であり、計算コストが大きい。しかし、個々の 2 電子積分は独立に計算できるため、図 1 で示した擬似コードのように、簡単に並列化が可能であり、高い並列化効率を持つことも示されている⁹⁾。そこで、大規模なクラスター型並列計算機である

表 1 AIST Super Cluster を構成する 3 つのクラスタ計算機

Table 1 Outline of three cluster computers constructing AIST Super Cluster.

| cluster name | processor | # processors | memory | network (bandwidth) |
|--------------|------------------------|----------------------|------------|---------------------------|
| P32 | Opteron (model 246) | 2,048 (2 × 1,024) | 6 GB/node | Myrinet (2 Gbps) |
| M64 | Itanium2 (1.3 GHz) | 512 (4 × 128) | 16 GB/node | Myrinet (4 Gbps) |
| F32 | Xeon (3.06 GHz) | 536 (2 × 268) | 4GB/node | Gigabit ether (1 Gbps) |

AIST Super Cluster (ASC)¹⁰⁾ を用いてタンパク分子に対する FMO-MO 計算を行い、その性能評価を行った。

3. 計算機環境

FMO 法の計算には、国立衛生研の中野らが開発した並列 FMO 計算プログラム、ABINIT-MP¹¹⁾ を用いた。ABINIT-MP は Fortran90 でコーディングされている。FMO-MO 計算における Fock 行列作成には、著者らが作成し、C 言語で記述された 2 電子積分プログラム⁹⁾ を用いた。これら 2 つのプログラムは、MPI¹²⁾ によって並列化されている。重なり行列が正定値対称であることを利用して、Cholesky 分解を用いた標準固有値問題への変換、Householder 変換、bisection 法、ならびに逆反復法などを用い、Fock 行列に対する一般化固有値問題を解いた^{13),14)}。それらの操作には、BLAS, LAPACK ルーチン¹⁵⁾ を用いた。対象とする分子は、卵白リゾチーム (PDB ID=1LZT, アミノ酸 129 残基, 1,961 原子) とした。リゾチームは脊椎動物の細胞や分泌物中に広く分布し、植物性細菌の細胞壁を破壊する、殺菌作用のある酵素である。機能がよく知られており、タンパクの中では比較的小規模ではあるが、それでも従来の MO 法では計算が非常に困難な大きさを持つ分子である。この卵白リゾチームに対する FMO-MO の計算を、6-31G 基底関数系^{16)~20)} を用いて行った。その場合の基底関数の総数 N は 10,969 である。今回の計算は、すべて ASC を用いて行った。ASC は表 1 に示すような 3 つのクラスタ計算機で構成されている。FMO 計算は通信量が計算量に比べて比較的小さいため、Gigabit ether で結合された F32 クラスタで計算を行った。また Fock 行列作成は、並列化効率の高い 2 電子積分計算で多くのプロセッサを利用したいこと、ならびに行列のリダクションの際に大きなネットワーク負荷がかかると予想されるため、2,048 プロセッサが高速なネットワーク Myrinet で結合された P32 クラスタを用いた。一般化固有値問題の求値は P32 クラスタ中の 1 プロセッサ

を用いて行った。F32 クラスタでは、コンパイラとして Intel Fortran Compiler (version 8.0)²¹⁾ を、また MPI ライブラリは MPICH (バージョン 1.2.5.2)²²⁾ を用いた。一方、P32 クラスタでは、コンパイラとして PGI コンパイラ (バージョン 5.1.3)²³⁾ を、また、MPI ライブラリは、高速なクラスタとしての利用を容易にするために P32 クラスタに導入されている Score クラスタシステムソフトウェア²⁴⁾ が提供するものを用いた。また、BLAS, LAPACK のライブラリとして、AMD 社が提供している数値計算ライブラリ ACML (バージョン 2.1.0)²⁵⁾ を用いた。

4. 結果と考察

4.1 FMO 計算

まず、FMO 法の計算時間を示す。この計算では、アミノ酸 2 残基を 1 つのフラグメントとして計算を行った。ただし、計算精度の問題から S-S 結合部分を切断していないことと、末端の 1 残基は 1 つのフラグメントとして処理していることもあるので、この分子のフラグメント数は 68 になる。今回は、F32 クラスタの 256 プロセッサを用い、SCF 計算には 1 フラグメント (フラグメントペア) あたり 16 プロセッサを使って計算を行ったが、全体の計算時間は 2 時間弱 (7,522 秒) であった。フラグメント (フラグメントペア) の SCF 計算では、より多くのプロセッサを用いて並列処理を行うことで、さらに短時間で計算することができる。したがって、プロセッサ数の多い P32 クラスタで 2,000 プロセッサを使用した場合には、リゾチーム程度の大きさの分子に対する FMO 計算を、20 分足らずで行うことも可能になる。

4.2 Fock 行列作成

次に、FMO-MO 計算で用いる大規模な Fock 行列作成の計算時間を表 2 に示す。ここでは、使用するプロセッサ数を 240 台、480 台と変化させて同じ計算を行い、その実行時間を計測した。また、表中には Fock 行列作成において通信にかかる時間も示してある。この表を見ると、計算時間はプロセッサ数の増加にとも

表 3 一般化固有値問題を解くための計算時間 (秒)
(行列サイズ 10,969 × 10,969, 40 個の固有値, 固有ベクトルを計算)

Table 3 Elapsed time (seconds) to solve the generalized eigenvalue problem for large scale Fock matrix (matrix size = 10,969 × 10,969, 40 eigenvalues and corresponding eigenvectors were calculated).

| Operation | subroutine ^a | Elapsed time |
|---|-------------------------|--------------|
| Cholesky decomposition of S | DPPTRF | 866 |
| Transformation to standard eigenproblem | DSPGST | 2,016 |
| Tri-diagonalization | DSPTRD | 2,512 |
| Eigenvalue calculation with bisection method | DSTEBZ | < 1 |
| Eigenvector calculation with inverse iteration | DSTEIN | < 1 |
| Eigenvector transformation for standard eigenproblem | DOPMTR | 27 |
| Eigenvector transformation for generalized eigenproblem | DSPTV | 18 |
| Total | | 5,439 |

^a; subroutine name of BLAS and LAPACK.

表 2 リゾチーム分子に対する Fock 行列作成時間 (秒)
(6-31G 基底関数系, 10,969 関数)

Table 2 Elapsed time (seconds) of Fock matrix generation for Lysozyme molecule (6-31G basis set, 10,969 basis functions).

| # processors | 240 | 480 |
|----------------------------|-------|-------|
| elapsed time | 5,884 | 3,150 |
| (for communication) | (117) | (145) |
| speedup ratio ^a | 1.00 | 1.87 |

^a; speedup ratio to the results using 240 processors.

なって、計算時間が大きく短縮されていることが分かり、Fock 行列計算時に行う 2 電子積分計算の独立性の高さが表れる結果となった。この Fock 行列作成の計算量は、先述のとおり $O(N^4)$ であり、最も計算コストがかかる部分ではあるが、今回の結果を見る限り、大規模並列化による高速化が容易であり、さらに他の高速 Fock 行列作成のためのアルゴリズム^{26)–28)} などと併用すれば、より短時間で計算が可能となるはずである。一方で、高速な Myrinet で接続された P32 クラスタを用いても、240 プロセッサ、480 プロセッサ使用時で、通信時間がそれぞれ 117 秒、145 秒かかっており、これは Fock 行列計算時間全体の、それぞれ、1.99%、4.60%を占める。より多くのプロセッサを用いると通信時間がより大きくなるが、今回の結果から類推すると、10,000 基底程度の分子における Fock 行列作成は、P32 クラスタの 2,000 プロセッサを使用すれば、20 分を切る時間で計算できると予想される。ところで、5,000 × 5,000 を超えるような大規模な Fock 行列を作成する場合には、2 電子積分と密度行列要素との積を Fock 行列に加算する部分での数値誤差混入に対して注意する必要がある²⁹⁾。今のところ、その精度検証は行っていないが、今回は、Fock 行列を数百プロセッサを用いた並列計算で求めているため、結果的に部分和をとっていること、ならびに、FMO 法で

得られた密度行列の要素には、正確に“0”となっている要素が多く存在し、その部分に関わる積分計算、ならびに Fock 行列への加算を行わないため、誤差の混入が抑えられている、と考えている。

4.3 一般化固有値問題

表 3 には、FMO-MO を求めるための最後のステップである、Fock 行列と重なり行列を用いた一般化固有値問題を解くために要した時間を示す。Fock 行列は比較的密な行列であり、大規模疎行列に対する固有値解法は使えないため、前述のとおり、Cholesky 分解、Householder 変換、bisection 法、および逆反復法を組み合わせた、密行列に対する一般化固有値問題の解法として広く使用されている計算手法による対角化を行った。また、フロンティア軌道 (3,810, 3,811 番目の軌道) 付近の 40 個の MO に対応する固有値、固有ベクトルを求めるために、bisection 法のエキスパートルーチン (DSTEBZ) における固有値のシリアル番号による範囲指定モード (RANGE = 'I', IL = 3,791, IU = 3,830) で計算を行った。さらに、メモリの使用量を減らすために、圧縮 ('U') 形式の行列を用いる BLAS と LAPACK のエキスパートルーチンを用いた。表 3 の “subroutine” の欄に、用いたサブルーチン名が併記してある。この結果を見ると、まず対角化全体で 5,439 秒と、プロセッサ数が異なるものの、FMO 計算や Fock 行列作成の計算時間に対して無視できない時間が必要であることが分かった。特に、並列化が困難である三重対角化で多くの時間を費やしていた。FMO 計算や Fock 行列作成の計算時間は大規模並列化による高速化が容易であるが、対角化は並列化による計算時間短縮が比較的難しい。FMO-MO 計算をより高速に行うためには、FMO 計算や Fock 行列作成だけでなく、並列化効率の良い一般化固有値問題解法の開発が望まれる。ところで、擬縮退している固

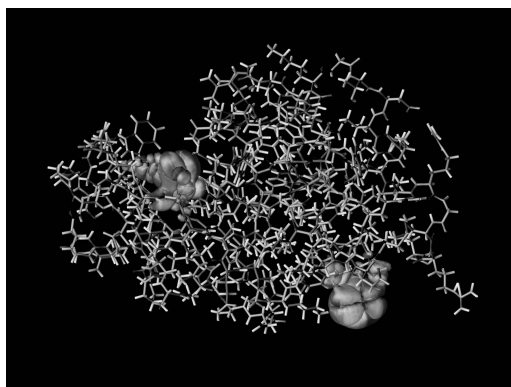


図 2 FMO-MO 計算で得られたリゾチーム分子のフロンティア軌道 (左側 HOMO, 右側 LUMO)

Fig.2 Highest occupied (left lobe) and lowest unoccupied (right lobe) molecular orbitals of Lysozyme molecule obtained by FMO-MO method.

有値に対応する固有ベクトルを逆反復法で求めた場合には、得られた固有ベクトル間の直交化が必要となる。今回のケースでは固有ベクトルがほぼ直交化していたため、改めて固有ベクトルの直交化を行わなかったが、大規模な Fock 行列の固有値は擬縮退していることが多いため、並列化が困難な固有ベクトルの直交化が計算時間を考える際に大きな問題になる可能性がある。しかし、FMO-MO 法では少数の固有ベクトルの直交化だけですむので、そのための計算量がかなり小さいことから、FMO-MO 計算で、固有ベクトル直交化の計算時間が問題になることはない、と考えている。

4.4 得られた MO について

最後に、求められた FMO-MO の等値面を図 2 に示す。この図は FMO-MO 計算で得られたフロンティア軌道の MO 係数 (固有ベクトル) を基に描いたものである。図にある 2 つの等値面のうち、左側は電子が詰まったフロンティア軌道 (Highest Occupied MO, HOMO) を、右側は電子が入っていないフロンティア軌道 (Lowest Unoccupied MO, LUMO) を表している。この図から、分子軌道は巨大な分子全体に広がっているのではなく、特定の部位に局在化していること、また、2 つのフロンティア軌道の位置が大きく離れているため、他の化学物質と反応する部位が、反応する相手分子の電子吸引力、電子供与性の違いによって大きく異なること、などが分かる。タンパク (酵素) の反応には周囲の水の存在が重要であるにもかかわらず今回の計算では考慮していないため、ここではこれ以上の化学的な考察は省略する。ただ、FMO 法で周囲に存在する多数の水分子をあらわに考慮して分子軌道計算を行うことも容易であるため、今後、周囲の溶媒

分子が反応に及ぼす影響についての議論も FMO-MO 計算で行うことができると期待される。

5. ま と め

今回、FMO-MO 法を用いたタンパクに対する MO 計算を大規模なクラスタ計算機を用いて行い、その性能を評価した。その結果、これまでは計算が困難であった、基底関数が 10,000 を超えるような巨大分子に対する MO 計算が、4 時間半程度という非常に短い時間で行うことができた。したがって、タンパクや DNA などの生体分子の反応に対して、分子軌道計算による理論的な反応性予測が、FMO-MO 法を用いることで比較的短時間に行えることが分かった。

2 電子積分の計算量は $O(N^4)$ であるが、効率の良い並列化が容易であるため、P32 クラスタのように高速ネットワークで接続された大規模クラスタ計算機を用いれば、基底関数の数が 10,000 程度の分子に対する Fock 行列は、1 時間足らずで計算が可能なが分かった。一方で、計算量が $O(N^3)$ であり、従来の MO 計算では 2 電子積分計算 (Fock 行列作成) の計算量に紛れて見えていなかった一般化固有値問題の計算時間が、分子サイズが大きくなると大きなウェイトを占めることが分かった。今回の卵白リゾチーム分子に対する計算では、約 34% の計算時間を一般化固有値問題求値に費やしていた。並列化効率の高い FMO 計算、ならびに Fock 行列作成のために、より多くの計算機資源を使った場合には、さらにこの割合が増加することになる。したがって、高速な FMO-MO 計算のためには、Fock 行列作成だけでなく、一般化固有値問題を高速に解く手法が求められる。

謝辞 本研究は独立行政法人科学技術振興機構 (JST) が行う戦略的創造研究推進事業 (CREST) の研究領域「シミュレーション技術の革新と実用化基盤の構築」の研究プロジェクト「グリッド技術を用いた大規模分子シミュレーションプログラムの開発」の支援による。また、すべての計算は独立行政法人産業技術総合研究所 (AIST) の AIST Super Cluster を用いて行った。

参 考 文 献

- 1) Szabo, A. and Ostlund, N.S.: *MODERN QUANTUM CHEMISTRY (Introduction to Advanced Electronic Structure Theory)*, Dover Publications Inc., Mineola, New York.
- 2) 米澤貞次郎, 永田親義, 加藤博史, 今村 詮, 諸熊奎治: 三訂 量子化学入門 (上), 化学同人 (1990).

- 3) Kitaura, K., Sawai, T., Asada, T., Nakano, T. and Uebayasi, M.: Pair interaction molecular orbital method: An approximate computational method for molecular interactions, *Chemical Physics Letters*, Vol.312, p.319 (1999).
- 4) Kitaura, K., Ikee, E., Asada, T., Nakano, T. and Uebayasi, M.: Fragment molecular orbital method: An approximate computational method for large molecules, *Chemical Physics Letters*, Vol.313, p.701 (1999).
- 5) Nakano, T., Kaminuma, T., Sato, T., Akiyama, Y., Uebayasi, M. and Kitaura, K.: Fragment molecular orbital: Application to polypeptides, *Chemical Physics Letters*, Vol.318, p.614 (2000).
- 6) Inadomi, Y., Nakano, T., Kitaura, K. and Nagashima, U.: Definition of molecular orbitals in fragment molecular orbital method, *Chemical Physics Letters*, Vol.364, p.139 (2002).
- 7) Sekino, H., Sengoku, Y., Sugiki, S. and Kurita, N.: Molecular orbital analysis based on fragment molecular orbital scheme, *Chemical Physics Letters*, Vol.378, p.589 (2003).
- 8) Fukuzawa, K., Kitaura, K., Nakata, K., Kaminuma, T. and Nakano, T.: Fragment molecular orbital study of the binding energy of ligands to the estrogen receptor, *Pure and Applied Chemistry*, Vol.75, p.2405 (2003).
- 9) Inadomi, Y., Nakano, T., Kitaura, K. and Nagashima, U.: Increased efficiency of parallel calculations of fragment molecular orbitals by using fine-grained parallelization on a HITACHI SR8000 supercomputer, *Proc. High-Performance Computing and Networking, 9th International Conference, HPCN Europe 2001*, Amsterdam, The Netherlands (June 2001).
- 10) AIST Super Cluster homepage.
<http://unit.aist.go.jp/tacc/en/supercluster.html>
- 11) ABINIT-MP homepage.
<http://moldb.nihs.go.jp/abinitmp/>
- 12) MPI: A Message-Passing Interface Standard, *MPI forum* (1995).
- 13) 戸川隼人: マトリックスの数値計算, オーム社 (1971).
- 14) 小国 力(訳): 行列計算パッケージ LAPACK 利用の手引, 丸善 (1996).
- 15) Lapack homepage.
<http://www.netlib.org/lapack/>
- 16) Ditchfield, R., Hehre, W.J. and Pople, J.A.: *Journal of Chemical Physics*, Vol.54, p.724 (1971).
- 17) Hehre, W.J., Ditchfield, R. and Pople, J.A.: *Journal of Chemical Physics*, Vol.56, p.2257 (1972).
- 18) Hariharan, P.C. and Pople, J.A.: *Molecular Physics*, Vol.27, p.209 (1974).
- 19) Gordon, M.S.: *Chemical Physics Letters*, Vol.76, p.163 (1980).
- 20) Hariharan, P.C. and Pople, J.A.: *Theoretica Chimica Acta*, Vol.26, p.213 (1973).
- 21) Intel Fortran Compiler version 8.0, Copyright © 2001–2004 Intel Corporation.
- 22) MPICH homepage.
<http://www-unix.mcs.anl.gov/mpi/mpich/>
- 23) PGI Workstation 5.1, Copyright © 2003 The Portland Group™ Compiler Technology STMicroelectronics, Inc.
- 24) Score homepage.
<http://www.pcccluster.org/index.html.en>
- 25) AMD Core Math Library (ACML) Version 2.1.0, Copyright © 2003, 2004 Advanced Micro Devices, Inc., Numerical Algorithms Group Ltd.
- 26) Schwegler, E. and Challacombe, M.: Linear scaling of the Hartree-Fock exchange matrix, *Journal of Chemical Physics*, Vol.105, p.2726 (1996).
- 27) Challacombe, M. and Schwegler, E.: Linear scaling computation of the Fock matrix, *Journal of Chemical Physics*, Vol.106, p.5526 (1997).
- 28) Schwegler, E., Challacombe, M. and Head-Gordon, M.: Linear scaling computation of the Fock matrix II. Rigorous bounds on exchange integrals and incremental Fock build, *Journal of Chemical Physics*, Vol.106, p.9708 (1997).
- 29) Takashima, H., Kitamura, K., Tanabe, K. and Nagashima, U.: Is Large-Scale *Ab Initio* Hartree-Fock Calculation Chemically Accurate? Toward Improved Calculation of Biological Molecule Properties, *Journal of Computational Chemistry*, Vol.4, p.443 (1999).

(平成 16 年 10 月 4 日受付)

(平成 16 年 11 月 15 日採録)



稲富 雄一

1998年筑波大学大学院化学研究科博士課程修了。博士(理学)。同年同大学化学系技官,2000年(株)富士総合研究所入社,2002年独立行政法人産業技術総合研究所特別研究員,2004年独立行政法人科学技術振興機構戦略的創造事業研究員,現在に至る。専門分野は理論化学。



梅田 宏明

1970年生。1997年東北大学大学院理学研究科修了。理学博士。科学技術振興機構,CREST研究員。独立行政法人産業技術総合研究所グリッド研究センターにてグリッド向け大規模分子軌道計算プログラムの開発に従事。日本化学会会員。



渡邊 寿雄

2000年筑波大学大学院化学研究科博士課程修了。博士(理学)。同年同大学化学系助手,2002年科学技術振興機構計算科学技術研究員。2004年同戦略的創造事業研究員,現在に至る。日本化学会会員。



櫻井 鉄也(正会員)

1986年名古屋大学大学院工学研究科博士課程前期課程修了。同年名古屋大学工学部助手。1993年筑波大学電子・情報工学系講師。1996年同大学電子・情報工学系助教授。現在,筑波大学大学院システム情報工学研究科助教授。独立行政法人産業技術総合研究所客員研究員。博士(工学)。方程式の反復解法と精度保証,大規模固有値問題の並列解法,数理ソフトウェアの利用支援環境の研究に従事。1996年日本応用数理学会論文賞受賞。日本応用数理学会会員。



長嶋 雲兵(正会員)

1983年北海道大学大学院理学研究科博士後期課程化学第二専攻修了。理学博士。同年岡崎国立共同研究機構分子科学研究所電子計算機センター助手。1992年お茶の水女子大学理学部情報科学科助教授,1996年同教授を経て,1998年通産省工業技術院物質工学工業技術研究所基礎部理論化学研究室長。1999年同産業技術融合領域研究所計算科学研究グループ長,2001年独立行政法人産業技術総合研究所先端情報計算センター情報基盤研究開発室長,2002年より同グリッド研究センター総括研究員。筑波大学連携大学院大学教授。計算化学,情報化学,大規模数値計算,広域分散並列処理の研究開発に従事。日本化学会,IEEE,応用数理学会,計算工学会,化学工学会,日本コンピュータ化学会各会員。