

## 量子化学シミュレーションの高速化

田宮 豊      濱湊 真      石原 輝雄      池 敦      吉川 隆英

**概要:** 量子化学シミュレーションで一般的に用いられる密度汎関数法 (DFT: Density Functional Theory) では、二電子積分に多くの計算時間が費やされることが知られている。カットオフ半径以遠の電子ペア間の相互作用を無視する近似によって計算量が削減するも、依然として二電子積分の計算時間の多さが問題になっている。本論文では、FPGA による二電子積分の高速化を提案する。更に、FPGA へのデータ供給を効率化させるため、ウィンドウによる電子集合の効率的な管理を提案する。シミュレーション空間内においてカットオフ半径を 1 辺の長さとするウィンドウをスキャンし、ウィンドウに出入りする電子を効果的に管理することにより、カットオフ半径の判定回数自体を削減させ、二電子積分の時間を削減する。量子化シミュレーションソフトである CP2K について提案手法の有効性を確認した。

**キーワード:** 密度汎関数法, ハートリーフォック法, 二電子積分, FPGA, CP2K

## An Accelerating Method for Quantum Chemistry Simulation

TAMIYA Yutaka\*      HAMAMINATO Makoto\*      ISHIHARA Teruo\*  
IKE Atsushi\*      YOSHIKAWA Takahide\*

**Abstract:** The Density Functional Theory (DFT), which Quantum Chemistry simulation generally uses, is known as a time-consuming process. The reason is it must calculate an enormous number of two-electron integrals. In order to speed up the simulation, it exploits the approximation, where it ignores electron pairs located beyond 'cutoff radius'. Although applying this approximation, the two-electron integrals are still time-consuming. In this paper, we propose FPGA acceleration of two-electron integrals. To supply input data to the FPGA efficiently, we also propose a windowing method, which scans the window with the same size of the cutoff radius all over the simulation space and efficiently enumerate electron pairs located within the cutoff radius. We have found the efficiency of the proposed method with the public-domain quantum chemistry simulation: CP2K.

**Keywords:** Density Functional Method, Hartree-Fock Method, Two-Electron Integral, FPGA, CP2K

## 1. はじめに

創薬や材料物性の分野においては、分子や原子の振る舞いや相互作用を、実験よりも環境制約の少ないコンピュータ・シミュレーションにおいて解明しようとする試みが数多くなされるようになってきている[1][2]。

高効率燃料電池[3]やアンモニア常温合成[4][4]など、近年の地球規模のエネルギー・環境問題の解決に向けた様々な新素材研究においても、様々な環境下での化学反応の詳細を解析するために、コンピュータ・シミュレーションの活用への期待は大きい。

しかし、一般的な古典的力学モデルでは、自由電子が密接に関与する化学反応や溶媒との相互作用を精密にシミュレーション上で再現することが出来ない。そこで、電子構造までを考慮した量子力学モデルを導入して分子間相互作用をシミュレーション上で精密に再現する試みが行われるようになってきた。

量子化学モデルのシミュレーションにおいてはシュレーディンガー方程式を解いて、原子内の電子状態や分子の位置と速度を計算する。この計算は経験的な化学知識を必要としない第一原理(ab initio)[5]に基づくものの、計算コストが非常に大きくなってしまふ。そのため、そのままでは数百原子程度の極めて小さい系しか取り扱うことが出来ない。そこで、この計算を高速化するためにアルゴリズムの最適化やハードウェア・アクセラレータの開発などが試みられている[7]。

本論文では量子化学シミュレーションを高速化するアクセラレータの実現に向け、とくに計算コストの高い二電子積分に着目し、その処理をハードウェア・アクセラレータで効率的に実行するための要素技術の提案を行う。具体的には、粒子データの管理方法を見直し、計算対象のスキップを効率的に行う手法の提案を行う。本論文の構成は次のようになる。第2章では、量子化学シミュレーションの概要を説明する。第3章では、その性能ボトルネックの詳細とそれを解消する手法の提案を行う。第4章では、その

提案手法の効果を見積もる。そして第5章でまとめる。

## 2. 量子化学シミュレーションの概要

3 個以上の電子に対する量子化学シミュレーションにおいては、シュレーディンガー方程式を直接解くことは困難であるため、数々の近似計算が導入されている。その代表的なものが、密度汎関数法(DFT: Density Functional Theory)とハートリーフォック法(Hartree-Fock Method)[6]である。

複数の電子が存在する場合、個々の電子は、他電子へのクーロン力を及ぼすと同時に、電気作用を受ける。この関係を、個々の電子に対する連立シュレーディンガー方程式として同時に解くことは、技術的な困難を伴う。そこで、密度汎関数法では、個々の電子が他電子から受けるクーロン力を電子密度関数として表す。これにより、個々の電子に対するシュレーディンガー方程式を独立に解くことが可能となり、これを結果が収束するまで計算を繰り返すアルゴリズム(SCF 法: Self-Consistent-Field)が一般的になった。

更にハートリーフォック法では、電子の波動関数を基底関数の線形和で表現することにより、全電子に対するシュレーディンガー方程式を、フォック行列と呼ばれる行列の方程式に帰着する手法が導入された。

以上の近似手法の導入により、量子化学シミュレーションは現実時間で解けるようになったものの、依然として計算に時間が掛かっている。

最も時間が掛かる計算は二電子積分である。図1に、水分子 64 個を量子化学シミュレーションの 1 つである CP2K[8]を CPU 単スレッドで実行した例のコールグラフの一部を示す。図中の箱には、Fortran のサブルーチン名(モジュール名で修飾された識別名)と CPU 使用率(自身のみの場合と、自身配下のサブルーチンを含む場合)を示す。サブルーチン `integrate_four_center` は、二電子積分を計算し、その結果をフォック行列の更新に使う。配下サブルーチンを含めた CPU 使用率は 95.2% である。この例のように、二電子積分に掛かる CPU 時間は 60%から 90%まで掛かる重い処理であることが分かる(図1)。

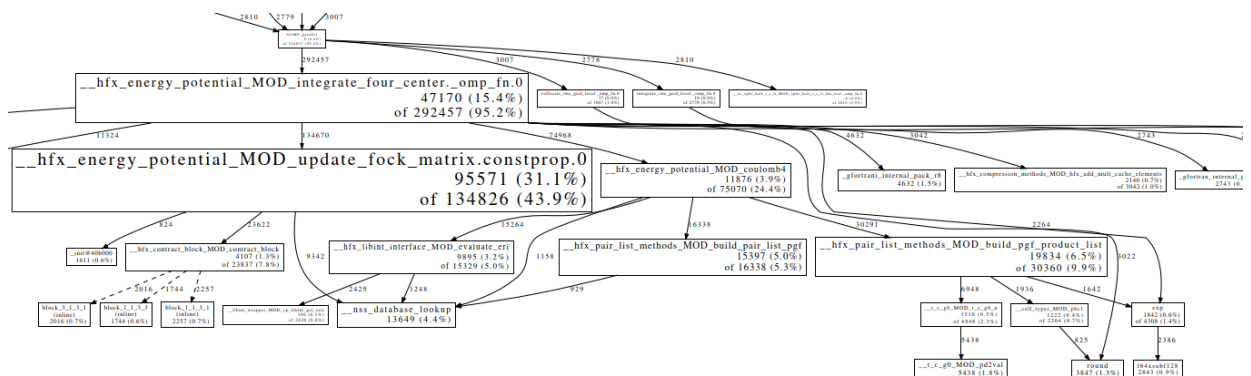
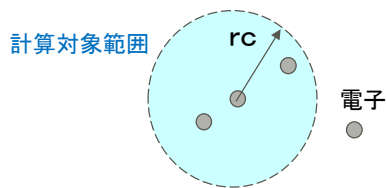


図 1 CP2K のコールグラフと CPU 使用率(一部)

$$\langle \phi_1 \phi_2 | \phi_3 \phi_4 \rangle = \iint \phi_1^*(\mathbf{r} - \mathbf{a}_1) \phi_2^*(\mathbf{r}' - \mathbf{a}_2) \frac{1}{|\mathbf{r} - \mathbf{r}'|} \phi_3(\mathbf{r} - \mathbf{a}_3) \phi_4(\mathbf{r}' - \mathbf{a}_4) d\mathbf{r} d\mathbf{r}'$$

数式 1



※図は2次元で表現しているが、実際は3次元空間上の距離を扱う

図 2 カットオフ半径

二電子積分は数式 1 で定義される。これは 4 つの電子波動関数の基底関数 ( $\phi_i, i=1..4$ ) 毎に計算する必要がある。 $\mathbf{a}_i$  は各基底関数の中心座標を表す。基底関数はいずれかの原子の電子軌道であり、基底関数の中心座標として、電子が属する原子の中心座標を使う(説明を簡単にするため、本論文ではこの座標を「電子の座標」と呼ぶ)。CP2K では、 $\phi_1$  と  $\phi_2$ 、および、 $\phi_3$  と  $\phi_4$  のそれぞれの電子ペア列挙を二電子積分の計算の前に行っている。

これまで二電子積分の高速化が試みられてきた。クーロン力は遠距離では弱まるため、カットオフ半径として定数  $rc$  を導入してカットオフ半径以遠の電子ペアを無視する近似手法もその 1 つである(図 2)。更に、CP2K では電子を領域分割管理している。電子を、複数の計算機ノードに割り当て、各計算機ノードは、割り当てられた電子について、その近傍を探索してペアとなる電子を抽出する。

しかしながら、CP2K の電子の割り当ては、シミュレーションで扱う電子のリストを先頭から等分割しているだけで、電子の座標情報を使った高速化は使っていない。原田ら[7]は二電子積分計算の専用 LSI 化を行った。基底関数がガウス型関数で表される場合は二電子積分を漸化式として計算できることを利用し、この漸化式計算をハードウェア化している。しかし、ハードウェアへの入力データをどのように効率的に供給するか、その入力データの元になるペア電子をどのように列挙するかについては考慮されていない。

本論文では、量子化学シミュレーションを高速化するために、二電子積分の FPGA へのオフロードを検討した。二電子積分の漸化式計算自体を FPGA のパイプライン回路で高速化可能だとしても、FPGA へのデータ供給が非効率的ならば高速化は望めない。そこで、カットオフ半径内のペア電子の列挙について、座標情報を用いたウィンドウスによって効率化する手法を提案する。

### 3. 性能ボトルネックとその解消方法提案

前章で述べた量子化学シミュレーションは、汎用 CPU や GPU での高速化が困難なアプリの 1 つである。その 1 つの理由として、二電子積分のような、BF 比(演算量対するデータ量)が大きい計算が大量に必要なことである。そのため、我々は専用ハードウェアによる高速化を検討している(図 3)。

二電子積分とフォック行列計算の重い処理は高位合成で合成した回路を FPGA にオフロード実行[9]する。FPGA を用いる理由は、今後量子化学シミュレーションで提案される新たな計算アルゴリズムにも対応するためである。その目的には、プログラマブルなハードウェアとして FPGA が適している。更に、ソフトウェアで記述されたアルゴリズムを短時間で実装するために、高位合成ツールを用いる。

一方、動的に収束条件を扱う SCF 計算は、汎用 CPU で柔軟に実行する。FPGA と汎用 CPU が、それぞれに適した処理を担当することにより、全体アプリの効果的な高速化が可能になると考える。

従来ソフトウェアの二電子積分計算では、総当たりで全電子ペアを列挙して、それらの距離がカットオフ半径内かの判定を行っている。明らかに遠いペアについても無駄なカットオフ半径判定を行っており、実行時間の増大を招いている。

そこで、我々は、電子の座標情報を用いたウィンドウスを導入し、 $x,y,z$  方向のいずれかの座標差がカットオフ半径内の電子を管理する手法を提案する。

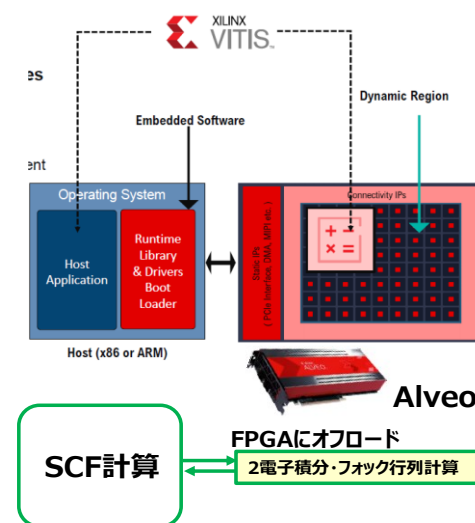


図 3 FPGA オフロード構成

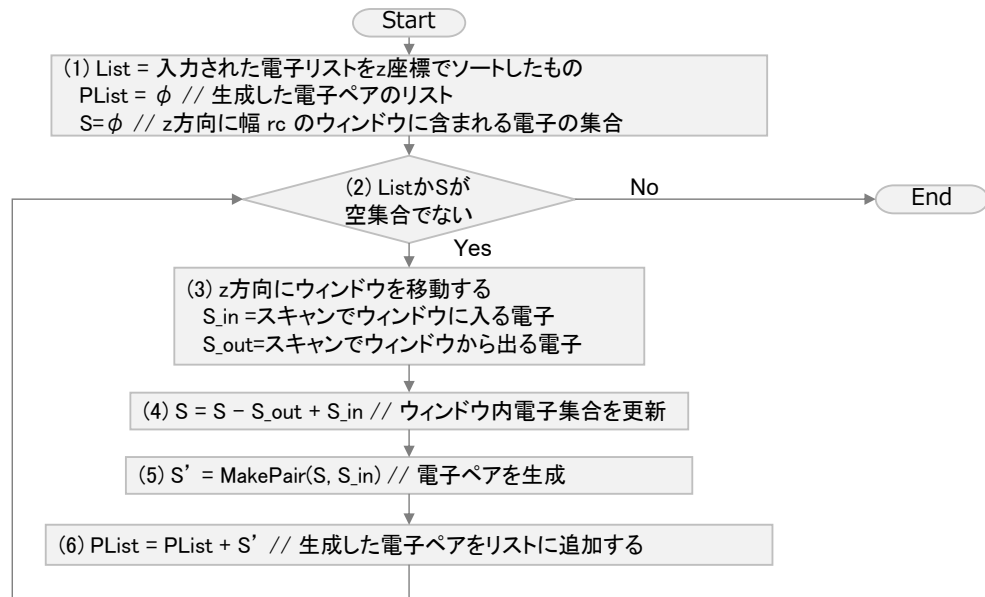


図 4 提案手法フロー

ウィンドウを空間内にスキャンしながら、ウィンドウ内の電子集合を効率的に更新することにより、電子ペア間距離のカットオフ半径内判定の回数を削減できる。また、電子ペア候補をウィンドウから予測できるので、電子データをワークメモリに予めロードでき、FPGA の漸化式計算パイプライン回路へのデータ供給を効率的になると期待できる。従来の二電子積分計算では、全ての電子ペアを列挙して、それらの距離がカットオフ半径内かの判定を行っている。明らかに遠いペアについても無駄なカットオフ半径判

提案手法では、電子座標情報を用いたウィンドウを導入し、 $x, y, z$  方向のいずれかの座標差がカットオフ半径内の電子を管理する。ウィンドウを空間内にスキャンしながら、ウィンドウ内の電子集合を効率的に更新することにより、電子ペア間距離のカットオフ半径内判定の回数を削減できる。また、電子ペア候補をウィンドウから予測できるので、電子データをワークメモリに予めロードでき、FPGA の漸化式計算パイプライン回路へのデータ供給を効率的になると期待できる。

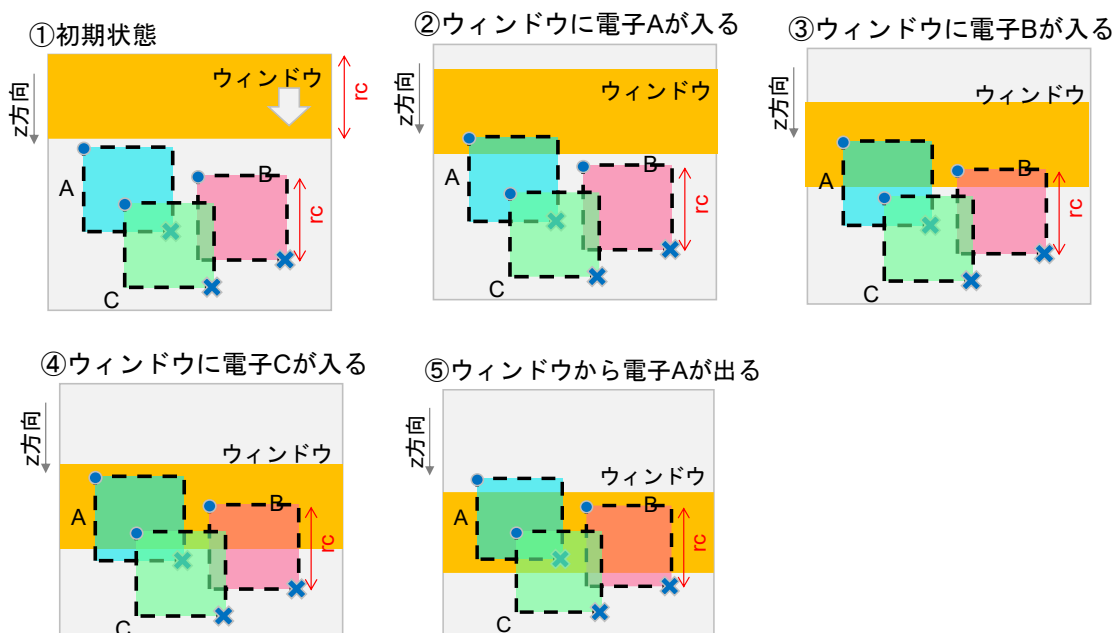


図 5 ウィンドウスキャン例

(List)を入力とする。この電子リストは予め  $z$  座標でソートされているものとする。次に、 $z$  方向に  $rc$  の幅を持ち、 $x$ - $y$  平面に広がるウィンドウを設定する。このウィンドウは、初期状態では  $z$  座標の最小値を取り、 $z$  方向に順次移動(スキャン)させる。

ウィンドウ移動に従って、ウィンドウに入る電子 ( $S_{in}$ )、および、ウィンドウから出る電子 ( $S_{out}$ ) を求める。

ウィンドウに入る電子 ( $S_{in}$ )、および、ウィンドウから出る電子 ( $S_{out}$ ) を使って、ウィンドウ内電子集合 ( $S$ ) を更新する。ウィンドウ内電子集合 ( $S$ ) は、現在位置のウィンドウに含まれる電子を集約として常に保持する。

ウィンドウに入る電子 ( $S_{in}$ ) とウィンドウ内電子集合 ( $S$ ) が更新されたタイミングで電子ペア候補を生成する。すなわち、 $S_{in}$  と  $S$  の両集合から電子を1つずつ取り出すような全ての組合せについて、両者の距離を計算する。その距離がカットオフ半径  $rc$  以下の場合、電子ペアとしてリスト変数 PList に追加する。

提案手法の動作を、3 個の電子(A, B, C) の例で説明する(図 5 ①)。説明を簡単にするため、図では  $y$  方向を省略する( $y$  方向を含めた手法の拡張は自明である)。

各電子の座標(始点: ● で図示)と、座標に  $x, y, z$  方向に  $rc$  (カットオフ半径) を加算した座標(終点: × で図示) をプロットする。2 電子の始点と終点を頂点とする立方体(図では正方形)が重なりを持たない時、その 2 電子間の距離は  $rc$  より大きいことが保証される。逆に重なりを持つ場合は電子ペア候補として列挙しておき、後処理で 2 電子間距離がカットオフ半径以内の電子ペアかどうかを判定するため、最終的な電子ペアのリストは、ウィンドウを導入しない従来法のと変わらないことが保証される。本例では、A と C、B と C の領域が重なりを持つため、電子ペア候補となる。

本例に対してウィンドウを移動させた際の動作は以下の通りである。

- ① 初期状態では、ウィンドウは幅  $rc$  で最小  $z$  座標に位置する。この時点でウィンドウは電子を含まない。
- ② ウィンドウが  $z$  方向に移動して、電子 A がウィンドウに入る。この時点では A 以外に電子がウィンドウに無いので、電子ペア候補は生成しない。
- ③ 更にウィンドウが移動して、電子 B がウィンドウに入る。この時点では  $x$  方向(図では横方向)に A と B は  $rc$  以上の距離があるため、電子候補ペアは生成しない。
- ④ 更にウィンドウが移動して、電子 C がウィンドウに入る。C と A が  $x$  方向  $rc$  以内にあることが分かる。C と A の距離を計算して  $rc$  以下ならば、電子ペア候補を生成する。C と B も同様に処理する。
- ⑤ 電子 A がウィンドウから出る時は、ウィンドウ内電子集合を更新するのみで、電子ペア候補を生成しない。

ウィンドウ内の電子集合 PList は、実際は  $x$  座標でソートした XList と  $y$  座標でソートした YList の 2 つのリストとして実装する。

電子がウィンドウから出る時、電子の  $x$  および  $y$  座標を XList と YList のそれぞれから検索できる。ウィンドウ内の電子数を NS とすると、その検索時間は  $O(\log_2 NS)$  であり、効率的である。

同様に、電子がウィンドウに入る場合も、XList と YList の入るべき位置を  $O(\log_2 NS)$  で計算でき、効率的である。また、電子ペア生成において、ウィンドウに入る電子の  $x, y$  座標に  $\pm rc$  の範囲にある電子の検索は、XList と YList を使って  $O(\log_2 NS)$  で計算できるため、電子ペア生成を効率的に行える。

#### 4. 提案手法の評価

提案手法の効果を見積もるため、64 水分子シミュレーションにおける CP2K のサブルーチン build\_pair\_list\_pgf で生成される電子ペアを調べた。何も工夫が無い場合、全電子ペア数は 12,59,111,210 である。CP2K では、この個数と同じ全電子ペアを列挙するループを有している。従来のカットオフ半径の条件でスキップされるペア数は 10,680,724,032 であった。CP2K の全電子ペア列挙ループは、実に 85.2% ( $= 10,680,724,032 / 12,529,111,210$ ) が無駄である。

一方、提案手法を使った場合の電子ペアのスキップ数は 8,760,911,910 であった。これは総当たりの全電子ペアに対して 69.9% であり、カットオフ半径でスキップされるべきペア数の大部分をカバーできる。これより、サブルーチン build\_pair\_list\_pgf に提案手法を実装した場合、最大 3.3 倍 ( $= 100 / (100 - 69.9)$ ) の高速化が見込めることになる。

#### 5. おわりに

本論文では、FPGA による二電子積分の高速化とともに、FPGA へのデータ供給を効率化させるためにウィンドウをスキャンさせる手法を提案した。

ウィンドウによって、シミュレーション空間内の電子位置を効率的に管理できるため、電子ペアの列挙に掛かる処理は、従来の総当たりの生成法に比べて、最大 3.3 倍となることが分かった。

今後は FPGA への実装と CP2K への組み込みを行って提案手法の高速化効果を評価する予定である。

#### 参考文献

- [1] Atsushi Ishikawa, Yoshitaka Tateyama, "A First-Principles Microkinetics for Homogeneous-Heterogeneous Reactions: Application to Oxidative Coupling of Methane Catalyzed by

- Magnesium Oxide”, ACS Catalysis, 2021.
- [2] 宮島 豊生, 他, “既存のパラダイムを革新する新材料・デバイス”, 雑誌富士通 Vol. 69-5, 2018.
- [3] 岡崎 進, “ポスト「京」重点課題5 「エネルギーの高効率な創出、変換・貯蔵、利用の新規基盤技術の開発」について”, サイエンティフィック・システム研究会 HPC フォーラム 2016.
- [4] Yutong Gong, Hideo Hosono, et al. “Ternary Intermetallic LaCoSi as a Catalyst for N<sub>2</sub> Activation”, Nature Catalysis, 1, pp. 178–185, 2018.
- [5] 中田謙吾, “産業利用に役立つ第一原理計算コードの選び方”, [http://support.spring8.or.jp/Doc\\_workshop/PDF\\_20150728/5.nakada.pdf](http://support.spring8.or.jp/Doc_workshop/PDF_20150728/5.nakada.pdf)
- [6] David S.Sholl(著), Janice A.Steckel(著), 佐々木 泰造(訳), “密度汎関数理論入門—理論とその応用”, 吉岡書店, 2014.
- [7] 原田 宗幸, 小原 繁, 村上 和彰, 他, “Eric(二電子積分計算専用プロセッサ)LSI の開発”, ARC 研究会報告, pp.133-138, 2003-03.
- [8] CP2K マニュアル, <https://manual.cp2k.org/>
- [9] Xilinx マニュアル “Get Moving with Alveo”, UG1352, 2019.