Vol.2018-MPS-118 No.36
Vol.2018-BIO-54 No.36
2018/6/15

蛍光X線ホログラムからの原子像再生の並列化

窪田 昌史[1,a)]　松下 智裕[2]　八方 直久[1]

概要：蛍光 X 線ホログラフィーは，特定元素周辺の数 nm にわたり 3 次元原子配列を再生できる 3 次元中距離局所構造解析技術として注目されている．本手法では計算機を用いた 3 次元原子像再生に長時間を要することが問題となっているため，本研究では，その処理の並列化による高速化を試みた．179x360 点の入力ホログラムデータから 3 次元の各軸 192 点の直交座標の原子像を求める処理を 8 ノード 96 コアの Xeon X5660 2.8GHz の PC クラスタ上で実行したところ，逐次実行の 94 倍まで高速化された．

Parallelization of Atomic Image Reconstruction from X-ray Fluorescence Holograms

Abstract: X-ray fluorescence holography is a three-dimensional middle range local structural analysis method, which can provide three-dimensional atomic images around specific elements within a radius of a few nanometers. Presently, it is an issue that it takes long time to reconstruct three-dimensional atomic images on computers. In this study, the reconstruction is parallelized to reduce the processing time. The reconstruction, whose input is a hologram data of 179×360 points and output is a three-dimensional atomic image of 192^3 points, is executed on the PC cluster which consists of 8 nodes of Intel Xeon X5660 processors and 96 cores in total and we confirmed that the parallelized reconstruction is 94 times faster than the sequential execution.

1. はじめに

蛍光 X 線ホログラフィーは，特定元素周辺の 3 次元原子配列をホログラムとして記録することができる構造解析法として注目されている．原子レベル構造解析手法としては X 線回折法や X 線吸収微細構造法 (XAFS) などが確立されているが，これらの従来手法に比べた蛍光 X 線ホログラフィーの特徴として，数 nm にわたる広い範囲の局所構造を 3 次元原子配列として可視化できることが挙げられる．また，原子の本来のサイト (理想位置) からのずれに対して非常に敏感であり，再生された原子像を詳細に解析することにより，局所的な格子歪みに対する定量的な情報が得られるなどの特徴もある．

科研費新学術領域研究「3D 活性サイト科学」では，蛍光 X 線ホログラフィー，光電子ホログラフィー，CTR 散乱測定法，回折イメージングなどのコア測定技術を手法班が提供し，第一原理計算などによる理論計算や原子像再生アルゴリズムを理論班が担い，加えて試料班，応用班なども連携し，物質の中で機能発現を担う特殊な構造を持つ原子である「活性サイト」に関する研究を行っている．蛍光 X 線ホログラフィーや光電子ホログラフィーといった手法では，SPring-8 や KEK-PF といった大規模実験施設で得られた実験データに対し，Barton 法 [1], [2] に基づく 3 次元原子像再生処理が行なわれてきた．

しかし，Barton 法では，実験データであるホログラムデータから原子像を再生する処理に長時間を要していることが問題となっている．そこで我々は，マルチノード構成の PC クラスタやスーパーコンピュータ上でこの処理を並列実行することで，原子像再生の高速化を試みている [3], [4], [5].

PC クラスタやスーパーコンピュータなどの分散メモリ向けには，C 言語や Fortran 言語のプログラムについて，ノード間は Message Passing interface(MPI)[6] のライブラリ呼び出し，ノード内は並列処理 API である OpenMP[7] による並列化を行うプログラミングスタイルが広く用いられている．しかしながら，MPI のプログラムは記述が冗長

[1] 広島市立大学
Hiroshima City University
[2] (公財) 高輝度光科学研究センター
Japan Synchrotron Radiation Research Institute
[a)] kubota@hiroshima-cu.ac.jp

Vol.2018-MPS-118 No.36
Vol.2018-BIO-54 No.36
2018/6/15

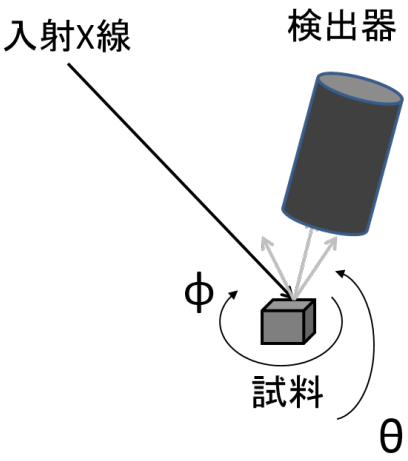

図 1　インバースモード

であること，間違いを犯しやすいことなどの問題点が指摘されている．そのため，並列プログラミングの生産性向上を目的として様々な提案がなされてきた．

Fortran 言語や C 言語については，High Performance Fortran(HPF)[8] や，Fortran2008 で採用された CoArray[9]，C 言語をベースとした UPC[10]，指示文によってデータの分散とループの並列化を行う XcalableMP[11] などがある．また，X10[12] や Chapel[13] などの並列プログラミング言語の提案もある．

本研究では，C 言語で記述された既存の Barton 法の原子像再生プログラムからの修正が容易であることから，ノード間を XcalableMP，ノード内を OpenMP によって並列化することとした．

以下，2 章で蛍光 X 線ホログラフィー法とその 3 次元原子像再生法について説明する．次いで，3 章で 3 次元原子像再生の並列化について述べ，4 章でその実行結果を示す．5 章で今後の研究の展開について触れ，6 章でまとめる．

2. 蛍光 X 線ホログラフィー

2.1 蛍光 X 線ホログラフィー法の概要

蛍光 X 線ホログラフィーでは，ノーマルモードとインバースモードの 2 種類が存在し，近年の実験では主にインバースモードが用いられているため，本稿でもインバースモードを中心に説明する．詳しくは解説記事 [14] などを参照されたい．インバースモードでは，**図 1** に示すように入射 X 線に対して試料の角度を変化させ，試料中の原子からの蛍光 X 線の強度を検出器で測定する．

図 2 に示すように，原子 A に近づく入射X線と，原子 B によって散乱された後で原子 A に近づく入射 X 線とが原子 A の周辺で X 線定在波を形成する．この X 線定在波のパターンは，入射 X 線の方位によって変化し，結果として，原子 A からの蛍光 X 線の強度変化をもたらす．この蛍光 X 線の強度を測定した実験データが，原子像のホログラムとなる．

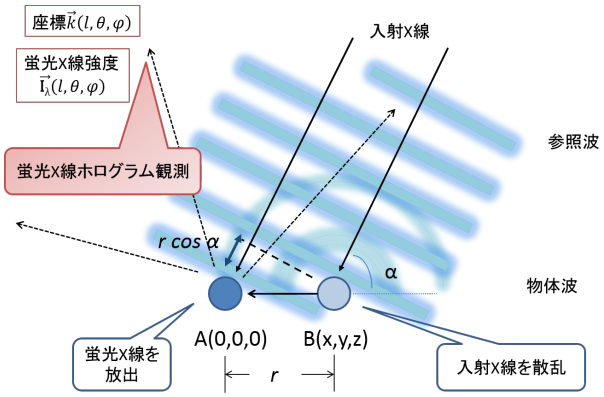

図 2　インバースモードの光路差

通常のホログラムにあてはめると，原子 A に直接近づく入射 X 線の波が参照波，原子 B によって散乱された後で原子 A に近づく入射 X 線の波が物体波の役割を果たしていることになる．

2.2 原子像再生

波長 λ の入射 X 線によって得られる球面上のホログラムの極座標 $\boldsymbol{k}(l,\theta,\phi)$ における値を $I_\lambda(l,\theta,\phi)$ としたときの試料中の原子の位置は以下のように求められる．

試料中の原点である原子 A から見て，原子 B が $\boldsymbol{r}(x,y,z)$ の位置にあるとすると，その原子像は 3 次元離散フーリエ変換 (DFT) と同様の処理で再生できる．この際，数種類の波長のホログラムから原子像を再生することで，真の原子像のみを強調させ，ゴーストイメージを低減できることが知られている [14]．我々は，処理時間を短縮するため，ホログラムの球面の半径 l を固定して以下の式 (1) のように原子像を求めるようにしている．

$$\chi(x,y,z) = \sum_{\theta}\sum_{\phi}\sum_{\lambda} -I_\lambda(\theta,\phi)\exp(i2\pi(|\boldsymbol{r}| - \boldsymbol{kr})/\lambda) \times \sin\theta\omega(\lambda) \quad (1)$$

ここで $\omega(\lambda)$ は，数種類の波長のホログラムからの原子像を重ね合わせる際の，波長 λ の入力 X 線から得られるホログラムの重みである．

入力データは倍精度浮動小数点形式の実数としてファイルに保存されている．球面上のデータの位置を極座標で表し，$\theta = 1^\circ$から　179°，　$\phi = 0^\circ$から　359°を，それぞれ 1 度きざみとした格子上のホログラムデータを入力とする．出力データである原子像は直交座標系の x, y, z それぞれ-10.0 Åから 10.0 Å程度の範囲で 0.1 Åきざみの格子点上の複素数値であり，倍精度浮動小数点形式で出力ファイルに保存される．今回は，4 章で述べるように，使用する PC クラスタ上での並列化の容易さを考慮し，直交座標系で-9.6 Åから 9.6 Åの範囲で 0.1 Å間隔，各軸 192 点をとっている．

Vol.2018-MPS-118 No.36
Vol.2018-BIO-54 No.36
2018/6/15

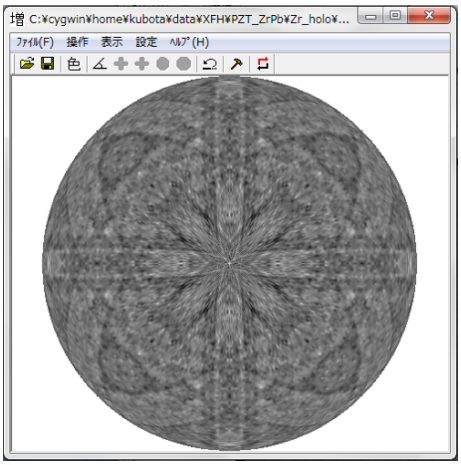

図 3 実験データから得られたホログラム例

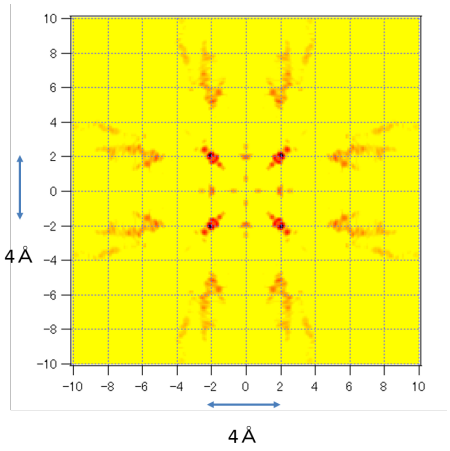

図 4 2 次元原子像の再生例
チタン酸ジルコン酸鉛 (PZT) を Zr を中心として測定し，Z=2.0 の xy 平面を再生

入力データが極座標系上，出力データが直交座標系上の格子点値であるため，式 (1) に現れる複素指数関数，つまり三角関数の引数が複雑であり，離散フーリエ変換と同様の処理ではあるものの，高速フーリエ変換 (FFT) を適用することは難しい．そのため，この計算には長時間を要することになる．

本来は，実験で得られた測定データであるホログラムから 3 次元の原子像の再生が可能であるが，再生には長時間を要し，通常の PC では数日かかるものと見積もっている．そのため，現状では 2 次元像を選択的に再生することも行われている．おおよその結晶構造は他の測定方法によって確認できているため，結晶格子の原子間の間隔は既知とすることができる．たとえば，原子間の間隔が 2 Å であれば，z = -4 Å, -2 Å, 0, 2 Å, 4 Å などの xy 平面の 2 次元像を再生することで，3 次元像を推定している．再生した 2 次元原子像を確認し，必要に応じて z = 1.9 Å, 2.1 Å などの 2 次元像を再生し，活性サイト近辺の原子位置のずれを確認することになる．そのため，従来の実験データの解析では，2 次元像を再生し，像を確認した上で追加の 2 次元像を再生するという手順を繰り返しており，解析に時間を要することが問題となっている．

2.3 蛍光 X 線ホログラフィー法の解析手順

蛍光 X 線ホログラフィー法では，以下に示すような手順で実験データを解析する．実験データから原子像を再生する際に時間を要するのは原子像再生であるが，その前後の処理も必要となっている．

(1) 実験
(2) バックグラウンド処理
(3) 完球処理
(4) 原子像再生
(5) 原子像表示

実験データにバックグラウンドと呼ばれる低周波のノイズが含まれていることが経験的に知られているため，原子像再生前に除去している．また，実験では，図 1 の θ を 0 ˚から 75 ˚程度になるように試料を傾け，各 θ について試料を回転させて ϕ を 0 ˚から 360 ˚まで変化させ，試料から発せられる蛍光 X 線の強度を計測する．θ を 75 ˚より大きな角度で測定しない理由は，試料の表面と入射 X 線が平行に近くなり有用なデータが得られなくなるためである．そのため，θ を 0 ˚から 180 ˚として全球面上のホログラムを得るため，結晶構造の対象性を利用してデータを補完する完球処理と呼ばれる処理を行っている．**図 3** は，これらバックグラウンド処理や完球処理といった前処理を行って得られた球面上のホログラム例である．

再生された原子像は，x,y,z の直交座標系の各格子点の原子が存在する確率に比例した値が得られるに過ぎない．そのため，原子の存在する確率が低い格子点の値はノイズであり，現在は画面にグラフィカルに表示された値を見ながら手動でしきい値を設定して，**図 4** のような再生画像を得ている．

3. 3 次元原子像再生の並列化

前章で述べたように，従来の実験データの解析では，2 次元像を再生し，像を確認した上で追加の 2 次元像を再生するという手順を繰り返しており，解析に時間を要することが問題となっている．そのため，本研究では，マルチノード構成の PC クラスタやスーパーコンピュータ上で並列プログラミング言語 XcalableMP や並列処理 API である OpenMP を用いてこの原子像再生を並列化することにより高速化を試みた．

OpenMP[7] は，逐次プログラムに指示文を挿入することで並列化を行うことができる並列プログラミング API である．特に，データ並列性を持つループを容易に並列化し，共有メモリ型並列計算機上で実行することができる．一方，XcalableMP[11] は，OpenMP と同様に逐次プログ

ラムに指示文を挿入することで並列化を行うことができる並列プログラミング言語であるが，特に，マルチノードの分散メモリ環境下での並列実行が可能なように，分散メモリへの配列の分割配置が指定できるという特徴を持つ．OpenMP, XcalableMP ともに，逐次プログラム向けのコンパイル時は，ほとんどの並列化の指示文をコメントとして無視できるように設計されている．そのため，逐次プログラムと並列プログラムを単一のソースコードとして管理することができるという利点がある．また，XcalableMP と OpenMP を併用したハイブリッド並列化も可能である．

本章では，以下の段階に分けて原子像再生の並列化について説明する．

(1) 2 次元原子像の再生を OpenMP を用いて PC の 1 ノードで並列化
(2) 3 次元原子像の再生を XcalableMP と OpenMP を用いてマルチノードで並列化

3.1 2 次元原子像の OpenMP による並列化

2 次元像の再生では，多重ループ内で繰り返し計算される三角関数を配列参照に置き換えた．そのため，多重ループに入る前に必要な三角関数の値を計算しておき，配列に保存しておく．

また，従来は式 (1) に従い，z を固定した xy 平面の 2 次元原子像を再生する場合は，x, y, θ, ϕ, λ の順に外側ループから内側ループへとネストした 5 重ループとしていた．なお，本稿では，ある配列の次元，たとえば配列の θ の次元の配列要素を連続してアクセスするループを，単に θ のループなどと表記する．

今回，この 5 重ループに対してループ交換を行って λ, x, y, θ, ϕ の順に変更した．これは，λ,θ, ϕ 方向の 3 次元からなる入力データ配列が 10MB 程度のサイズとなり，プロセッサの 2 次キャッシュにのりきらなくなる恐れがあるため，λ を固定した θ, ϕ の 500KB 程度の配列要素がキャッシュにのり，x, y, θ, ϕ の 4 重ループで繰り返し参照されるようになることを意図したものである．また，最内ループのループ長が，λ のループでは高々 20 程度であるのに対し，ϕ のループでは 360 まで長くなるため，Intel の AVX といった汎用プロセッサの SIMD 命令を適用しやすくなることも期待している．

これらの最適化を行った上で，この 5 重ループの x のループを OpenMP 指示文を挿入して並列化することで，2 次元原子像の再生の高速化を行う．

3.2 3 次元原子像の XcalableMP による並列化

3 次元原子像の再生では，高速化と OpenMP による並列化を行った 2 次元原子像の再生コードを拡張する形で，式 (1) を外側から順に λ, z, x, y, θ, ϕ の 6 重ループとしている．この 6 重ループに入る前に，三角関数の値を計算しておくことも，2 次元の場合と同様である．

z, x, y の 3 つのループうち，外側の z のループを XcalableMP によるマルチノードの並列化，その 1 つ内側の x のループを OpenMP によるノード内の並列化の対象とする．

z のループを XcalableMP を用いて並列化し，ブロック分割した．入力データは，各ノードですべてのデータをファイルから読み込む．原子像再生では z 方向に分散された原子像データを計算し，最後に原子像データを 1 ノードに集約し，3 次元原子像データとしてファイルに出力する．

4. 評価

本章では，前章で説明した 2 次元原子像の再生の OpenMP を用いた 1 ノードでの並列化と，3 次元原子像の再生の XcalableMP と OpenMP を用いたマルチノードでの並列化に対応して，その実行結果を示す．また，マルチノードの並列化の手段として，XcalableMP と MPI を比較した結果についても示す．

使用した PC クラスタは 8 ノードで構成され，各ノードには 6 コアをもつ Xeon X5660 2.8GHz を 2 基と 24GB のメインメモリを搭載している．ノード間は InfiniBand DDR(4Gbps) と Gibabit Ethernet で接続されている．Xeon X5660 のスマートキャッシュの容量は 12MB である．使用したコンパイラなどのソフトウェアは XcalableMP 1.2.2, Intel Compiler 18.0.1, 最適化オプションは -O3 -XHOST である．

4.1 2 次元原子像再生の実行結果

2 次元原子像再生のプログラムを PC クラスタの 1 ノードを用いて実行した．入力データは，入力 X 線波長が 21 種類の試料 (チタン酸ジルコン酸鉛; PZT) に対する実験データであり，$\theta = 1$ ˚から　179 ˚，　$\phi = 0$ ˚から　359 ˚でそれぞれ 1 ˚きざみである．出力データは　12 コアでの並列化が容易となるように，z を固定した [x][y]=[192][192] の 2 次元配列としている．

表 1 に示すように，オリジナルの実行時間 972.473 秒に対し，三角関数の配列参照とループ交換によって 914.301 秒に高速化された．さらにプログラムを OpenMP によって並列化し，1 ノード内の 6 コアの Xeon2 基計 12 コアを使用するように 12 スレッドで実行した場合に 75.814 秒まで高速化された．オリジナルに対する性能向上率は 12.8 倍である．

以下，ループ交換の効果について考察してみる．入力データの 3 次元倍精度浮動小数点配列 (λ, θ, ϕ) のサイズは 10MB 程度となる．ループ交換前は，x, y を外側ループとし，内側の λ, θ, ϕ の 3 重ループ内で 10MB の入力データを繰り返し参照することになるが，スマートキャッシュが 12MB であるため，使用しているデータがキャッシュからあふれ，キャッシュミスが頻発していると推測している．

Vol.2018-MPS-118 No.36
Vol.2018-BIO-54 No.36
2018/6/15

表 1　OpenMP による 2 次元原子像の並列化

オリジナル (s)	配列参照・ループ交換 (s)	OpenMP(s) 12 スレッド	性能向上率
972.473	914.301	75.814	12.8

これに対し，ループ交換によって λ のループを最外ループとすることで，θ,ϕ の内側ループでは入力データのうち 500KB 程度が繰り返し参照されることになる．これはスマートキャッシュには十分おさまることになる．

4.2　3 次元原子像再生の実行結果

3 次元像の再生では，先にのべた 2 次元像再生の実行結果をもとに，λ, z, x, y, θ, ϕ の 6 重ループで実行する．

入力データは $[\lambda][\theta][\phi] = [21][179][360]$ の 3 次元配列で変わらず，出力データはノード数 8 とノード内コア数 12 を考慮し，並列化が容易となるように $[z][x][y] = [192][192][192]$ の 3 次元配列としている．PC クラスタの各ノードに z 次元を分割して配置する．

この PC クラスタ上で，z のループを XcalableMP で並列化し，x のループを OpenMP で並列化して実行した結果を**表 2** と**表 3** に示す．出力のデータサイズを $[z][x][y] = [192][192][192]$ とした場合，逐次実行では長時間を要することが予想されたため，表 2 に示すとおり，総スレッド 8 以上でのみ実行した．この表のスレッド数の列は，総スレッド数 (ノード数×ノードあたりのスレッド数) を表している．それぞれ，全実行時間に加え，入力データをファイルから読み込む時間，ノード間に分散された計算結果を 1 ノードに集約する時間，集約したデータをファイルに出力する時間も示している．

また，z=-0.4 から 0.3 までの 0.1 きざみの 8 スライスの xy 平面の原子像再生のみを実行し，同様に全体の実行時間や入出力，データ集約の時間を示したものが表 3 である．逐次実行でも数時間で終了するため，1 スレッドから 96 スレッドまでスレッド数を変化させて実行した．

z 方向に 192 スライス，あるいは 8 スライス実行した表 2 と表 3 の性能向上率を**図 5** に示す．このグラフは両軸とも対数となっている．Z8 のグラフは表 3 の 1 スレッドの実行に対する性能向上率となっている．Z192 のグラフは 8 スレッドの実行に対する性能向上率に 8 を乗じたものである．どちらも理想的な linear の直線に近い結果が得られており，96 スレッド実行時の性能向上率の値は Z8 が 94.21，Z192 が 94.23 である．

入力データや出力データのサイズが比較的小さいため，実行時間に対するファイル入出力の時間の割合は非常に小さい．また，XcalableMP の gmove を用いて計算結果を 1 ノードに集約する処理時間も非常に短いといえる．そのため，並列ファイル出力を行うことなく高い性能を得ることができている．

表 2　XcalableMP による z:192 の並列化の実行結果

スレッド数	実行時間 (s)	入力 (s)	集約 (s)	出力 (s)
8(8x1)	21,683.076	0.781	0.176	9.772
48(8x6)	3,623.352	0.721	0.163	9.731
96(8x12)	1,840.942	0.767	0.174	9.701

表 3　XcalableMP による z:8 の並列化の実行結果

スレッド数	実行時間 (s)	入力 (s)	集約 (s)	出力 (s)
1(1x1)	7,214.325	0.745	0.007	0.301
8(8x1)	923.830	0.473	0.009	0.303
12(1x12)	603.962	0.459	0.005	0.310
48(8x6)	151.574	0.458	0.008	0.303
96(8x12)	76.576	0.455	0.009	0.310

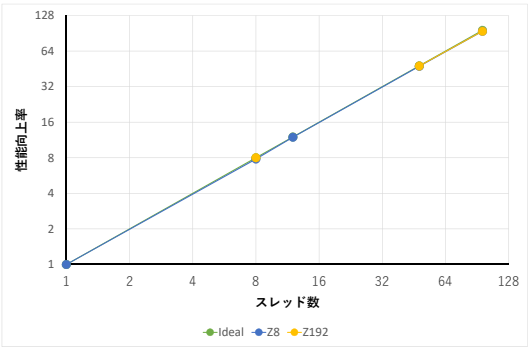

図 5　XcalableMP による並列化の性能向上率

表 4　XcalableMP と MPI による実行時間の相違 (z:192, 96 スレッド)

並列化	実行時間 (s)	追加修正行数
XcalableMP	1,840.942	32
MPI	1,817.042	53

4.3　MPI による並列化との比較

並列プログラミングにおける生産性の比較のため，XcalableMP と同様の並列処理を行うプログラムを MPI でも記述した．すでに OpenMP で並列化された 350 行の C 言語のプログラムに対し，マルチノードの並列化のために追加または変更した行数は，XcalableMP では 32 行，MPI では 53 行となっており，XcalableMP の方が追加または変更した行数は少ない．

表 4 に $[z][x][y] = [192][192][192]$ として，8 ノードの PC クラスタで総スレッド数 96 での XcalableMP と MPI の実行時間と，並列化に必要とした追加修正行数をまとめている．実行時間を比較しても，大きな違いはない．計算結果のデータ集約を行う XcalableMP の gmove 文にあたる処理を MPI では MPI_Gatherv 関数を用いて記述しているが，XcalableMP の処理系が内部で適切に MPI の関数へと変換しているもの考えられる．

5. 今後の課題

2.3 節で述べたように，再生された原子像は図 4 のように表示する際に，手動でしきい値を設定してノイズを除去している．ノイズの除去については，現在，L1 正則化を用いた手法も試みている [15] が，計算量が多く，直交座標系の x, y, z の格子間隔を 0.1 Å よりも間引き，重ね合わせるホログラム数を削減することで，計算量を削減して計算している．そこで，大規模な PC クラスタなどで実行することにより，このような計算量の削減を行わずに L1 正則化を用いたノイズ除去を実現することを計画している．大規模な PC クラスタとしては，九州大学情報基盤研究開発センターなどに設置されている GPU クラスタの利用なども予定している．

6. おわりに

本稿では，3 次元原子配列の構造解析法である蛍光 X 線ホログラフィーの 3 次元原子像再生処理の並列化について述べた．PC クラスタやスーパーコンピュータの利用を前提として，ノード間は XcalableMP で，ノード内は OpenMP によって並列化を行った．入力 X 線波長が 21 種類のエネルギーの 179 × 360 点のホログラムデータから 3 次元の各軸 192 点の直交座標系上の原子像データを求める処理を，8 ノード 96 コアの Xeon X5660 2.8GHz の PC クラスタ上で 30 分強の 1,841 秒で実行することができた．逐次処理では数日を要すると見積もっていたが，PC クラスタ上でほぼコア数に比例する 94 倍までの高速化が達成され，実用的な実行時間まで高速化されたといえる．また，XcalableMP によるノード間並列の記述は，性能やプログラミングの生産性の点で有用であることを示した．

謝辞 本研究は一部，JSPS 科研費新学術領域研究 26105013 の助成を受けました．ここに謝意を表します．

参考文献

[1] Barton, J. J.: Photoelectron Holography, *Physical Review Leters*, Vol. 61, No. 12, pp. 1356–1359 (1988).
[2] Barton, J. J.: Removing Multiple Scattering and Twin Images from Holographic Images, *Physical Review Letters*, Vol. 67, No. 22, pp. 3106–3109 (1991).
[3] 窪田昌史，八方直久，松下智裕：蛍光 X 線ホログラムからの原子像再生の高速化，ハイパフォーマンスコンピューティングと計算科学シンポジウム (HPCS) 予稿集，pp. 95–95 (2015).
[4] Matsushita, T., Kubota, A., Happo, N., Akagi, K., Yoshinaga, N. and Hayashi, K.: Fast Calculation Algorithm Using Barton's Method for Reconstructing Three-Dimensional Atomic Images from X-ray Fluorescence Holograms, *Zeitschrift für Physicalishe Chemie*, Vol. 230, No. 4, pp. 449–456 (2016).
[5] 窪田昌史，松下智裕，八方直久：蛍光 X 線ホログラムからの 3 次元原子像再生の XcalableMP による並列化，情報処理学会論文誌プログラミング， Vol. 10, No. 1, pp. 30–30 (2017).
[6] Message Passing Interface Forum: *MPI: Message Passing Interface.*
[7] OpenMP Architecture Review Board: *OpenMP Application Program Interface.*
[8] High Performance Fortran Forum: *High Performance Fortran Language Specification (Ver. 2.0)* (1997).
[9] ISO/IEC JTC 1/SC 22/WG 5/N1830: *Fortran 2008 standard* (2010).
[10] UPC Consortium: *Berkeley UPC – Unified Parallel C.*
[11] XcalableMP Specification Working Group: *XcalableMP Language Specification.*
[12] Saraswat, V., Bloom, B., Peshansky, I., Tardieu, O. and Grove, D.: *X10 Language Specification Version 2.5* (2015).
[13] Cray Inc.: *The Chapel Parallel Programming Language.*
[14] 林 好一，八方直久，細川伸也：蛍光 X 線ホログラフィーによる局所格子歪みの評価，放射光，Vol. 26, No. 4, pp. 195–205 (2013).
[15] Matsushita, T.: Atomic Image Reconstruction from Atomic Resolution Holography Using L_1-regularized Linear Regression, *e-JSSNT*, Vol. 14, pp. 158–160 (2016).