

超並列クラスタにおける 3D-RISM への Volumetric 並列三次元 FFT の適用と性能評価

多田野 寛人^{†1} 高橋 大介^{†1} 佐藤 三久^{†1}
吉田 紀生^{†2} 丸山 豊^{†2} 平田 文男^{†2}

本論文では、3D-RISM 法に対して volumetric 並列三次元 FFT アルゴリズムを適用し、大規模並列クラスタ上における性能を評価する。3D-RISM 法の反復計算過程において三次元 FFT が必要となり、この部分が計算時間の大半を占める。volumetric 並列三次元 FFT アルゴリズムを適用した 3D-RISM 法のコードを T2K-Tsukuba システム上で実装し、同コードの性能を評価した。データ数が 256^3 の 3D-RISM に対して、T2K-Tsukuba システムの 2048 コアを用いて OpenMP/MPI ハイブリッド並列を行った結果、16 コアを用いた場合と比較して約 60 倍の速度向上率が得られた。

Application and Performance Evaluation of the Volumetric Parallel 3D-FFT to 3D-RISM on Massively Parallel Cluster

HIROTO TADANO,^{†1} DAISUKE TAKAHASHI,^{†1}
MITSUHISA SATO,^{†1} NORIO YOSHIDA,^{†2}
YUTAKA MARUYAMA^{†2} and FUMIO HIRATA^{†2}

In this paper, we apply the volumetric parallel 3D-FFT algorithm to the 3D-RISM method and evaluate its performance on massively parallel cluster. The iteration part of the 3D-RISM method requires a 3D-FFT. Most of the time for the iteration part of the method is spent on the 3D-FFT. We implemented the code of 3D-RISM with the volumetric parallel 3D-FFT algorithm and evaluated the performance of the code on the T2K-Tsukuba system. The elapsed time for 3D-RISM calculation with 256^3 grid points on the T2K-Tsukuba system with 2048 cores and with OpenMP/MPI hybrid parallelization was about 60 times faster than the elapsed time on the system with 16 cores and with the same parallelization method.

1. はじめに

液体の状態や構造を記述する理論として、積分方程式理論が提案されている。たんぱく質や DNA、ナノチューブなどの巨大で複雑な分子の溶媒構造を取り扱うことのできる積分方程式理論として、three-dimensional reference interaction site model (3D-RISM) 法^{1),5)}がある。同法では、三次元空間を x 方向、 y 方向、および z 方向に分割して格子点上で計算を行う。各格子点上における計算はそれぞれ独立に計算できるため、並列計算に適している。しかしながら、3D-RISM 法では三次元空間内の全データに対して三次元高速 Fourier 変換（以下、三次元 FFT）が必要となり、この部分が演算の大半を占める。並列化のアプローチとして、三次元空間の 1 方向（例えば z 方向）に分割して並列化するのが最も単純であるが、大規模並列環境においては最大でも z 方向のデータ数までしか並列化ができないという問題点がある。

並列度を向上させるアプローチとして、三次元空間を x 方向、 y 方向、 z 方向それぞれに分割することが考えられる。しかしながら、全方向に対して分割を行った場合、各方向への FFT を行う度に全対全通信を行う必要がある。近年、高橋は三次元空間を y 方向、 z 方向に分割し三次元 FFT を行う volumetric 並列三次元 FFT アルゴリズム³⁾を提案した。同アルゴリズムを用いることにより、全方向に分割して並列化した場合と比較して通信量を減少させることが可能となる。

本論文では、3D-RISM 法で必要となる三次元高速 Fourier 変換に対して、volumetric 並列三次元 FFT アルゴリズムを適用することを考える。また、マルチコア超並列クラスタである T2K-Tsukuba システム上で実装し、性能評価を行う。

本論文の構成は以下の通りである。次章では、3D-RISM 法の計算アルゴリズムについて簡単に述べる。第 3 章では、三次元空間を y 方向、 z 方向に分割して三次元高速 Fourier 変換を行う volumetric 並列三次元 FFT アルゴリズムについて述べる。第 4 章において、3D-RISM 法に volumetric 並列三次元 FFT アルゴリズムを適用し、T2K-Tsukuba システム上で性能評価を行う。最後にまとめと今後の課題を述べる。

^{†1} 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
^{†2} 分子科学研究所
Institute for Molecular Science

2. 3D-RISM 法

本章では、three-dimensional reference interaction site model (3D-RISM) 法の計算アルゴリズムについて、簡単に述べる。3D-RISM 法の計算では、レナード・ジョーンズポテンシャル、およびクーロンポテンシャルが必要となる。レナード・ジョーンズポテンシャルは、

$$u_{\gamma}^{(LJ)}(\mathbf{r}) = \sum_{\alpha}^{solute} \left\{ 4\varepsilon_{\alpha\gamma} \left[\left(\frac{\sigma_{\alpha\gamma}}{|\mathbf{r} - \mathbf{r}_{\alpha}|} \right)^{12} - \left(\frac{\sigma_{\alpha\gamma}}{|\mathbf{r} - \mathbf{r}_{\alpha}|} \right)^6 \right] \right\}$$

で求められる。ここで、 $\varepsilon_{\alpha\gamma}$ 、および $\sigma_{\alpha\gamma}$ はそれぞれ、レナード・ジョーンズ相互作用のエネルギー、および直径パラメータを表す。クーロンポテンシャルはエwald法を用いて

$$u_{\gamma}^{(es)}(\mathbf{r}) = \frac{1}{V_{Cell}} \sum_{k \neq 0} \sum_{\alpha} q_{\alpha}^u q_{\gamma}^v \frac{4\pi}{k^2} \exp\left(ikr_{\alpha\gamma} - \frac{k^2\eta^2}{4}\right) + \sum_{\alpha} q_{\alpha}^u q_{\gamma}^v \frac{\text{erfc}(|\mathbf{r} - \mathbf{r}_{\alpha}|/\eta)}{|\mathbf{r} - \mathbf{r}_{\alpha}|}$$

で与えられる。ここで、 i は虚数単位、 $r_{\alpha\gamma}$ は溶質 α のサイトと溶媒 γ の距離を表す。

3D-RISM 法では、レナード・ジョーンズポテンシャル、およびクーロンポテンシャルを用いて、以下の反復計算を行う。

Step 0 $\tau_{\gamma}(\mathbf{r})$ の初期値を決定する。

Step 1 $c_{\gamma}(\mathbf{r}) = \exp\left[-\beta\left(u_{\gamma}^{(LJ)}(\mathbf{r}) + u_{\gamma}^{(es)}(\mathbf{r}) + \tau_{\gamma}(\mathbf{r})\right)\right] - 1 - \tau_{\gamma}(\mathbf{r})$ 。

Step 2 $\tilde{c}_{\gamma}(\mathbf{r}) = \int c_{\gamma}(\mathbf{r}') e^{i\mathbf{k}\mathbf{r}} d\mathbf{r}'$ 。

Step 3 $\tilde{h}_{\gamma}(\mathbf{k}) = \sum_{\gamma'} \tilde{c}_{\gamma'}(\mathbf{k}) \tilde{X}_{\gamma\gamma'}(k)$ 。

Step 4 $h_{\gamma}(\mathbf{r}) = \int \tilde{h}_{\gamma}(\mathbf{k}) e^{-i\mathbf{k}\mathbf{r}} d\mathbf{k}$ 。

Step 5 $\tau_{\gamma}(\mathbf{r}) = h_{\gamma}(\mathbf{r}) - c_{\gamma}(\mathbf{r})$ 。

Step 6 外挿法により $\tau_{\gamma}(\mathbf{r})$ を更新し、**Step 1** に戻る。

Step 7 得られた結果から溶媒和自由エネルギーを

$$\Delta\mu = \sum_{\gamma} \frac{\rho}{\beta} \int \left(\frac{1}{2} h_{\gamma}(\mathbf{r})^2 - c_{\gamma}(\mathbf{r}) - \frac{1}{2} h_{\gamma}(\mathbf{r}) c_{\gamma}(\mathbf{r}) \right)$$

で評価する。

ここで、 β は温度の逆数、 $h_{\gamma}(\mathbf{r})$ は全相関関数、 $c_{\gamma}(\mathbf{r})$ は直接相関関数であり、 ρ は溶媒の数密度を表す。**Step 3** で現れる $\tilde{X}_{\gamma\gamma'}(k)$ は溶媒 - 溶媒相関関数を表し、反復計算前に事前に計算を行う。**Step 6** では、外挿法として Modified direct inversion in the iterative subspace (MDIIS) 法²⁾ が用いられ、 $\tau_{\gamma}(\mathbf{r})$ が更新される。上記アルゴリズムの **Step 1** から **Step 6** では反復計算が行われ、**Step 3** の畳み込み積分を評価するために、その前後の **Step2**、および **Step 4** において三次元 Fourier 変換が必要となる。3D-RISM 法では、**Step 1**~**Step 6** の反復部分が演算の大部分を占め、特に三次元 Fourier 変換に計算時間を要する。

3. volumetric 並列三次元 FFT

前章では、3D-RISM 法の反復計算部分において三次元高速 Fourier 変換が必要となり、この部分に多くの計算時間を要することを述べた。本章では大規模並列環境で並列度を向上させるための、volumetric 並列三次元 FFT アルゴリズム³⁾ について述べる。

データ数 $n = n_1 \times n_2 \times n_3$ の場合の三次元離散 Fourier 変換は、以下の式で定義される。

$$y(k_1, k_2, k_3) = \sum_{j_1=0}^{n_1-1} \sum_{j_2=0}^{n_2-1} \sum_{j_3=0}^{n_3-1} x(j_1, j_2, j_3) \omega_{n_3}^{j_3 k_3} \omega_{n_2}^{j_2 k_2} \omega_{n_1}^{j_1 k_1}. \quad (1)$$

但し、 $\omega_{n_r} = \exp(-2\pi i/n_r)$ ($1 \leq r \leq 3$) である。式 (1) は、文献⁴⁾ の multicolumn FFT アルゴリズムに基づく三次元 FFT で計算することができる。この三次元 FFT では、行列の転置が 3 回必要となる。

並列三次元 FFT において最も単純な配列の分散方法は、図 1 に示すような z 方向のみの分散である。この分散方法を用いた場合は、 z 方法のデータ数までの並列化しか行うことができず、大規模並列環境で並列度を上げることが困難である。

次に、配列を 2 方向に分散して計算を行う volumetric 並列三次元 FFT アルゴリズム³⁾ について述べる。volumetric 並列三次元 FFT アルゴリズムでは、図 2 に示すように入力データが y 方向、 z 方向の 2 方向に分散される。そのため、 z 方向分散のみの場合と比較して、並列度を上げることが容易である。

全データ数を $N = N_1 \times N_2 \times N_3$ とし、 N 個のデータを $P \times Q$ 個のプロセッサをもつメモリ分散型並列計算機で保持することを考える。このとき、 N 個のデータを格納している配列

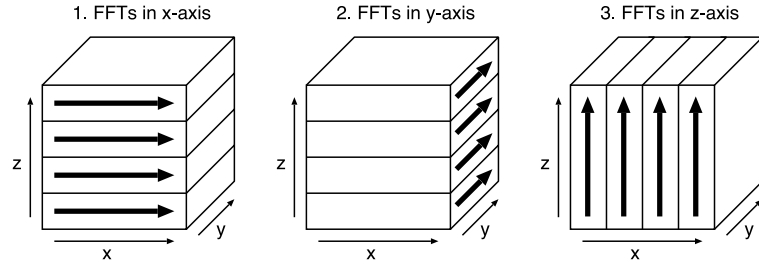


図1 一次元ブロック分割した場合の並列三次元 FFT

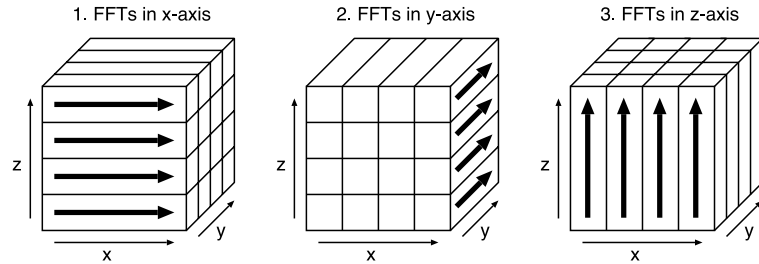


図2 二次元ブロック分割した場合の並列三次元 FFT

$x(N_1, N_2, N_3)$ は二次元目 (N_2)、および三次元目 (N_3) に沿って分割される。 N_2 が P で、 N_3 が Q で割り切れるとすると、各プロセッサに格納されるデータ数は $N_1 \cdot (N_2/P) \cdot (N_3/Q)$ 個となる。

分散された三次元配列を表すために、記法 $\hat{N}_r \equiv N_r/P$ 、および $\hat{\hat{N}}_r \equiv N_r/Q$ を定義する。また、インデックス J_r に沿ったデータが P 個のプロセッサに分散されることを示す記法を \hat{J}_r とし、同様に、インデックス J_r に沿ったデータが Q 個のプロセッサに分散されることを示す記法を $\hat{\hat{J}}_r$ とする。添字 r は、次元 r にインデックスが属していることを表す。

以上の記法を用いると、分散された三次元配列は $\hat{x}(N_1, \hat{N}_2, \hat{\hat{N}}_3)$ と表される。 y 方向で第 l 番目のプロセッサにおけるローカルインデックス $\hat{J}_r(l)$ と、グローバルインデックス J_r の間には以下の関係式が成り立つ。

$$J_r = \hat{J}_r(l) \times P + l, \quad 0 \leq l \leq P-1, \quad 1 \leq r \leq 3.$$

また、 z 方向で第 m 番目のプロセッサにおけるローカルインデックス $\hat{\hat{J}}_r(m)$ とグローバル

インデックス J_r の間には

$$J_r = \hat{\hat{J}}_r(m) \times Q + m, \quad 0 \leq m \leq Q-1, \quad 1 \leq r \leq 3$$

の関係が成り立つ。

全対全通信を表すために、記法 $\tilde{N}_i \equiv N_i/P_i$ 、および $\tilde{\tilde{N}}_i \equiv N_i/Q_i$ を定義する。これらの記号を用いると、 i 次元目のデータ数 N_i は \tilde{N}_i と P_i 、および $\tilde{\tilde{N}}_i$ と Q_i の二次元表現で表される。ここで、 P_i 、 Q_i の添字 i は、 P 、 Q が次元 i に属していることを示している。

初期データを $\hat{x}(N_1, \hat{N}_2, \hat{\hat{N}}_3)$ としたときの volumetric 並列三次元 FFT アルゴリズムは、以下で与えられる。

Step 1 $(N_2/P) \cdot (N_3/Q)$ 組の N_1 点 multicolumn FFT

$$\hat{x}_1(K_1, \hat{J}_2, \hat{\hat{J}}_3) = \sum_{J_1=0}^{N_1-1} \hat{x}(J_1, \hat{J}_2, \hat{\hat{J}}_3) \omega_{N_1}^{J_1 K_1}$$

Step 2 転置

$$\hat{x}_2(\hat{J}_2, \hat{\hat{J}}_3, \hat{K}_1, P_1) \equiv \hat{x}_2(\hat{J}_2, \hat{\hat{J}}_3, K_1) = \hat{x}_1(K_1, \hat{J}_2, \hat{\hat{J}}_3)$$

Step 3 y 方向の P 個のプロセッサ間で全対全通信を Q 組行う

$$\hat{x}_3(\hat{J}_2, \hat{\hat{J}}_3, \hat{K}_1, P_2) = \hat{x}_2(\hat{J}_2, \hat{\hat{J}}_3, \hat{K}_1, P_1)$$

Step 4 プロセッサ内再配置

$$\hat{x}_4(J_2, \hat{\hat{J}}_3, \hat{K}_1) \equiv \hat{x}_4(\hat{J}_2, P_2, \hat{\hat{J}}_3, \hat{K}_1) = \hat{x}_3(\hat{J}_2, \hat{\hat{J}}_3, \hat{K}_1, P_2)$$

Step 5 $(N_3/Q) \cdot (N_1/P)$ 組の N_2 点 multicolumn FFT

$$\hat{x}_5(K_2, \hat{\hat{J}}_3, \hat{K}_1) = \sum_{J_2=0}^{N_2-1} \hat{x}_4(J_2, \hat{\hat{J}}_3, \hat{K}_1) \omega_{N_2}^{J_2 K_2}$$

Step 6 転置

$$\hat{x}_6(\hat{\hat{J}}_3, \hat{K}_1, \tilde{\tilde{K}}_2, Q_2) \equiv \hat{x}_6(\hat{\hat{J}}_3, \hat{K}_1, K_2) = \hat{x}_5(K_2, \hat{\hat{J}}_3, \hat{K}_1)$$

Step 7 z 方向の Q 個のプロセッサ間で全対全通信を P 組行う

$$\hat{x}_7(\tilde{\tilde{J}}_3, \hat{K}_1, \hat{K}_2, Q_3) = \hat{x}_6(\hat{\hat{J}}_3, \hat{K}_1, \tilde{\tilde{K}}_2, Q_2)$$

Step 8 プロセッサ内再配置

$$\hat{x}_8(J_3, \hat{K}_1, \hat{K}_2) \equiv \hat{x}_8(\tilde{\tilde{J}}_3, Q_3, \hat{K}_1, \hat{K}_2) = \hat{x}_7(\tilde{\tilde{J}}_3, \hat{K}_1, \hat{K}_2, Q_3)$$

Step 9 $(N_1/P) \cdot (N_2/Q)$ 組の N_3 点 multicolumn FFT

$$\hat{x}_9(K_3, \hat{K}_1, \hat{K}_2) = \sum_{J_3=0}^{N_3-1} \hat{x}_8(J_3, \hat{K}_1, \hat{K}_2) \omega_{N_3}^{J_3 K_3}$$

Step 10 転置

$$\hat{x}_{10}(\hat{K}_1, \hat{K}_2, K_3) = \hat{x}_9(K_3, \hat{K}_1, \hat{K}_2)$$

Step 11 z 方向の Q 個のプロセッサ間で全対全通信を P 組行う

$$\hat{x}_{11}(\hat{K}_1, \tilde{K}_2, \hat{K}_3, Q_2) = \hat{x}_{10}(\hat{K}_1, \hat{K}_2, \tilde{K}_3, Q_3) \equiv \hat{x}_{10}(\hat{K}_1, \hat{K}_2, K_3)$$

Step 12 プロセッサ内再配置

$$\hat{x}_{12}(\hat{K}_1, \hat{K}_3, K_2) \equiv \hat{x}_{12}(\hat{K}_1, \hat{K}_3, \tilde{K}_2, Q_2) = \hat{x}_{11}(\hat{K}_1, \tilde{K}_2, \hat{K}_3, Q_2)$$

Step 13 y 方向の P 個のプロセッサ間で全対全通信を Q 組行う

$$\hat{x}_{13}(\tilde{K}_1, \hat{K}_3, \hat{K}_2, P_1) = \hat{x}_{12}(\hat{K}_1, \hat{K}_3, \tilde{K}_2, P_2) \equiv \hat{x}_{12}(\hat{K}_1, \hat{K}_3, K_2)$$

Step 14 プロセッサ内再配置

$$\hat{y}(K_1, \hat{K}_2, \hat{K}_3) \equiv \hat{y}(\tilde{K}_1, P_1, \hat{K}_2, \hat{K}_3) = \hat{x}_{13}(\tilde{K}_1, \hat{K}_3, \hat{K}_2, P_1)$$

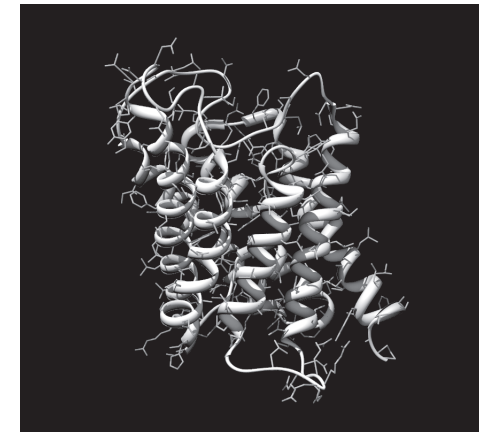


図 3 アクアポリンの構造図

以上の手順を行うことにより、三次元 FFT を計算することができる。プロセッサ間の全対全通信は、Step 3, 7, 11, および 13 の 4 回行う必要がある。 x 方向, y 方向, および z 方向全てに分割した場合、全部で 6 回の全対全通信を行う必要があるが、volumetric 並列三次元 FFT では x 方向は分割されていないため、全対全通信を 2 回減らすことができる。

上記のアルゴリズムでは、入力・出力がともに y 方向, z 方向に分割されたものになるが、**Step 10** で計算を停止すると出力は x 方向, y 方向に分割されたものが得られる。volumetric 並列三次元 FFT を用いるアプリケーションにおいて、正変換後は x 方向, y 方向に分割された状態で計算を行い、逆変換後は y 方向, z 方向に分割された状態で処理を行うことが可能な場合、全対全通信の回数を半分に減少させることができる。

4. 性能評価

本章では、 y 方向, z 方向に分割して計算を行う 3D-RISM コードを実装し、同コードに対して volumetric 並列三次元 FFT アルゴリズムを適用したときの性能を評価する。3D-RISM で必要となる FFT のデータは全て実数であるため、実数 FFT を用いることによりデータ量を半分に減らすことができる。しかしながら、この場合は三次元 FFT の全対全通信の他にデータを生成するための通信が必要となる。そのため、FFT を行うデータのみを倍精度複素数型で保持することで、通信量を減少させた。

評価実験では、水分子の透過に関係したたんぱく質であるアクアポリンを対象とした。図 3 にアクアポリンの構造図を示す。 x 方向, y 方向, および z 方向への分割数は 256 とした。そのため、三次元 FFT で扱う全データ数は $256^3 = 16,777,216$ である。

実験はマルチコア超並列クラスタである T2K-Tsukuba システムを用いた。T2K-Tsukuba システムは、ApproXtremeServer-X3 Server 648 ノードから構成され、10,368 個のコアを持つ。ノード間は multi-rail の InfiniBand を用いた Fat Tree 相互接続網によって接続されている。通信ライブラリとして、MVAPICH2 1.2RC2 を用いた。コンパイラは Intel Fortran 10.1 を使い、コンパイルオプションは `-O3 -x0` とした。

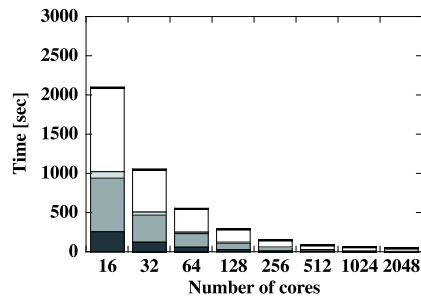
T2K-Tsukuba システムの各ノードは、4 つの Quad-Core AMD Opteron 8356 2.3GHz から構成されており、1 つのソケット内を OpenMP で並列化するハイブリッド並列を用いることも可能である。本論文では、ソケット内を MPI で並列化を行う flat MPI と、ソケット内を OpenMP で並列化したハイブリッド並列の比較を行う。Quad-Core AMD Opteron は NUMA アーキテクチャであるため、`numactl` を用いることにより使用するソケット、およびメモリ割り当てをコントロールすることができる。実験で用いた flat MPI 用スクリプト、および OpenMP/MPI ハイブリッド並列用スクリプトを、それぞれ図 4, 5 に示す。これらのスクリプトを用いて、使用コア数を変化させた場合の 3D-RISM 法の計算時間変化を図 6 に示す。グラフには 3D-RISM 法における、レナード・ジョーンズポテンシャル計算、

```
#!/bin/bash
RANK=$PMI_ID
VAL=$(( $RANK / 4 ))
SOCKET=$(( $VAL % 4 ))
numactl --cpunodebind=\$SOCKET \
--membind=\$SOCKET \@$@
```

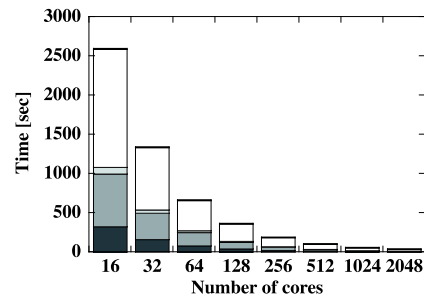
図 4 flat MPI 用スクリプト

```
#!/bin/bash
RANK=$PMI_ID
SOCKET=$(( $RANK % 4 ))
numactl --cpunodebind=\$SOCKET \
--membind=\$SOCKET \@$@
```

図 5 OpenMP/MPI ハイブリッド並列用スクリプト



(a) flat MPI



(b) OpenMP/MPI ハイブリッド並列

図 6 使用コア数に対する 3D-RISM 法の計算時間変化と割合 (■: レナード・ジョーンズポテンシャル計算, ■: クーロンポテンシャル計算, ■: $\tau_\gamma(\mathbf{r})$ の初期値計算, □: Step 1 ~ Step 6 の反復計算, ■: その他)

クーロンポテンシャル計算, $\tau_\gamma(\mathbf{r})$ の初期値計算, Step 1 ~ Step 6 の反復計算部分, およびその他の時間を記載した. 3D-RISM 法の反復計算前に必要な溶媒-溶媒相関関数の計算は, その他の中に含まれている.

図 6 に示すように, 3D-RISM 法で多くの割合を占めるのは Step 1 ~ Step 6 の反復計算部分であり, この部分において三次元 FFT が多用されている. 次いで, レナード・ジョーンズポテンシャル計算, およびクーロンポテンシャル計算も多くの割合を占める.

表 1 に使用コア数に対する 3D-RISM 法の計算時間変化を示す. 使用コア数が 16 の場合, flat MPI を用いることにより OpenMP/MPI ハイブリッド並列と比較して約 500 秒高速に計算が終了した. 使用コア数を 512 まで増加させた場合でも, この傾向は変わらなかった. しかしながら, 使用コア数が 1024, および 2048 の場合は OpenMP/MPI ハイブリッド並列が flat MPI よりも高速になった.

表 1 使用コア数に対する 3D-RISM 法の計算時間変化 (秒)

#cores	flat MPI	OpenMP/MPI
16	2106.8	2599.6
32	1060.6	1344.2
64	559.7	668.2
128	298.0	367.6
256	158.6	193.6
512	94.3	108.8
1024	72.1	62.5
2048	57.5	43.4

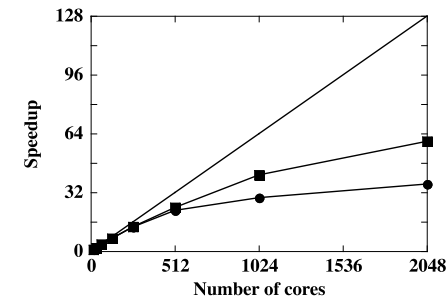


図 7 使用コア数に対する 3D-RISM 法の速度向上率 (●: flat MPI, ■: OpenMP/MPI ハイブリッド並列)

図 7 に, 使用コア数に対する 3D-RISM 法の速度向上率を示す. 同図は, 16 コアを用いた時の速度向上率を 1 として算出した. 16 コアから 512 コアを用いた計算においては, flat MPI, OpenMP/MPI ハイブリッド並列ともにほぼ同等の速度向上率を示した. 一方, 1024 コアを用いた場合の速度向上率は, flat MPI: 29.2, OpenMP/MPI ハイブリッド並列: 41.6 であり, 2048 コアを用いた場合は, flat MPI: 36.6, OpenMP/MPI ハイブリッド並列: 59.9 となった. 以上より, 並列度が高い場合は OpenMP/MPI ハイブリッド並列を用いることで高い速度向上率が得られた.

表 2 に, volumetric 並列三次元 FFT に要した全体時間と通信時間, および通信時間の割合を示す. 使用コア数が 512 までの場合は, 表 1 と同じく flat MPI の方が少ない時間で計算が終了している. しかしながら, 1024 コア, および 2048 コアを使用した場合は OpenMP/MPI ハイブリッド並列の方が時間が短くなっており, 特に通信時間が短縮されている. 通信時間の割合に着目すると, OpenMP/MPI ハイブリッド並列を用いた場合は,

表 2 volumetric 並列三次元 FFT に要した全体時間と通信時間 (秒), および通信時間の割合 (%)

#cores	flat MPI			OpenMP/MPI		
	全体時間	通信時間	(割合)	全体時間	通信時間	(割合)
16	658.0	284.6	(43.3)	736.9	362.4	(49.2)
32	339.1	151.9	(44.8)	382.6	193.9	(50.7)
64	185.3	91.7	(49.5)	205.2	108.5	(52.9)
128	103.0	55.6	(54.0)	116.2	62.8	(54.0)
256	51.4	30.7	(59.7)	63.6	38.0	(59.7)
512	33.2	24.5	(73.8)	37.8	23.2	(61.4)
1024	31.0	25.2	(81.3)	21.5	12.8	(59.5)
2048	24.3	15.1	(63.1)	15.7	9.5	(60.5)

使用コア数を増加させても通信時間の割合は約 60%で推移している。全対全通信において、通信量が小さい場合はレイテンシが支配的になると考えられるが、OpenMP/MPI ハイブリッド並列を用いることで flat MPI よりも通信 1 回あたりの通信量が増加する。そのため、並列度が高い場合はハイブリッド並列を用いることで、通信におけるレイテンシの影響を緩和できたと考えられる。

5. まとめと今後の課題

本論文では、3D-RISM 法に対して、 y 方向、および z 方向に分散して計算を行う volumetric 並列三次元 FFT アルゴリズムを適用し、その性能を検証した。 z 方向のみに分散を行う場合は、 256^3 のデータ点に対して三次元 FFT は最大 256 並列までしか並列化ができないが、volumetric 並列三次元 FFT アルゴリズムを用いることにより、2048 コアを用いた計算も可能であることを確かめた。また、並列度が低い場合は flat MPI、並列度が高い場合は OpenMP/MPI ハイブリッド並列が有効であることが分かった。

3D-RISM 法で必要な溶媒 - 溶媒相関関数の計算部分は並列化がなされていないため、高並列計算においてはボトルネックになりつつある。この部分の並列化を行うことは今後の課題である。また、本論文における計算では volumetric 並列三次元 FFT の入力・出力が全て y 方向、 z 方向に分割されていたが、出力が x 方向、 y 方向に分割された状態で 3D-RISM 法の畳み込み積分を行うことにより、通信時間を半分に減少させることが可能である。通信量を減少させて更なる高速化を図ることも今後の課題の 1 つである。

参考文献

- 1) Kovalenko, A. and Hirata, F.: Self-consistent description of a metal-water interface by the Kohn-Sham density functional theory and the three-dimensional reference interaction site model, *J. Chem. Phys.*, Vol.110, No.20, pp.10095–10112 (1999).
- 2) Kovalenko, A., Ten-no, S. and Hirata, F.: Solution of three-dimensional reference interaction site model and hypernetted chain equations for simple point charge water by modified method of direct inversion in iterative subspace, *J. Comput. Chem.*, Vol.20, No.9, pp.928–936 (1999).
- 3) 高橋大介: マルチコア超並列クラスタにおける Volumetric 並列三次元 FFT の実現と評価, 情報処理学会 研究報告, Vol.2009-HPC-119, pp.19–24 (2009).
- 4) Van Loan, C.: *Computational Frameworks for the Fast Fourier Transform*, SIAM Press, Philadelphia, PA (1992).
- 5) Yoshida, N., Imai, T., Phongphanphane, S., Kovalenko, A. and Hirata, F.: Molecular Recognition in Biomolecules Studied by Statistical-Mechanical Integral-Equation Theory of Liquids, *J. Phys. Chem. B*, Vol.113, No.4, pp.873–886 (2009).