

TSUBAME ESJ.



マルチフェーズフィールド法を用いた 金属多結晶組織形成シミュレーションの 大規模 GPU 計算

Multi-GPU Computation of Multi-phase Field Simulation of the Evolution of
Metallic Polycrystalline Microstructure

量子モンテカルロ法に基づく 振動状態解析の大規模並列計算

A Large-scale Parallel Computation for Vibrational State
Analysis Based on Quantum Monte Carlo method

個別要素法による粉体の大規模シミュレーション

Large-scale DEM Simulations for Granular Dynamics

マルチフェーズフィールド法を用いた 金属多結晶組織形成シミュレーションの大規模GPU計算

中山 晃徳* 岡本 成史** 下川辺 隆史*** 青木 尊之***

*東京農工大学大学院工学研究院先端機械システム部門 **東京農工大学大学院工学府機械システム工学専攻 ***東京工業大学学術国際情報センター

金属材料の内部で生じるミクロな多結晶組織形成のシミュレーション法として注目されているマルチフェーズフィールド(MPF)法を複数のGPUで高速かつ大規模に並列計算する方法を開発した。本稿では、数値計算と並列処理のためのデータ通信を非同期実行するためのオーバーラッピング法を導入したMPF法の計算手法を紹介する。さらに、開発したGPU計算手法による多結晶粒成長の大規模3次元シミュレーションをGPUスーパーコンピューターTSUBAME2.5に実装し、良好な実行性能が得られたことを示す。

はじめに

1

地球環境に与える負荷の低減のために、ハイブリッド自動車などの燃費の良い自動車の開発が活発である。自動車の燃費をさらに向上させるためには、エンジンやモーターの効率向上の他に、車体の軽量化も重要な技術課題とされている。自動車車体の軽量化は、使用的する金属板材を薄肉化することにより可能であるが、衝突安全性を確保するために強度も高めが必要とされる。しかし一般には、金属材料は強度を高めると加工性は低下する。すなわち、薄くプレス加工する際に割れ易くなる。そのため従来の材料研究開発では、絨毯爆撃的な実験を行い、材料の特性を支配する材料内部のミクロな構造の空間分布や形態を制御するための適切な合金組成や製造プロセスを見つけ出す努力を重ねてきた。しかし近年では、材料研究開発の国際競争が激化しており、如何にして効率良く、無限の組み合せが存在する合金元素や製造プロセスの中から、所望の特性を具備する材料を作るための条件を見つけ出しが極めて重要となっている。

これに対して現在では、材料開発のコストや期間を飛躍的に短縮するために、計算材料学に基づく数値シミュレーションにより材料のミクロ組織や特性を予測する計算技術の開発が世界的に活発となっている。その中でも、材料中のミクロ組織を予測するための強力な数値シミュレーション方法としてフェーズフィールド法が注目されている。特に、実用金属材料のほとんどで観察される多結晶構造を有するミクロ組織の形成過程を計算するための方針として、マルチフェーズフィールド(MPF)法^[1]が世界標準となりつつある。しかしながら、2011年にゴードンベル賞を受賞した通常のフェーズフィールド法を用いた合金凝固の計算^[2]とは異なり、MPF法では複雑な非線形項を含んだ複数の偏微分方程式を解く必要があるため、数値演算とメモリの両面で計算コストが大きいことが問題である。

そこで我々の研究では、GPUを用いてMPF法を高速かつ大規模に並列計算する技術を開発している。本稿では、MPF法を多数のGPUを用いて計算するために必要なオーバーラッピング法^[3](数値

演算とデータ通信を非同期に行うアルゴリズム)と、それを東京工業大学学術国際情報センターのGPUスーパーコンピューターTSUBAME2.5に実装して実施した、金属多結晶組織形成の大規模3次元シミュレーションを紹介し、我々が開発したGPU計算法の実行性能評価を示す^[4]。

マルチフェーズフィールド法

2

本研究で使用したMPF法は、1999年にSteinbachらによって提案された^[1]。MPF法は、次式で表される材料の全自由エネルギーが単調に減少するようにミクロ組織形成が進行するとの仮定に基づいている。

$$G = \int_V \left[\sum_{i=1}^N \sum_{j=i+1}^N \{W_{ij}\phi_i(\mathbf{r}, t)\phi_j(\mathbf{r}, t)\} + \sum_{i=1}^N \sum_{j=i+1}^N \left\{ -\frac{a_{ij}^2}{2} |\nabla \phi_i(\mathbf{r}, t)|^2 \right\} \right] dV \quad (1)$$

ここで、右辺第1項はポテンシャルエネルギー、第2項は勾配エネルギーと呼ばれる。 W_{ij} や a_{ij} は界面エネルギーや界面幅に関係づけられるパラメータである。 $\phi_i(\mathbf{r}, t)$ は、フェーズフィールド変数と呼ばれ、異なる N 個の結晶粒からなる系を考えたとき、空間座標 \mathbf{r} と時間 t において、ある i 番目($i = 1 \sim N$)の結晶粒が存在する確率を表す。したがって、たとえば番号1の結晶粒(結晶粒1と呼ぶ)が存在する座標では $\phi_1(\mathbf{r}, t) = 1$ 、結晶粒1が存在しない座標では $\phi_1(\mathbf{r}, t) = 0$ と定義する。また、結晶粒1と他の結晶粒との界面を有限厚さの領域と定義し、そこで0から1へと滑らかに変化するものと定義する。以下では各変数について、 (\mathbf{r}, t) は省略して記する。

式(1)に示した全自由エネルギーの時間に対する単調減少を仮定すると、 ϕ_i の時間発展方程式は、Allen-Cahn方程式より次式で与えられる。

$$\frac{\partial \phi_i}{\partial t} = -\frac{2}{n} \sum_{j=1, j \neq i}^n M_{ij}^{\phi} \sum_{k=1}^n \left\{ (W_{ik} - W_{jk}) \phi_k + \frac{1}{2} (a_{ik}^2 - a_{jk}^2) \nabla^2 \phi_k \right\} \quad (2)$$

ここで、 n は任意座標において非零となるフェーズフィールド変数の個数、 M_{ij}^ϕ はフェーズフィールド変数の変化を規定するモビリティである。

なお数値シミュレーションにおいては、式(2)に示す ϕ_i の偏微分方程式を時間と空間について離散化し、3次元直交格子上でステンシル計算を行う。本研究では、空間微分は2次精度の有限差分法、時間積分は1次精度のオイラー法を用いて計算した。

MPF 法の複数 GPU 計算

3

3.1 Active Parameter Tracking 法

本研究で用いたMPF法では、 N 個すべての結晶粒ではなく、任意座標に存在する n 個の結晶粒に対応する ϕ_i についてのみ時間変化を計算すればよい。したがって、 N 個すべての ϕ_i の値や結晶粒番号 i の値をメモリ上に保存しておく必要はない。そこで本研究では、メモリ効率化手法としてActive Parameter Tracking (APT)法を用いた^[5]。APT法の詳細は文献^[5]を参照されたいが、MPF法で多結晶組織形成の3次元シミュレーションを高速に行うためには、メモリアクセス数やデータ通信量を低減するためにも、APTの利用は必須である。本稿では、APT法の処理のうち隣接差分格子点のデータを必要としない処理をAPT1、隣接差分格子点のデータを必要とする処理をAPT2と記する。

3.2 領域分割

MPF法を複数のGPUで計算するために、全計算領域を複数のサブ領域に分割し、各サブ領域の計算にCPUとGPUを1基ずつ割り当てる、領域分割法を採用した。例えば、図1に示すように、 $NX \times NY \times NZ$ 個の差分格子からなる全計算領域を3次元分割により x 、 y 、 z 方向にそれぞれ X 、 Y 、 Z 分割する場合には、サブ領域に含まれる差分格子点は $NX/X \times NY/Y \times NZ/Z$ 個となる。本研究では、ステンシル計算を行う為、隣接するサブ領域間のデータ通信が必要となる。そのため、各サブ領域の x 、 y 、 z 方向の端面に境界領域を作成する。すなわち、サブ領域の x 、 y 、 z 方向の端に、それぞれ $1 \times NY \times NZ$ 、 $NX \times 1 \times NZ$ 、 $NX \times NY \times 1$ の境界領域を作成する。境界領域のデータ通信には、MPIを用いる。なお、本研究で実施した実行性能の評価では、全計算領域サイズを変えるごとに、1次元分割、2次元分割および3次元分割でテスト計算を実施し、実行性能が最も高い領域分割方法を採用了。

3.3 スレッドブロックの設定

GPUの性能を十分に発揮させてシミュレーションを行うためにはスレッドおよびブロックの割当て方法が非常に重要となる。本研究では、図2のように、サブ領域の格子点数が $nx \times ny \times nz$ の場合、 x 、 y 方向にそれぞれ X' 、 Y' に分割した。つまり、サブ領域を $nx/X' \times ny/Y' \times nz$ の小領域に分割し、 $nx/X' \times ny/Y' \times 1$ のスレッドブロックが z 方向に nz 回移動するように計算した。スレッドとブロックの割り当て数については、計算領域サイズや領域分割数に応じて、試行錯誤により最適値を決定した。

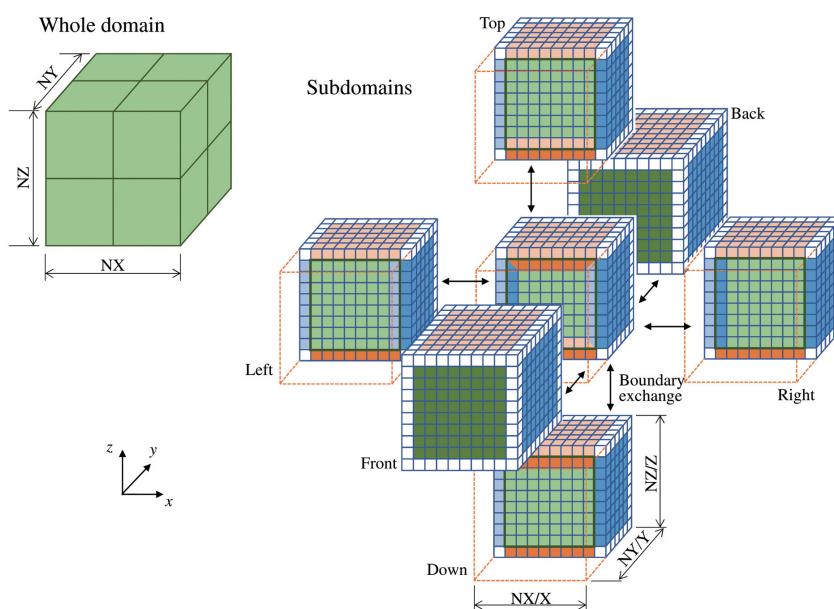


図1 3次元分割による計算領域の分割

マルチフェーズフィールド法を用いた 金属多結晶組織形成シミュレーションの大規模GPU計算

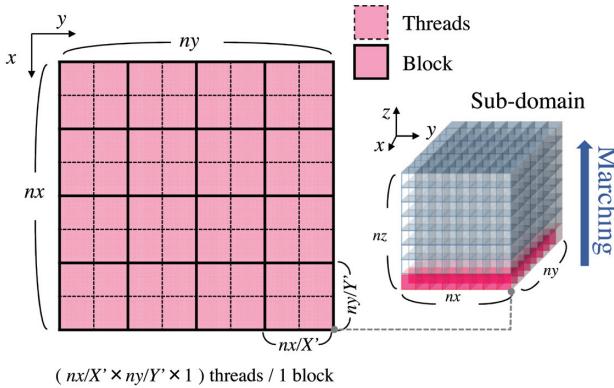


図2 スレッドブロックの設定

3.4 オーバーラッピング法

複数のGPUとCPUを用いて並列計算する場合、並列数を増やすほど境界領域のデータ通信のオーバーヘッドが発生し、実行性能が低下する。そこで本研究では、CUDAの複数カーネルの非同期実行機能とGPU-CPU間の非同期通信機能およびこれらの同時実行機能を利用して、数値演算とデータ通信をCPUとGPUの両方で同時に実行し、オーバーヘッドを隠蔽するためのアルゴリズム(オーバーラッピング法)を開発した^[4]。

図3に、オーバーラッピング法を適用したMPF法の計算ダイアグラムを示す。本方法では、GPUで行う計算を4つのストリーム(Stream 1～Stream 4)として用意する。まず、Stream 1では、各GPUに割り当てられたサブ領域内部に対して式(3)の時間発展方程式の計算とAPT1の処理を行う。これと並行して、Stream 2～Stream 4では、境界領域に対して式(3)の時間発展方程式の計算を行う。Stream 2～Stream 4での計算が終り次第、GPUのグローバルメモリにある計算結果をCPU側のホストメモリへ転送する。ここでは、CUDAの非同期通信関数(cudaMemCpyAsync)を用いる。

一方、CPUは1計算ステップ前に計算されたGPUのグローバルメモリ上にある境界領域の計算結果をcudaMemCpyAsyncによりホストメモリへ転送した後、Stream 2～Stream 4の処理が完了するのを待機する。Stream 2～Stream 4の処理が完了すれば、境界領域についてAPT1とAPT2の処理とそれに必要なデータ通信を行う。以上の処理により更新された境界領域のデータをcudaMemCpyAsyncによりGPUのグローバルメモリへ転送する。

CPUでの計算とデータ通信、および全てのStreamの実行完了を確認してから、GPUが境界領域に対してAPT2の計算とデータ更新を行い、1計算ステップが完了する。なお、ホストメモリ上のデータ更新は、Stream 1～Stream 4を実行している間に行なうことができる。

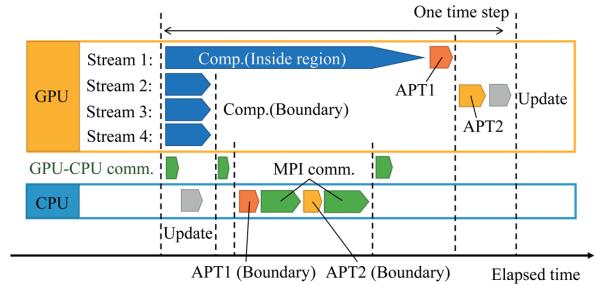


図3 オーバーラッピング法の計算ダイアグラム

GPU スーパーコンピューター TSUBAME2.5 における実行性能評価

4

4.1 多結晶粒成長シミュレーション

本研究で開発したMPF法の複数GPU計算法のベンチマークとして、東京工業大学学術国際情報センターのGPUスーパーコンピューターTSUBAME2.5において、多結晶粒成長の大規模3次元シミュレーションを実施し、実行性能の評価を行った。実行性能の評価は、すべて単精度計算にて行った。

図4に、TSUBAME2.5で計算した多結晶粒成長シミュレーションの結果の一例を示す。この例では、 0.512^3mm^3 の計算領域を 1024^3 の規則差分格子で分割し、256台のGPUを用いて計算した。初期結晶粒数は32768個であり、図4では各結晶粒に異なる色を付けて多結晶粒組織を可視化した。多結晶粒成長は、金属材料の強度を調節するために行われる熱処理(焼純)などで生じる現象であり、その際に観察される結晶粒の粗大化と収縮して消滅する挙動を良好に再現できている。このシミュレーションのような大規模な計算領域で、かつ非常に多数の結晶粒で構成されるミクロ組織の変化を考慮したシミュレーションを行うことで初めて、正しい多結晶組織の統計的評価(例えば、結晶粒サイズの平均値の変化やその分布の評価)が可能となる。

図4で示したような、多結晶粒成長シミュレーションの実行性能を評価するために、1GPUで計算する差分格子点数を 256^3 、結晶粒数を512で一定とし、使用するGPU数を変えることで計算領域サイズを大きくした際の浮動小数点演算性能(FLOPS)の変化を評価した。この評価により、弱スケーリング性能を評価することができる。図5に、弱スケーリング性能の評価結果を示す。オーバーラッピング法を導入せざると、FLOPS値はGPU数に比例して増加しており、良好なスケーリング性能が得られているが、オーバーラッピング法を導入すれば、実行性能は全体的に3～4倍は向上する。この結果、GPU数を729、差分格子点数を 2304^3 、結晶粒数を373248とした計算では、実行性能は1.9 TFLOPSとなった。

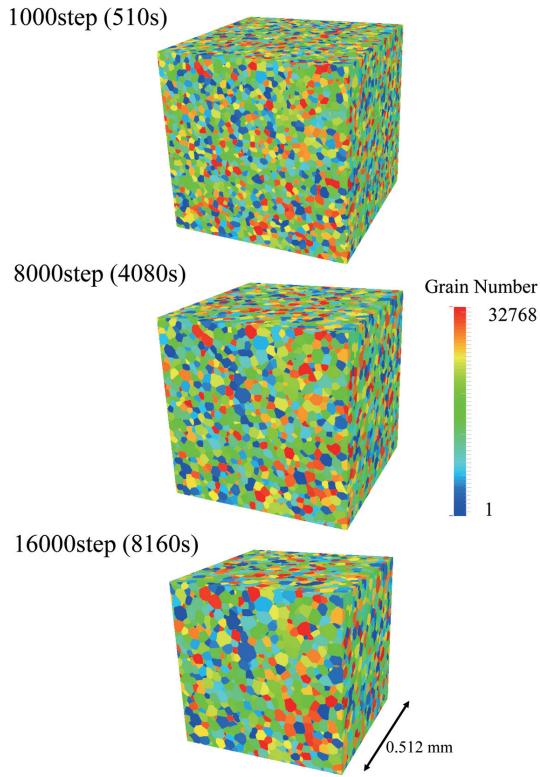


図4 TSUBAME2.5で計算した多結晶粒成長シミュレーションの結果。差分格子点数は 1024^3 であり、256GPUを使用した。結晶粒ごとに異なる色を付けて多結晶組織を可視化した。

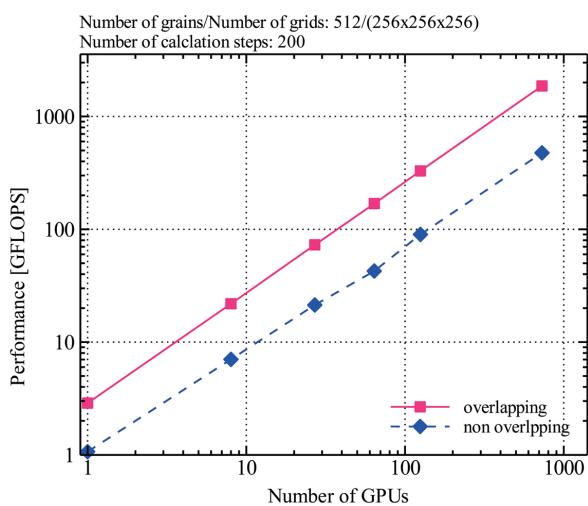


図5 多結晶粒成長シミュレーションの複数GPU計算の弱スケーリング性能。

次に、計算領域サイズと結晶粒数は一定とし、領域分割数(GPU数)を変化させてシミュレーションを実施した際のFLOPS値の変化、すなわち強スケーリング性能を評価した。ここでは、 256^3 差分格子点の計算領域に含まれる結晶粒数を512で一定とし、計算領域を 256^3 、 512^3 、 1024^3 差分格子点とした。図6に、強スケーリング性能の測定結果を示す。基本的には、どの計算領域サイズを用いてもGPU数の増加によりFLOPS値は線形に増加し、良好な強スケーリング性能が得られているが、 256^3 差分格子点の計算領域を128GPUで計算した場合のように、設定した計算領域に対してGPU数が過多となると、実行性能が低下する。これは、GPUで行う計算時間がCPUで行うデータ通信などの処理に要する時間よりも短くなることで、オーバーラッピング効果が得られなくなつたためである。しかしながら、計算領域を 512^3 および 1024^3 差分格子とした計算では、GPU数が100を超えて良好なスケーラビリティが得られることがわかる。

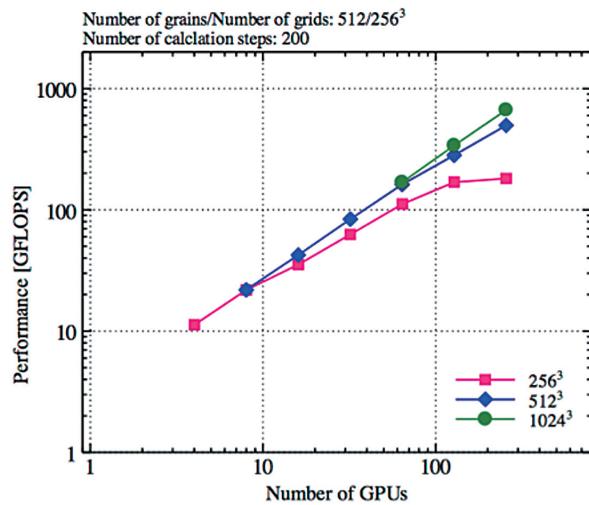


図6 多結晶粒成長シミュレーションのMPF法の複数GPU計算の強スケーリング性能。

マルチフェーズフィールド法を用いた 金属多結晶組織形成シミュレーションの大規模GPU計算

おわりに

5

MPF法を用いて、金属材料中の多結晶組織形成の高速かつ大規模なシミュレーションを行うために、複数のGPUで並列計算する方法を開発した。本稿では、開発した計算法をGPUスーパーコンピューター TSUBAME2.5に実装し、多結晶粒成長の大規模3次元シミュレーションを行い、優れた実行性能を得られることを示した。現代の材料制御技術を利用して、実験的な研究開発を繰り返して、所望の結晶粒サイズを得るための最適条件を決定することは時間のかかる作業となる。我々が開発したMPF法の複数GPU計算法により、金属多結晶組織の変化を効率的かつ高精度に解析可能となれば、材料開発期間の短縮、実験コストの削減に大きく貢献できるものと期待している。

謝 辞

本研究は、平成25年度および平成26年度の学際大規模情報基盤共同利用研究・共同研究拠点の支援のもと行ったものである。関係者の皆様に感謝申し上げる。また、本研究の一部は、科学研究費補助金・挑戦的萌芽研究（課題番号 25630322）から支援を頂いた。記して謝意を表す。

参考文献

- [1] I. Steinbach, F. Pezzola: A Generalized field method for multiphase transformations using interface fields, *Physica D*, Vol. 45, pp.385-393 (1999)
- [2] T. Shimokawabe, T. Aoki, T. Takaki, A. Yamanaka, A. Nukada, T. Endo, N. Maruyama, S. Matsuoka: Peta-scale phase-field simulation for dendritic solidification on the TSUBAME2.0 supercomputer, *Proceedings of the 2011 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis, SC'11*, IEEE Computer Society, Seattle, WA, USA, (2011)
- [3] 岡本成史, 山中晃徳, 下川辺隆史, 青木尊之: マルチフェーズフィールド法による多結晶粒成長シミュレーションの複数GPU計算, 日本計算工学会論文集, Vol. 2013, p.20130018 (2013)
- [4] A. Yamanaka, M. Okamoto, T. Shimokawabe, T. Aoki: Large scale 3D multi-phase-field simulation of microstructure evolution using TSUBAME2.5 GPU-supercomputer, *Proceedings of 2nd. International Congress on 3D Materials Science, The Minerals, Metals & Materials Society*, pp.59-64, (2014)
- [5] S. G. Kim, D. I. Kim, W. T. Kim, Y. B. Park: Computer simulation of two-dimensional and three-dimensional ideal grain growth, *Phys. Rev. E*, Vol. 74, p.061605 (2006)

量子モンテカルロ法に基づく 振動状態解析の大規模並列計算

中山涼太* 藤岡蔵** 北幸海** 立川仁典**

* 横浜市立大学国際総合科学部 ** 横浜市立大学大学院生命ナノシステム科学研究科

本稿では、我々が開発した振動量子モンテカルロ(vibQMC)法、およびその並列化の実装法について概説する。vibQMCプログラムの並列化率は99.9981%であり、東京工業大学のTSUBAME2.5スーパーコンピューターシステムを用いたベンチマーク計算(最大5376 cores)による実行並列化効率は約91%である。また大規模並列計算の応用事例として、大気中のエアロゾル形成の前駆体となる負イオン核一水和物(H_3O_2^-)に対する振動状態解析の結果についても報告を行う。

序論

1

量子モンテカルロ(Quantum Monte Carlo; QMC)法^[1]は乱数を用いた確率論的な量子化学計算手法であり、原子や分子を問わず適用可能な超高精度第一原理計算の一つである。その中でも拡散モンテカルロ(Diffusion Monte Carlo; DMC)法は、高精度な変分エネルギーを与えることで知られている。しかしながら、(i) 配置数変動による誤差(population control bias)が不可避、(ii) ハミルトニアンと非可換な演算子の固有値として与えられる物理量の計算が困難である、といった問題があり、事実上、DMC法の適用範囲は変分エネルギーの解析に制限されている。

Reptation Monte Carlo(RMC)法^[2]は、DMC法と同様に虚時間発展を利用した高精度な計算手法でありながら、上記の問題を解決することが出来る方法である。しかしRMC計算は計算コストが大きいため、その適用範囲は水分子^[3]など極めて小さな系に留まっている。

本稿では、大規模系に対するRMC計算の実現に向けて実装したRMCプログラムの並列化の実装案を概説し、東京工業大学のTSUBAME 2.5スーパーコンピューターシステムを用いたベンチマーク計算(最大5376 cores)の結果について報告を行う。また大規模並列計算の応用計算例として、大気中のエアロゾル形成の前駆体となる負イオン核一水和物(H_3O_2^-)に対する振動状態解析の結果について報告を行う。

方法

2

本研究では多原子分子の振動状態解析に、変分モンテカルロ(VMC)法とReptation Monte Carlo(RMC)法という2種類の量子モンテカルロ法を用いた。本節ではこれら二手法について簡単に説明を行う。

2.1 変分モンテカルロ(VMC)法

VMC法では、変分パラメーターを含む任意の試行反動関数 Ψ_T を用いて、次式で定義されるハミルトニアン演算子 $\hat{H} = \hat{T} + \hat{V}$ (\hat{T} は運動エネルギー演算子、 \hat{V} はポテンシャルエネルギー演算子)の期待値 E を考える:

$$E = \frac{\int \Psi_T(\mathbf{Q}) \hat{H} \Psi_T(\mathbf{Q}) d\mathbf{Q}}{\int \Psi_T^2(\mathbf{Q}) d\mathbf{Q}} \quad (1)$$
$$= \frac{\int \Psi_T^2(\mathbf{Q}) \epsilon_L(\mathbf{Q}) d\mathbf{Q}}{\int \Psi_T^2(\mathbf{Q}) d\mathbf{Q}}$$

ここで $\mathbf{Q} \equiv \{q_1, q_2, \dots, q_N\}$ は多次元の一般化座標であり、配置あるいはwalkerと呼ばれる。また $\epsilon_L \equiv \Psi_T^{-1} \hat{H} \Psi_T$ は局所エネルギーであり、試行波動関数には実関数を仮定した。(1)式はエネルギー期待値 E が局所エネルギー ϵ_L の Ψ_T^2 重み付き平均によって算出されることを意味している。VMC法では、 Ψ_T^2 に従う配置のセット $\{\mathbf{Q}_i\}(i=1 \sim M)$ をメトロポリス法^[4]によって生成することでエネルギー期待値の算定を行う。

VMC法では全エネルギーだけでなく、任意の演算子の期待値を与えることが可能であるが、その計算精度は試行波動関数 Ψ_T の質に強く依存する。多電子系のQMC計算では、分子軌道計算で得られる単一(Hartree-Fock法)あるいは複数(配置間相互作用法)のSlater行列式にJastrow因子と呼ばれる相関因子を付与した、Slater-Jastrow型試行波動関数を用いることが一般的である。多原子分子の振動状態解析を主目的とした本研究では、試行波動関数には次式で表されるVibrational Self-Consistent Field(VSCF)

量子モンテカルロ法に基づく振動状態解析の大規模並列計算

型関数^[5]を用いた：

$$\Psi_{\text{VSCF}} = \varphi_1(q_1) \times \varphi_2(q_2) \times \cdots \times \varphi_N(q_N) \quad (2)$$

ここで φ_i は i 番目の振動モードの modal function である。本研究では調和振動子の固有関数の線形結合によって φ_i を展開し、Umrigarらによって開発された線形法^[6]を用いて展開係数の最適化を行った。また最適化したVSCF関数は次節で説明するRMC法における試行波動関数として用いる。

2.2 Reptation Monte Carlo (RMC) 法

以下、表記の簡単化のため問題を1次元とする。時間依存 Schrödinger方程式の虚時間発展を利用したQMC法では、任意の試行波動関数 Ψ_T に虚時間プロパゲータを作用させることで正確な波動関数 Ψ_0 を得る：

$$|\Psi_0\rangle = \lim_{\tau \rightarrow \infty} e^{-\frac{\tau \hat{H}}{2}} |\Psi_T\rangle \cdots \times \varphi_N(q_N) \quad (3)$$

ここで τ は虚時間である。RMC法では以下に示す仮想的な分配関数 Z_0 を導入する：

$$Z_0 \equiv \langle \Psi_0 | \Psi_0 \rangle = \lim_{\tau \rightarrow \infty} \langle \Psi_T | e^{-\tau \hat{H}} | \Psi_T \rangle \quad (4)$$

虚時間ステップを $\Delta\tau = \tau / N$ として虚時間プロパゲータ $e^{-\tau \hat{H}}$ を N 個に短時間分解し、二次の短時間近似 (Suzuki-Trotter分解^[7]) を用いることで、分配関数に対する以下の式を得る：

$$Z_0 = \int \cdots \int dq^{(0)} \cdots dq^{(N)} P(q^{(0)}, \dots, q^{(N)}) \times W(q^{(0)}, \dots, q^{(N)}) \quad (5)$$

ここで、

$$P \equiv \Psi_T(q^{(0)}) \left[\prod_{i=0}^{N-1} g(q^{(i)}, q^{(i+1)}; \Delta\tau) \right] \Psi_T(q^{(N)}) \quad (6)$$

$$W \equiv \prod_{j=0}^{N-1} e^{-\frac{\Delta\tau}{2} \{ \epsilon(q^{(j)}) + \epsilon(q^{(j+1)}) \}} \quad (7)$$

である。 $q^{(i)}$ は虚時間 $\Delta\tau \times i$ における配置であり、各虚時間における配置のセット $\{q^{(i)}\}$ ($i=0 \sim N$) を虚時間Pathあるいは reptile と呼ぶ。また二次近似として次式を用いている：

$$\langle q | e^{-\tau \hat{H}} | q' \rangle \approx e^{-\frac{\Delta\tau}{2} \epsilon_L(q)} \langle q | e^{-\Delta\tau \mathcal{H}} | q' \rangle e^{-\frac{\Delta\tau}{2} \epsilon_L(q')} \quad (8)$$

ただし、 $\hat{H} = \mathcal{H} + \epsilon_L(q)$ 、 $\mathcal{H} \equiv \hat{T} - \Psi_T^{-1}(q) \hat{T} \Psi_T(q)$ 、および $g(q_i, q_{i+1}; \Delta\tau) \equiv \langle q_i | e^{-\Delta\tau \mathcal{H}} | q_{i+1} \rangle$ である。

(5) 式において、 $P(q^{(0)}, \dots, q^{(N)})$ は Ψ_T^2 に従うreptileを生成するプロパゲータ、 $W(q^{(0)}, \dots, q^{(N)})$ はそのreptileの重みと解釈することができます。本研究では、このreptileの生成にLangevin方程式

$$q^{(i+1)} = q^{(i)} + 2Dv(q^{(i)})\Delta\tau + \chi(2D\Delta\tau) \quad (9)$$

を用いた。ここで、 $D=1/2\mu$ 、 μ は振動モードの換算質量、 $\chi(2D\Delta\tau)$ は分散 $2D\Delta\tau$ を持つ正規分布乱数、 $v \equiv \Psi_T^{-1} \nabla \Psi_T$ は drift velocity である。また重み W によるreptileの採択・棄却にはメトロポリス法を用いた。

十分長い虚時間長 τ の下、reptile に対して採択・棄却を繰り返すことで、reptile の両端の配置 $q^{(0)}, q^{(N)}$ の集合は分布 $\Psi_T \times \Psi_0$ へ、reptile の中央の配置 $q^{(N/2)}$ は分布 Ψ_0^2 へと収束する。本研究では、エネルギー期待値などハミルトニアンと可換な演算子の期待値は、分布 $\Psi_T \times \Psi_0$ を用いて次式で定義されるmixed estimatorによって算出した：

$$\begin{aligned} \langle \hat{H} \rangle_{\text{mix}} &\equiv \lim_{\tau \rightarrow \infty} \frac{\langle \Psi_T | \hat{H} e^{-\tau \hat{H}} | \Psi_T \rangle}{\langle \Psi_T | e^{-\tau \hat{H}} | \Psi_T \rangle} \\ &= \frac{\int \Psi_T \Psi_0 \epsilon_L(q) dq}{\int \Psi_T \Psi_0 dq} \end{aligned} \quad (10)$$

またポテンシャルエネルギーや分子の幾何構造など、ハミルトニアンと非可換な演算子に対しては、分布 Ψ_0^2 を直接解析することで期待値の算定を行った。

並列化

3

前節で概説したように、QMC法による物理量の算定では、VMC法ではwalkerに対して、RMC法ではreptileに対してメトロポリス法による多重マルコフ連鎖を発生させる。その際、サンプリング回数が十分多く、計算するサンプルの総量が変わらなければ、その生成方法に寄らず統計学的に等価な結果を得ることができる。したがって、QMCアルゴリズムの並列化に向けた最も単純かつ有効な戦略は、統計的に独立した複数のwalker/reptileを用いた配置空間の同時サンプリングである。具体的には、単一のwalker/reptileを用いてサンプリング数を増やす代わりに、サンプリングに用いるwalker/reptile数を増大させる(図1参照)。

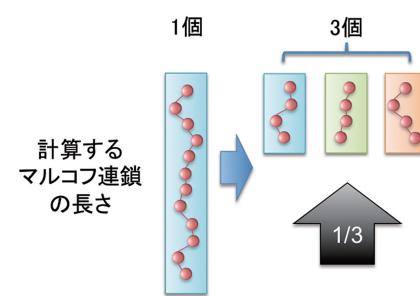


図1 マルコフ連鎖の分割の模式図

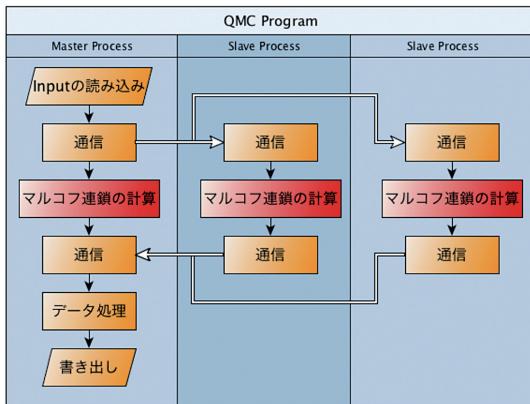


図2 並列化QMCプログラムの概略図

本研究では、複数になったマルコフ連鎖を複数のプロセスを用いて並列処理する。並列化されたQMCプログラム概略を図2に示す。ここでデータ処理用に用意される1つのプロセスをマスター プロセス (Master Process)、マルコフ連鎖の計算を並列的に処理するために用意される1つもしくは複数のプロセスがスレーブ プロセス (Slave Process) である。並列化の実装にはMPI (Message Passing Interface) を用い、ライブラリーにはOpenMPI 1.4.2を用いた。

結果と考察

4

4.1 並列化効率

本研究では、東京工業大学のTSUBAME2.5スーパーコンピューターシステム上で、最大5376プロセス (1 process/core × 12 cores/node × 448 nodes)を用いて、並列化効率に関するベンチマーク計算を実行した。ベンチマーク計算に使用した分子系は、正イオン H_3O^+ の一水和物 H_5O_2^+ の振動基底状態であり、ポテンシャルエネルギーにはBowmanらによってCCSD(T)/aug-cc-pVTZレベルの第一原理計算を再現するように作成された解析的なポテンシャル関数^[8]を用いた。

12並列(1node)計算を基準にした際のVMC計算とRMC計算のベンチマーク結果を図3に示す。5376 coresを用いた並列計算におけるspeed up (=Time[12 process]/Time [Nprocess])は、VMC計算で4868 (=405.70 × 12) 倍、RMC計算では2307 (=192.25 × 12) 倍である。各計算における問題サイズは不变 (アムダールの法則^[9])と仮定し、これらの結果とから両手法の基準計算 (12並列)における逐次実行部分の割合を見積もると、VMC計算では $r_{\text{VMC}}^{(12)} = 0.023\%$ 、

RMC計算では $r_{\text{RMC}}^{(12)} = 0.3\%$ となった。これは非並列計算 (1core) 時における処理の内、VMC計算では99.9981%が、RMC計算では99.975%が並列化されたことに相当する。

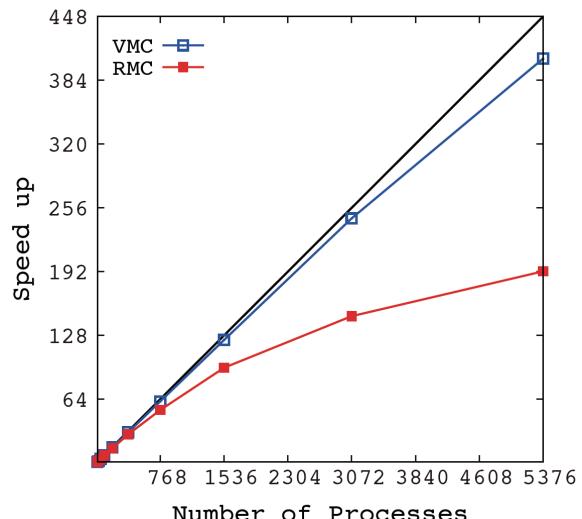


図3 VMC法、RMC法のSpeed up (12並列時の計算を1とした)

4.2 負イオン一水和物 H_3O_2^- の解析

本手法を用いて、大気中のエアロゾル形成の前駆体となる負イオン核一水和物 (H_3O_2^-) の振動状態解析を行った。 H_3O_2^- は大気中に存在する負イオン核 OH-に水分子が1つ配位した系である。原子核の量子効果を含めない通常の第一原理計算では、図4(a)に示すように中央のプロトンが片側の酸素原子に寄った構造が最安定となる。しかし、このプロトン移動の遷移状態におけるエネルギー障壁は極めて小さいことから (0.88 kJ/mol)、振動基底状態においては中央のプロトンが酸素原子間の中央に位置した遷移状態構造が最安定構造となることが示唆されている (図4(b))^[10]。

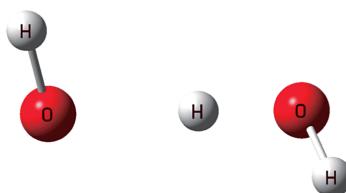
本研究では、 H_3O_2^- 及びその重水素化 (D) 体、三重水素化 (T) 体に対して、振動基底状態と中央のプロトン移動に関する振動モード (以下、架橋振動) の基音準位の解析を行った。試行波動関数はVSCF波動関数、PESにはHuangらによってCCSD(T)/aug-cc-pVTZの第一原理計算を再現するように作成された解析ポテンシャルを用いた^[8]。

VMC法とRMC法によって得られた各系の零点振動エネルギー (ZPE) と架橋振動の基本振動数を表1に示す。変分エネルギーであるZPEに注目すると、全ての系においてRMC法による値がVMC法よりも低いことから、RMC法によって計算精度が改善されていることがわかる。また架橋振動の基本振動数の実験値 (H体) は 697cm^{-1} と報告されているが、RMC法は誤差 12cm^{-1} の精度で実験値を再現している。

各振動状態における水素結合プロトン (H^*) の振る舞い解析するため構造パラメーター $\delta_{OH^*} = R_{O1H^*} - R_{O2H^*}$ に注目した。ここで R_{O1H^*}, R_{O2H^*} は図4 (b) で定義されている 2 つの OH^* 間距離であり、 $\delta_{OH^*} = 0$ のとき H^* は酸素原子間の中央に位置することを意味する。RMC 法から得られた振動基底状態と架橋振動の基音準位における δ_{OH^*} の一次元分布を図5 に示す。振動基底状態では H^* は酸素原子間の中央に分布しており、同様の振る舞いが先行研究においても報告されている^[10,12]。また D 体・T 体でも D^*, T^* は酸素原子間の中央に位置しているが、その分布は H 体と比較して局在化している。経路積分分子動力学法を用いた Suzuki らの解析^[12]では、報告されている中で最も低温である 50Kにおいて、D 体と T 体に有意な差は得られていないが、本解析により振動基底状態 (0 Kelvin)において T 体の分布は、D 体と比較してわずかに局在化することがわかった。

架橋振動の基音準位においては、波動関数の節構造を反映した結果、 δ_{OH^*} はダブルピーク構造を持つ。2 つの分布の内、片側一方に注目すると、その分布は原子核の質量が軽いほど非局在化しており、かつピーク位置は $|\delta_{OH^*}|$ が大きい領域へシフトしている（片側の酸素原子へ寄る）。 $\delta_{OH^*} = 0$ はプロトン移動の遷移状態に対応しているが、基底状態・基音準位ともに原子核の質量が重いほど遷移状態付近の存在確率が増大するという興味深い結果が得られた。

(a) Equilibrium geometry



(b) Average geometry at the vibrational ground state

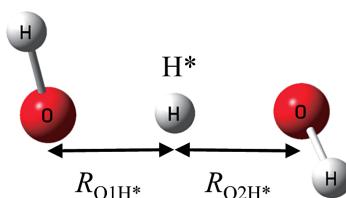


図4 $H_3O_2^-$ の平衡構造 (a) と振動平均構造 (b)

図5 (c) は、振動基底状態と架橋振動の基音準位の二状態のみを考慮した場合の 600K における δ_{OH^*} の一次元分布である。H 体における δ_{OH^*} は基底状態よりさらに非局在化したシングルピーク構造を、D 体・T 体はわずかに分裂したダブルピーク構造を有している。同様の分裂が Suzuki らによる 600K における経路積分法の

解析で報告されているが、本研究により D 体・T 体におけるピークの分裂は、主に架橋振動の基音準位の寄与であることが明らかになった。

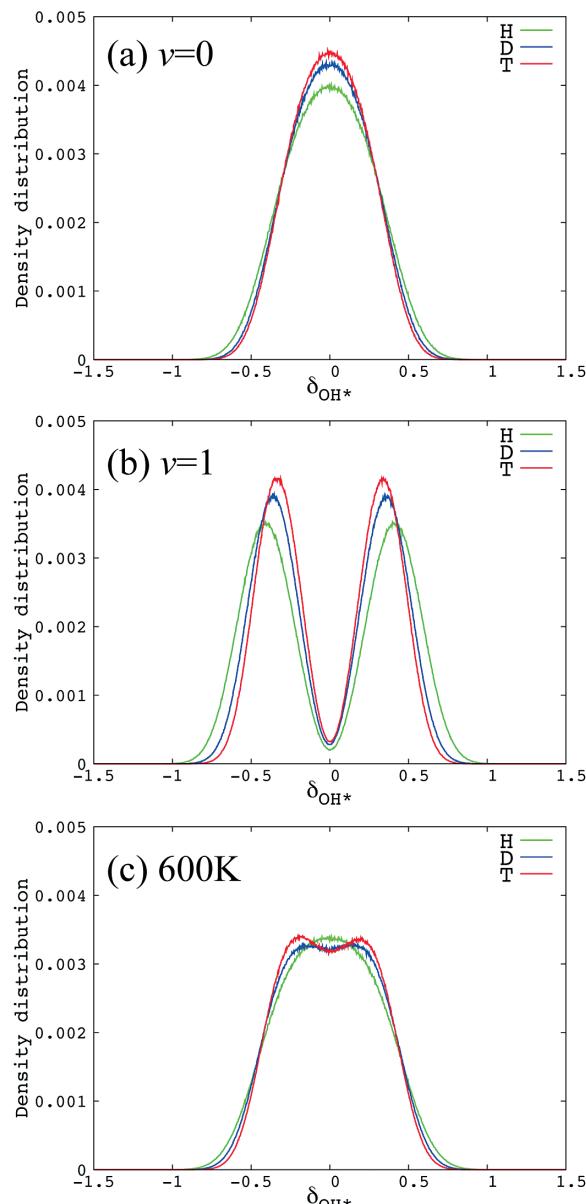


図5 $H_3O_2^-$ の δ_{OH^*} の一次元分布。(a) 振動基底状態、(b) 架橋振動の基音準位、(c) 600K。
単位は Bohr。

手法		ZPE	基本振動数
VMC	H 体	6881.6(1)	763.9(1)
	D 体	5046.4(0)	488.4(1)
	T 体	4261.7(0)	383.7(1)
RMC	H 体	6702.2(2)	685.3(3)
	D 体	4923.8(1)	420.7(2)
	T 体	4160.1(1)	324.7(2)
Expl.[11]		697	

表1 H_3O_2^- の零点振動点エネルギー (ZPE) と架橋振動の基本振動数 (単位は cm^{-1})

結論

5

本稿では、変分モンテカルロ (VMC) 法と Reptation Monte Carlo (RMC) 法という 2 種類の量子モンテカルロ法、およびその並列化の実装方法に関する概略を示した。東京工業大学の TSUBAME2.5 スーパーコンピューターシステムを用いたベンチマーク計算 (最大 5376 cores) を実行した結果、Speed up は VMC 法では 4868 倍 (並列化効率 91%) を、RMC 法では 2307 倍 (並列化効率 43%) となった。プログラム全体の並列化率は VMC 法では 99.9981%、RMC 法では全体の 99.975% である。また大規模並列計算の応用例として、大気中のエアロゾル形成の前駆体となる負イオン核一水和物 (H_3O_2^-) に対する振動状態解析を行い、振動状態解析に対する QMC 法の有効性を示した。

本研究では多次元のポテンシャルエネルギー曲面の算定に、計算コストが安価な解析的ポテンシャル関数を使用したが、第一原理計算を併用した on-the-fly 法の利用により、今後さらなる並列化効率の向上が期待される。

謝 辞

本研究における計算は「平成 26 年度秋期 TSUBAME グランドチャレンジ大規模計算制度 (カテゴリー B)」の採択により、東京工業大学学術国際情報センターの大型計算機 (TSUBAME2.5) を利用したものである。本制度を利用させて頂いた東京工業大学学術国際情報センターの関係各位に深く感謝致します。また本研究の一部は科学研究費補助金から支援を頂きました。

参考文献

- [1] B.L. Hammond, W.A. Lester Jr. and P.J. Reynolds, "Monte Carlo Methods in Ab Initio Quantum Chemistry" (World Scientific, 1994).
- [2] S. Baroni and S. Moroni, Phys. Rev. Lett., 82, 4745 (1999).
- [3] D.G. Oblinsky, W.K. Yuen, S.M. Rothstein, J. Mol. Struct. (THEOCHEM) 961, 219 (2010).
- [4] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, J. Chem. Phys., 21, 1087 (1953).
- [5] J.M. Bowman, J. Chem. Phys. 68, 608 (1978).
- [6] J.Toulouse, C.J. Umrigar, J. Chem. Phys. 126, 084102 (2007).
- [7] M. Suzuki, Proc. Japan Acad., 69, 161 (1993).
- [8] Huang et al. J. Am. Chem. Soc. 126, 5042 (2004).
- [9] Gene M. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In AFIPS Conference Proceedings, pp. 483-485 (1967).
- [10] A.B. McCoy, X. Huang, S. Carter, and J.M. Bowman, J. Chem. Phys. 123, 064317 (2005)
- [11] E.G. Diken, J.M. Headrick, J.R. Roscioli, J.C. Bopp, and M.A. Johnson, A.B. McCoy J. Phys. Chem. A, 109, 8 (2005).
- [12] K.Suzuki, M. Shiga, and M. Tachikawa, J. Chem. Phys. 129, 144310 (2008).

個別要素法による 粉体の大規模シミュレーション

都築 恵理 渡辺 勢也 青木 尊之

東京工業大学・学術国際情報センター

砂や粉の振る舞いは液体などと似ている場合もあれば大きく違うときもある。このような粉体をシミュレーションするために、粒子間の接触相互作用をバネと摩擦のモデルで表現した個別要素法が良く使われる。スプーン1杯に砂糖が何粒あるかを想像すれば分かる通り、実際の粒子サイズで粉体をシミュレーションしようとすると100万個を超える粒子数が必要になる。粗視化モデルのアプローチもあるが限界があり、スパコンによる大規模計算で実際に近い粒子サイズで粉体シミュレーションを行う意義は大きい。粉体は空間的な広がりが時間的に変化するため、大規模計算で良く使われる領域分割がそのままでは使えない。粒子分布に合わせて時々刻々と領域分割を変更しながら、動的に計算負荷のバランスを取ることが重要である。本稿では動的負荷分散をやり難いと言われているGPUスパコンにおいて、大規模粉体シミュレーションを行い、いくつかの実用計算とゴルフのバンカーショットのような砂粒子からゴルフボールへの運動伝達を含む複雑な粉体計算を行った例を示す。

はじめに

1

粉体（粉粒体）はさまざまな場面で現れ、その振る舞いが明らかにされていない現象も多い。工学的にもプリンターのトナーや製薬プロセス、化学工学プラントなどにおいて、粉体シミュレーションの需要は高い。粉体計算は粒子計算の1つであり、天体の重力多体計算と類似した印象を受ける場合がある。しかし、粉体の相互作用は後で述べるように接触による相互作用であるため、1つの粒子が相互作用する粒子は高々数個程度であり、メモリ律速の計算である。一方、重力多体計算や分子動力学計算などは相互作用する粒子数が非常に多く、浮動小数点演算のコストが支配的である。

粉体シミュレーションも粒子数を増やして行くと（メモリアクセスを含めた）計算負荷が非常に高くなり、これまで大規模粉体シミュレーションは余り行われてこなかった。近年、スパコンの性能は飛躍的に向上し、実サイズの粒子を用いた大規模粉体シミュレーションが可能な性能を有している。しかし、プロセッサはマルチコア化し、数千～数万計算ノードから構成されているため並列計算が必須の条件である。粉体シミュレーションにおける粒子間相互作用は接触による反発と摩擦であるため、粒子番号による並列化は現実的でなく、領域分割による並列化を行うことが必要である。時間・空間的に粒子分布が変化する粉体シミュレーションに対し、計算負荷の分散とメモリ分散の目的で動的領域分割を導入する必要がある。GPU (Graphics Processing Unit) は演算性能、メモリバンド幅や電力効率の点で有利な反面、メモリ階層が深くデータ移動のオーバーヘッドが大きい。本稿では、そのGPUを用いて大規模粉体シミュレーションを実行するためのアルゴリズムの開発と実装を行い、実用問題への適用例を示す。

GPUによるDEM計算

2

GPUは階層的なメモリ構造と2,000を超える演算コア(CUDAコア)を持っている。GPUのアーキテクチャを考慮した超細粒度・超多スレッドの計算アルゴリズムと実装を行うことにより、高い性能を引き出すことができる。GPUのプログラミングはNVIDIA社が提供する統合開発環境のCUDAを用いる。

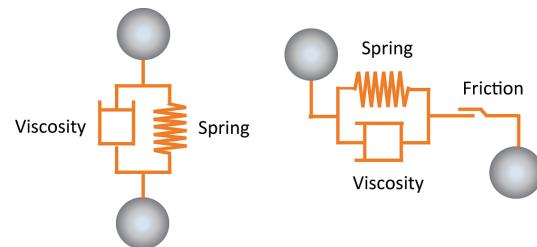


図1 個別要素法の物理モデル

粉体シミュレーションの計算モデルを図1に示す。本研究では、粉体粒子同士が接触すると反発力とダンパー（バネとダッシュポット）が作用する個別（離散）要素法 DEM (Discrete Element Method) を用いている。せん断方向にはバネとダッシュポットに加えて摩擦力も発生する。粒子運動は式(1)に従い、加えられた力とトルクに対して2次精度のルンゲクッタ法か leap-frogなどにより時間積分される。

$$m\ddot{x}_i = \sum_{j \neq i}^N (-kx_{ij} - \gamma\dot{x}_{ij}) \quad (1)$$

GPU計算では、各粒子の持つ速度や座標、運動量などの従属変数は粒子構造体の中のメンバ変数としてGPUボード上のDeviceメモリ (CUDAプログラミング上では、グローバル・メモリ) に保持され、1スレッドが1粒子を計算する。

DEMのように接触した粒子とのみ相互作用する局所性の高い計算では、全粒子との接触判定を行うことは非効率的である。計算領域を仮想的に格子に分割し、自身の属する格子（セル）及び隣接するセルに属する粒子とのみ相互作用計算を行う「セル分割法」を用いて計算する。粒子が属するセルに全ての粒子番号を登録する通常のセル分割法ではメモリ不足を引き起こす可能性がある。図2のように各セルでは1つの粒子の番号のみを登録し、同一セルに属する他の粒子は粒子番号を数珠つなぎに保持する Linked-list を導入する^{[1][2]}。これにより空間格子のメモリ使用量は静的な空間格子を用いる場合と比較して8分の1に削減できる。

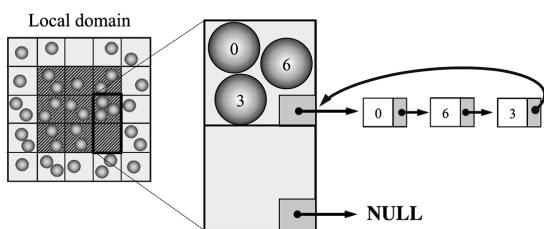


図2 Linked-list を用いた近傍粒子探索

動的負荷分散を用いた複数 GPU 計算

3

複数のGPUで粉体計算を行うには、計算領域を小領域に分割し、分割された各小領域にGPUを割り当てて計算する。図3の左側のように均等分割した場合、小領域内の粒子数に偏りが生じる。そこで、図3右図のように小領域内の粒子数を一定にするために、小領域の境界を計算途中で移動させるスライスグリッド法を導入する^[3]。

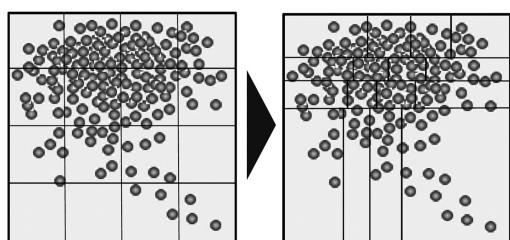


図3 スライスグリッド法による負荷分散

境界線の移動にともない隣接小領域に移動する粒子をGPU上で探索するための効率的な手法を提案する。粒子は小領域のどこに分布しているか不明なので、図4のように小領域を微小幅 Δl で分

割し、各分割幅内の粒子を数え上げる。境界を移動すべき距離が分かると、境界が移動したために小領域外にはみ出る粒子をGPU上でパッキングし、隣接小領域のGPUにホスト計算機を介してデータ転送する。複数ノードに分散するGPU間ではMPIライブラリによる通信が必要になる。受け取る側の小領域はパッキングされた粒子を小領域の粒子に加え領域の再分割が完了する。

粒子は運動するため、時間積分後に境界を横切り領域外に出た粒子の通信や、隣接小領域内の接触する可能性がある粒子のデータ転送も必要になる。GPUではメモリの確保や解放に非常に時間がかかるため、粒子データを格納するメモリはある程度静的に保持する必要がある。一方、このような頻繁な小領域境界を横切る粒子移動が起こると、使用するメモリの断片化が発生する。そこで図5のように定期的に粒子の再整列を行い、メモリの断片化を解消する。

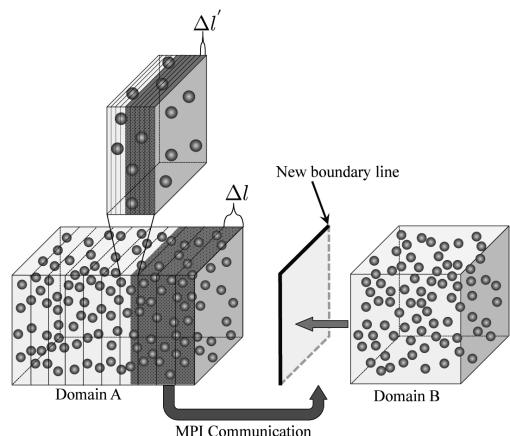


図4 粒子の数え上げと境界線の変更

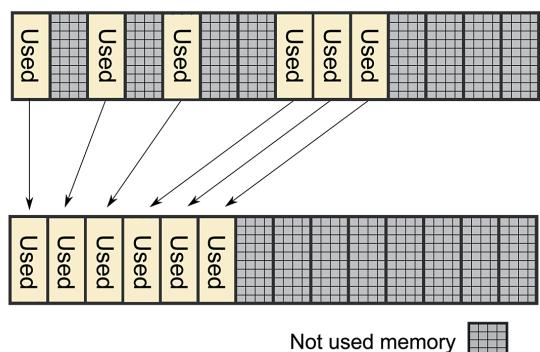


図5 Deviceメモリ上の粒子データの再整列

TSUBAME2.5における強・弱スケーリング

4

スライスグリッド法による動的負荷分散を導入したDEMのGPUコードを用いて、TSUBAME 2.5のGPU(K20X)で大規模粉体シミュレーションの実行性能を強スケーリングで検証した。図6のスクリューが粉体中で回転する搅拌計算を約200万個、約1,600万個、約1億2,900万個の粒子により計算する場合の実行性能の測定結果を図7に示す。

縦軸の実行性能は用いた粒子数を計算時間で割ったもので、1秒当たり何個の粒子を計算できるかを意味している。図7の×印で表された2つの測定点が、それぞれ1億2,900万粒子を用いた場合の256GPU、512GPUに対する実行性能の測定結果である。同一線上の実行性能の変化が強スケーリングを示している。1,600万個程度までの粒子数であれば、8～16倍まではGPU数の増加に応じて性能向上が期待できるが、それ以上は性能が飽和していくことが分かる。弱スケーリングは4GPUによる200万粒子の計算、32GPUによる1,600万粒子の計算、256GPUによる1億2,900万粒子の計算に対する実行性能を比較することにより検証でき、理想的な直線と比較して大きく性能低下していることがわかる。スライスグリッド法の欠点である分割された小割領域の形状(アスペクト比)の悪化による領域間通信量の増大が原因となり並列化効率が低下したためである。DEM計算へのスライスグリッド法の適用限界が256GPU～512GPU程度であることが確認できる。

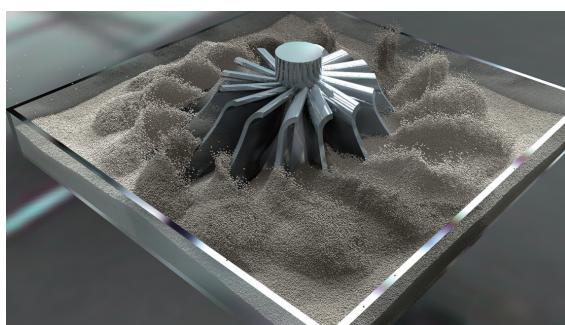


図6 64台のGPUを使い412万個の粒子を使った搅拌計算

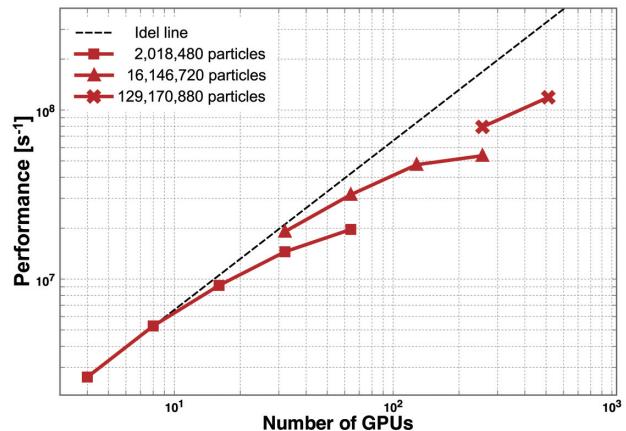


図7 粉体シミュレーションの強・弱スケーリング

提案手法の実用問題への適用

5

工業的、産業的に重要ないくつかの典型的な粉体现象に対して、動的負荷分散を用いた複数GPUによる大規模シミュレーションを行った。実用的なDEM計算には任意形状の物体との相互作用を取り扱う必要がある。図8のようにCADデータから物体表面からの符号付距離関数(Level Set関数)を事前計算しておくことにより^[4]、そのLevel Set関数を参照するだけで物体-粒子間の距離が求まり、粒子-物体間の相互作用を効率的に計算できる。

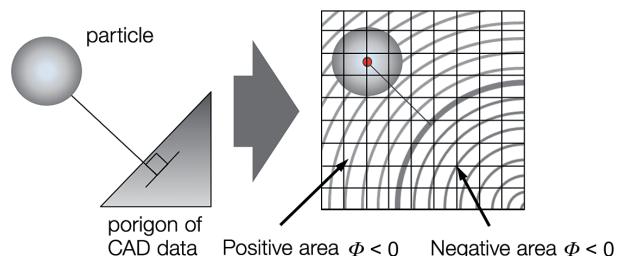


図8 Level Set 関数による物体形状表現

433万個の粒子による粉体の搬送計算を64 GPUを用いて計算した結果をそれぞれ図9に、416万個の粒子による螺旋すべり台の計算を32 GPUを用いて計算した結果を図10に示す。

ゴルフのバンカーショットはサンドウェッジのスイングによる砂のかき上げと、かき上げられた砂によるゴルフボールへの運動

伝達を含む複雑な問題である。解析手法であるDEMは計算コストが高いため、これまでには10万個程度の粒子による2次元計算にとどまっている^[5]。実際の砂と同程度のサイズの粒子を数千万個～1億個用いることにより、実現象のスケールでの3次元バンカーショット・シミュレーションを実行することができた。

バンカー砂に含まれる粗砂を想定し、粒子半径0.4 mmの粒子1,670万個を用いた大規模バンカーショット・シミュレーション

を64台のGPUを用いて行った。サンドウェッジの軌道は回転及び二重振り子モデルから決定している。バンカーショットに特徴的な砂上のゴルフボールの「目玉」の初期状態は、実際と同じようにボールを落とし、DEM計算で64,000ステップかけて生成している。サンドウェッジの先端の最大速度を5.0 m/sとしてスイングを開始した。計算結果を図11に示す。

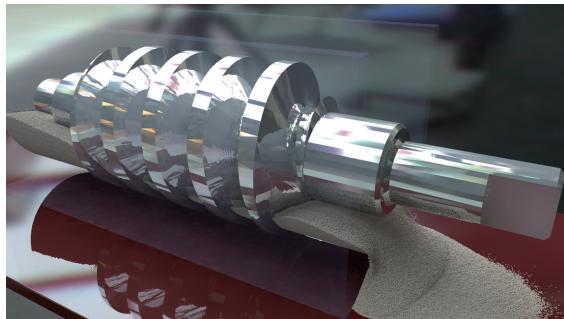


図9 64台のGPUを使い433万個の粒子を使った搬送計算

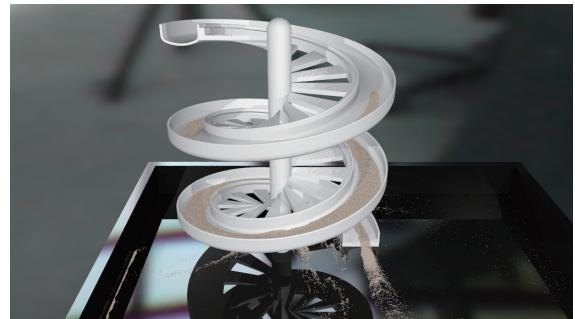


図10 32台のGPUを使い416万個の粒子を使った螺旋すべり台のシミュレーション算

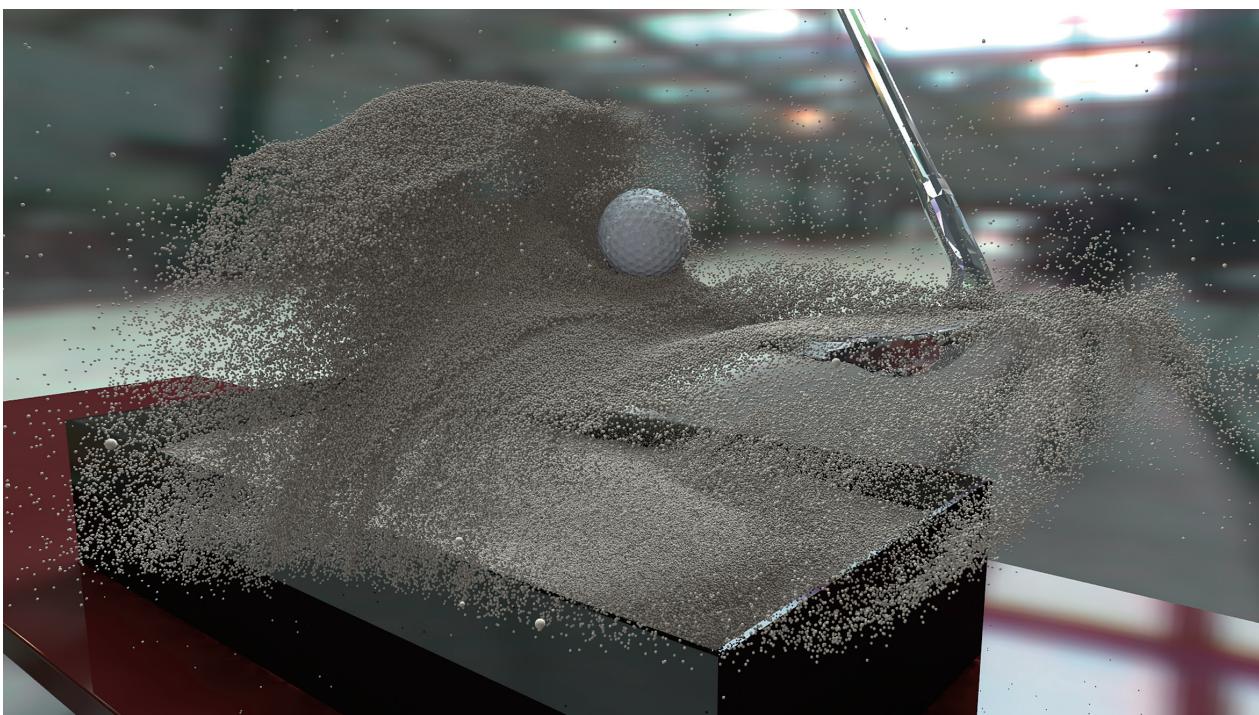


図11 1,670万粒子のバンカーショット計算

非球形粒子を用いた DEM 計算

6

複数の粒子を剛体連結させた非球形の粒子モデルを用いることで、球形粒子よりも粒子間の摩擦を正確に表現でき、より現実の現象に近い粉体シミュレーションが可能となる^[6]。1つの非球形粒子を複数の粒子を用いて表現するため球形粒子を用いたDEMに比べて多くの粒子が必要となり計算コストが増大するため、GPU計算の必要性が高まる。1台のGPUで数十万～数百万個の非球形粒子による粉体シミュレーションを実行することができた。

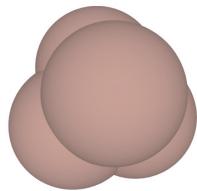


図12 テトラポッド型非球形粒子

実問題への適用例として、長靴の足跡のシミュレーションを実行した。図12のように、4つの粒子をテトラポッド型に連結させた非球形粒子を約40万個用いた。図13に(a)球形粒子、(b)テトラポッド型の非球形粒子を用いた場合のシミュレーション結果を示す。非球形粒子を用いた場合は粒子間のインターロックの影響が表われ、靴底の溝が確認できるほど明確な足跡が砂に残った。非球形粒子で計算することで粒子間の摩擦を正確に表現できることを確認した。



(a) 球形粒子による計算



(b) テトラポッド型の非球形粒子による計算

図13 長靴の足跡のシミュレーション

まとめ

7

DEM計算に動的負荷分散を導入することにより、大規模粉体シミュレーションをGPUスーパーコンで実行することができた。スライスグリッド法によるGPU間の動的領域分割と近傍粒子探索におけるLinked-list法の導入により、効率的に粉体計算を実行する手法を提案した。複雑形状物体を含む問題や非球形粒子のDEM計算も可能になり、今後は流体と連成させるなどより実用問題への展開が期待できる。

謝辞

本研究の一部は、科学研究費補助金・基盤研究(S)課題番号26220002「ものづくりHPCアプリケーションのエクサスケールへの進化」、科学技術振興機構CREST「ポストペタスケール高性能計算に資するシステムソフトウェア技術の創出」、学際大規模情報基盤共同利用・共同研究拠点、および革新的ハイパフォーマンス・コンピューティング・インフラから支援を頂いた。記して謝意を表す。

参考文献

- [1] G. S. Grest, B. D. unweg, and K. Kremer, "Vectorized link cell Fortran code for molecular dynamics simulations for a large number of particles," *Computer Physics Communications*, vol. 55, pp. 269–285, Oct. 1989.
- [2] Gomez-Gesteira, M., Crespo, A., Rogers, B., Dalrymple, R., Dominguez, J. and Barreiro, A.: fSPHysicsg development of a free-surface fluid solver Part 2: Efficiency and test cases, *Computers and Geosciences*, Vol. 48, No. 0, pp. 300 – 307 (2012).
- [3] S, Tsuzuki, and T, Aoki: Large-scale granular simulations using Dynamic load balance on a GPU supercomputer, in Poster at the 26th IEEE/ACM International Conference on High Performance Computing, Networking, Storage and Analysis (SC) 2014, New Orleans, US-LA, USA (2014).
- [4] J. A. Bærentzen and H. Aanæs, "Computing discrete signed distance fields from triangle meshes," *Informatics and Mathematical Modeling*, Technical University of Denmark, DTU, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, Tech. Rep., 2002.
- [5] 堀井宏祐, 小泉孝之, 辻内仲好, 三木光範, 日高重助, 折戸啓太; "並列粒子要素法によるバンカーショット解析", 情報処理学会論文誌, 数理モデル化と応用, Vol. 44, No. 14, pp. 91-99 (2003)
- [6] Ikuya Ono, Hiroshi Nakashima, Hiroshi Shimizu, Juro Miyasaka, Katsuaki Ohdoi, Investigation of elemental shape for 3D DEM modeling of interaction between soil and a narrow cutting tool, *Journal of Terramechanics*, Volume 50, Issue 4, August 2013, Pages 265-276, ISSN0022-4898.

● TSUBAME e-Science Journal vol.13

2015年3月10日 東京工業大学 学術国際情報センター発行 ©

ISSN 2185-6028

デザイン・レイアウト：キックアンドパンチ

編集： TSUBAME e-Science Journal 編集室

青木尊之 渡邊寿雄 佐々木淳 仲川愛理

住所：〒152-8550 東京都目黒区大岡山2-12-1-E2-6

電話：03-5734-2085 FAX：03-5734-3198

E-mail：tsubame_j@sim.gsic.titech.ac.jp

URL：http://www.gsic.titech.ac.jp/

TSUBAME

TSUBAME 共同利用サービス

『みんなのスパコン』 TSUBAME共同利用サービスは、
ピーク性能 5.7PFlops、18000CPUコア、4300GPU搭載
世界トップクラスの東工大のスパコンTSUBAME2.5を
東工大以外の皆さんにご利用いただくための枠組みです。

課題公募する利用区分とカテゴリ

共同利用サービスには、「学術利用」、「産業利用」、「社会貢献利用」の3つの利用区分があり、
さらに「成果公開」と「成果非公開」のカテゴリがあります。

ご利用をご検討の際には、下記までお問い合わせください。



他大学や公的研究機関の研究者の [学術利用](#) [有償利用]

民間企業の方の [産業利用](#) [有償・無償利用]

その他の組織による社会的貢献のための [社会貢献利用](#) [有償利用]

共同利用にて提供する計算資源

共同利用サービスの利用区分・カテゴリ別の利用課金表を下記に示します。TSUBAMEにおける計算機資源の割振りは口数を単位としており、1口は標準1ノード(12 CPUコア、3GPU、55.82GBメモリ搭載)の3000時間分(≈約4ヶ月)相当の計算機資源です。

1000 CPUコアを1.5日利用する使い方や、100 GPUを3.75日利用する使い方も可能です。

利用区分	利用者	制度や利用規定等	カテゴリ	利用課金(税抜)※
学術利用	他大学または研究機関等	共同利用の利用規定に基づく	成果公開	1口:120,000円
産業利用	民間企業を中心としたグループ	「先端研究基盤共用・プラットフォーム形成事業」に基づく	成果公開	トライアルユース(無償利用) 1口:120,000円
			成果非公開	1口:480,000円
社会貢献利用	非営利団体、公共団体等	共同利用の利用規定に基づく	成果公開	1口:120,000円
			成果非公開	1口:480,000円

※ 平成27年度の利用課金です。最新の利用課金については、下記 URL をご参照ください。
<http://www.gsic.titech.ac.jp/kyodou/kakin>

産業利用トライアルユース制度（先端研究基盤共用・プラットフォーム形成事業）

東工大のスパコンTSUBAMEを、より多くの企業の皆さんにご利用いただくため、初めてTSUBAMEをご利用いただく際に、無償にてご試用いただける制度です。

(文部科学省先端研究基盤共用・プラットフォーム形成事業による助成)

詳しくは、下記までお問い合わせください。

お問い合わせ

- 東京工業大学 学術国際情報センター 共同利用推進室
- e-mail kyoyo@gsic.titech.ac.jp Tel. 03-5734-2085 Fax. 03-5734-3198
- 詳しく述べ <http://www.gsic.titech.ac.jp/tsubame/> をご覧ください。