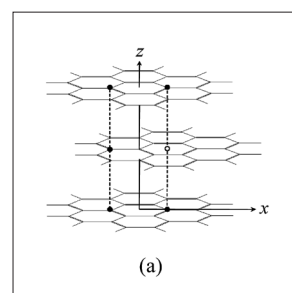
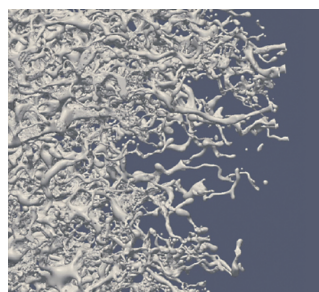


TSUBAME

ESJ.



大規模GPUコンピューティングによる 天然岩石内の二相流シミュレーション

Two-phase Porous Flow Simulation for Natural Sandstone on GPU Supercomputer

大規模画像データセットの機械学習のための 分散コンピューティング

Distributed Computing for Machine Learning on Large-Scale Image Dataset

マルチGPU超並列クラスタシステムを用いた 大規模ナノ炭素分子の電子状態計算

Electronic structure calculation of large nano carbon molecules
using multi-GPU massively parallel cluster system

大規模GPUコンピューティングによる天然岩石内の二相流シミュレーション

辻健 蔣飛

九州大学カーボンニュートラル・エネルギー国際研究所

GPUスパコン (TSUBAME 2.5) を用いることにより、格子ボルツマン法 (LBM) による2相の間隙流体シミュレーションを、巨大なデジタル岩石 (1000³グリッド) に適用することが可能となった。計算領域のサイズアップによって、これまで間隙スケール (μm スケール) で実施していた計算を、岩石スケール (mmスケール) で実施できるようになった。それによりLBMの計算結果を、実験結果と比較し、計算結果を検証することが可能となった。岩石間隙内の2相流を扱う例として、CO₂地中貯留がある。このプロジェクトでは、貯留層内での水—CO₂の2相流挙動が重要となる。その挙動は貯留層の条件 (例えば界面張力、間隙形状、濡れ性、水圧など) に強く依存する。今回開発したアプローチを用いて、様々な貯留層条件でCO₂の挙動を計算することで、効果的にCO₂を貯留できる条件を明らかにした。さらに貯留層に圧入したCO₂の挙動を考慮して、CO₂が鉱物化する過程をモデル化し、それが浸透率といった水理特性に与える影響を評価することも可能となった。

はじめに

1

多孔質媒質内の流体挙動は、理学から工学にわたる様々な分野で取り扱われる。特に、岩石内部の2相流の挙動は、石油の増進回収 (EOR) や、二酸化炭素 (CO₂) の回収・貯留 (CCS)、難水溶性汚染物質 (NAPL) の除去、地熱開発等で重要となる。ここでは、2相流シミュレーションを、CCSの諸問題に適用した事例を紹介する。

CCSプロジェクトは、CO₂の大規模排出源でCO₂を回収し、地下深部の貯留層 (岩石) の間隙にCO₂を注入することで、永久的にCO₂を地下に閉じ込める技術である^[1]。CCSは、大気中のCO₂を削減できる近未来的な技術として注目されている。貯留層に圧入 (貯留) したCO₂の挙動は、多孔質媒質内の2相流として扱われる。CO₂の圧入は、非濡れ性流体のCO₂が、岩石間隙内の水 (濡れ性流体) を押し出す排水過程 (Drainage process) と考えることができる。一方で、CO₂圧入が終了した後は、地層水がCO₂を押し出す浸潤過程 (Imbibition process) を考える必要がある。このCO₂貯留において、間隙スケールでみられるHaines jumpや、スナップオフ、フィンガリング現象は、貯留層内でのCO₂の安定度、浸透特性、飽和度などに関係する。そのため、間隙スケールのCO₂の挙動を評価することは、CO₂の貯留能力、漏洩リスクの評価、効率的なCO₂貯留を実現する上で、不可欠であると考えられている。

近年、格子ボルツマン法 (Lattice Boltzmann Method; LBM) が、多孔質内の流体挙動を計算する手法として注目されている^[2]。この手法は、一般的な数値力学的手法とは異なり、流体を衝突と移動を繰り返す多数の粒子の集合体と捉え、粒子の動きを計算することによって流体の運動を求める数値計算法である。

既存の多相流モデルとして、Level set 法^[3]、Volume of fluid (VOF) 法^[4]、Phase-field 法^[5]などがあるが、一般に界面を取り扱うために複雑なアルゴリズムを必要とする。一方、LBMでは界面を分離することが容易であり、また界面を操作する必要もない。さらにLBMでは、複雑な境界を取り扱うことができる。このような特徴から、LBMは複雑な間隙形状を持つ天然岩石内の多相流挙動を計算する上で、適当なアプローチといえる。

しかし多くのLBMシミュレーションでは、計算効率の制限から、低解像度かつ小さなスケールで実施されることが多かった。例えば、Ahrenholz et al.^[6]は、11 μm の解像度で200³グリッドの砂岩モデルに対してLBMの計算を実施している。またRamstad et al.^[7]も、6 μm の解像度で256³グリッドの砂岩などのデジタルロックに対して、LBMを適用している。

近年、Graphics Processing Unit (GPU) が科学計算にも利用され、その大規模並列計算が一般的になってきている。GPUを利用した計算では、並列計算用アーキテクチャであるCUDAを利用する。過去数年間に、単相のLBM計算をGPUに実装した研究が実施され、CPUを用いた場合に比べて、パフォーマンスが大きく向上することが示されている^[8,9,10]。

本研究では、2相のLBMの計算コードをGPUに実装し、三次元多孔質媒質 (天然岩石のデジタルモデル; 図1) 内の多相流挙動を調べた。デジタル岩石モデルの構築は、高解像度のマイクロCTスキャナを用いて岩石サンプルの間隙構造の断面画像 (2D) を取得し、補間することによって解像度を落とさずにバイナリー三次元計算格子を生成した。この巨大岩石モデルに対して、GPUスーパーコンピューターを用いた2相のLBMシミュレーションを実施した。

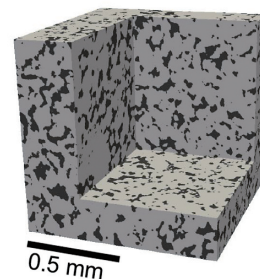


図1 マイクロX-CT画像から構築されたベリア砂岩のデジタルロック。黒い部分が間隙であり、CCSプロジェクトでは、この間隙にCO₂を貯留する。

格子ボルツマン法と GPU 実装

2

格子ボルツマン法 (LBM) は近年、多孔質媒体内の流体挙動の有効な計算手法として注目されている。今回の計算では、RK color 2相のLBMモデル^[11]を用いた。このモデルでは、MRT (multiple-relaxation-time) モデルなどの改良を加えることで、2相流の粘性比の高い条件でも安定して計算することができる。境界条件は、Non-slip境界条件をBounce-backスキームで実装した。濡れ性は、固体格子の擬似カラー関数で再現した。

LBMには、大規模GPU並列計算を行う上で、いくつかの利点がある。まずPoisson方程式を解かない陽解法であること、かつ局所的な計算がメインであることである。また界面の追跡にフェーズフィールド法の概念を取り込み、並列化した計算が容易である。

マルチGPUの実装は、CUDAコアレベルと、GPU間のレベルの2段階に分けて実施した。シングルGPUチップ内部の並列計算アルゴリズム (CUDAコアレベル) は、マルチスレッド処理と呼ばれる。CUDAはソフトウェア上で、大量のコアを、スレッドとブロックの構造で管理している。並列演算が可能な最小単位をスレッド、スレッドの集合体をブロックと呼ぶ (図2)。計算では、各スレッドが担当する格子を指定することが重要である。すなわち、スレッドIDと、更新される格子の空間座標を示さなければならない。本研究では、図2aに示すように、z方向の一行の格子点の更新を1スレッドで行うこととした。1ブロックが担当する計算領域は、blockDimx, blockDimyのパラメータによって決められる。図2aに示すように、ブロックをx、y平面にマッピングした。従って、x、y方向のブロック数は、それぞれNx/blockDimx, Ny/blockDimyである (Nx, Nyは計算領域x、y方向のメッシュ数)。これにより、関数内のx、yループが必要なくなり、GPUスケジューラが自動的に複数のコアを利用して、並列計算を行うことが可能となる。スレッド、ブロックID (threadID.x, threadID.y, blockID.x, blockID.y) と格子座標 (jx, jy) の関係は、次式で表される。

$$jx = blockDim.x \times blockIdx.x + threadIdx.x; \quad (1)$$

$$jy = blockDim.y \times blockIdx.y + threadIdx.y. \quad (2)$$

GPUメモリ内に確保されている配列のデータレイアウトも、計算を効率的に行う上で重要となる。GPUのメモリバンド幅を無駄なく活用するため、コアキャッシングとよばれるデータまとめて転送する方式の利用が重要となる。ここでLBMの分布関数は、以下のレイアウトで保存される。

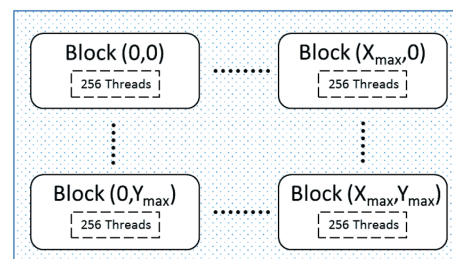
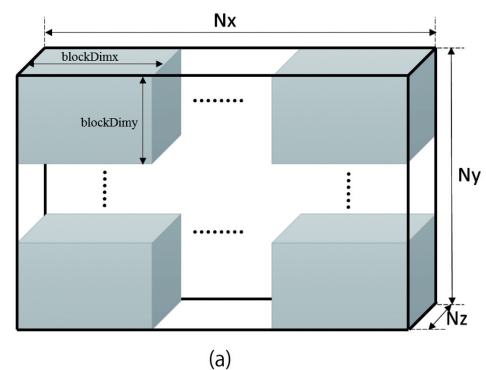
$$i \times L_z \times L_y \times L_x + jz \times L_y \times L_x + jy \times L_x + jx \quad (3)$$

またデータアクセス量を減らすため、衝突過程と移流過程を統合した。これにより、計算効率を、30%も向上することに成功した。

大規模計算を行う際には、メモリを節約するため、計算する必要のない固体メッシュの情報を保存しない方法を用いた。さらにストリームパターンを工夫し、一つだけの配列に分布関数を保存する方法をとった。

マルチGPU計算では、はじめに計算領域を分割する必要がある。今回は、流体格子数を均等に割り、z方向に分割する方法を用いた。分割した領域の境界では、一層のGhostレイヤーが設けられ、これは領域間のデータ転送かつカラー勾配の計算に用いられる。GPU間のデータ転送は、2つのパターンがある (図3)。同一のマザーボード (ノード) に装着しているGPUボード間のデータ転送はPeerToPeer方式で行う。一方で、異なるノード間のデータのやり取りはMessage Passing Interface (MPI) に準じて実施する。データ転送を効率するために、CPUホスト側のメモリ保存は、Pinnedメモリ方式で行った。このようなコードの最適化によって、2相流シミュレーションの計算効率を大幅に向上させることに成功した。

以上のようにLBMの計算コードを効率化し、さらにTSUBAME 2.5スパコンの20ノード (60枚Tesla-K20 GPU) を利用することで、1000³グリッドという大きな領域でCO₂圧入シミュレーションを実現した (図4)。このような巨大グリッドモデルの計算を行うことができれば、長さ数ミリメートルの岩石内部のCO₂の挙動を調べることができるようになる。このように計算領域のサイズアップを行うことで、間隙スケール (μmスケール) から岩石スケール (mmスケール) の数値シミュレーションを実施することが可能となり、実験室の結果と比較した議論を行うことができるようになった。



(b)

図2 GPU内部のマルチスレッド化。

大規模GPUコンピューティングによる天然岩石内の二相流シミュレーション

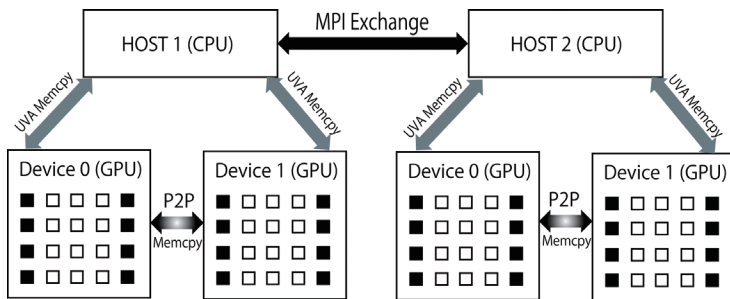


図3 マルチGPU間のデータ転送モデル。
黒い四角は、Ghostレイヤーを示している。

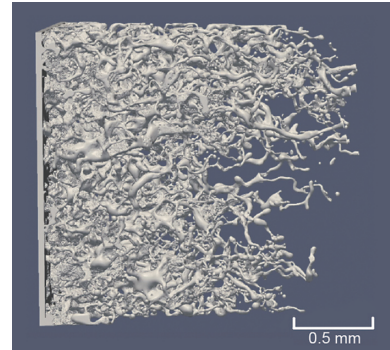


図4 2相のLBMシミュレーション結果（グリッドサイズ：1000³）。白い部分が空隙内に圧入したCO₂（非濡れ性流体）である。ここでは、岩石構成粒子と水は透明にしている。右側から左側に向かってCO₂を圧入している。

適用例

3

2相のLBMシミュレーションを使って、CO₂ 地中貯留における課題を扱った研究成果を紹介する。ここでは (1) 貯留層条件が、CO₂-水の挙動に与える影響を調べた研究、(2) CO₂ を圧入した岩石に再び水を圧入する湿潤過程を計算して、効果的にCO₂ 残留トラップや溶解トラップを引き起こす貯留層条件を調べた研究、(3) CO₂ の挙動から鉱物化をモデリングし、その鉱物化が流体挙動や浸透率に与える影響を明らかにした研究を紹介する。

3.1 様々な貯留層条件での多相流の挙動

貯留層内の流体挙動をコントロールするパラメータ（例えば温度、界面張力、空隙形状、濡れ性、水圧など）は多い。ここではデジタル岩石を用いたアプローチにより、貯留層条件に依存した流体挙動特性（浸透率など）を定量的に評価した研究を紹介する（図5）^[12,13]。

界面張力を変化させてCO₂の挙動などを調べたところ、界面張力の小さい場合（図5a）のほうがCO₂は速く流れることが分かった。つまり、浸透率が高く、圧入効率が良いことになる。実際に、CO₂の挙動から相対浸透率を計算すると、界面張力が増加するにつれて、浸透率が減少することが明らかになった（図5c）。特に、非濡れ性流体であるCO₂の浸透率のほうが、界面張力の影響を強く受けることが分かった。一方で、界面張力が大きい場合（図5b）のほうが、最終的なCO₂の飽和度は大きくなることが分かった。つまり、界面張力が大きいほうが、限られた空隙スペースを有効に使ってCO₂を貯留できることになる。ここでは、界面張力を変化させた例（図5）だけを紹介したが、これ以外にも様々な条件下で、CO₂の

挙動を推定できている。これらの結果から、今回開発した2相のLBMシミュレーションを用いれば、様々な貯留層の条件におけるCO₂の飽和度や浸透率を定量的に推定できることが示された。

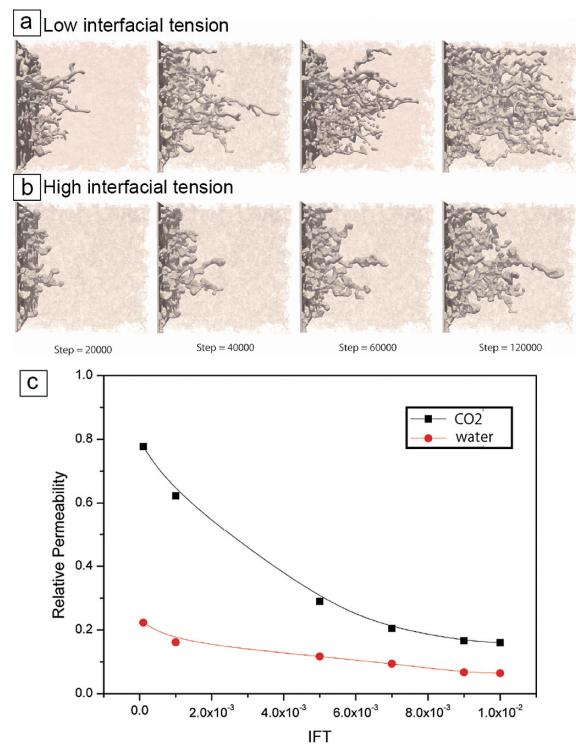


図5 界面張力 (IFT) に依存したCO₂の挙動^[12]。
(a) 低い界面張力の場合。
(b) 高い界面張力の場合。
(c) 水とCO₂の飽和度が50%の場合の相対浸透率と界面張力の関係

3.2 残留・溶解トラップが効果的に働く貯留層条件の推定

貯留層内でのCO₂の残留トラップ量を定量的に調べるため、LBMシミュレーションを用いて界面張力が残留CO₂クラスターに与える影響を調べた。CO₂の圧入(図5)が終了した後に、水を圧入するシミュレーションを実施し、空隙内に残留するCO₂の量と分布を調べた(図6)^[14]。この残留CO₂の量が多いほど、安定的に岩石空隙内にCO₂を貯留できていることになる。また残留CO₂クラスターのサイズ、長さ、接触面積の分布を計算し、界面張力がこれらの性質に対する影響について検討を行った。

計算の結果、界面張力が増えるにつれて、残留CO₂クラスターのサイズが大きくなり、トラップ量が増加する傾向があることが分かった(図6c)。一方で、低い界面張力では、クラスターが小さく、CO₂は広い表面積を持つことが分かった(図6a)。つまり、この条件は、CO₂の地層水への溶解(溶解トラップ)にとって有利であると考えることができる。この方法によって、異なった貯留層条件における残留CO₂の状態を予測することができる。つまり、安定したCO₂地中貯留を実施する上で、適当な貯留層条件を明らかにすることができる。

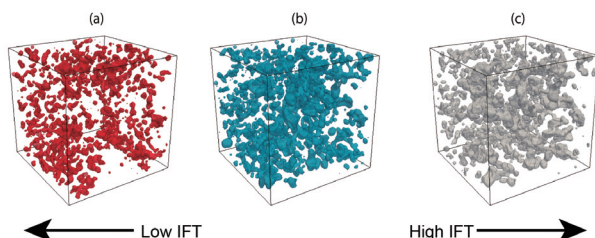


図6 残留したCO₂の分布^[14]。左側から右側へ界面張力が高くなる。ここでは、岩石構成粒子と水は透明にしている。

3.3 CO₂の鉱物化モデリング

圧入したCO₂は、最終的に鉱物化すると考えられている。ここではCO₂の鉱物化をモデル化し、それが空隙流体挙動(例えば浸透率)に与える影響を調べた^[15]。まずLBMで計算された流速場から、移流拡散方程式を用いることで、各グリッドでのCO₂濃度を計算した。その濃度が、ある閾値を超えた場合に、そのグリッドを鉱物(炭酸塩鉱物)に変化させた。その後、鉱物化した岩石モデルに対して、再度、LBMの流体シミュレーションを実施した。この操作を各時間ステップで行い、鉱物化の進行を再現した(図7)。なお、この鉱物沈殿の計算結果を、実験結果と比較することによって、鉱物化モデルに用いるパラメータを正確に決定することができた。

鉱物化の計算結果から、鉱物化が生じる場所は、流速だけに依存しているのではなく、不均質な空隙形状にも依存していることが明らかとなった(図7)。鉱物化に伴う浸透率の変化を計算すると、絶対浸透率は鉱物化によって大きく影響を受けること、CO₂の相対浸透率は大きく低下することが分かった。一方、水の相対浸透率は、あまり変化しないことが分かった。これは鉱物化によって、空隙(Pore throat)の径が小さくなり、そこを通過するのに要するCapillary pressureが増加する。それにより、CO₂といった非濡れ性流体の移動が制限され、浸透率が大きな影響を受けると考えられる^[16]。

今回、デジタル岩石内で生じる鉱物化や、それが水理特性に与える影響を調べることができた。この鉱物モデリングで得られる情報を、一般的に実施されるフィールド(km)スケールの貯留層シミュレーションに組み込めば、鉱物反応を考慮して、長期的なCO₂の挙動予測が可能となる。

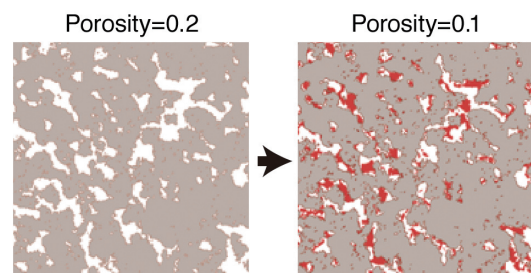


図7 空隙流体挙動を考慮した鉱物化シミュレーションの結果^[15]。赤色部はCO₂が化した炭酸塩鉱物を示し、白色部は空隙を示す。左側は鉱物化前の空隙形状である。

まとめ

4

GPU大規模計算により、2相の格子ボルツマン法を巨大なデジタル岩石モデルに適用し、間隙内の流体挙動を計算することに成功した。今回のようなLBMによる流体挙動シミュレーションでは、マルチGPUの利用が効果的である。

今回開発したアプローチを、CO₂貯留プロジェクトの課題に適用した結果、次のことが明らかになった。(1) 貯留層の状態が、多相流の挙動に与える影響を明らかになった。(2) CCSプロジェクトにおいて残留CO₂トラップや溶解CO₂トラップが効果的に働く条件を明らかにすることができた。(3) CO₂の間隙内での挙動からCO₂の鉱物化をモデル化し、その鉱物化が浸透率に与える影響を調べることができた。また、ここでは紹介しないが、本研究で用いたデジタル岩石の弾性定数を、弾性波動シミュレーションで推定することも可能となってきた^[17]。

このようにデジタル岩石を用いた大規模計算によって、貯留層内でのCO₂の挙動や反応を再現することが可能となってきた。近い将来、CO₂地中貯留をはじめとした地下流体を扱うプロジェクトでは、貯留層全体をデジタル化し、貯留層内で生じている挙動や反応を正確にモデル化することが可能になると考えられる。それにより、貯留層を有効に利用または正確にコントロールできるようになると考えられる。

謝辞

本研究は、日本学術振興会 (JSPS) の新学術領域公募研究 (no. 15H01143)、基盤研究A (no. 24246148)、スタートアップ (no. 26887028)、エジプトMOSRとの二国間共同研究、JICA/JSTのSATREPSプロジェクト、文部科学省の世界トップレベル研究拠点 (WPI) プログラムの助成を受けて実施したものである。

参考文献

- [1] Metz, B., et al. eds. (2005), IPCC Special Report, Carbon Dioxide Capture and Storage. Cambridge University Press, Cambridge.
- [2] Boek, E.S., M. Venturoli (2010), Lattice -Boltzmann studies of fluid flow in porous media with realistic rock geometries. *Comput. Math. Appl.* 59, 2305–2314.
- [3] Sussman, M., A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome (1999), An adaptive level set approach for incompressible two-phase flows. *J. Comput. Phys.* 148, 81–124.
- [4] Hirt, C.W., B.D. Nichols (1981), Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.* 39(1), 201–225.
- [5] Badalassi, V.E., H.D. Ceniceros, S. Banerjee (2003), Computation of multiphase systems with phase field models. *J. Comput. Phys.* 190(2), 371–397.
- [6] Ahrenholz, B, J. Tölke, P. Lehmann, A. Peters, A. Kaestner, M. Krafczyk, W. Durner (2008), Prediction of capillary hysteresis in a porous material using lattice-Boltzmann methods and comparison to experimental data and a morphological pore network model, *Advances in Water Resources*, 31 (9), 1151–1173.
- [7] Ramstad, T, N. Idowu, C. Nardi, P.E. Øren (2012), Relative permeability calculations from two-phase flow simulations directly on digital images of porous rocks, *Transport in Porous Media*, 94 (2), 487–504.
- [8] Tölke, J., M. Krafczyk (2008), TeraFLOP computing on a desktop PC with GPUs for 3D CFD. *Int. J. Comput. Fluid Dyn.* 22(7), 443–456.
- [9] Tölke, J. (2010), Implementation of a lattice Boltzmann kernel using the compute unified device architecture developed by nVIDIA. *Comput. Vis. Sci.* 13(1), 29–39.
- [10] Kuznik, F., C. Obrecht, G. Rusaouen, J.J. Roux (2010), LBM based flow simulation using GPU computing processor. *Comput. Math. Appl.* 59(7), 2380–2392.
- [11] Tölke, J., S. Freudiger, M. Krafczyk (2006), An adaptive scheme using hierarchical grids for lattice Boltzmann multi-phase flow simulations. *Comput. Fluid* 35, 820–830.
- [12] Jiang, F., T. Tsuji, and C. Hu (2014), Elucidating the role of interfacial tension for hydrological properties of two-phase flow in natural sandstone by an improved lattice Boltzmann method, *Transport in Porous Media*, 104, 1, 205–229.
- [13] Yamabe, H., T. Tsuji, Y. Liang, and T. Matsuoka (2015), Lattice Boltzmann simulations of supercritical CO₂-water drainage displacement in porous media: CO₂ saturation and displacement mechanism, *Environmental Science & Technology*, 49 (1), 537–543.
- [14] Jiang, F., and T. Tsuji (2015), Impact of interfacial tension on residual CO₂ clusters in porous sandstone, *Water Resources Research*, 51, 1710–1722.
- [15] Jiang, F., and T. Tsuji (2014), Changes in pore geometry and relative permeability caused by carbonate precipitation in porous media, *Physical Review E* 90, 053306.
- [16] Kitamura, K., F. Jiang, A.J. Valocchi, S. Chiyonobu, T. Tsuji, and K.T. Christensen (2014), The study of heterogeneous two-phase flow around small-scale heterogeneity in porous sandstone by measured elastic wave velocities and lattice Boltzmann method simulation, *J. Geophys. Res.* 119, 7564–7577.

- [17] Yamabe, H., T. Tsuji, Y. Liang, T. Matsuoka (2016), Influence of fluid displacement patterns on seismic velocity during supercritical CO₂ injection: Simulation study for evaluation of the relationship between seismic velocity and CO₂ saturation, International Journal of Greenhouse Gas Control, 46, 197-204.

大規模画像データセットの 機械学習のための分散コンピューティング

佐藤 育郎* 渡邊 隆太郎* 西村 裕紀** 野村 哲弘*** 松岡 聡***

*株式会社デンソーアイティラボラトリ **株式会社デンソー ***東京工業大学

ディープニューラルネットワーク(DNN)と称される機械学習手法を使って、大規模データセットの教師あり学習を行うことで、非常に高い識別能力を獲得できることが、最近の研究によって明らかとなってきた。本研究は、大規模データセットにおいて、複数のDNNモデルを学習・評価することで性能の良いモデルの条件を定量化し、また優良な複数のモデルをアンサンブル化することで、高い識別能力を持つ画像識別器を生成することを目的としたものである。学習には独自開発した非同期型分散学習プログラムを使用し、1モデルあたり最多で96GPUを利用して高速な学習を行った。学習はTSUBAME グランドチャレンジを利用して行われ、ピーク時で1146台のGPUを使い、微分計算部分においては1GPUあたり約1TFLOPSの単精度浮動小数計算性能を引き出した。

はじめに

1

画像識別は情報技術分野における主要なテーマのひとつである。産業応用の例としては、監視カメラにおける人の識別、生産設備における製品の外観検査、自動車の自動緊急ブレーキ装置における障害物検知などが挙げられる。特に人の安全にかかわる分野においては、極めて高い識別精度が要求されるため、技術的な革新が求められている。

一般的に、機械学習を用いた識別器の性能は、使用するアルゴリズム、データセットのサイズ、計算機の能力に依存する。アルゴリズムにおいては、あらかじめ決められた手順で画像特徴を抽出する従来のアプローチよりも、学習によってその手順を決定するDNNの方が、物体識別のタスクにおいて圧倒的に高い識別能力を有することが、Krizhevskyらによって実証された^[1]。このブレイクスルーの背景にデータセットの大規模化がある^[2]ことは注目に値する。事実、小さなデータセットではDNNは従来のアプローチに性能で劣ることがあることが報告されている^[3]。DNNの最適化には多大な演算量が要求されるため、GPUの利用が現在では標準的である。それでもなお、大規模データセットの学習に、単体のGPUで1ヶ月またはそれ以上の期間を費やすことはそれほど珍しいことではない。

こうした背景から、我々は自動車の走行安全への応用に向けて、ノード分散型のDNN学習プログラムを開発し、単体の計算機に比較して数十倍の速度で反復計算を実行可能であることをTSUBAME2.5上で確認した。

本研究では、ノード分散型の学習プログラムを大規模データに適用し、(1) DNNのモデル選択とアンサンブル化による高精度画像識別器を、TSUBAME2.5で比較的短期間に生成した。また、(2) モデルの学習パラメータ数と識別性能との関係を定量的に明らかにした。(2)については現段階では予備的結果であるものの、パラメータ数を広範囲に分布させた結果、パラメータ数と識別性能の間にある興味深いトレンドが見えてきた。

ソフトウェア

2

DNNの学習アルゴリズムの分散には、一般的にモデル並列とデータ並列の二種の方針がある。モデル並列とは、1つのモデルを複数のGPU(ここでの議論はGPUの利用を前提とする)に分散させ必要に応じて変数を相互に通信し同期を取って処理することで、コスト関数の微分計算を行うものである^[1,4,5]。微分計算に必要とされる全変数の容量が1台のGPUのメモリを超える場合、この方針が取られることになる。つまり比較的大きなネットワークを使用する際に不可欠となる並列化方法である。もう一方のデータ並列は、一度の微分計算に使用する複数のデータを、複数のGPUに分散させ処理するものである^[4,5]。プログラムにおいては、モデル並列のみ^[1]、または本研究のようにデータ並列のみを実現することも可能であるし、またその両方を実現することも可能である^[4,5]。

データ並列には、さらに同期型^[6]のもの、非同期型^[4]のものが有り得る。同期型の場合、全ワーカーGPUが、それぞれ画像を読み込み、その画像に基づいた微分計算を終えたのち、通信によって微分を加算し、重みパラメータを更新し、最新の重みを全ワーカーに配布するサイクルを反復することとなる。利点は、単体のGPUで行う学習に比べて単位時間あたりに処理できる画像枚数が大きく出来る点と、収束の保証がある点にある。一方、非同期型の場合、計算機のリソースを間断なく利用するため、同期型に比べてさらに反復計算の速度を向上させることが可能である利点がある一方、「古い」重みパラメータを使った微分を使うことから理論的に収束の保証がないといった欠点が挙げられる。もっとも、後者の欠点は実質的には問題を起こさないという指摘もある^[4]。

データ並列には、同期/非同期に関わらず、さらに、重みパラメータの更新のための「パラメータサーバ」と呼ばれる特殊ノードを置く方式^[4]と、パラメータサーバの機能をワーカーノードに分散させた、特殊ノードを置かない方式^[5,6]の2種類がある。これ以上の説明は、本文の主旨とは若干隔たりがあるため、割愛する。

本研究では、特殊ノードを置かない方式の非同期型データ並列のみを実現した、独自開発のプログラムを使用した。分散アルゴ

リズムの詳細と学習速度の定量評価については、別の機会で論ずる予定である。

データセット

3

本研究で使用したILSVRC2012データセット^[2]は、ImageNetと呼ばれるデータセットのサブセットであり、128万学習サンプルと5万検証サンプルと10万枚のテスト画像を含んでいる。これは、公開の画像識別データセットとしては最大級のもののひとつである。ここでいうサンプルとは、画像と正解クラスの組みを指している。クラス数は1000である。なおクラスあたりの画像枚数は一律ではない。テスト画像に対する正解クラスは開示されていない。テストサーバに識別結果を投稿することで、テストセット全体の誤識別率が公表される仕組みとなっている。

DNN モデル

4

本検討におけるモデル設計の指針について述べる。パラメータ数と識別性能の関連を明らかにしたいため、パラメータ数は学習サンプル数の1倍から30倍程度の広い範囲で設定した。活性化関数による非線形写像を含む畳み込み層を連続的に反復させることで、ネットワークの表現能力を高められるという近年の知見^[7-9]に従って、畳み込み層は11層から48層の範囲で設定した。畳み込みのフィルタサイズは一律3ピクセル四方とした。各層の特徴マップ(畳み込みなどの操作によって得られる2次元信号のこと)の数は、畳み込み層の層数とパラメータ数の設計を元に、手で設定した。本検討においては、全結合層は最終隠れ層と出力層の間の一層のみとした。入力画像サイズは、目視による検査の結果200ピクセル以上に設定することとした。これは、特徴マップの解像度を下げるプーリング層を畳み込み層と畳み込み層の間に適当に挟むことで実現できる。プーリング層の層数は2から5の範囲である。参考までに、本検討で使用したモデルをひとつ取り上げ、そのネットワーク構造を表1に記載する。

全結合層について補足する。通常、畳み込みネットワークでは、マップの解像度を十分に低下させたのち、全ニューロンを使った線形結合の層(全結合層)を複数層(3層程度)設けるのが標準的である(例えば^[1,8])。これらの層では、ニューロン毎に特定のパラメータが学習されるため、多量のパラメータを必要とし、結果として過学習を生じさせやすくする傾向がある。こうした過学習を防ぐた

めに、全結合層にdropout^[10]などの正則手段を適用することがよく見受けられる(例えば^[1,8])。本検討では全結合層の使用は最後の一層のみとし、また、最後のマップサイズは、フィルタサイズと同じ3×3とおいた。このようにして、全結合に必要なパラメータ数を抑えた。我々の行った予備実験において、表1のようなネットワークは、複数の全結合層を持つ似たようなネットワークに比べて、パラメータが少なくも関わらず高い汎化性能を持つことを確認している。

最適化の詳細については表2に記載する。分散計算には1モデルあたり最少で24GPU、最多で96GPUを使用した。本研究で行ったモデルの多くは、TSUBAMEのグランドチャレンジを利用して学習したものである。モデルあたりの分散学習日数はおおむね7日である。グランドチャレンジでは、ピーク時で1146台のGPUを使い、13モデルを同時に学習した。特定のDNNの学習部分(順伝搬と逆伝搬)において、1GPU(NVIDIA Tesla K20X, 単精度浮動小数計算理論性能3.95TFLOPS)あたり0.98TFLOPSの単精度浮動小数計算性能を実測した。

layer	1	2	3	4	5	6	7	8	9
#maps	3	32	38	54	64	90	106	150	178
map size	396 ²	394 ²	196 ²	194 ²	96 ²	94 ²	46 ²	44 ²	42 ²
operation	C	CP	C	CP	C	CP	C	C	CP

layer	10	11	12	13	14	15	-
#maps	212	298	354	421	593	704	1000
map size	20 ²	18 ²	16 ²	7 ²	5 ²	3 ²	1 ²
operation	C	C	CP	C	C	C	S

表1 本検討で使用した、畳み込み層15層、プーリング層5層、パラメータ数16M個を持つネットワーク構造。
Cは畳み込みを、CPは畳み込みとプーリングを、Sはソフトマックス関数を意味する。活性化の記載は省略している。層の番号は、パラメータを持つ層に対して与えられている。

活性化関数	ReLU [11]
コスト関数	クロスエントロピー (ソフトマックスによる正規化あり)
Dropout (any variants)	なし
最適化方法	ミニバッチによる確率的勾配法
ミニバッチサイズ	アンサンブルに使用したものの中央値: 132
反復回数	アンサンブルに使用したものの中央値: 515 エポック
データ拡張	ランダムなホモグラフィ変換・クロッピング・左右反転・radial distortion・弱い弾性変形など
識別則	APAC[12]

表2 最適化の詳細

定性評価結果

5

適切に識別できている検証用画像の例を図1に提示する。図1 (a)の正解ラベルは「一輪車」である。このサンプルでは、注目物体である一輪車の車輪部分が画像外にあり、シャフトがかるうじて見える程度という点に難しさがある。しかしながらアルゴリズムは高い確度で識別に成功できている。これは、上を見上げる群集と、

中央にいるパフォーマーのポーズなどから、注目物体の種別を推測した結果であると思われる。つまり、DNNが画像全体から、シーンのコンテキストを読み取れていることが分かる。同様の分析は図1 (b)の画像（正解ラベルは「パドル」）にも当てはまる。図1 (c)はシャボン玉の識別に成功した事例を示している。従来、半透明物体の識別は難しいとされてきた。透けた部分にある背景の画像特徴が混入してしまうことで、識別対象の特徴が「汚染」されてしまうことが識別を困難にしていると考えられる。しかしながら我々のモデルはシャボン玉を高スコアで識別できており、こうした困難さをDNNが克服していることが窺える。

図1 識別に成功できたImageNet検証データの例。



(a) unicycle



(b) paddle



(c) bubble

識別に失敗している検証用画像の例を図2に提示する。図2 (a)では、「アラビアンキャメル」が正解であるところを、「犬ゾリ」と誤答している。これは、DNNがコンテキストを過剰に評価した結果、画像にうつるアラビアンキャメルの頭部の特徴が過少評価されたものと推測される。図2 (b)は「トリケラトプス」が正解であるが、我々のモデルはこれを識別できていない。正面方向から見たトリケラトプスの形状が識別できていないことから、汎化に限界があることが窺える。

図2 識別に失敗したImageNet検証データの例。



(a) Arabian camel



(b) triceratops

定量評価結果

識別性能とパラメタ数の関係を図3に記載する。これによると、パラメタ数が $8 \times 10^6 \sim 40 \times 10^6$ の範囲にあるとき識別性能は同等程度に良好だが、パラメタ数が 8×10^6 を下回るにつれ識別性能が急速に劣化する傾向が確認できる。使用したモデルのネットワーク構造は様々であるにも関わらず、このような傾向が確認できたということは、識別性能においてパラメタ数の影響が支配的であることを示唆している。分散計算環境における計算速度を考慮に入れると、パラメタ数の少ないモデルの使用が好ましい。このため、今回の検討からは、 8×10^6 個程度のパラメタ数を持ったモデルの適用が理に適うこととなる。

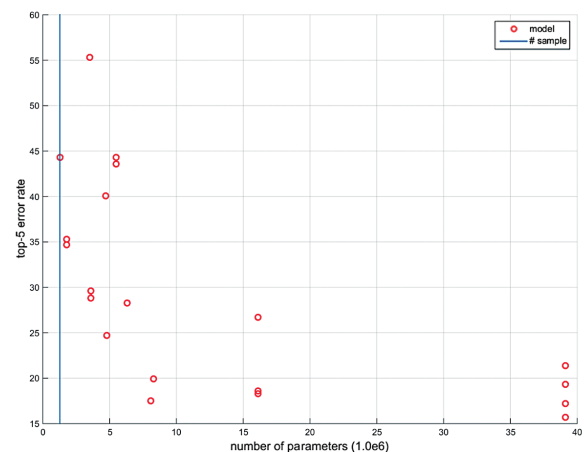


図3 識別性能とパラメタ数の関係。
青線は学習サンプルの個数を示す。

この結果が、機械学習の古典的な考えとはいくぶん乖離した結果であることは指摘に値する。古典的には、モデルのパラメタ数は学習サンプル数を下回るよう設定することが、過学習による性能劣化を防ぐために重要であると考えられている。しかしながら今回の結果を見る限り、パラメタ数が学習サンプル数を下回ることで良い識別性能が得られるとは推論しがたく、学習サンプル数の6倍程度またはそれ以上のパラメタ数を持つモデルが良好であることが確認できる。DNNがこれほど多数のパラメタ数を必要とする原因はまだ明らかにできていないため、今後の調査課題としたい。

DNNモデルを網羅的に組み合わせて生成したアンサンブル識別器のうち、表3に記載した6モデルからなるものが、ImageNetの検証用データにおいて優良な識別性能を示した。ここでのアンサンブル識別器とは、ある入力に対するソフトマックス出力の対数の期待値

の大きさに従ってクラス分類を決定するものである。アンサンブル化によりおよそ2%の性能向上が確認できた。これは、各モデルがそれぞれ少しずつ違った特徴を抽出していることを意味している。

モデル			Top-5 error rate (%) on validation set	
畳み込み層数	プーリング層数	パラメタ数	モデル毎	アンサンブル
11	4	16M	18.62	13.67
11	4	39M	17.20	
11	4	39M	15.68	
15	5	16M	18.26	
16	4	8M	17.51	
16	5	16M	26.65	

表3 本研究で使用したアンサンブル識別器の構成モデル。

代表的な手法との比較結果について述べる。テストサーバによる識別性能の評価結果を表4に記載する。予測された上位5個のクラスの中に正解が含まれない割合を評価指標としている。現時点の最良のスコアは3.57%^[9]であり、これは人で試験した結果^[2]を上回るものである。手法の特色は、複数の層をまたいだ結合を随所に持つネットワークを利用して、下層の信号との残差に対する学習を行う点にあり、この工夫によって、これまででは困難とされていた100層を超えるような非常に深いネットワークであっても効率的に学習が行えるようになっている。GoogLeNet^[7]は、Inception Moduleと呼ばれる異なるサイズの畳み込み演算を並列的に行う層を持つことが特色で、これにより従来に比べ少ないパラメタ数で高いスコアを獲得することに成功している。OxfordNet^[8]は、畳み込み層を多層に渡って重ねた比較的シンプルな構造となっており、多量のパラメタにより表現力を高める一方で、dropoutによる正則を採用し過学習を抑えている。AlexNet^[1]は2012年に性能におけるブレイクスルーを起こしたモデルとして有名である。本研究で使用したモデルのパラメタ数はGoogLeNetに次いで少なく、層数はAlexNetに次いで少ない。スコアはAlexNetより良好であり、概ね妥当な範囲に収まっていると思われる。

出典	アンサンブルのモデル数	内積を含む層の数	パラメタ数	Top-5 error rate (%) on test set
AlexNet [1]	5	8	60M	16.64
GoogLeNet [7]	7	22	5M	6.66
OxfordNet [8]	7	19	144M	7.33
MSRA [9]	6	152	57M*	3.57
本研究	6	16	39M	13.89

表4 DNNを使った手法の識別性能の比較評価結果。内積を含む層の数とパラメタ数は、アンサンブルに使用したモデルのうち最大のものとした。(*開示されているネットワーク構造からパラメタ数を計算)

今後

7

以前は困難とされてきた、深いネットワークの学習を効率的に行うための知見が積み重なってきたため、今後はこれらの知見を活用して、パラメタ数を低く抑えつつネットワークを多層化する検討を進めたい。パラメタ数の低減は学習の高速化にとって極めて重要である。現状のDNNに必要とされるパラメタ数は学習サンプル数を大きく上回っており、パラメタ数を低減させる余地はまだ多く残されていると推測できるため、今後の研究課題としたい。

謝辞

本研究で使用した、マルチノード対応の非同期分散学習プログラムは、TSUBAME産業利用トライアルユース制度によるTSUBAME2.5の無償利用を通じて開発したものである。また、本研究成果は平成27年度秋季TSUBAMEグランドチャレンジを利用したものである。ここに感謝の意を表する。

参考文献

- [1] A. Krizhevsky, et al., "ImageNet Classification with Deep Convolutional Neural Networks", NIPS 2012.
- [2] O. Russakovsky, et al., "ImageNet Large Scale Visual Recognition Challenge", IJCV 2015.
- [3] M. Oquab, et al., "Learning and transferring mid-level image representations using convolutional neural networks", CVPR 2014.
- [4] J. Dean, et al., "Large Scale Distributed Deep Networks", NIPS 2012.
- [5] R. Wu, et al., "Deep Image: Scaling up Image Recognition", arxiv:1501.02876, 2015.
- [6] F. N. Iandola, et al., "FireCaffe: Near-Linear Acceleration of Deep Neural Network Training on Compute Clusters", arxiv:1511.00175, 2015.
- [7] C. Szegedy, et al., "Going Deeper with Convolutions", CVPR 2015.
- [8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", ICLR 2015.
- [9] K. He, et al., "Deep Residual Learning for Image Recognition", arxiv:1512.03385, 2015.
- [10] G. E. Hinton, et al., "Improving neural networks by preventing co-adaptation of feature detectors", arxiv:1207.0580, 2012.
- [11] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines", ICML 2010.
- [12] I. Sato, et al., "APAC: Augmented PAttern Classification with Neural Networks", arxiv:1505.03229, 2015.

マルチGPU 超並列クラスタシステムを用いた 大規模ナノ炭素分子の電子状態計算

河東田 道夫* 成瀬 彰** 中嶋 隆人*

* (国) 理化学研究所 計算科学研究機構 量子系分子科学研究チーム ** NVIDIA Japan

本研究では、最近新規にマルチGPGPUを施したRI-MP2エネルギー計算プログラムを用い、大規模ナノ炭素分子集合体の電子状態計算をTSUBAME 2.5で実施した。本稿ではTSUBAME 2.5でのマルチGPUジョブの性能評価結果を報告する。特に、TSUBAME 2.5全ノード規模で実施した最大規模のナノ炭素分子集合体の計算(ナノグラフェン2量体(C₉₆H₂₄)₂, 1,349ノード・4,047 GPU使用)のRI-MP2計算では、515 TFLOPsの実測ピーク性能を達成した。本稿ではさらに、 π - π スタッキングしたナノグラフェン2量体に働く分子間相互作用の解析結果も報告する。

はじめに

1

ナノ炭素分子集合体は、有機分子機能材料への応用を視野に入れて実験的、理論的、計算科学的アプローチによる研究が盛んに行われている。有機分子機能材料の分子設計や機能評価を行うためには、分子集合体の局所的な構造や安定性の理解が不可欠である。電子状態計算はこれらの機能を評価する上で極めて有効な手法であると広く認識されている。Møller-Plesset 2次摂動(MP2)法^[1]はファンデルワールス相互作用などの弱い分子間相互作用の記述に重要な役割を果たす電子相関を正しい振る舞いで取り扱うことが可能な手法であり、ナノ炭素分子集合体の安定性の議論の際には有用な手法となる。しかしながら、MP2計算は前段階に実施されるHartree-Fock(HF)計算よりも高コストな計算となるため、計算が可能分子のサイズに制約がある。そのため、京コンピュータやTSUBAME 2.5などの大規模計算システムを活用し、大規模分子のMP2計算を実施することが求められている。

我々の研究チームでは、京コンピュータなどのペタスケールスーパーコンピュータを活用し大規模分子の電子状態の超並列計算を実現することを目的として、NTChemプログラム^[2,3]の開発を進めている。NTChemには様々な電子状態理論や効率的な計算アルゴリズムが実装されている(図1)。代表的な計算手法として、(1) HF法や密度汎関数理論(DFT)に基づき、分子の基底状態と励起状態の電子状態を計算する機能、(2) より高精度な電子相関理論であるMP2法、結合クラスタ法、量子モンテカルロ法に基づく基底状態と励起状態の電子状態計算機能、などが挙げられる。

これらの計算方法のうち、HF法、DFT法、MP2法のコードについては京コンピュータでの超並列計算のためにMPI/OpenMPハイブリッド並列化が施されており、京数千ノードから数万ノード規模を用いた超並列計算が可能である。

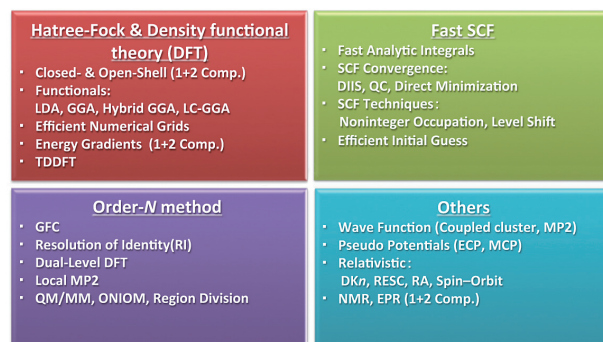


図1 NTChemに実装されている主要な機能。

最近の研究にて、著者らはResolution-of-Identity MP2(RI-MP2)法^[4,5]に基づく電子相関エネルギー計算の超並列アルゴリズムと計算プログラムの開発を行い、京8,911ノードを用いて360原子、9,840原子軌道系の大規模分子のRI-MP2エネルギー計算に成功している。^[6]

また、最近になり、graphics processing unit (GPU) などの加速器を搭載したヘテロジニアスなシステム構成のスーパーコンピュータが年々増加しており、これらのシステムに搭載されている加速器を活用した大規模電子状態計算のニーズも日増しに高まっている。NTChemの開発においても、このニーズに応えるため、プログラムのgeneral-purpose computing on GPU (GPGPU) 実装を進めている。最近の研究にて、著者らはRI-MP2エネルギー計算の超並列アルゴリズムの改良とGPGPU実装を行い^[7]、マルチノード・マルチGPUシステムを活用しMP2エネルギー計算を実現可能とする計算基盤の整備を行った。

本稿では、TSUBAME 2.5上で上記実装を用い、ナノ炭素分子複合体のRI-MP2エネルギー計算をマルチノード・マルチGPU環境で実施した結果を報告する。2章ではNTChemプログラムにおけるRI-MP2計算のマルチGPGPU実装の概略を説明する。3章および4章では、初めにTSUBAME 2.5でのマルチGPGPUコードの性能評価結果を報告する。5章では、 π - π スタッキングしたナノグラフェン2量体層間に働く分子間相互作用解析を行った結果を報告する。

NTChem プログラムにおける RI-MP2 計算のマルチ GPGPU 実装

2

2.1 RI-MP2 法

RI-MP2法^[4,5]は4中心電子反発積分 (electron repulsion integrals (ERIs)) のRI近似 (あるいは密度フィッティング近似) を適用することでMP2計算を高速化することが可能な計算手法である。

$$(ia|jb) = \sum_n B_n^{ia} B_n^{jb} \quad (1)$$

$$B_n^{ia} = \sum_{\mu\nu} (\mu\nu|l)(l|n)^{-1/2} C_{\mu} C_{\nu a} \quad (2)$$

ここで、式 (1) および (2) に対し原子軌道ペアに対して補助基底関数 (インデックス (l, n)) の線形展開を行う近似を導入している。この近似によりMP2計算のプレファクタを大きく減らすことが可能となる。RI-MP2計算において、最もボトルネックとなる処理は式 (1) の行列-行列積計算であるが、この処理は最適化されたBLAS^[6] に実装されているDGEMMを用いることにより、非常に高い実行効率およびスレッド並列性能での処理が可能である。MP2電子相関エネルギーは式 (1)、(2) から計算された近似積分を用いて式 (3) より計算される。

$$E^{(2)} = \sum_{ij}^{\text{occ.}} \sum_{ab}^{\text{vir.}} \frac{(ia|jb)[2(ia|jb) - (ib|ja)]}{\epsilon_j + \epsilon_i - \epsilon_a - \epsilon_b} \quad (3)$$

2.2 RI-MP2 計算の超並列実装

本研究を実施する以前の研究において、RI-MP2エネルギー計算プログラムの超並列アルゴリズム開発とNTChemプログラムへのMPI/OpenMPハイブリッド並列実装を行った。^[6,7] 本実装では、計算のボトルネックとなる (1) 式の4中心電子反発積分計算の並列化を行う際に仮想分子軌道のインデックス (a, b) を分散することにより、MPI並列化を行っている。この並列化手法により、占有分子軌道のインデックス (i, j) をMPI並列化していた従来の並列アルゴリズム^[5,9] と比較して大幅な並列スケールアップが可能である。さらに、分散メモリの有効活用を目的として、式 (1) と (2) に現れる中間データをインコアメモリに保持するアルゴリズムを選択した。本手法によりMP2従来法で大きな負荷となっていたI/O処理を除去することが可能となり、性能の向上に寄与している。

また、最近になり、数万MPIプロセスを用いた超並列計算を実行可能とするため、局所的なMPI communicator を活用し二次元的なMPI階層並列化を導入したアルゴリズム改善を実施した。この

改善では、最外ループとなる仮想軌道インデックス (a, b) の第1軸のMPI並列化に加え、このループ内で実行される積分計算と行列演算に関わる行列をブロック分割し第2軸の並列化を行った。このアルゴリズムとコードの改良により、前述の3,360原子、9,840原子軌道系の大規模分子の計算を最大で京80,199ノードを用いて実施することに成功している。^[7]

2.3 RI-MP2法のマルチGPGPU実装

本研究を実施する以前の研究において、2.2章で説明したRI-MP2エネルギー計算プログラムのCUDA^[10]を用いたGPGPU実装を行った^[7]。GPGPU実装を行うにあたり、行列-行列積計算に必要なデータをCPUからGPUに転送し、計算をGPUで実施するオフロードモデルを採用した (図2)。

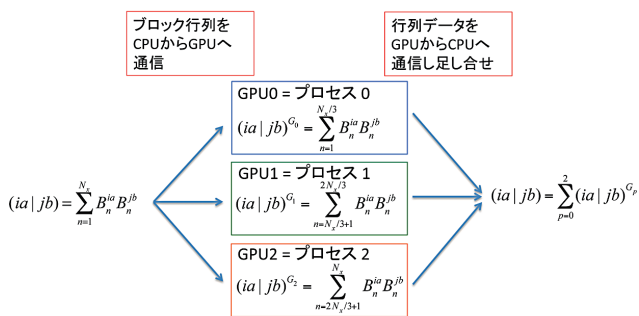


図2 RI-MP2計算のGPGPU実装の概要

各GPUへのタスク割り付けは2.2章で言及した行列-行列積演算の並列化に用いているMPIプロセスを各GPUに割り当てることで実現されている。行列-行列積演算はNVIDIAにより最適化されたcuBLAS^[11]ライブラリcublasDgemmルーチンを用いて実装している。このGPU計算を行う際にはCPU-GPU間のデータ転送が処理の大きなボトルネックとなる。この処理のオーバーヘッドを減らすために、異なる仮想軌道のインデックス (a, b) の複数の行列演算をまとめて実施するバッチ処理を実装した。このバッチ処理では、同じバッチに属するデータ転送と演算を1度にまとめて行い、さらにページロック(ピン)メモリを活用することにより、大きなオーバーヘッド削減が可能である。ページロックメモリはCPUのホストメモリとのデータ転送をする際にGPUからのダイレクトメモリアクセスを実現することができるため、CPUを使用せずにデータ転送が可能であり、性能向上に大きく寄与する。

本研究で使用したCPUコードのコンパイルにはIntelコンパイラを、GPUコードはCUDA 5.0コンパイラを使用した。また、数値計算ライブラリとして、Intel Math Kernelに実装されているBLASおよびLAPACK、cuBLASライブラリを使用した。

TSUBAME2.5 における RI-MP2 法の マルチ GPU 実装の性能評価： GPU による高速化と並列性能

3

本研究では、まず TSUBAME 2.5 における RI-MP2 計算ジョブの実行時間、並列性能、GPU による加速効果に関する性能評価を行った。性能評価のための計算ジョブとして、ナノグラフェン $C_{96}H_{24}$ の RI-MP2 計算を TSUBAME 2.5 の 64 から 512 ノードを用いて行った。これらの計算には cc-pVTZ 基底関数^[12]と cc-pVTZ 補助基底関数^[13]を用いた (120 原子、3,216 原子軌道、300 占有軌道、2,916 仮想軌道、8,496 補助基底関数、対称性なし)。MP2 計算の前に行われる HF 計算は京コンピュータで実行し、得られた分子軌道データを MP2 計算の入力データとして用いた。また、GPU 計算では 1 ノードあたり 3 MPI プロセスと 4 OpenMP スレッドを、CPU 計算では 1 ノードあたり 1 MPI プロセスと 12 OpenMP スレッドを用いた。表 1 に実行時間、並列性能、GPU による加速率を示す。

ノード	GPU ジョブ		CPU ジョブ		GPU による加速率
	実行時間 [秒]	並列加速率	実行時間 [秒]	並列加速率	
64	198	64	1277	64	6.4
128	123	104	806	101	6.6
256	92	137	531	154	5.8
512	86	148	349	234	4.1

表 1 TSUBAME 2.5 における RI-MP2 計算の実行時間、並列加速率、GPU による加速率 (ナノグラフェン $C_{96}H_{24}$ RI-MP2/cc-pVTZ)

いずれのノード数を用いた場合でも、GPU ジョブは CPU ジョブと比較して 4.1-6.6 倍の高速に計算が可能であることを確認した。また、GPU ジョブの並列性能は CPU ジョブと比較して低いものの、並列性能ノード数に対して良好にスケールすることを確認した。

TSUBAME2.5 全ノード規模を用いた マルチ GPU 大規模 RI-MP2 計算

4

3 章で述べた性能評価に続いて、TSUBAME 2.5 全ノード規模を用いて大きなナノ炭素分子の RI-MP2 エネルギーマルチ GPU 計算を実施した。本計算では、ナノグラフェン 2 量体 ($C_{96}H_{24}$)₂ の RI-MP2 計算を 1,349 ノードおよび 4,047 GPU を用いて実行した。本計算には cc-pVTZ 基底関数と cc-pVTZ 補助基底関数を用いた (240 原子、6,342 原子軌道、600 占有軌道、5,832 仮想軌道、16,992 補助基底関数、対称性なし)。3 章の計算と同様に GPU 計算では 1 ノードあたり 3 MPI プロセスと 4 OpenMP スレッドを、CPU 計算では 1 ノードあたり 1 MPI プロセスと 12 OpenMP スレッドを用いた。

図 3 に GPU ジョブと CPU ジョブの実行時間の詳細プロファイルを示す。図 3 に示されるように、GPU 利用により実行性能の大幅な向上が達成されている。特に、4 中心積分を計算するための行列-行列積演算を GPU へオフロードするにより、この演算が非常に短時間で完了したため (図 3 の赤色の部分)、GPU ジョブは CPU ジョブと比較して 5.9 倍の高速化を達成した。実行時間は GPU ジョブが 419 秒、CPU ジョブが 2047 秒であった。また、実測ピーク性能は GPU ジョブが 514.7 TFLOPs であったのに対して、CPU ジョブは 87.5 TFLOPs であり、大幅な実測ピーク性能向上を達成した。その一方で、GPU ジョブの実測ピーク性能 / 理論ピーク性能比 (10%) は CPU ジョブの結果 (42%) よりも遥かに低い結果となった。これは GPU ジョブで最も時間を要する処理が 4 中心積分のノード内集団通信と 3 中心積分のノード間 1 体 1 通信となったのが原因である。更なる性能向上のためにはこれらの処理の性能向上が必要となる。

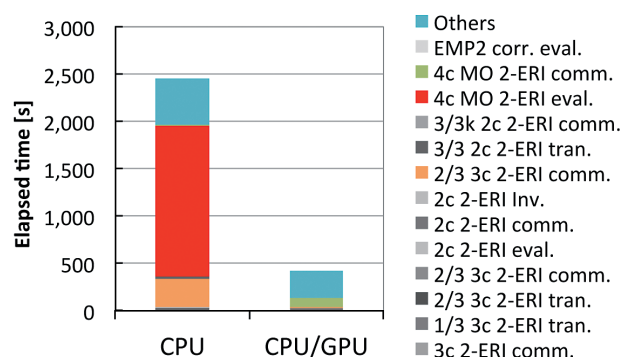


図 3 ナノグラフェン 2 量体 ($C_{96}H_{24}$)₂ の RI-MP2/cc-pVTZ 計算の GPU ジョブと CPU ジョブの TSUBAME 2.5 における詳細プロファイル

ナノグラフェン2量体の $\pi-\pi$ 相互作用解析

5

本研究で実施した計算結果より、ナノグラフェン2量体間に働く $\pi-\pi$ 相互作用エネルギーの解析を行った。モデル分子には $(C_{24}H_{12})_2$ 、 $(C_{54}H_{18})_2$ 、 $(C_{96}H_{24})_2$ を用いた(図4)。モデルを作成する際にはABスタッキング構造(図4(a))を対象とし、C-C結合距離(1.45 Å)と層間距離(3.35 Å)はバルクグラファイトの実験値を、C-H結合距離はベンゼンの実験値を用いた(1.1 Å)。相互作用エネルギーはMP2法、Spin-component scaled MP2(SCS-MP2)法^[14]、経験的分散力補正を施したダブルハイブリッドDFT法であるB2PLYP-D3法^[15]を用いて相互作用エネルギーの計算を行った。基底関数と補助基底関数はcc-pVTZを用いた。基底関数重ね合わせ誤差はCounter poise法^[16]で補正した。炭素原子1原子あたりの $\pi-\pi$ 相互作用エネルギーの結果を表2にまとめる。

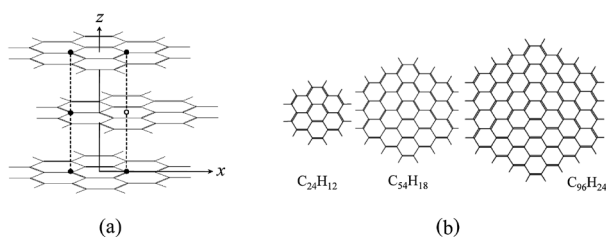


図4 2層ナノグラフェン2量体のモデル分子

モデル	MP2	SCS-MP2	B2PLYP-D3
$(C_{24}H_{12})_2$	68.6	44.4	42.7
$(C_{54}H_{18})_2$	87.6	59.4	54.6
$(C_{96}H_{24})_2$	85.7	68.0	61.2

表2 ナノグラフェン2量体の炭素原子当たりの
 $\pi-\pi$ 相互作用エネルギー (meV/atom)

これらの結果は、 $\pi-\pi$ 相互作用エネルギーはモデルのサイズが大きくなるにつれてバルクの2層グラファイトの相互作用エネルギーへ収束していくことを示している。最大のモデルである $(C_{96}H_{24})_2$ における相互作用エネルギーの計算結果は、SCS-MP2法では68.0 meV/atom、B2PLYP-D3法では61.2 meV/atomであり、バルクグラファイトの解離エネルギーの実験値 52 ± 5 meV/atom^[17]をよく再現している。一方で、MP2法の結果は、これらの方法の結果や実験値よりはるかに過大評価した結果(85.7 meV/atom)となった。

おわりに

6

本研究では、最大でTSUBAME 2.5 1,349ノード、4,047GPUを使用してナノ炭素分子集合体のRI-MP2エネルギー計算を実施した。初めにGPUによる計算の高速化の効果と並列性能の評価をTSUBAME 2.5 512ノードまで用いて実施した。GPU実装によりRI-MP2計算は大幅(4.1-6.6倍)な高速化が達成された。また、GPU実装の並列性能はノード数の増加に対して良好なスケーリングであった。次に、本研究で実施した最大のナノ炭素分子(ナノグラフェン2量体 $(C_{96}H_{24})_2$ 、240原子、6,342原子軌道)のRI-MP2エネルギーをTSUBAME 2.5全ノード規模の1,349ノードを用いて実施した。GPUジョブでは514.7 TFLOPsの実測ピーク性能を達成し、CPUジョブのピーク性能(87.5 TFLOPs)からの大幅な性能向上を達成した。

さらに、2層ナノグラフェン2量体に対して $\pi-\pi$ スタッキング相互作用エネルギーの解析を行った。 $\pi-\pi$ スタッキング相互作用エネルギーは分子のサイズが大きくなるにつれて、バルクの2層グラファイトの相互作用エネルギーに収束するのを確認した。また、SCS-MP2法、B2PLYP-D3法の計算結果はバルクグラファイトの解離エネルギーの実験値を良く再現した。一方、MP2法の結果はこれらの方法よりも大きな相互作用エネルギーとなった。

本研究で使用したコードは3中心原子軌道積分計算ルーチンのGPGPU実装も実施しておらず、またTSUBAME 2.5のネットワークに対応するように通信の最適化も実施していない。今後の研究では、TSUBAME 2.5のシステムに適するようにこれらの実装を進める予定である。

謝辞

本研究を実施するに際して、多くの有益な研究議論と助言をいただいたNVIDIA Japanの平野幸彦氏に深く謝意を表す。本研究実施に伴うTSUBAME2.5利用においては、TSUBAMEグランドチャレンジプログラムから全面的な支援をいただいた。また、本研究で利用したプログラムの開発およびテストでは「京」プロジェクト共通基盤研究の支援をいただいた。本研究を実施するにあたって事前に行った予備計算では京コンピュータと自然科学研究機構岡崎共同利用施設計算科学研究センターを利用した。記して謝意を表す。

参考文献

- [1] C. Møller and M. Plesset: Note on an Approximation Treatment for Many-Electron Systems, Phys. Rev., Vol. 46, pp. 618-622 (1934)
- [2] T. Nakajima, M. Katouda, M. Kamiya, and Y. Nakatsuka: NTChem: A high-performance software package for quantum molecular simulation, Int. J. Quant. Chem., Vol.

- 115, pp. 349-359 (2015)
- [3] NTChem, http://labs.aics.riken.jp/nakajimat_top/ntchem_e.html
- [4] M. Feyereisen, G. Fitzgerald, and A. Komornicki: Use of approximate integrals in ab initio theory. An application in MP2 energy calculations, *Chem. Phys. Lett.*, Vol. 208, pp. 359-363 (1993)
- [5] D. E. Bernholdt and R. J. Harrison, Large-scale correlated electronic structure calculations: the RI-MP2 method on parallel computers, *Chem. Phys. Lett.*, Vol. 250, pp. 477-484 (1996)
- [6] M. Katouda and T. Nakajima: MPI/OpenMP hybrid parallel algorithm of resolution of identity second-order Møller–Plesset perturbation calculation for massively parallel multicore supercomputers, *J. Chem. Theory Comput.*, Vol. 9, pp. 5373-5380 (2013)
- [7] M. Katouda, A. Naruse, and T. Nakajima: Massively parallel algorithm and implementation of RI-MP2 energy calculation for peta-scale many-core supercomputers, manuscript in preparation.
- [8] BLAS, <http://www.netlib.org/blas/>
- [9] M. Katouda and S. Nagase: Efficient parallel algorithm of second-order Møller–Plesset perturbation theory with resolution-of-identity approximation (RI-MP2), *Int. J. Quant. Chem.*, Vol. 109, pp. 2121-2130 (2009)
- [10] NVIDIA CUDA ZONE, <https://developer.nvidia.com/cuda-zone/>
- [11] cuBLAS <http://docs.nvidia.com/cuda/cublas/>
- [12] T. H. Dunning Jr.: Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen, *J. Chem. Phys.*, Vol. 90, pp. 1007-1022 (1989)
- [13] F. Weigend, A. Köhn, and C. Hättig: Efficient use of the correlation consistent basis sets in resolution of the identity MP2 calculations, *J. Chem. Phys.*, Vol. 116, pp. 3175-3183 (2002)
- [14] S. Grimme: Improved second-order Møller–Plesset perturbation theory by separate scaling of parallel- and antiparallel-spin pair correlation energies, *J. Chem. Phys.*, Vol. 118, pp. 9095-9102 (2003)
- [15] L. Goerigk and S. Grimme: Efficient and Accurate Double-Hybrid-Meta-GGA Density Functionals—Evaluation with the Extended GMTKN30 Database for General Main Group Thermochemistry, Kinetics, and Noncovalent Interactions, *J. Chem. Theory Comput.*, Vol. 7, pp. 291-309 (2011)
- [16] S. F. Boys and F. Bernardi: The calculation of small molecular interactions by the differences of separate total energies. Some procedures with reduced errors, *Mol. Phys.*, Vol. 19, pp. 553-566 (1970)
- [17] R. Zacharia, H. Ulbricht, and T. Hertel: Interlayer cohesive energy of graphite from thermal desorption of polyaromatic hydrocarbons, *Phys. Rev. B*, Vol. 69, pp. 155406 (2004)

● **TSUBAME e-Science Journal vol.14**

2016年2月29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 板倉有希

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京工業大学のスパコンTSUBAME2.5を
東京工業大学以外の皆さまにもご利用いただくための制度です。

TSUBAME 共同利用サービスの利用区分とカテゴリ

TSUBAME 共同利用サービスには「学術利用」と「産業利用」の利用区分があります。
「学術利用」は、HPCI(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)、
JHPCN(学際大規模情報基盤共同利用・共同研究拠点)の採択により無償でご利用になれる制度と、
東京工業大学学術国際情報センターが実施する有償の制度がご利用になれます。
「産業利用」は、HPCIの採択により無償でご利用になれる制度と、
東京工業大学学術国際情報センターが実施する有償の制度があり、
有償の制度には利用目的や利用成果を非公開とする「成果非公開」のカテゴリもあります。

TSUBAME 共同利用サービスの利用料金

TSUBAMEにおける計算資源は口数を課金単位としております。1口は3000ノード時間積で、
1計算ノード(12CPUコア、3GPU、58GBメモリ搭載)を3000時間、
あるいは300計算ノードを10時間というように、ご利用の用途に合わせて自由にご利用になれます。
TSUBAME 共同利用サービスの利用区分・カテゴリ・利用料金を下表に示します。

利用区分	利用者	制度	募集期間	申請および審査	成果	料金(税別)	
学術利用	他大学 または 研究機関 等	HPCI	年1回 10月頃	HPCI 運用事務局 (高度情報科学技術研究機構)	公開	無償	
		JHPCN	年1回 1月頃	JHPCN 拠点事務局 (東京大学 情報基盤センター)	公開	無償	
		TSUBAME 学術利用	随時 募集中	東京工業大学 学術国際情報センター	公開	1口 120,000円	
産業利用	民間企業	HPCI	実証利用	年1回 10月頃	HPCI 運用事務局 (高度情報科学技術研究機構)	公開	無償
			トライアル ユース	随時 募集中			
		TSUBAME 産業利用	随時 募集中	東京工業大学 学術国際情報センター	公開	1口 120,000円	
					非公開	1口 480,000円	

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
- 詳しくは <http://www.gsic.titech.ac.jp/tsubame/> をご覧ください。