

Advance/NeuralMD Proにおけるニューラルネットワーク力場のGPU化

山本 舜* 西原 慧径*

GPU Acceleration of Neural Network Potential by Advance/NeuralMD Pro

Shun Yamamoto* and Satomichi Nishihara*

本記事では、ニューラルネットワーク分子動力学システム Advance/NeuralMD を GPU 化した Advance/NeuralMD Pro におけるベンチマーク計算について紹介する。ニューラルネットワーク力場は、その計算精度と汎用性から、材料シミュレーションの分野において近年大きな注目を集めている。一方、計算速度の面では課題が残されているが、弊社製品の Advance/NeuralMD Pro ではニューラルネットワーク力場を GPU 化しており、実用上十分な速度での計算が実行可能である。

Keywords: ニューラルネットワーク力場、GPU、分子動力学、LAMMPS、クラウドコンピューティング、Advance/NanoLabo、Advance/NeuralMD

1. はじめに

分子動力学計算を実施する際に、力場は必須である。力場は与えられた原子座標におけるポテンシャルエネルギーを定義するものであり、一階微分することで原子に働く力が得られる。この力を使用して運動方程式を解くことで、分子動力学計算が遂行される。従来の力場では、簡単な関数形を用いてポテンシャルエネルギーを表現していた。簡単な関数形を使用することで、少ない計算コストで効率的なシミュレーションが実施できる。しかしながら、その計算精度は関数形の中で使用されるパラメータに大きく依存する。また、既報文献などで公開されているパラメータが存在すれば、その値を参照することで直ちに分動力学計算を実施できるが、既知のパラメータがなければ適正なパラメータを自作する必要がある。パラメータの作成には第一原理計算の結果を参照する必要があり、極めて多くの手間がかかる。そのような中で、2010年代に入ると、これらの問題を解決する力場として、ニューラルネットワーク力場が開発された。ニューラルネットワークによってポテンシャルエネルギーを表現するニュー

ラルネットワーク力場は、第一原理計算と同程度の計算精度で分子動力学計算を実施することができ、また、未知の系に対してもシステムティックに力場の作成が可能である。一方で、ニューラルネットワーク力場は第一原理分子動力学計算に比べると計算コストは低いものの、従来の古典力場よりも計算時間がかかるという問題点を抱えていた。そこで、弊社製品のニューラルネットワーク分子動力学システム Advance/NeuralMD の Pro 版では、ニューラルネットワーク力場の学習及び分子動力学の実行を GPU 化し、計算の高速化を実現した。本記事では、GPU 化したニューラルネットワーク力場を用いたベンチマーク計算の結果を紹介する。

2. ニューラルネットワーク力場学習の高速化

次の計算機環境のもとで、ニューラルネットワーク力場学習のベンチマーク計算を行った。

- CPU : Intel Xeon Silver 4310 (12cores, 2.1GHz) x 2
- メモリ : 8GB DDR4-3200 ECC RDIMM x 8
- GPU : NVIDIA A30 x 1
- OS : AlmaLinux 8.6

*アドバンスソフト株式会社 第1事業部

1st Computational Science and Engineering Group,
AdvanceSoft Corporation

教師データとしては、リチウムイオン伝導体 Li7La3Zr2O12 (LLZO) のプリミティブセル(96 原子系)を 4681 構造を使用した。また、ニューラルネットワーク力場の計算条件は以下の通りとした。

- ・対称関数：Chebyshev 多項式
- ・動径成分：50 個
- ・角度成分：30 個
- ・カットオフ半径：6.0 Å
- ・ニューラルネットワーク：2 層 x 40 ノード
w/ twisted tanh
- ・最適化アルゴリズム：L-BFGS
- ・エポック数：100

CPU に関しては、MPI と OpenMP のハイブリッド並列を、MPI 並列数 x OpenMP 並列数 = CPU コア数となるように適用した。また、GPU に関しては、CUDA における 1 ブロック当たりのスレッド数を 256 とした。

MPI 並列数は、GPU を使用しない場合には総 CPU コア数と等しくなるように設定した。また、GPU を用いる場合の MPI 並列数は 2 とした。

各計算条件において計算実行に要した時間を表 1 に示す。

表 1 ニューラルネットワーク力場学習の計算時間

計算時間 / s	w/o GPU	w/ GPU
CPU x 1 (12cores)	506	189
CPU x 2 (24cores)	379	171

以上の結果から、GPU を用いることでニューラルネットワークの学習が 2~3 倍程度高速化されることが分かる。

なお、以上の計算では MPI 並列数や CUDA のスレッド数などを様々に変化させてチューニングを行い、最も良い条件の結果のみを記載している。チューニングの詳細に関しては弊社ホームページの解析事例記事をご参照いただきたい[1]。

3. 分子動力学計算の高速化

次に、いくつかのマルチ GPU 環境の下で分子動力学計算のベンチマークを行った。ソルバーとしては、弊社製品の Advance/NanoLabo に搭載している、弊社で独自に改修した LAMMPS[2]を用いた。

分子動力学計算の対象となる系としては、硫化物リチウムイオン伝導体 Li10GeP2S12(LGPS) のスーパーセルモデルを用いた。原子数は 21,600 個としており、Neural Network 力場を適用するには比較的大きな系である。Neural Network 力場および分子動力学計算の計算条件を以下に示す。

- ・対称関数の種類 Chebyshev 多項式
- ・対称関数の動径成分 50 個
- ・対称関数の角度成分 30 個
- ・カットオフ半径 6.0 Å
- ・ Δ -NNP 法 適用あり
- ・NN の構造 2 層 x 40 ノード(twisted tanh)
- ・アンサンブル NVT (T = 500K)
- ・時間刻み 0.5 fs
- ・MD ステップ数 100

3.1. Mat3ra におけるベンチマーク

Mat3ra[3]は Exabyte.io 社によって提供されているクラウド環境である。Mat3ra では Amazon AWS および Microsoft Azure が利用可能で、両クラウド環境にてベンチマークを実施した。計算リソースの手配には Mat3ra の代理店である伊藤忠テクノソリューションズ (CTC) 社[4]にご協力いただいた。LAMMPS のコンパイラなどの仕様を以下に示す。

- ・コンパイラ：GCC11.2.0
- ・MPI ライブラリ：OpenMPI4.1.1
- ・行列演算ライブラリ：OpenBLAS
- ・CUDA：11.5

3.1.1. Amazon AWS におけるベンチマーク

CPU 単体で計算する場合の CPU としては Intel Xeon Platinum / 72core を、GPU を使用する場合の

CPU としては Intel Xeon E5-2686-v4 を用いた。また、GPU としては NVIDIA V100 を用いた。

計算結果を表 2 に示す。

表 2 Amazon AWS における分子動力学の計算時間

	MPI 並列数	OpenMP 並列数	計算時間 / s
CPU x 1	72	1	150.8
GPU x 1	4	2	21.6
GPU x 2	8	2	11.2
GPU x 4	16	2	6.9
GPU x 8	32	2	3.9

以上の結果からは、GPU1 デバイスで 7 倍、GPU8 デバイスで約 40 倍の高速化が実現できていることが分かる。

3.1.2. Microsoft Azure におけるベンチマーク

CPU 単体で計算する場合の CPU としては Intel Xeon Platinum 8168/44core を、GPU を使用する場合の CPU としては Intel Xeon E5-2690-v4 を用いた。また、GPU としては NVIDIA P100 を用いた。

計算結果を表 3 に示す。

表 3 Microsoft Azure における分子動力学の計算時間

	MPI 並列数	OpenMP 並列数	計算時間 / s
CPU x 1	44	1	195.4
GPU x 1	3	2	30.5
GPU x 2	6	2	16.4
GPU x 4	12	2	8.5

以上の結果から、GPU1 デバイスで約 6 倍、GPU4 デバイスで約 23 倍の高速化が実現できていることが分かる。2 万原子系では GPU が 4 デバイスほどあれば十分に実用的な計算が実施可能であり、原子数が 5000 以下であれば、GPU 1 デバイスでも十分高速に MD 計算が実施できると推測される。

3.2. HPC システムズ社提供の計算機環境におけるベンチマーク

HPC システムズ社[5]に提供していただいた環境におけるベンチマーク計算の結果を紹介する。

3.2.1. 8GPU 搭載マシンにおけるベンチマーク

CPU としては AMD EPYC 7742 64-Core を、GPU としては NVIDIA A100 SXM4 を用いた。また、コンパイラには GCC を、行列演算ライブラリには OpenBLAS を用いた。

表 4 に計算結果を示す。

表 4 HPC システムズ社提供の 8GPU 搭載マシンにおける分子動力学の計算時間

	MPI 並列数	OpenMP 並列数	計算時間 / s
CPU x 1	64	1	56.4
GPU x 1	4	2	16.0
GPU x 2	8	2	8.3
GPU x 4	16	2	4.4
GPU x 8	32	2	2.5

以上の結果から、GPU 1 デバイスで約 4 倍、8 デバイスで約 23 倍の高速化が実現できていることが分かる。

これまでに示した 3 つのマルチ GPU 環境での相対計算速度を図 1 にまとめた。なお、相対計算速度は、HPC 環境での CPU 単体による計算速度を 1 として算出した。いずれの環境においても、GPU デバイス数の増加に合わせて等比的に計算速度が増加していく様子が確認できる。

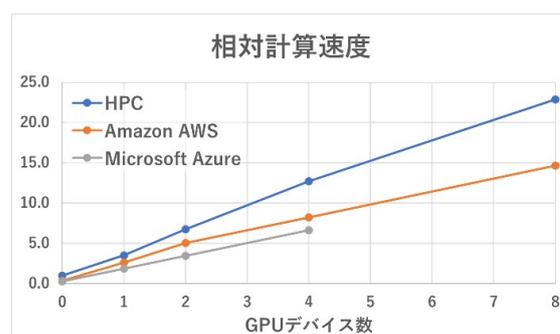


図 1 マルチ GPU 環境における相対計算速度

3.2.2. 32GPU 搭載クラウドにおけるベンチマーク

最後に、これまで用いた LGPS の 21600 原子系スーパーセルモデルに加えて、同一組成の 98000 原子系スーパーセルモデルについてもベンチマーク計算を行った。

HPC システムズ社に提供していただいた計算機の 1 ノードあたりのスペックは以下の通りである。また、総ノード数は 4 である。

- CPU : AMD EPYC 7J13 (64cores, 2.55GHz) x2
- メモリ : 2048GB
- GPU : NVIDIA A100 ×8
- ストレージ : 120GB NVMe + 27.2TB NVMe SSD (7.68TB ×4)
- ネットワーク : 2 × 50Gbps + 16 × 100Gbps RDMA
- コンパイラ : Intel oneAPI 2021.7.1
- MPI ライブラリ : OpenMPI 4.1.3
- 行列演算ライブラリ : Intel MKL 2022.2.1
- CUDA : 11.4

1GPU あたりの MPI プロセス数は 4 とした。また、全ての計算において OpenMP スレッド数は 1 とした。

表 5 に計算結果を示す。

表 5 HPC システムズ社提供の 32GPU 搭載マシンにおける分子動力学の計算時間

計算時間 / s	21600 原子	98000 原子
GPU x 1	15.35	69.12
GPU x 2	7.84	35.64
GPU x 4	4.16	18.08
GPU x 8	2.29	9.84
GPU x 16	1.41	4.97
GPU x 24	1.14	3.42
GPU x 32	1.03	2.63

3.1.1 節で行った AWS 上の CPU を 1 つのみ用いた計算と比較すると、クラウド上の GPU を 32 個使用することで、98000 原子系の計算が約 260 倍

高速化されていることが分かる。ただし、AWS 上での計算では 21600 原子系を用いているため、計算時間が原子数に比例することを仮定して比較を行った。

また、32GPU 搭載のクラウド環境で 1 つの GPU デバイスのみを用いた場合の計算速度を 1 とした相対速度を図 2 に示す。

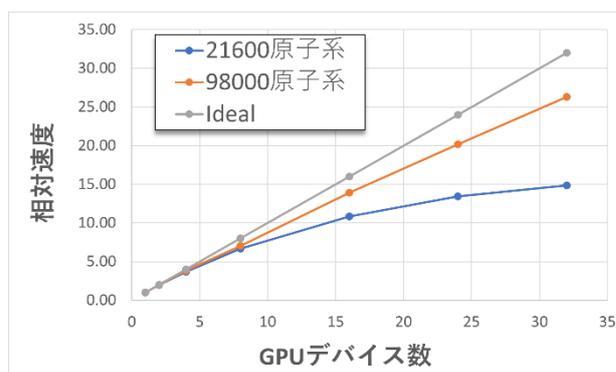


図 2 32GPU 環境における相対計算速度

図 2 には、並列効率が理想的な場合の相対速度についてもプロットしている。21600 原子系では GPU デバイス数が増えるにつれて並列効率が下がっている一方で、98000 原子系では GPU デバイス数が 32 でも十分な高速化が実現できていることが分かる。GPU デバイスを 32 個使用した場合、各 GPU に割り当てられる原子数が 21600 原子系では約 700 原子、98000 原子系では約 3000 原子となっており、前者に比べて後者では 1GPU あたりの原子数が十分大きいいため並列性能を維持できていると考えられる。

以上の結果をもとに見積もりを行うと、ニューラルネットワーク力場を用いた 1 ns の分子動力学計算を 10 万原子程度の系に対して実施する場合でも、クラウド環境の 32 個の GPU を使用すれば 15 時間程度で計算を実行できると考えられる。

4. まとめ

以上のベンチマークの結果から、GPU 化されたニューラルネットワーク力場を使用することで、大きなモデルでの長時間のシミュレーションでも、より現実的な短い計算時間で実行できることが明らかになった。

参考文献

- [1] <http://case.advancesoft.jp/NeuralMD/GPU-benchmark/index.html>
- [2] <https://www.lammps.org>
- [3] <https://www.mat3ra.com>
- [4] <https://ctc-mi-solution.com/exabyte-io>
- [5] <https://www.hpc.co.jp/>

※ 技術情報誌アドバンスシミュレーションは、アドバンスソフト株式会社 ホームページのシミュレーション図書館から、PDF ファイル（カラー版）がダウンロードできます。（ダウンロードしていただくには、アドバンス/シミュレーションフォーラム会員登録が必要です。）