

機械学習の第一原理計算への応用

西原 慧径*

Applying Machine Learning for First-Principles Calculation

Satomichi Nishihara*

近年における機械学習技術の発展は目覚ましく、毎月のように新しい手法が提案され続けている。機械学習の応用先は自然言語処理や画像処理などに留まらず、構造解析・流体解析・材料解析などのシミュレーション分野にも広がっている。本稿では、材料シミュレーションに大きな力を発揮する第一原理計算において、機械学習を応用した手法を2つほど紹介する。具体的には、第一原理計算のポテンシャル地形を模擬する機械学習力場、および、密度汎関数理論における汎関数を機械学習で再現する手法(機械学習汎関数)である。両手法の紹介を通じて、量子性を機械学習で再現する際の有用性と問題点について考察する。また、最後に機械学習のカジュアルな応用例として、生成 AI を活用して材料モデルを自動的に作成する GUI ツールを紹介する。

<https://doi.org/10.69290/j.001172-vol32>

Keywords: Density Functional Theory, Molecular Dynamics, Machine Learning Potential, Machine Learning Functional

1. 機械学習力場

第一原理計算において、入力データは系を構成する各原子の座標 $\mathbf{R}_i$ と原子番号 Z_i のみであり、出力データはエネルギー E_{tot} である。種々の $\mathbf{R}_i$ に対して E_{tot} を計算すると、系のポテンシャルエネルギーを定義する曲面が得られる。これがポテンシャル地形である。ポテンシャル地形が既知であれば、構造最適化計算や分子動力学計算が可能となり、弾性係数や熱伝導率といった物理量の解析もできる。バンド構造、状態密度、HOMO/LUMO、原子電荷量などの電子状態に関する解析を無視するのであれば、原子レベルのシミュレーションの多くはポテンシャル地形のみで議論することができる。第一原理計算にてポテンシャル地形を評価するのは計算コストが極めて高く、系のサイズが大きい場合には現実的ではない。そこで、ポテンシャル地形を近似的な関数形で表現する“力

場”という手法がよく使われる。従来の力場の例として、有機分子系における共有結合はバネ(二次関数)で表現され、金属や半導体などの無機材料における原子間相互作用は Lennard-Jones や Buckingham などの2体関数で表現されてきた。このような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(バネ定数など)。そこで、第一原理計算と同等の精度を担保した力場として、2010年代より盛んに使用されるようになってきたのが機械学習力場である。

一般に、連続で滑らかな関数であれば、機械学習にて容易に再現できる。ポテンシャル地形は基本的には連続で滑らかである。したがって、ポテンシャル地形は機械学習で再現可能である。 $\mathbf{R}_i$ および Z_i を入力情報として、 E_{tot} を出力するような機械学習モデルを用意すれば良いのである。よく勘違いされる方も多いのだが、この機械学習モデルはエネルギーを出力するのであり、既存の力場

*アドバンスソフト株式会社 第6事業部

6th Computational Science and Engineering Group,
AdvanceSoft Corporation

モデルにおける力場パラメータを出力するのではない。つまり、機械学習で力場パラメータを最適化するのではなく、機械学習モデルそのものが力場となっているのである。機械学習モデルとしては種々のものが提案されているのだが、近年では Neural Network を使うものがほとんどであり、機械学習力場のことを Neural Network 力場と称することもある。Neural Network 力場の定式化として最も標準的なのが Behler によって提案された High-Dimensional Neural Network Potential (HDNNP)である[1]。

HDNNP では、系のエネルギー E_{tot} を各原子のエネルギー E_i の総和で表現する：

$$E_{tot} = \sum_{i \in \{\text{all atoms}\}} E_i \quad (1)$$

各原子のエネルギーは、Neural Network にて計算される：

$$E_i = E_{NN}(G_{i,1}, G_{i,2}, G_{i,3}, \dots) \quad (2)$$

E_{NN} は E_i を出力する多層パーセプトロン(層状に積み重なった Neural Network)であり、隠れ層の数は2~3程度、各層のノード数は20~50程度である。 E_{NN} の入力層は、対称関数 G_i である。対称関数は、 i 番目の原子の周辺にある他の原子の立体構造(化学環境)を表現したものである。対称関数にも多くのバリエーションが存在するのだが、原子番号 Z_i で重み付けされた Behler 関数[2]：

$$G_i^{rad} = \sum_j e^{-\eta(R_{ij}-R_s)^2} f_c(R_{ij}) Z_j \quad (3)$$

$$G_i^{ang} = 2^{1-\zeta} \sum_{j,k} (1 + \lambda \cos(\theta_{ijk}))^\zeta \times e^{-\eta[(R_{ij}-R_s)^2 + (R_{ik}-R_s)^2 + (R_{jk}-R_s)^2]} \times f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \sqrt{Z_j Z_k} \quad (4)$$

および、Chebyshev 関数[3]：

$$c_{i,\alpha}^{(2)} = \sum_j T_\alpha\left(\frac{2R_{ij}}{R_c} - 1\right) f_c(R_{ij}) Z_j \quad (5)$$

$$c_{i,\alpha}^{(3)} = \sum_{j,k} T_\alpha\left(\frac{2\theta_{ijk}}{\pi} - 1\right) \times f_c(R_{ij}) f_c(R_{ik}) \sqrt{Z_j Z_k} \quad (6)$$

などがよく使われる。 $f_c(R_{ij})$ カットオフ関数であ

る。いずれも、原子間距離 R_{ij} および結合角 θ_{ijk} に依存した情報を表現した関数である。式中の各種パラメータの詳細は文献[2,3]を参照されたし。力場の関数形に Neural Network を使用しているため、従来の力場における力場パラメータに相当するものは Neural Network における重みとバイアスとなる。したがって、重みとバイアスを系に応じて最適化する必要がある。この最適化のプロセスこそが、Neural Network の学習である。学習には教師データが必要であるのだが、この教師データに第一原理で計算されたエネルギー E_{tot} を用いる(最近は力やストレスを含めるケースが多い)。教師データに使う第一原理計算の結果は、数千~数万個ほどである。

第一原理計算の結果を教師データとして学習した HDNNP は、第一原理計算における E_{tot} に相当するエネルギーを出力し、ポテンシャル地形を模擬する。しかしながら、その計算コストは第一原理計算よりも低く、数千原子系の分子動力学シミュレーションも容易に実施できる。HDNNP の応用例として、リチウムイオン伝導体 $\text{Li}_{10}\text{GeP}_2\text{S}_{12}$ のイオン伝導率の計算を示す。イオン伝導率を精度よく計算するには、ポテンシャル地形の正確さに加えて、十分に大きなモデルサイズでの分子動力学計算が必須となる。HDNNP であれば、この要件を満たすことができる。1600 原子系のスーパーセルモデル(図1)を用いて、350~800K の温度範囲にて 1.0ns の動力学計算を実行して、Li 原子の拡散係数の値からイオン伝導率を評価した。図2は、イオン伝導体の対数を、逆温度に対してプロットしたものである。第一原理計算ではモデルサイズが小さく高温領域しか計算できないのだが、HDNNP であれば常温領域でも精度よく実験値を再現できていることがわかる。

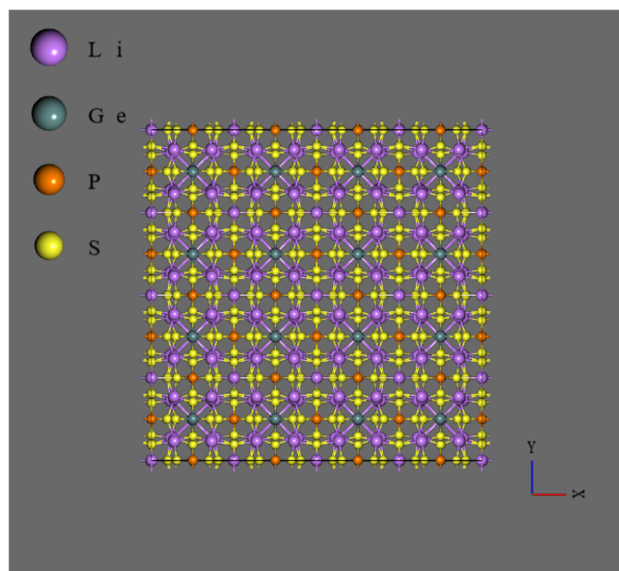
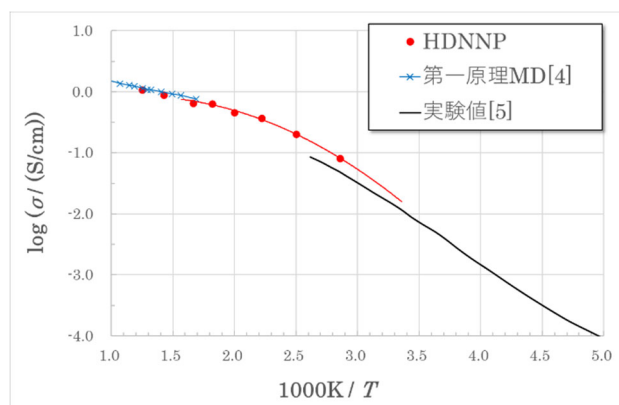
図1 $\text{Li}_{10}\text{GeP}_2\text{S}_{12}$ のスーパーセルモデル

図2 Li イオンのイオン伝導率

最近では HDNNP のように対称関数を固定するのではなく、Graph Neural Network (GNN) を応用してより深層な機械学習モデルを利用するケースが主流となりつつある。GNN を使う場合、Materials Project[7]などの大規模データベースの計算データを事前に学習させて、種々の系に適用可能な汎用性を持たせることが多い。この方法では、ユーザーが個々の系について教師データを用意したり学習させたりする必要がなく、非常に使い勝手が良い。ただし、モデルの深層化に伴う計算コストの増大と個別の系に対する精度低下という欠点もある。オープンソースの GNN 力場としては、M3GNet[8]、CHGNet[9]、SevenNet[10]などがある。

2. 機械学習汎関数

前章の機械学習力場では、エネルギー E_{tot} を座標 $\mathbf{R}_i$ の関数として、その関数形を機械学習モデルで表現した。このとき、電子状態はブラックボックス化されている。このブラックボックス化の弊害としては、(1)系が複数の電子状態をとり得る場合に両状態のポテンシャル地形の違いを再現できない、(2)電子状態に関する解析ができない、といった問題がある。(1)については、酸素分子のスピナー重項状態と三重項状態の違いがよい例である。(2)については、リチウムイオン電池の電極材料のシミュレーションなどで各原子の電荷量や電位などを評価することができないといった問題に相当する。これらの問題を解決するためには第一原理計算を実施すればよいわけだが、そもそも第一原理は計算コストが高いため代替手法として機械学習力場が使われているのである。逆に、第一原理計算を機械学習モデルで高速化して、機械学習力場と同等の速度で計算できるようになればどうか。これを実現する方法の一つが機械学習汎関数にある。

具体的には、Orbital Free DFT(OF-DFT)[11]における運動エネルギー汎関数を機械学習で表現するのである。現在広く使われている密度汎関数理論(DFT)は Kohn Sham DFT(KS-DFT)と呼ばれるものであり、Kohn Sham 軌道と呼ばれる一電子波動関数を用いて運動エネルギーを表現している。この波動関数による運動エネルギーの表現方法は、実用に耐えうる精度の運動エネルギー汎関数を *a priori* に見つけ出すことが出来なかったための苦肉の策に過ぎない。波動関数の導入により運動エネルギーの精度は担保できたものの、計算コストが高くなった。原子数の3乗に比例する計算時間である。波動関数を使わずに本来の DFT を実現しようという試みが OF-DFT であり、その課題は運動エネルギー汎関数の精度にある。機械学習モデルを使って運動エネルギー汎関数を高精度に再現できれば、OF-DFT の問題は解決される。そして、OF-DFT の計算時間は原子数に比例するため、機械学習力場と同程度の速度で計算が実施できるのである。OF-DFT は第一原理計算の中でもか

なりニッチな手法であるため、まだまだ発展途上であるものの、既に機械学習を使った運動エネルギー汎関数 $T[\rho]$ がいくつか提案されている。例としては、電子密度 ρ 、電子密度の勾配 $\nabla\rho$ およびラプラシアン $\Delta\rho$ を引数(特長量)とした運動エネルギー密度 τ を機械学習モデルにて予測して、これを積分するという方法がある[12][13]：

$$T[\rho] = \int d\mathbf{r} \tau(\rho(\mathbf{r}), \nabla\rho(\mathbf{r}), \Delta\rho(\mathbf{r}), \dots) \quad (7)$$

この方法では、座標 $\mathbf{r}$ における運動エネルギー密度は、同じ座標 $\mathbf{r}$ とその近傍の電子密度のみで定義されており、いわゆる準局所近似である。その他に、 τ の引数として固定された非局所の特長量を含めるような方法も提案されている[14][15]。準局所モデルに比べれば精度改善は期待できるものの、汎用性を担保することは難しい。DeepMind21[16]では類似の方法で交換相関汎関数の再現に成功しているが、固定された特長量の物理的妥当性がこれまでの Hybrid-GGA の研究過程にて十分に既知であることに起因する。一方、運動エネルギー汎関数については未知な部分が多く、ちょうどよい特長量を見出すことが難しい。そこで当社(アドバンスソフト株式会社)では、任意の汎関数を再現可能とすべく、場の関数を深層学習できる新手法を開発して運動エネルギー汎関数へと適用した。既に広範な教師データについて学習可能な汎用性を有することを確認済みであり、そう遠くない将来にアルゴリズムを公開するとともに、製品またはサービスとしてリリース予定である。いずれの機械学習モデルにおいても、教師データは運動エネルギーおよび運動エネルギーを電子密度で汎関数微分したものである。この汎関数微分は、KS-DFT の計算結果より計算される：

$$\frac{\delta T}{\delta \rho} = \frac{1}{\rho} \sum_i n_i \left[-\frac{1}{2} \phi_i^* \Delta \phi_i + (\varepsilon_F - \varepsilon_i) \phi_i^* \phi_i \right] \quad (8)$$

ϕ_i 、 ε_i 、 n_i はそれぞれ波動関数とそのエネルギーおよび占有数である。 ε_F はフェルミ準位であり、中性の絶縁体の場合にはバンドギャップ中の任意のエネルギー準位となる。しかしながら、 $\delta\rho$ がイオン化を伴わない場合には、このエネルギー準位

の任意性は $\delta T/\delta\rho$ には寄与しない。しかしながら、イオン化を伴う場合(半導体へのドーピングなど)には、エネルギー準位の任意性が $\delta T/\delta\rho$ に影響を与え、 $\delta T/\delta\rho$ は不連続となる。残念ながら機械学習モデルでは不連続性は再現できないため、絶縁体および半導体の教師データには少々工夫が必要となる。

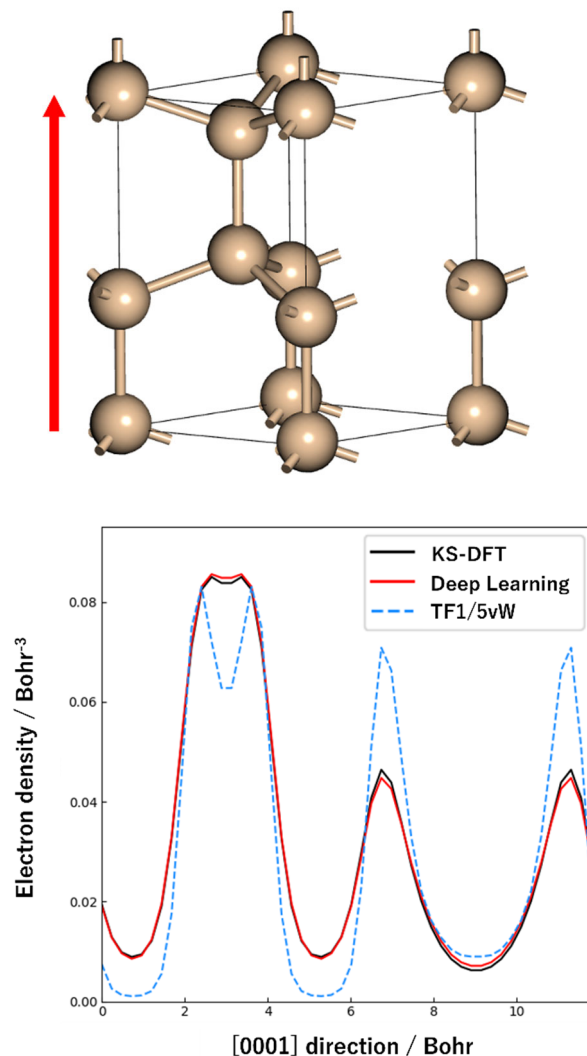


図3 六方晶 Si の構造(上)および電子密度(下)

本稿では、当社で開発した深層学習運動エネルギー汎関数を使った OF-DFT の簡単な計算結果を紹介する。具体的には、文献[13]を参考にして 10 種の立方晶の半導体結晶を教師データとして、運動エネルギー汎関数を学習させた。この運動エネルギー汎関数を用いた OF-DFT 計算を、六方晶の Si 結晶に適用して電子密度を最適化した。最適化された電子密度を図 3(下)に示す。図 3(上)の矢印に沿った(0001 方向)の電子密度を描画している。

KS-DFT での計算結果、および、TF1/5vW 汎関数による準局所密度近似での OF-DFT の計算結果についても併せて描画している。機械学習で構築された運動エネルギー汎関数により、KS-DFT の結果を良く再現できていることが確認できる。さらに、教師データとして Materials Project に収録された多数の結晶を用いることで、より汎用的な運動エネルギー汎関数となることが期待される。多数の教師データを使って学習させる際には、機械学習モデルの深層化が重要性を増す。

3. 量子性と機械学習に関する考察

機械学習力場や機械学習汎関数など、機械学習モデルを第一原理計算へ応用することは非常に有用である。しかしながら、その限界も存在する。第一原理計算は量子力学に基づいているわけだが、量子力学における“量子性”とはエネルギー準位が不連続であることを意味する。いわゆるエネルギー量子のことである。この不連続性の典型的なものがバンドギャップであり、式(8)における $\delta T / \delta \rho$ の不連続点がバンドギャップに相当するのである。ポテンシャル地形においては、ヤン-テラー効果における極大点や異なるスピン多重度のエネルギーが切替わる点などが不連続となる。機械学習モデルは連続的なものを表現するのは得意だが、不連続なものは全く表現できない。つまり、機械学習では量子性は扱えないのである。一般に、系のサイズが小さければ量子性は大きく、逆に系のサイズが大きければ量子性は小さくなる。したがって機械学習モデルの有用性が極大となるのは、系のサイズが大きくなり量子性が失われつつあるものの、完全には古典論的な方程式では表現しきれない局面である。ただし、バンドギャップや磁性材料など、マクロスケールでも量子性が残るような問題については、別途特別な取り扱いが必要となる。

4. 生成 AI を活用したモデリング

前章までは小難しい話が続いたが、もう少しカジュアルな機械学習の応用例として、近年流行りの生成 AI を使ったモデリングについて紹介した

い。当社では Advance/NanoLabo[16]という材料シミュレーション用のソフトウェアを開発および販売しており、当該ソフトウェアの機能の一つとしてモデリングがある。モデリング機能では、表面・界面や分子吸着など含む種々のモデル構造を作成することができる。複雑なモデル構造の場合には、多段階のプロセスを経てモデルを作成する必要があるため、初心者にとっては少々手間のかかる作業が要求される。この手間を省いた効率的なモデリングを実現すべく、2024 年 6 月頃に生成 AI による支援機能を実装した。具体的には、画面上の入力フィールドに日本語または英語にて作成したいモデルの詳細を記載すると、自動的に該当のモデルが生成されるというものである(図 4)。ジルコニアに Pt/Rh 合金触媒を担持したモデルを自動生成する例を、YouTube にて公開している[17]。図 5 の QR コードより、動画を閲覧できる。この自動モデリングの機能は GPT-4o mini モデル[18]にて実装されており、画面の操作方法を生成 AI に学習させるために fine-tuning および many-shot prompting を使っている。興味のある方は、ぜひ 1 ヶ月の無料トライアルをご申請下さい[16]。

5. まとめ

第一原理計算をはじめとした材料シミュレーション技術と機械学習や AI 技術の融合は急速に進み続けている。特に機械学習力場の発展は著しく、この 5 年ほどで新しい手法が次々に誕生した。現状では汎用 GNN 力場の普及に伴って、当該技術の進展もようやく安定期に入ったように見受けられる。それと同時に、機械学習汎関数などの他の手法が今後発展していくことが予測されるわけだが、実際にどの手法が“次の GNN 力場”となるのかを言及することは難しい。しかしながら、ただ手を拱いて海外研究機関が成果物を公開してくれるのを待つだけでは、シミュレーションの専門家であっても直ちに時代遅れとなってしまう。逆に、シミュレーションを専門としない実験研究者にとっては、より大きなモデルをより小さい計算リソースで扱うことができるため、シミュレーションの敷居が下がりがつつある。さらに、生

成 AI を活用してモデリングなどの作業を自動化することで、理論計算研究者と実験研究者の差はより一層に小さくなっていく。

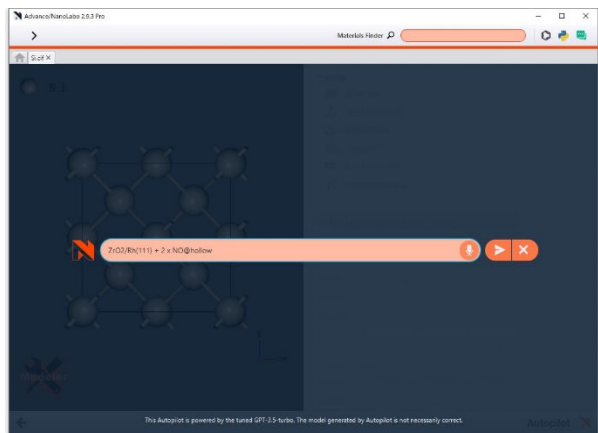


図 4 Advance/NanoLabo の AI 機能画面



図 5 担持触媒の表面モデルの自動生成動画

参考文献

- [1] J. Behler, IJQC 115, 1032 (2015).
- [2] M. Gastegger, et al., J. Chem. Phys. 148, 241709 (2018)
- [3] N. Artrith, et al., Phys. Rev. B 96, 014112 (2017)
- [4] A. Marcolongo, et al., ChemSystemsChem 2020, 2, e1900031.
- [5] 菅野了次, Electrochemistry, 85(9), 591-596 (2017)
- [6] Q. Xu, et al., WIREs Comput Mol Sci. 2024;14:e1724.
- [7] <https://next-gen.materialsproject.org>
- [8] C. Chen, et al., Nat Comput Sci 2, 718 (2022)
- [9] B. Deng, et al., Nature Machine Intelligence 5, 1031 (2023)
- [10] Y. Park, et al., J. Chem. Theory Comp. 20, 4857

(2024)

- [11] F. Imoto, et al., Phys. Rev. Research 3, 033198

(2021)

- [12] J. Lüder, et al., Electron. Struct. 6 045002

(2024)

- [13] L. Sun, M. Chen, Phys. Rev. B 109, 115135

(2024)

- [14] L. Sun, M. Chen, Electron. Struct. 6 045006

(2024)

- [15] J. Kirkpatrick, et al., Science 374, 1385 (2021)

- [16] <https://www.nanolabo.advancesoft.jp>

- [17] <https://www.youtube.com/watch?v=1wQcmTh5zOo>

- [18] <https://chatgpt.com>

※ 技術情報誌アドバンスシミュレーションは、それぞれの文献タイトルの下に記載した DOI から、PDF ファイル（カラー版）がダウンロードできます。また、本雑誌に記載された文献は、発行後に、JDREAMⅢ（日本最大級の科学技術文献情報データベース）に登録されます。