

材料研究を加速するデータ駆動手法：プロセス最適化、最小の実験数で全体を掴む

はじめに

機械学習・人工知能技術の活用による研究革新は、多くの材料研究分野で達成されつつある。特に、「次に行うべき実験条件選択をアシストするアルゴリズム」は、材料研究に対する人工知能（AI for Materials Science）とも呼ぶことができる。適切な実験条件を決められるアルゴリズムとして、ブラックボックス最適化手法がある。この最適化手法では、手持ちのデータから機械学習モデルを構築し、まだ実験していない未知材料に対する特性値を予測することで、次に行うべき実験条件を適切に選択できる。ブラックボックス最適化は多種多様な研究ニーズに対応するために複数の手法が考案され、多くの手法がオープンソースとして公開されている。これらの手法をうまく利用することで、様々な材料研究分野で自律駆動型（研究プロセスを人工知能やロボット実験で加速した）材料研究が実施できるようになってきた。そこで、本稿では、研究ニーズに依存したいくつかのブラックボックス最適化手法および、それらを同じプラットフォームで利用できるように整備した自律駆動型材料研究を支援するためのミドルウェア NIMO^(1,2)を紹介する。

まず初めに、ブラックボックス最適化を実施する際に用意するデータセットについて説明する。次に、研究ニーズの異なる3種類のアルゴリズムについて紹介する。1つ目は、良い特性値（機械特性や材料収率など）を得るためのベイズ最適化手法である。これを利用することで、プロセスや材料組成の最適化が実現できる。2つ目は、材料系を理解することを目的とした無目的探索手法である。これは、プロセスや材料組成の最適化を目的とせず、特性値がどのように分布するかを最小の実験数で把握するために考案されたアルゴリズムである。3つ目は、状態図・相図を少ない実験で描くためのアルゴリズムである。これも特性値の向上を狙っておらず、ベイズ最適化では直接扱えない問題である。最後に、これら3つのアルゴリズムを容易に利用できるように整備したミドルウェア NIMO の利用方法を紹介する。

探索空間データセット

ブラックボックス最適化を用いた材料開発で最も重要なことは、材料探索空間の策定であり、これを明記したデータセットの準備方法について紹介する。一般的には、探索空間は連続値、離散値どちらも扱えるが、ここでは NIMO での利用を念頭に置き、実験条件を全てリストアップした探索リストを用意し、それを材料探索空間として扱う場合を考える。例えば、温度、圧力の2パラメタのプロセスを最適化する場合、探索空間データセットは図1のようになる。各行が実験条件の1セットであり、温度、圧力は離散化して格納されている。最後の列に対象とする特性値を格納する列を用意している。すでに実験済みの実験条件に対しては、この列に特性値を格納する。これがブラックボックス最適化における初期学習データとなる。もし全く実験したデータがない場合、機械学習モデルを学習することができないため、いくつかの実験条件に対して、ランダムに選択、または研究者の直感で実験をし、初期データを用意する必要がある。一方で、未実験の実験条件に対しては空欄としておく。ブラックボックス最適化手法は、最後の列が空欄となっている実験条件から、次に行うべき実験条件を選択する。ここで注意すべきなのは、最適化したいパラメタが増えることによりリストアップする実験条件が膨大になってしまう点である。同様に、材料組成に関する探索空間も用意することができ、その例も図1に記載した。この探索空間を簡単に作ることでできる Web アプリとして、Materials Search Space Generator⁽³⁾を開発し公開している。これを利用することで、ブラックボックス最適化を実施するためのデータセットを容易に用意することができる。

プロセス最適化

温度	圧力	特性値
1450	5	
1450	6	100
1450	7	
1450	8	
1450	9	
1500	5	23
1500	6	
1500	7	
1500	8	
1500	9	19.8
1550	5	
:	:	:

← 実験済み
実験条件

← 未実験の
実験条件の場合
特性値は空欄

組成最適化

Ni	Co	Fe	特性値
0	0	1	
0	0.1	0.9	10.4
0	0.2	0.8	
0	0.3	0.7	
0	0.4	0.6	
0	0.5	0.5	
0	0.6	0.4	
0	0.7	0.3	30.5
0	0.8	0.2	
0	0.9	0.1	
0	1	0	
0.1	0	0.9	
:	:	:	:

図1 ブラックボックス最適化で利用する探索空間データセット。最後の列には特性値を入力する。実験済みの実験条件のみ特性値を入力し、未実験のものは空欄としておく。特性値が空欄となっている実験条件から選択が行われる。

良い特性値を得るためのベイズ最適化

良い特性値（機械特性や材料収率など）を得るために、プロセスや組成の最適化を行うことができるブラックボックス最適化手法が、ベイズ最適化である。

ベイズ最適化では、探索空間データセットにリストアップされた各実験条件を表す数値ベクトルを入力とし、ターゲットとする特性値を予測する機械学習モデルとしてガウス過程回帰を利用する。学習に使用するデータは、すでに実験が終わっており、探索空間データセットに特性値が記載されている実験条件である。ガウス過程を利用する利点として、特性値の予測値（予測値の平均値）およびその予測の不確かさ（予測値の分散）を評価できる点にある。ベイズ最適化では、「探索」と「活用」の考え方が重要である。「探索」では、近くに実験データが少ない領域では、良い材料が潜んでいるかもしれないと考え、予測の不確かさが大きい実験条件を選択する。一方で、「活用」では、特性値の予測値が向上する実験条件を選択する。ベイズ最適化では、これらを同時に取り込んだ選択を行うために、獲得関数と呼ばれる予測値とその分散を利用した関数を導入する。そして、獲得関数を最大化する実験条件を選定する。最も単純な獲得関数は、UCB (Upper Confidence Bound) と呼ばれ、予測値と分散の線型結合で定義される。これにより、予測値も分散値も大きい実験条件を適切に選択することができる。

選択された実験条件に従い、実験を実施し、得られた特性値

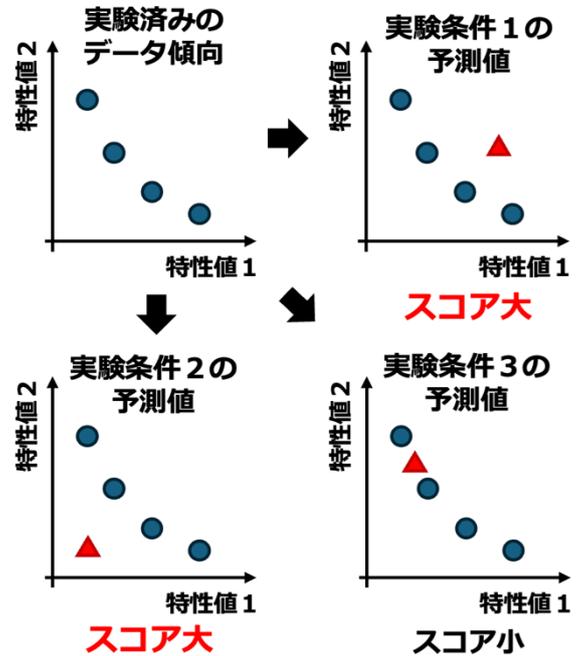


図2 無目的探索 BLOX における実験条件選択。予測値が実験済みのデータ傾向から外れた場合、スコアが大きくなり、次の実験条件として選ばれる。傾向と外れない実験条件 3 は選択されにくい。

を記載することで、材料空間データセットを更新する。そして、ガウス過程を再学習することで、次の実験条件が選択される。この実験とガウス過程による選択を繰り返すことで、徐々に特性値が向上していき、プロセスや組成の最適化が実施できる。ベイズ最適化用の Python パッケージとして、PHYSBO を開発している⁽⁴⁵⁾。PHYSBO では、ハイパーパラメータを完全に自動で決定でき、初期値の設定も必要がないため、探索空間データセットを用意できれば、他に決めないといけないパラメータはない。上記説明では、獲得関数が最大となる 1 つの実験条件を選択し、1 回の実験を行うという最適化を想定した説明をしたが、複数実験条件の提案も可能であり、複数の実験を並列で行う場合にも PHYSBO は対応している。さらに、複数の特性値を扱う多目的最適化問題も扱える。その一方で、PHYSBO を利用するためには、Python 環境の構築が必須であるが、環境構築が必要なく、Windows コンピュータ上で手軽にベイズ最適化が実行できる PHYSBO.exe も開発している⁽⁶⁾。図 1 に示す探索空間データセットを csv 形式で用意することで、次に実験すべき実験条件の提案が得られる仕様となっている。

ベイズ最適化を使用したプロセスパラメタ最適化の例として、ガスアトマイズ装置のプロセスパラメタ最適化が実施されている⁷⁾。超合金粉末の収率を向上させるために、溶湯温度とガス圧力を最適化した例である。詳しくは、日本語の解説記事⁸⁾を参照されたい。

全体を掴むための無目的探索

材料研究を行うモチベーションは最適化だけではないだろう。ここで紹介する無目的探索手法 BLOX は、特性値がどのように分布するかを少ない実験回数で知りたい場合に利用できる手法である⁹⁾。

BLOX ではまず、ベイズ最適化と同様に、特性値を予測する機械学習モデルを用意するところからスタートする。BLOX では、予測の分散値を使わないため、使用する機械学習モデルは回帰モデルであればどのようなものでも利用可能である。ここでは、簡単のために2つの特性値を対象とする場合を考える。特性値を予測するための機械学習モデルを用いることで、未実験の実験条件に対する特性値を予測することができる。この予測結果から、実験済みの特性値の傾向から大きく外れたものを次の実験条件として選定するのが BLOX である。傾向から外れているかどうかは、Stein discrepancy と呼ばれる指標を用いてスコアリングしている。例えば、図2のようなケースに対して、実験条件1および2のスコアが大きくなるように定義されている。ここでポイントとなるのが、特性値が大きくなる場合（図2の実験条件1）も小さくなる場合（図2の実験条件2）も傾向から外れていれさえすればスコアが大きくなるという点である。つまり、特性値が大きくなる実験条件を「最適化」するわけではなく、特性値空間の傾向からずれた多くの材料を見つけたいというアルゴリズムとなっている。実験と選択を繰り返すことにより、特性値のバラエティに富んだ材料を探索することができる。それにより、少ない実験回数で、特性値の分布を俯瞰的に描くことができ、材料系の理解に貢献することができる。

一点特記したい、この BLOX の特徴は、特性値の空間でバラエティに富んだ材料を探索する点である。一方で、実験条件の空間でバラエティに富んだ実験を行う場合、ベイズ最適化における「探索」のみを実施すれば良い。これは不確実性サンプリングと呼ばれる手法であり、バラエティを持たせたいのが、実験条件なのか、特性値なのかによってアルゴリズムを適切に

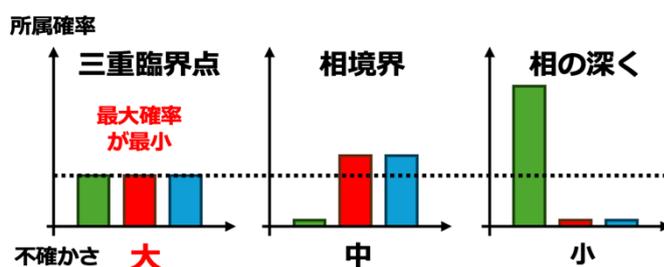


図3 PDCにおけるスコアリング例。LCを利用すると、最大の確率が最小となる実験条件が選ばれるため、三重臨界点を選択されやすいスコアである。

選択する必要がある。

状態図を描くための最適化手法

状態図を描くためのアルゴリズムとして、PDC(Phase Diagram Construction)を開発している¹⁰⁾。PDCを利用する場合も利用する探索空間データセットは図1と同様である。異なる点としては、特性値を入力する列に、各相を表す整数値のインデックスを入力する点である。初期データとして、 P 個の相がある場合、 $1, \dots, P$ の整数値をそれぞれの相に与え、探索空間データセットを作成する。

PDCでは、ラベル伝搬法 (Label propagation method) という機械学習手法を利用して、状態図を予測する。このアルゴリズムは、学習データとして入力された相情報を状態図内で伝播し、まだ実験していない実験条件においてどの相に所属するかの確率が計算できる手法である。その確率を利用し、状態図中

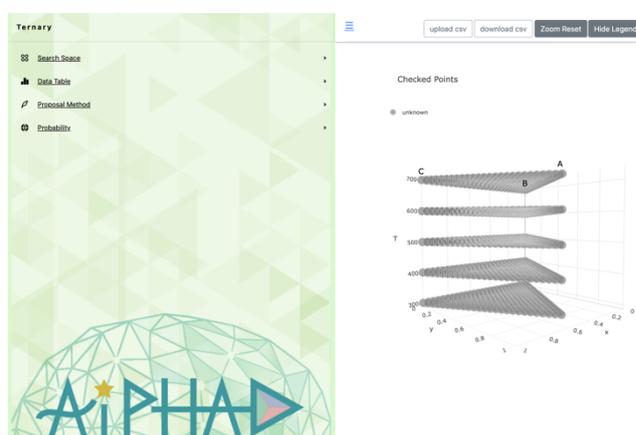


図4 状態図作成用 Web アプリケーション AIPHAD の操作画面。温度依存の3元系状態図作成の例を示した。

で最も不確かな実験条件を選択する手法である。不確かさを表すスコアリングとして、Least Confidence (LC)法がある。LCは、所属確率のうち最大の確率が最小となる実験条件が選ばれるように定義される。例えば、3つの相が初期データにある場合、図3のような確率が得られたとする。その場合、LCを利用すると、三重臨界点と予測される実験条件で最もスコアが大きく、続いて相境界、最後に相に深く入り込んだ実験条件という順でスコアが小さくなる。つまり、PDCでは、三重臨界点や相境界と予測された箇所が次に実験すべき実験条件として選択されるように設計されている。このように選択された実験条件に従い、実験により相を同定し、材料空間データセットに相情報を更新する。そして、次の実験条件の選択を繰り返すことで、状態図が徐々に完成していき、少ない実験回数で詳細な状態図を描くことができる。もし、初期データには含まれていない、新しい相が発見できた場合、相を表すインデックスを一つ増やし、 $1, \dots, P + 1$ とすれば良い。

PDCの材料研究への応用例として、薄膜成長条件相図作成が実施されている。分子線エピタキシー (MBE) による Zn-Sn-P 膜成長に関する新規相図を PDC により作成した。詳しくは日本語の解説記事⁽¹⁾を参照されたい。また、PDC アルゴリズムをより直感的に利用してもらうことを目的として、Web アプリケーション AIPHAD⁽²⁾を開発、公開している(図4)。状態図は、可視化が重要であるため、可視化機能も搭載し、作成される状態図を見ながら、ブラックボックス最適化からの提案も得られる仕様となっている。

自律材料開発を支援するミドルウェア

これまで3種類のアルゴリズムを説明してきた。これらのアルゴリズムを選択するだけで容易に利用できるように開発したミドルウェアが NIMO である。NIMO ではさらに、これらのアルゴリズムとロボット実験装置との連携を見据えた仕様となっており、実験を担当するのが人間であってもロボットであっても、自律型材料研究プラットフォームを容易に実行できるソフトウェアである(図5)。

NIMO は Python 言語で実装されており、次の実験条件を提案するアルゴリズムとして、PHYSBO (ベイズ最適化)、BLOX (無目的探索)、PDC (相図探索) を指定することで、探索空間データベースから、それぞれのアルゴリズムに従った提案を受けることができる。このとき、探索空間データベース



図5 自律駆動型研究を支援するためのミドルウェア NIMO のイメージ。NIMO を利用することで、ベイズ最適化、無目的探索、相図探索といった複数のアルゴリズムが同じプラットフォームで利用可能となる。

は candidates.csv として NIMO に読み込ませることとなる。複数の目的関数を扱う多目的最適化問題や、複数の実験候補を一度に実験するバッチ実験にも対応した材料選択も行える。選択された結果は、proposals.csv というファイルに記載されるため、その提案に従い実験を実施し、candidates.csv を更新することで、実験と AI のクローズドループを容易に形成することができる。NIMO を利用することで、全く同じプラットフォームで異なる研究ニーズを扱うことができるため、様々なタイプの自律型材料研究を同一の装置で実施できる。

また、自律自動実験用の機能として、次の実験条件が提案された後、ロボット実験を実施させる機能も備えている。ロボット実験装置に対する入力ファイルを作成し、ロボット実験をスタートさせるプログラムである。また、実験が終わると、実験結果を解析し、探索空間データベースファイルを自動でアップデートするプログラムも実装されている。一方で、これらロボット実験装置を利用するためのプログラムは、使用するロボットに依存するため、新しいロボット実験装置を利用する場合は、それぞれ作成する必要がある。現状では、NIMS 電気化学自動実験ロボット (NAREE) 用のプログラムが標準実装されており、それを参考に、できるだけ少ない修正で他のロボット実験装置も使用できるようにプログラムを用意している。オリジナルの新しい AI アルゴリズムやロボット実験装置用モジュールを利用したい場合にも簡単に対応できるような仕様を導入している。一度、NIMO とロボット実験装置の連携が完成すると、様々なアルゴリズムが使用できるだけでなく、新しいアルゴリズムの導入もスムーズとなる利点がある。

NIMO を利用して、NAREE をコントロールすることで、電解液探索を自律自動実験として行なった研究例がある。実行例

は、日本語解説記事や紹介動画⁽¹³⁾があり、これらを参照されたい。

おわりに

本稿では、材料研究に対する人工知能として利用できるブラックボックス最適化手法を紹介した。研究ニーズによってこれらのアルゴリズムは使い分ける必要があるが、容易にこれらを利用するためのミドルウェア NIMO も開発が進んでおり、プログラムにあまり詳しくない研究者・技術者でもこれらの手法の恩恵を得られるように整備を進めているところである。「プロセスを最適化したい」、「材料系の全体をいち早く掴みたい」、「高速に状態図が欲しい」などのニーズがある場合、ぜひ一度、機械学習・人工知能の力を借りてみるのはどうだろう。

執筆者プロフィール

<非会員>

田村 亮

◎国立研究開発法人 物質・材料研究機構 マテリアル基盤研究センター チームリーダー

◎専門：マテリアルズインフォマティクス

〒305-0044

茨城県つくば市並木 1-1

tamura.ryo@nims.go.jp



参考文献（本文記載順）

- (1) R. Tamura, K. Tsuda, and S. Matsuda, *Sci. Technol. Adv. Mater. Meth.* 3, 2232297 (2023).
- (2) <https://github.com/NIMS-DA/nimo> (参照日2025年1月26日)
- (3) <http://app.aiphad.org/mssg.html> (参照日2025年1月26日)
- (4) Y. Motoyama, R. Tamura, K. Yoshimi, K. Terayama, T. Ueno, and K.

- Tsuda, *Comput. Phys. Commun.* 278, 108405 (2022).
- (5) <https://www.pasums.issp.u-tokyo.ac.jp/physbo/> (参照日2025年1月26日)
- (6) https://isspns-gitlab.issp.u-tokyo.ac.jp/physbo-dev/physbo-gallery/-/tree/master/tools/physbo_exe (参照日2025年1月26日)
- (7) R. Tamura, T. Osada, K. Minagawa, T. Kohata, M. Hirosawa, K. Tsuda, and K. Kawagishi, *Mater. Des.* 198, 109290 (2021).
- (8) 田村亮, 金属材料開発に活用できるブラックボックス最適化手法の開発, 溶射, 第59巻, 第4号, pp.217-223 (2022)
<https://doi.org/10.11330/jtss.59.217>
- (9) K. Terayama, M. Sumita, R. Tamura, D. T. Payne, M. K. Chahal, S. Ishihara, and K. Tsuda, *Chem. Sci.* 11, 5959-5968 (2020).
<https://doi.org/10.1039/d0sc00982b>
- (10) K. Terayama, R. Tamura, Y. Nose, H. Hiramatsu, H. Hosono, Y. Okuno, and K. Tsuda, *Phys. Rev. Mater.* 3, 033802 (2019).
<https://doi.org/10.1103/PhysRevMaterials.3.033802>
- (11) 田村亮, 寺山慧, 勝部涼司, 野瀬嘉太郎, 機械学習による相図作成の効率化, 応用物理, 第91巻, 第2号, p.96 (2022)
https://doi.org/10.11470/oubutsu.91.2_96
- (12) <https://aiphad.org/> (参照日2025年1月26日)
- (13) <https://www.nims.go.jp/press/2023/07/202307200.html> (参照日2025年1月26日)