OUKA
The University of Osaka
Institutional Knowledge Archive

Title	AIとCAEを連携させるPINNsの実装「NVIDIA Modulus」のご紹介
Author(s)	柴田, 良一
Citation	サイバーメディアHPCジャーナル. 2024, 14, p. 2-8
Version Type	VoR
URL	https://doi.org/10.18910/96512
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

AI と CAE を連携させる PINNs の実装「NVIDIA Modulus」のご紹介

柴田 良一
岐阜工業高等専門学校 建築学科

1. ものづくりでの PINNs の可能性と期待

ものづくりにおいては、「開発・設計・生産・管理」の様々な場面で、問題解決や意思決定が必要とされます。この４つの場面を実現する手法は、目的や条件によって適切な選択が不可欠です。現時点では「ものづくりにおける問題解決手法」を大きく分類すると、４つの手法があり、これらは中央大学 AI・データサイエンスセンター所長：樋口知之先生の資料を参考にすると、「演繹的と帰納的」の２つの側面で分類できます。

この考え方をもとに、著者が問題解決手法を図解した結果である図１を用いて説明します。今後が期待される「これからの問題解決」では、最も基本的な手法である「従来の問題解決」から始まり、様々な技術発展を取り込みながら展開して、今後は人工知能で対象とする情報、つまりデータを中心とした「データ指向・データ同化」が重要になってくると考えています。

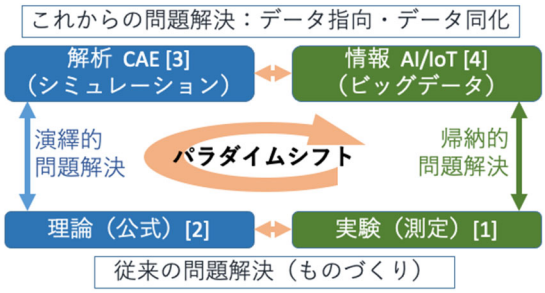

図１：ものづくりにおける問題解決手法の展開

まず、第１段階は、右下[1]の「実験（測定）」です。現実の対象物や模型などに条件を設定した実験を行い、その挙動を測定することで必要とする知見を得ようとする手法です。最も確実といえますが、想定外の影響が結果を支配することもあり、様々な誤差もあることに注意が必要です。

次に、第２段階は、左下[2]の「理論（公式）」です。実験や観察から得られた知見をもとに、数学的な記述を基本としながら、理論的に法則性を見つけ出し公式としてまとめることです。固有の現象から汎用な法則を導くことにより、広範囲の問題解決が可能ですが、適用範囲を意識する必要があります。

さらに第３段階は、左上[3]の「解析 CAE（シミュレーション）」です。「Computer Aided Engineering」は、コンピュータ支援による工学解析として、問題対象の数式によるモデル化を、コンピュータ内でプログラミングすることにより、膨大な計算を処理することが可能です。これにより現実的で複雑な問題を対象にできますが、実際は計算コストが制限となってきます。

最後の第４段階は、右上[4]の「情報 AI/IoT（ビッグデータ）」です。これまでの３つの段階で得られる「測定結果・物理法則・解析結果」などを前提に、新しいコンピュータの活用方法である人工知能により問題を解決することができます。この手法は現在最も注目され、試行錯誤の段階です。

この図をみると、右側が情報を中心とした「帰納的問題解決」で左側が理論を中心とした「演繹的問題解決」となっています。解析手段は時代の進展とともに、右下の[1]から時計回りに展開していき、段階的にパラダイムシフトを進めていることがわかります。

これらの問題解決手段は、４つの手法が置換されるのではなく、それぞれが高度化しながら目的の条件によって適材適所で選択されるものですが、これらの多面的な手段によって、幅広い問題解決が可能となっています。ここで、現時点で最も注目を集めるのは、[4]の情報を対象とした人

工知能技術を用いた問題解決手段です。他の３つの手段と連携することで、新しい可能性が日進月歩で開発されています。

現状の CAE に対する課題を解決するために、人工知能技術の１つである機械学習と連携して、ニューラルネットワークの活用による様々な研究が始まっています。この中で、特に下記の２つ「最適化設計・原因分析」は、ものづくりにおいて実現が期待される大きな目的です。

【目的１：最適化設計：パラメトリックな解析】
ものづくりにおける設計過程においては、製品の目的を満足するための設計変数の決定が必要です。この場合には、膨大な数のパラメータの組合せケースに対して、１ケースずつ設計評価を行う数値解析を繰返して実行する必要があります。

【目的２：原因分析：逆問題による要因の特定】
製品の活用において、想定外の条件により破損などが生じた場合は、その要因を明確にして、期待する目的を満足するために、設計変数の値を逆に推定する必要もあります。このような処理は、逆問題で実現可能であり、安全な製品設計を目指すものづくりでの活用が期待されています。

このような課題に対して、ニューラルネットワークを活用する場合、これまでは「データ中心の問題解決」が行われており、特に学習データにおいてビッグデータを前提として膨大な情報の準備が必要となっていました。確かに問題解決手段[1]の「実験（測定）」とニューラルネットワークとの連携であれば、センサーによる大量のデータを準備することは容易です。しかし問題解決手段[3]の「解析 CAE（シミュレーション）」とニューラルネットワーク（用語の記述では「NN」と省略）との連携の場合には、学習データの確保が不可欠です。従来の手法では、数値解析により多数の学習データを事前に準備する必要があり、膨大な計算時間がこの「CAE と NN との連携」の実用化において、大きな課題となっています。

この「CAE と NN との連携」において、残された可能性として問題解決手段[2]の「理論（公式）」に注目します。つまり、CAE の基礎となっている理論（方程式）に注目することで、従来のニューラルネットワークの前提であった膨大な学習データのみを活用するだけでなく、新しいニューラルネットワークの可能性が期待されます。

これが「PINNs（Physics Informed Neural Networks : 物理法則を活用したニューラルネットワーク）」です。具体的には、ニューラルネットワークの学習においてデータのみではなく、対象となる現象を記述する微分方程式や境界条件を活用することにより、データに依存しないニューラルネットワークの学習を実現しています。

この研究手法からみると、従来の学習データのみのニューラルネットワークの学習では、入力条件と出力結果の関係性のみを対象とするために、物理現象を規定する微分方程式や境界条件を、厳密に満足する保証がないという問題がありました。この問題に対して PINNs では、これらの物理的な法則や条件をニューラルネットワークの学習に組み込んでいるため、より合理的な出力結果が得られる特徴があります。

2. NVIDIA Modulus の仕組みと特徴

この PINNs の実装の１つとして、NVIDIA 社が開発し、2021 年より公開され現在はオープンソーツとなっている「Modulus : モデュラス」（参考文献１）についてご紹介します。

NVIDIA Modulus の本質は、次のように説明できます。「偏微分形式の支配方程式や境界条件で定義された物理法則の情報を、ニューラルネットワークの学習用データと統合することで、高い表現能力を持ちながらパラメータによる問題定義を可能にした、深層学習によるニューラルネットワークを用いたサロゲートモデルの構築を可能にするツールです。」

この Modulus を、問題解決の基盤（プラットホーム）として用いることにより、様々な分野で応用できるニューラルネットワークによる問題解決を実現することが可能になります。具体的に

は、ニューラルネットワークの学習処理における複雑さを軽減し、効率的な実装を実現しています。これにより、対象とする問題の挙動に関する物理現象の知識（物理法則）を、ニューラルネットワークモデルの学習に反映させて有効活用できます。結果的に、「CAE と NN との連携」の１つの可能性として、効率的に高性能な PINNs の構築を可能にします。

Modulus は、物理法則を活用した機械学習によるサロゲートモデルの開発に対して、各種機能を実現する Python のクラスや関数を提供しています。これらの高度な機能を提供するプログラミング支援機能により、PINNs の実装における学習や推論の複雑な処理を効率化することができます。このため、異なる問題領域や活用対象に対して容易に拡張可能となっており、ものづくりをはじめ工学や科学での応用において、人工知能技術の革新的な応用を可能にする有益なソフトウエア・プラットフォームです。

さらに Modulus は、NVIDIA の人工知能用の階層的ソフトウエア群の１つとして設計され、NVIDIA Omniverse などの外部ツールとの連携が考慮されています。対象問題の規模としては、単純で例題的な問題から複雑な現実世界の問題までを容易に拡張可能であり、対応するハードウエアとしては、単一 の GPU 利用から複数ノードの GPU 利用まで、制限のない GPU 拡張に対応しています。これにより、複雑な問題において多数のパラメータで表現された物理法則を対象として、機械学習によるサロゲートモデルの構築を可能にしています。この処理能力を活用して、対象とする物理現象の設計変数や設計空間を統合化してモデルを作成することで、利用者が複数の設計点や挙動の変化を、ほぼリアルタイムに推測することができます。

このような高度な GPU 活用の動作環境により、事前に一度だけ学習を行い PINNs によるサロゲートモデルを構築できれば、あとは何度でもリアルタイム（短い処理時間で入力に対して すぐに結果を得る）で予測を行うことができます。つまり、従来のシミュレーションと比較して、桁違いの高速化を達成しています。

Modulus のプログラミングにおいては、Python 言語を用いています。プログラミングにおいては、高度な処理を実現する API を用いることで、対象とする問題の高度な抽象化を可能にして、それぞれの問題領域の専門知識の活用を容易にしています。つまり、本書で解説する例題は、新しい問題に対しての既存解法を拡張するための重要な出発点になり、対象領域の支配法則の Python クラスを定義するために、既存の同様な支配法則の Python クラスを拡張することができます。

著者が想定する Modulus の利用者としては、下記の方々が考えられます。

・工学や科学でのシミュレーション技術を対象とし、新しい可能性を開拓すべく、従来の問題解決手法を改善するために、新しい人工知能ベースの手法を開発する研究者

・ものづくりのエンジニアリングと科学的シミュレーションを統合し、革新的な機能を持つ解析技術の実現のために、ニューラルネットワーク活用手法を実現する開発者

・本書が目標とする「CAE と NN との連携」を目指して、問題解決手法として最適化設計や逆問題解析を実現するような、デジタルツインアプリケーションを展開する技術者

産業分野としては、Modulus は自動車、航空宇宙、機械製造、石油ガス、医療医薬、電子製品などの幅広い業界を対象としています。これらの分野で必要となる問題解決技術として、流体解析、熱伝導解析、構造解析、電磁界解析などや、これらが連成したマルチフィジックスシミュレーションで使用することができます。

Modulus の解析機能の範囲をまとめた図２をみると、Modulus が実装する機械学習やニューラルネットワークの対応範囲は、概略的には、従来の「Category I : 物理法則による数値解析による順問題解法（Forward Solution）」、「Category III : 物

理法則は使わずデータのみを基本とするデータ指向解法（Data-Driven Solution）」に対して、新しい「Category II：物理法則とデータ同化の両方を含むようなハイブリッド解法（Inverse Solution / Data Assimilation）」、の３つのカテゴリーに分類することができます。Modulus の特徴であるハイブリッド解法としては、具体的には「逆問題解法とデータ同化」といえます。

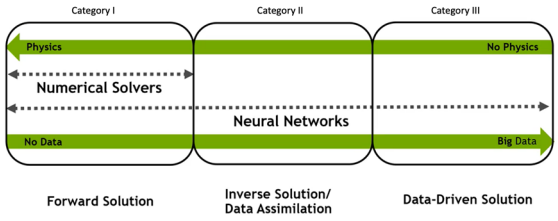

図２：Modulus の対応範囲

次に、Modulus のソフトウエア構成の概要を、図３を用いて簡単に説明します。Modulus は、物理法則を用いた学習で用いる対象形状の生成や入力のための Python 用 API を提供しているだけでなく、先の Category III : データ志向のニューラルネットワーク学習で用いる効果的なデータ導入を実現するための、いくつかのツールも用意しています。

例えば、効率的なニューラルネットワーク処理を実現するために、人工知能研究分野で開発された標準的で多彩なニューラルネットワーク構成に対応して、これらを実現するソフトウエア機能を実装しています。さらには、対象となる問題の物理法則の偏微分方程式を効率的に定義するため、参照可能な基盤となる方程式なども提供しています。

Modulus は実際のニューラルネットワーク処理において、学習対象や制約条件として定義される複雑な問題に対して、最適化された計算処理手順を生成する機能を実装し、これらを効果的に制御することを可能にしています。なお、データ指向の手法を採用せずに、物理法則で記述されたモデルによる Category I の処理も同様に、これらの最適化手法が適用されます。

最後に、対象とするデータの可視化の側面からも柔軟な機能を提供しており、利用者は Modulus の処理モデルを用いた操作において、Omniverse、TensorBoard、ParaView などのツールを連携して利用することができます。

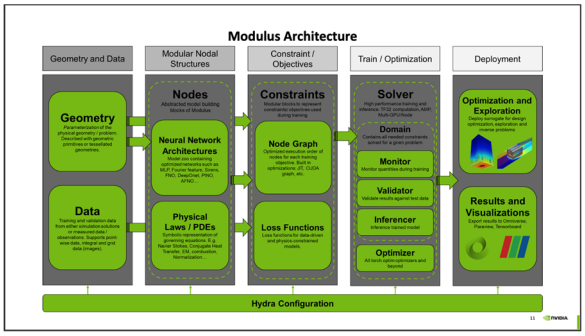

図３：Modulus のシステム構成

図４は、Modulus で開発を行う場合の標準的な処理手順を示しています。全ての PINNs の問題が、完全にこの処理手順どおりに実行するとは限りませんが、ここでは Modulus に関する標準的な情報として、この図を用います。

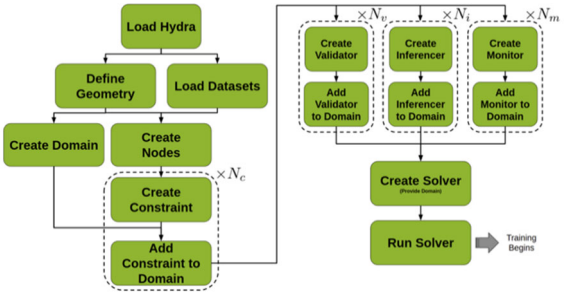

図４：Modulus で開発を行う時に用いる標準的な処理手順

この処理の重要な手順は以下の通りです。

・Load Hydra（ハイドラの読込）

YAML 形式の設定ファイルを読み込むための、Modulus のメインオブジェクトのデコレーターを用いてハイドラを初期化します。

・Load Datasets（データセットの読込）

処理において必要なデータを読み込みます。

・Define Geometry（形状の定義）

必要な対象システムの形状を定義します。

・Create Nodes（ノードの生成）
要求された任意のノードを生成し、ニューラルネットワークのモデルで利用されます。
・Create Domain（処理手続の生成）
学習用処理手続のオブジェクトを生成します。
・Create Constraint と Add Constraint to Domain
（制約条件生成と処理手続への制約条件追加）
各制約条件を順番に生成して、処理手続に対してこれを追加します。
・Create Solver（ソルバーの生成）
学習処理手続としてデータが設定されたソルバーを初期化します。
・Run Solver（ソルバーの実行）
準備したソルバーを実行します。学習処理の結果として、物理現象の問題を解くためのニューラルネットワークを最適化します。

3. NVIDIA Modulus で提供される例題の紹介

最初に、工学的な問題解決における Modulus の特徴や、それらの特徴が Modulus の持つ機能にどう関連しているかをまとめておきます。

微分方程式で定義された物理シミュレーションを解析するためには、問題の解析空間に関する情報と、対象となる物理現象の支配方程式や境界条件の定義が必要となります。これらの情報と定義を活用することで、目的とする結果を得るための解析空間に対応する支配方程式を設定し、これを解く Modulus のソルバーを選択することができます。さらに Modulus では、このような問題の定義や解法だけでなく、処理経過の確認（モニタリング）や結果の可視化（ポストプロセス）などを実行する各段階に対して、有用となる機能や関数が提供されています。

対象となる解析空間の定義では、Modulus が持つ統合可能なソリッド形状生成の機能モジュール、STL ファイルによる形状定義の機能モジュール、または CSV/NumPy/HDF5 などの配列形式によるデータファイル、外部ツールの標準的な形式のデータファイルなどの利用が可能です。

Modulus の PINNs の処理においては、この形状または点群（ポイントクラウド）が定義できれば、境界条件を定義するための境界上の点を部分的に指定することが可能になります。さらに、物理法則を記述した偏微分方程式や常微分方程式は、対象とする内部領域において誤差を最小化するように解法が定義されます。

ここでは、Modulus を活用した基本的な例題として、「2 次元矩形空間でのキャビティ流れの確認」を取り上げます。

この例題演習では、NVIDIA Modulus を活用して、上部壁駆動によるキャビティ（以下 LDC と略記）流れを対象として、２次元流れ解析の PINNs による問題解決の手順を段階的に説明します。具体的には、次の５ステップで問題解決を進めます。

1. 形状モジュールで２次元形状を生成する
2. 境界条件を設定する
3. 解析対象となる流れの支配方程式を選択する
4. 残差ロスを評価して PINNs を学習する
5. 基本的な結果の可視化を実行する

解析空間であるキャビティの形状を、図５に示しています。これは正方形の空間で、その辺長は縦横 0.1 m の長さとします。空間の正方形の中心を、直交座標系の原点として定義して、x 方向は右に向かって増加する座標で、y 方向は上に向かって増加する座標としています。矩形の空間を囲む４つの壁は、左と右と下の辺は静止した壁で、上部の壁は x 方向に１ m/s の速度で移動する設定です。

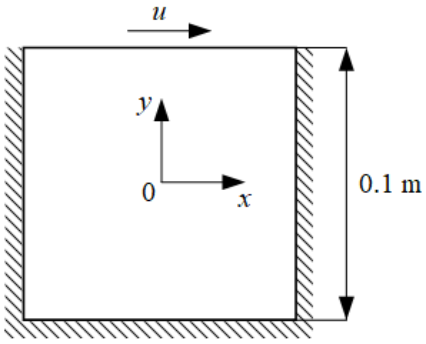

図５：上部壁駆動によるキャビティの形状

処理結果の例として、関数 add_validator()を用いて、問題空間に定義された点群上で学習処理されたデータを可視化してみます。この比較検討に用いる問題空間は、参照解と比較してデータを評価するために必要な定義です。この比較用の結果データは、拡張子.vtp や.npz 形式のファイルになっています。この拡張子.vtp ファイルは、ParaView のような可視化ツールを用いて確認できます。このファイルを含む validators ディレクトリにある拡張子.vtp と.npz ファイルは、選択された点群における予測として、正解値（検証用データ）、予測値（モデルの推論）などの値を提示しています。

例えば、ファイル./validators/validator.vtp は、正解値とネットワークによる予測値の変数として “u”, “v” と “p”に対応するように、true_u, true_v, true_p, と pred_u, pred_v, pred_p などの変数の値を保存しています。図６は、このような変数に関して正解値と Modulus の予測値との比較検討を示しています。

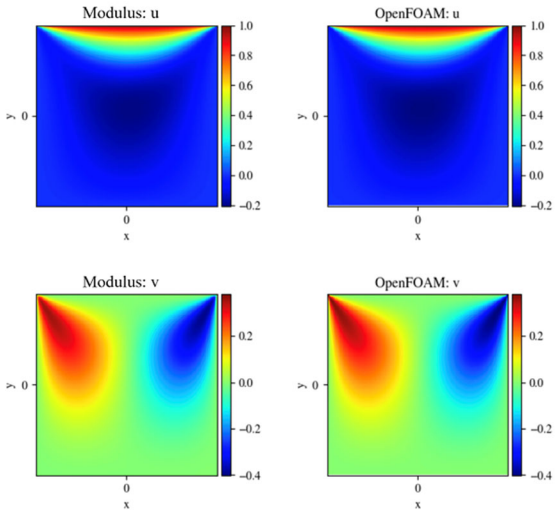

図６：OpenFOAM の結果との比較検討

Modulus の学習処理においては、その収束過程を把握するために、ニューラルネットワークの学習過程の可視化ツール TensorBoard は、機械学習の検証の可視化においてとても有用です。Modulus による PINNs の実行の過程で得られる情報（学習や評価の損失など）を可視化するために、図７に示すように TensorBoard を利用できます。

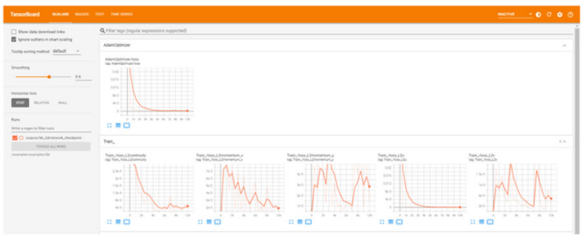

図７：可視化用の TensorBoard インターフェス

この TensorBoard に接続したブラウザでは、学習過程の各ステップにおける、様々な損失を表示します。このグラフの中で、１段目にある AdamOptimizer 損失は、ネットワークで計算される全てのロスを表わしています。さらに、２段目の左から３つの loss_continuity, loss_momentum_x と loss_momentum_y は、質量保存式とナビエ・ストークス方程式の x 軸と y 軸のそれぞれの方向に関する L2 損失を表示しています。また、２段目の右から２つの loss_u と loss_v は、境界条件がどれぐらい正しく満足しているかを示す L2 損失を表示しています。

4. おわりに

本紹介記事は、AI と CAE を融合した新しい問題解決技術である「Physics-informed Machine Learning (Physics-ML)」の中で、特に「Physics Informed Neural Networks (PINNs)」に興味を持った技術者の方が NVIDIA Modulus を活用するための概要を提供することを目的としています。

対象とする利用者としては、AI と CAE に関心があり、Ubuntu のコマンドライン操作に対応できて、Python の基礎的な知識があれば、NVIDIA Modulus による PINNs を実行できるようになります。皆さんの問題解決において、AI+CAE の最先端技術である NVIDIA Modulus が役立つことを期待しています。

参考文献

(1) はじめての NVIDIA Modulus: Physics-ML 物理に基づいた機械学習による工学シミュレーション，工学社，(2023).
（本原稿は、上記自著から引用しています）

謝辞

本原稿中の図版の引用に関しては、NVIDIA さまのご協力ご許可を得ましたことに、感謝申し上げます。