

ディープラーニングを用いた流体定常計算の収束・発散予測法の提案

～安全な設計者 CAE を目指して～

松原 大輔^{1†}

¹オープン CAE 勉強会@関西

Proposal of convergence or non-convergence prediction method of steady-state CFD

For the safe designner's CAE

Daisuke MATSUBARA^{*†}

^{*}OpenCAE Local user group@KANSAI

Abstract

Always, CAE (especially CFD) is difficult for designer because of the validity of the result. To help designer, this paper introduces an artificial neural network model to predict the convergence of the non-convergence of the CFD model. Input of the model is a residual history of OpenFOAM's tutorial cases, especially incompressible steady-state cases is used. At the end, future tasks are stated.

Keywords: OpenFOAM, simpleFoam, convergence prediction, Deep Larning, Designer's CAE

1. はじめに

近年、設計者 CAE を導入し納期短縮や品質向上を目指す企業が増えており、計算結果の妥当性を確認することは益々重要になっていくことが予想できる。特に CFD の場合は熟練技術が必要であり、数値振動した結果を設計に用いた場合、非常に危険である。そこで、本論文はディープラーニングを用いて、計算中の残差のふるまいから数値振動及び発散を予測することを目的としている。また、数ステップの計算で発散の予測ができれば、解析に要する時間のある程度短縮することができる。なお流体計算には OpenFOAM v5 を、学習には Keras(Tensor flow)を使用した。

2. 提案手法

収束および振動・発散した計算ケースに対して、空間残差（圧力 p 、流速 U 、乱流エネルギー k 、消散率 ϵ などの乱流パラメータ）の初期 100step の履歴を、収束した場合としなかった場合で教師あり学習させ、新規の計算ケースで収束および振動・発散を予測することを目的とする。

学習データは tutorials の simpleFoam を用いた計算ケース(全 10 ケース)を用いた。収束データはデフォルトの計算結果と、各種の安定化手法（①移流項の一次風上化、②緩和係数の低下）を組み合わせた計算結果を用いた。発散及び振動データは、デフォルトの計算条件に対して不安定化(①移流項の中心差分化、②緩和係数の増大、③代数方程式ソルバの残差の絶対値を増加させた場合)させ、これらの条件を組み合わせた計算結果を用いた。さらに乱流モデルを変えた場合の計算結果を用いることでデータ数の水増しを行った。データ数は全 500 個の時系列データとした(収束ケースは 4.5 割)。

学習データの前処理は、数値を 0 から 1 の間にするために、残差の対数を取り、マイナスを乗じたのち、それらの最大値の商とした。入力層は利用する残差の step 数と同じ 100 ノードとし、中間層は 7 層の 500 ノードとし、出力層は収束または振動・発散を 2 層のワンホットベクトルで表現した。中間層の活性化関数は ReLU を用いた。重みの初期値は He の正規分布で与えた。オーバーフィッティングと勾配消失を避けるために Batch normalization と dropout を確率 0.5 で適用した。Loss 関数は categorical-cross-entropy を、最適化アルゴリズムは Adam(学習率 0.001)とした。本手法の比較手法は中間層が 1 層の 500 ノードの単純なニューラルネットとした。

[†] E-mail address of corresponding author: matsunenekao@gmail.com

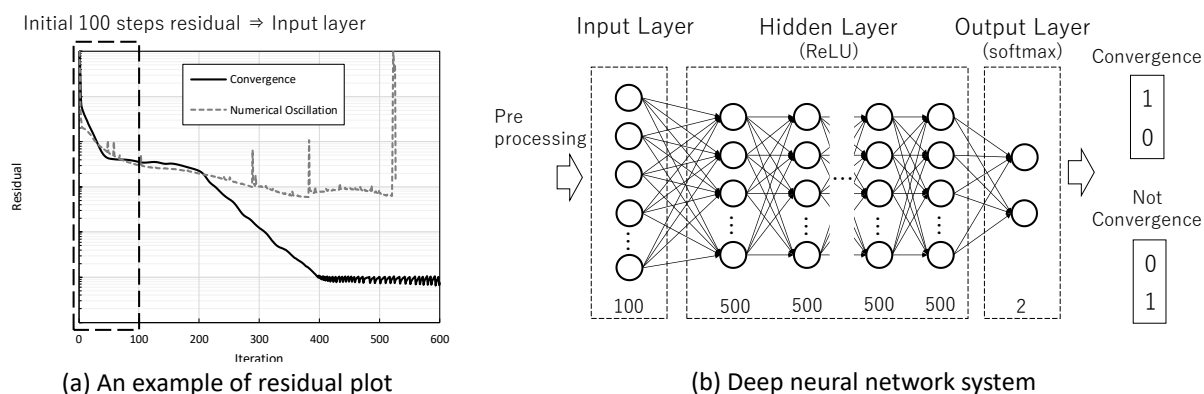


Fig. 1 Overview of the proposed method

3. 検証データと学習結果

表 1 にディープニューラルネットワークを用いた結果と、ニューラルネットワークを用いた結果を示す。検証データは、全データから無作為抽出した 20% の結果を用いた。表 1 からディープニューラルネットワークを用いた場合、ニューラルネットワークの場合と比べて、学習時の精度と損失関数の検証ケースとの差が小さく、過学習が抑えられている可能性がある。しかし、精度と損失関数は共に十分な値と言えず、さらなる改良が必要だと言える。今回誤差の発生した原因として学習データに様々な物理量の残差を混合したものをを用いたことが考えられる。物理量ごとに残差の変化の振舞いは全くことなるためである。今回は学習データ不足からこのような方法を選んだが、講演時にはさらに精度の高い結果を報告したい。

Table 1 Results of DNN and NN

	DNN		NN	
	学習ケース	検証ケース	学習ケース	検証ケース
精度	0.8725	0.85	0.90	0.81
損失関数	0.324	0.350	0.233	0.394

4. おわりに

本研究では、ディープラーニングを用いた流体定常計算に対する収束及び発散予測法の開発を行い、その適用性の確認を行った。今後の課題として、データ量を増やし、さらなる検証を行い、信頼性を高める必要がある。本手法が確立すれば、設計者が設定すべき緩和係数などのパラメータを減らせる可能性がある。

参考文献

- [1] 巢籠悠輔. 詳解ディープラーニング : TensorFlow・Keras による時系列データ処理. マイナビ出版, 2016. <https://ci.nii.ac.jp/ncid/BB23702347>, (accessed 2018-10-10).