

オートエンコーダを用いたチャネル乱流の速度場と壁面圧力変動場の相互推定

Estimation of the Wall Pressure Fluctuation and the Velocity Field in a Turbulent Channel Flow Using Auto-Encoder

- 鈴木雄平, 明大院, 神奈川県川崎市多摩区東三田 1-1-1, E-mail: ce182033@meiji.ac.jp
 中吉嗣, 明大理工, 神奈川県川崎市多摩区東三田 1-1-1, E-mail: naka@meiji.ac.jp
 Yuhei Suzuki, Meiji University, 1-1-1 Higashimita, Tama-ku, Kawasaki-shi, Kanagawa
 Yoshitsugu Naka, Meiji University, 1-1-1 Higashimita, Tama-ku, Kawasaki-shi, Kanagawa

A combination of auto-encoders is formulated to estimate the wall pressure fluctuation field from the velocity field in a turbulent channel flow. Each of velocity field data and the wall pressure fluctuation field data is compressed by auto-encoder. These auto-encoders exhibit successful results showing the output data are qualitatively in good agreement with the input data. Then the intermediate compressed velocity and pressure data of each encoder are connected through the layer networks to estimate the compressed pressure from the compressed velocity data. The convergence of the proposed network is unsatisfactory for the moment, i.e., the value of the loss function is more than one-order of magnitude larger than the auto-encoders. This might lead unphysical discontinuities in the estimated intermediate pressure data. The results would be improved by choosing better network configurations.

1. 結論

乱流能動制御を行う上で、制御対象となる流れの情報を得ることが重要となる。例えば物体周り流れの制御では、物体の表面に配置したセンサにより、壁から離れた位置の流れの情報を得ることができれば、制御性能の向上に繋がる可能性がある。Naka ら⁽¹⁾は壁面圧力変動が境界層外層の乱流構造と相関を有することを示している。このことは、壁面での圧力変動の測定結果から壁から離れた位置の乱流構造を推定することについての可能性を示唆している。

深層学習を流体力学の諸問題に適用する試みが多くある。Lee ら⁽²⁾は敵対的生成ネットワーク (GAN) を用いて円柱上の低レイノルズ数の渦放出流れを予測した。GAN を用いて予測された流れは、数値計算の解とよく一致することを示した。Jin ら⁽³⁾は複数の畳み込みニューラルネットワークを組み合わせることで、円柱後流の速度分布を予測した。Fukami ら⁽⁴⁾は、流入条件として用いる人工乱流場の生成にオートエンコーダを適用し、良好な結果を得ている。また、Ling ら⁽⁵⁾は深層学習を用いて、RANS モデルの予測性能を向上するために、レイノルズ応力非等方テンソルのモデルを学習し、予測性能が改善することを確認した。

本研究では、壁面圧力変動場から流れ場を推定する前段階として、複数のオートエンコーダを組み合わせることで、チャネル乱流の速度場データから、その速度場に対応する壁面圧力変動を推定する手法を提案し、その有効性を検証することを目的とする。

2. チャネル乱流の壁面圧力変動場の推定

2. 1. 数値計算

チャネル乱流の速度場、及び圧力変動場は、OpenFOAM v1906 の pisoFoam ソルバを用いた数値シミュレーションにより得た。格子点数は x, y, z 方向である流れ方向、壁の垂直方向、スパン方向に対して $256 \times 192 \times 256$ とし、計算領域は、 $L_x \times L_y \times L_z = 4\pi \times 2 \times 2\pi$ とした。時間刻み幅はクーラン数が 1 を超えないように設定した。摩擦レイノルズ数が約 180 となるよう、流れを駆動する一定の圧力勾配を与えた。

2. 2. データセット

Figure 1 にチャネル乱流の流れ場を示す。深層学習のネットワークに供する速度データは、計算領域を x, y, z 方向にそれぞれ 8,

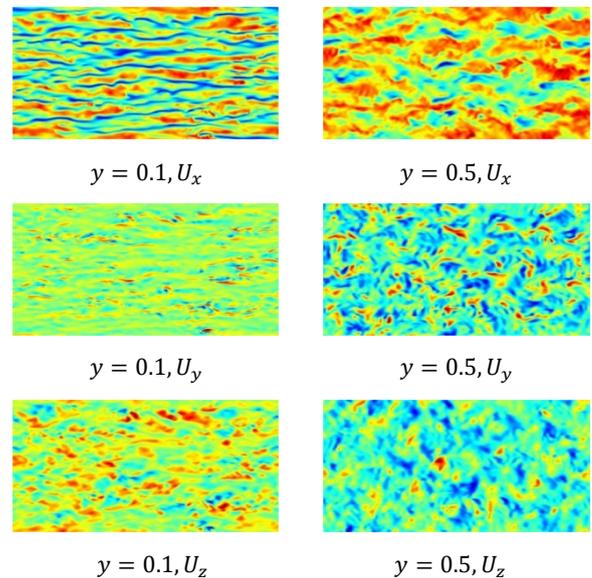


Fig. 1 Snapshot of x -velocity component of a turbulent channel flow. Color scale is arbitrary.

2, 8分割した、 $32 \times 96 \times 32$ の次元を持つデータを1単位とした。計算開始から流れが統計的に十分発達するまでの時間が経過した後の $t = 50$ から $t = 414$ までの 46720 個のスナップショットデータを抽出し、その内のランダムに抽出した 45000 個を訓練データ、残りの 1720 個を検証データとした。圧力変動場データは、速度場と同様に壁面領域を分割し、さらに上下面の対称性を用いることで、計 46720 個のデータを得た。訓練データ、検証データの個数や振り分けは、速度場データの場合と同様である。また、深層学習のネットワークに入力する際には、両データを数値範囲 0 から 1 となるように正規化した。時間 $t = 415$ から $t = 421$ までの 896 個のスナップショットデータを、テストデータとして抽出した。

2. 3. ネットワークの全体構成

Figure 2 に、チャネル乱流における壁面圧力変動を推定するネットワークを示す。ネットワーク構築には、ニューラルネットワー

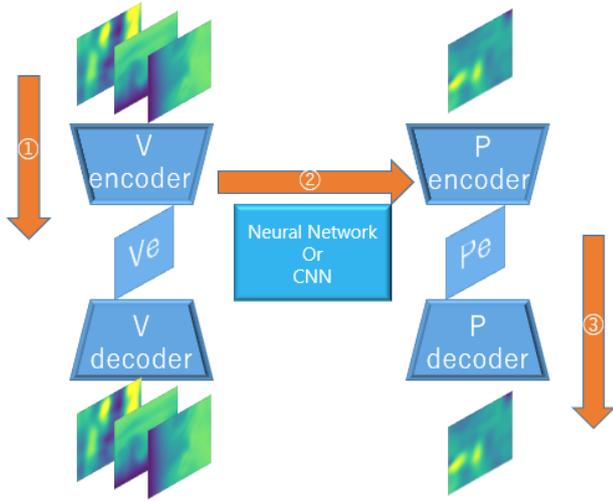


Fig. 2 Configuration of a network for predicting wall pressure fluctuation.

ライブラリである Keras を用いた。このライブラリは TensorFlow をバックエンドとしており、使用したバージョンは 2.1.5 である。ネットワークの学習の高速化のために、GeForce 社の GTX 1080 Ti を用いた。

本手法では、まず入力データである速度データから、エンコーダ (Fig. 2 の V encoder) によって次元圧縮された速度データ V_e を得る。その後、 V_e から、全結合もしくは畳み込みニューラルネットワーク (CNN) 層を介して、次元圧縮された壁面圧力変動データ P_e を推定する。最後に P decoder によって壁面圧力変動を推定する。

2. 4. 各ネットワークの詳細

2. 4. 1. 速度オートエンコーダ

Table 1 に速度データを圧縮するためのオートエンコーダの構成を示す。入力と出力次元は、速度成分に対応する $3 \times 32 \times 96 \times 32$ である。速度データは $3 \times 8 \times 24 \times 8$ へ圧縮され、その後、V decoder 部において元のデータサイズに復元される。このネットワークには、畳み込み層、マックスプーリング層、Batch Normalization 層⁹が含まれる。バッチサイズは 100 とし、活性化関数には ReLU 関数を用いた。損失関数には以下のように定義される平均 2 乗誤差関数 E を用いた。

$$E = \frac{1}{K} \sum_{k=1}^K (y_k - y_k^p)^2$$

K は学習に用いるデータ数、 y_k は正解ラベル、 y_k^p は出力ラベルである。最適化手法には Adam⁷ を採用し、ニューラルネットワークの重み w^t は以下のように更新される。

$$m_{t+1} = \beta_1 m_t + (1 - \beta_1) g_t,$$

$$v_{t+1} = \beta_2 v_t + (1 - \beta_2) g_t^2,$$

$$\hat{m} = \frac{m_{t+1}}{1 - \beta_1^t},$$

$$\hat{v} = \frac{v_{t+1}}{1 - \beta_2^t},$$

$$w^{t+1} = w^t - \alpha \frac{\hat{m}}{\sqrt{\hat{v} + \epsilon}},$$

ここで、 α は学習係数、 g_t は重みの勾配、 m_t と v_t はそれぞれ 1 次及び 2 次モーメントベクトルである。本研究では $\alpha = 1.00 \times 10^{-3}$ とした。 β_1 と β_2 は指数関数的減衰率である。これらのネットワーク構成は、画像分類にて一般的に用いられるものを採用し

Table 1 Autoencoder for velocity data.

Layer	Data size	Activation
Input	(3, 32, 96, 32)	-
Conv3D_1	(3, 32, 96, 32)	-
Batch Normalization	(3, 32, 96, 32)	ReLU
MaxPooling3D_1	(3, 16, 48, 16)	-
Conv3D_2	(3, 16, 48, 16)	-
Batch Normalization	(3, 16, 48, 16)	ReLU
MaxPooling3D_2	(3, 8, 24, 8)	-
Conv3D_3	(3, 8, 24, 8)	-
Batch Normalization	(3, 8, 24, 8)	ReLU
Upsampling3D_1	(3, 16, 48, 16)	-
Conv3D_4	(3, 16, 48, 16)	-
Batch Normalization	(3, 16, 48, 16)	ReLU
Upsampling3D_2	(3, 32, 96, 32)	-
Conv3D_5	(3, 32, 96, 32)	-
Output	(3, 32, 96, 32)	-

た。学習の過程では、訓練データを入力した際に望みの出力が得られるよう、損失関数を最小化するようにネットワーク内各要素の重みが調整される。適切に調整されたネットワークでは、訓練データ以外の同種のデータに対しても訓練データと同様の出力が得られる。一方で、訓練データ以外のデータで、損失関数が大きくなる場合には、過学習が起きていると考えられ、訓練データを増やす等の対策が必要となる。

学習 epoch 数の制御には early stopping⁸法を用い、損失関数の値が最低値を更新した後、最低 5 回学習するように設定した。5 回の学習中に損失関数の最低値が更新されない場合、学習が終了する。チャンネル乱流 DNS より得られた速度場データを入力、及び出力データとしてオートエンコーダの学習を行った結果、最終的な損失関数の値は、訓練データで 3.76×10^{-3} 、検証データで 3.83×10^{-3} となり、総 epoch 数は 51 であった。両者の値に差がないことから、過学習は抑制されていると判断した。テストデータに対する損失関数の値は 3.88×10^{-3} となった。このネットワークを用いて、次元圧縮された速度場データ V_e を得た。ネットワークの出力結果を Fig. 3 に示す。テストデータの損失関数の値と Fig. 3 のネットワークの出力結果より、速度場のオートエンコーダが速度場を概ね忠実に生成していることが分かる。

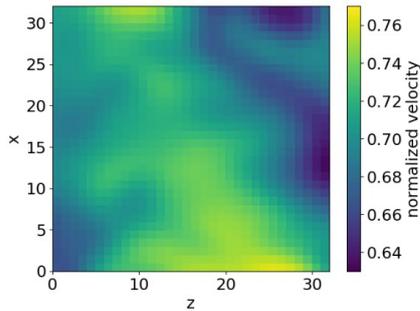
2. 4. 2. 壁面圧力変動オートエンコーダ

Table 2 に壁面圧力変動データを圧縮するためのオートエンコーダを示す。入力と出力次元は、壁面圧力変動場に対応する 32×32 であり、 $8 \times 8 \times 2$ の次元まで圧縮される。バッチサイズ、活性化関数、損失関数、最適化手法は 2.4.1 の場合と同様とした。このオートエンコーダにも early stopping 法を設定し、最低試行回数を 10 回とした。

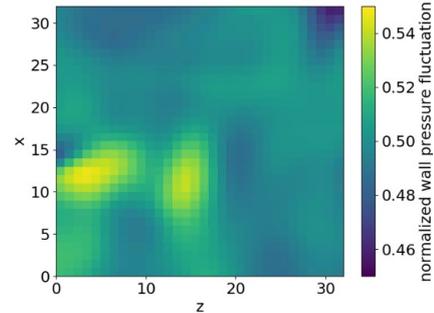
数値計算により得られた壁面圧力変動データを入力、及び出力データとして学習を行った結果、最終的な損失関数の値は、訓練データで 3.12×10^{-4} 、検証データで 3.65×10^{-4} 、テストデータで 6.26×10^{-5} となり、総 epoch 数は 74 であった。このネットワークを用いて、次元圧縮された壁面圧力変動データ P_e を得た。 P_e も V_e と同様、概ね正しく推定されたと判断した。また、入力データと P_e の分布を比べると、データ値の高低関係が反転しているものの、元のデータの分布における定性的な特徴は保持されていることがわかる。

2. 4. 3. V_e から P_e を推測するネットワーク

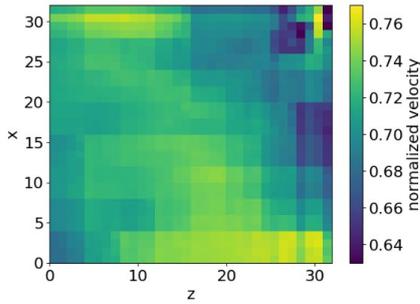
V_e から P_e を推定するネットワークのハイパーパラメータを最適



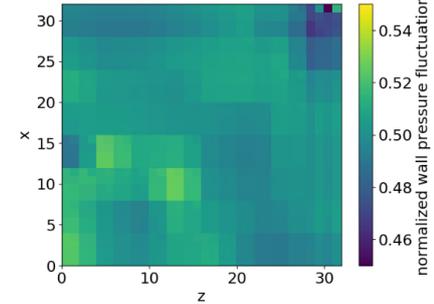
True velocity field



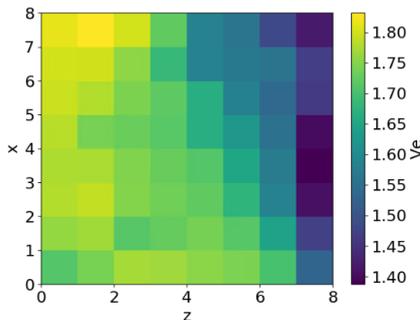
True wall pressure fluctuation field



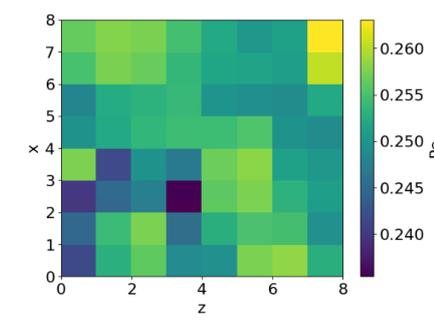
The output of V decoder



The output of P decoder



Result of V encoder(V_e)



Result of P encoder(P_e)

Fig. 3 Result of V auto-encoder.

Fig. 4 Result of P auto-encoder.

Table 2 Autoencoder for pressure fluctuation data.

Layer	Data size	Activation
Input	(32, 32, 1)	-
Conv2D_1	(32, 32, 2)	ReLU
MaxPooling2D_1	(16, 16, 2)	-
Conv2D_2	(16, 16, 2)	ReLU
MaxPooling2D_2	(8, 8, 2)	-
Conv2D_3	(8, 8, 2)	ReLU
Upsampling2D_1	(16, 16, 2)	-
Conv2D_4	(16, 16, 2)	ReLU
Upsampling2D_2	(32, 32, 2)	-
Conv2D_5	(32, 32, 1)	ReLU
Output	(32, 32, 1)	-

Table 3 Result of the learning in 2.4.3.

	The number of full connected layer	The number of convolutional layer	The value of loss function for training data	The value of loss function for validation data
Case 1	4	0	4.40×10^{-2}	4.42×10^{-2}
Case 2	3	0	4.43×10^{-2}	4.45×10^{-2}
Case 3	3	1	1.75×10^{-2}	1.75×10^{-2}
Case 4	2	2	1.13×10^{-2}	1.16×10^{-2}
Case 5	1	3	1.16×10^{-2}	1.16×10^{-2}

化する実験として、5 パターンのネットワークの検証を行った。入力データは 2.4.1 節に示したネットワークにより得られる V_e 、出力データは 2.4.2 節のネットワークから得られる P_e である。各ケースにおける層の種類と数、学習後の訓練データと検証データの損失関数の値を Table 3 に示す。5 つのケースの内、訓練データと検証データの損失関数の値は Case4 の場合に最小となった。このケースにおける、テストデータに関する損失関数の値は 1.00×10^{-2} であった。この推定では、訓練データ、検証データ、テストデータのすべてで、損失関数の値が速度と圧力のそれぞれの場合のオート

エンコーダと比較して 10 倍から 100 倍程度大きい値となった。推定された P_e の分布には、非物理的な不連続性が確認された。このネットワークを用いて、速度場から妥当な壁面圧力場を推定するためには、 V_e から妥当な P_e を推定する必要があると考えられる。より適切なネットワーク構成を選択することで、結果の改善が期待される。

4. 結論

本研究では、オートエンコーダを用いて、チャンネル乱流の速度場から壁面圧力変動場の推定を試みた。オートエンコーダの入力データと出力データは定性的に一致し、速度場と壁面圧力変動場の次元圧縮が良く行われたことが示された。

次元圧縮された速度場から次元圧縮された壁面圧力変動場の推定の際、損失関数の値は他のネットワークのものよりも高かった。

そのネットワークにおける全結合層と畳み込み層の最適な数を比較し、両者が2層の時に損失関数の値が最小となった。

参考文献

- (1) Naka, Y., Stanislas, M., Foucaut, J., Coudert, S., Laval, J. and Obi, S., "Space-time pressure-velocity correlations in a turbulent boundary layer," *Journal of Fluid Mechanics*, vol.771(2015), pp. 624-675
- (2) Lee, S. and You, D., "Prediction of laminar vortex shedding over a cylinder using deep learning," arXiv preprint (2017) arXiv:1712.07854.
- (3) Jin, X., Cheng, P., Chen, W. and Li, H., "Prediction model of velocity field around circular cylinder over various Reynolds numbers by fusion convolutional neural networks based on pressure on the cylinder," *Physics of Fluids*, 30 (2018), 047105.
- (4) Fukami, K., Nabae, Y., Kawai, K. and Fukagata, K., "Synthetic turbulent inflow generator using machine learning", arXiv:1806.08903.
- (5) Ling, J., Kurzawski, A. and Templeton, J., "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," *J. Fluid Mech.*, 807 (2016), pp. 155-166.
- (6) Ioffe, S. and Szegedy, C., "Batch Normalization: Accelerating deep network training by reducing internal covariate shift," arXiv preprint (2015), arXiv:1502.03167
- (7) Kingma, D.P. and Ba, J.L., "ADAM: A method for Stochastic Optimization," arXiv preprint (2014), arXiv:1412.6980.
- (8) Prechelt, L., "Automatic early stopping using cross validation: Quantifying the criteria," *Neural networks*, Volume11 (1998), pp. 761-767.