

2次元モデルのCFD結果の深層学習による解析の試み Deep learning-based analyses on CFD results of 2-dimensional models

- 伊藤雅仁, 立教大理, 豊島区西池袋 3-34-1, sigure3030@rikkyo.ac.jp:
齊藤天菜, 立教大理, 豊島区西池袋 3-34-1:
伊藤隆登, 立教大理, 豊島区西池袋 3-34-1:
奥脇弘次, 立教大理, 豊島区西池袋 3-34-1, okuwaki@rikkyo.ac.jp:
土井英男, 産総研 CD-Fmat, つくば市梅園 1-1-1 中央第二事業所, hideo-doi@aist.ac.jp
満野仁美, 立教大理, 豊島区西池袋 3-34-1:
○ 望月祐志, 立教大理, 豊島区西池袋 3-34-1, fullmoon@rikkyo.ac.jp
& 東大生産研, 目黒区駒場 4-6-1:
近藤修司, (株)ヴァイナス, 大阪市北区堂島 2-1-31:
小杉範仁, (株)ヴァイナス, 大阪市北区堂島 2-1-31, kosugi@vinas.com:
Masato Ito, Fac. Sci., Rikkyo Univ., 3-34-1 Nishi-ikebukuro, Toshima-ku, sigure3030@rikkyo.ac.jp:
Sona Saitou, Fac. Sci., Rikkyo Univ., 3-34-1 Nishi-ikebukuro, Toshima-ku:
Takato Ito, Fac. Sci., Rikkyo Univ., 3-34-1 Nishi-ikebukuro, Toshima-ku:
Koji Okuwaki, Fac. Sci., Rikkyo Univ., 3-34-1 Nishi-ikebukuro, Toshima-ku, okuwaki@rikkyo.ac.jp:
Hideo Doi, CD-Fmat, AIST, Central 2, 1-1-1 Umezono, Tsukuba City, hideo-doi@aist.go.jp:
Hitomi Mitsuno, Fac. Sci., Rikkyo Univ., 3-34-1 Nishi-ikebukuro, Toshima-ku:
○ Yuji Mochizuki, Fac. Sci., Rikkyo Univ., 3-34-1 Nishi-ikebukuro, Toshima-ku, fullmoon@rikkyo.ac.jp
& IIS, The Univ. Tokyo, 4-6-1 Komaba, Meguro-ku
Shuji Kondo, Vinas Co. Ltd., 2-1-31 Dojima, Kita-ku, Osaka City:
Norihito Kosugi, Vinas Co. Ltd., 2-1-31 Dojima, Kita-ku, Osaka City, kosugi@vinas.com:

CFD simulations were performed for several two-dimensional NACA wing models by using OpenFOAM and HELYX-OS. A series of flow fields were labeled by human with respect to the vortex situations, and these supervised results were cast into Google's TensorFlow as a representative deep learning toolkit. After the training, TensorFlow could automatically detect the generation of vortex. The transfer learning was performed with Keras wrapper as well. Applications of deep learning to post-processing in CFD simulations should be promising.

1. はじめに

シミュレーション・計算科学は、コンピュータの発達とプログラムの改良によって大規模計算を行えるようになってきました。しかし、出力される数値結果は適当な可視化ツールを用いたとしても、人が把握するにはあまりに膨大になり、妥当な解釈を迅速に得ることは必ずしも容易でない状況も生じています。こうしたことから、文責者の望月の研究室では、がんの画像診断や宇宙観測データの解析等でも応用が急速に進んでいる深層学習(DL)の技術⁽¹⁾を用い、計算化学分野におけるフラグメント分子軌道(FMO)法の計算結果を自動的に解釈する試みを行っています^(2,4)。文献⁽⁴⁾では、タンパク質のアミノ酸残基間の安定化/不安定化の相互作用エネルギーを2次元マップとして可視化(IFIE-map)した結果を、代表的な—それ故にネット上の公開情報も多い—DL ツールキットである Google の TensorFlow⁽⁵⁾に学習させ、タンパク質内の α ヘリックス、 β シートの構造的特徴を自動判定させることに成功しています。一方、FMO 計算に基づいて有効相互作用パラメータを決め、高分子やベシクル・膜などの粗視化シミュレーションを非経験的に行えるマルチスケール手法の開発と応用も進めており、ナノ〜メゾの橋をかけつつあります⁽⁶⁻¹⁰⁾。さらに、メゾよりも上のスケールを考える時、格子ボルツマン法などマクロの流体力学的な計算手法への接続も自然に視野に入ってきます。

上記を背景に、数値流体力学(CFD)計算と DL を連携させてみてはどうか、と考えるに至りました。ここで、もう一つの背景についても記しておく必要があります。それは、望月が東京大学生産技術研究所において展開された国産シミュレーションソフトの開

発プロジェクト群⁽¹¹⁾(東京大学生産技術研究所の小林敏雄先生, 加藤千幸先生がリーダー)に FMO 計算の立場で参画させていただき、成果報告会や報告書などを通じて長年に渡って CFD の重要性・有用性に接してきた経験です。2016 年の秋に加藤先生に本アイデアをご相談し、いけそうかな、と自分なりに判断しましたので、2017 年度の伊藤(隆)の卒論テーマに設定し、OpenFOAM⁽¹²⁾と HELYX-OS⁽¹³⁾をサービス展開されている(株)ヴァイナスとの共同研究の形で立ち上げることになりました。CFD 計算の対象は、対称性が比較的高い NACA 翼⁽¹⁴⁾を2次元化したモデルです。その理由は、(i) コストを下げて DL 用の多数の画像を揃えるため、(ii) (教師役)の人が見ての分かりやすさ、からでした。この試みは試金石として成功し、計算工学会での発表⁽¹⁵⁾と「機械設計」への寄稿⁽¹⁶⁾としてまとまりました。2018 年度は、伊藤(雅)が卒論テーマとして Keras⁽¹⁷⁾を使った転移学習⁽¹⁸⁾まで進めて展開しています。

これ以下本稿では、望月研究室で行っている CFD の DL 解析について時系列的にまとめさせていただきます。また、今後の課題と研究の方向性についても触れます。

2. 試行の第一段階：概念実証

先ず、2017 年度に伊藤(隆)がメインプレーヤーとなって行った概念実証(Proof of Concept)について記します。NACA 翼⁽¹⁴⁾は対称性のよい{0018, 0025, 1408}の3種、迎角は{14°, 15°, 15.5°, 16°, 17°}の5パターンとしました。PC 上で OpenFOAM⁽¹²⁾と HELYX-OS⁽¹³⁾を使い、2次元化された翼の CFD 計算を実行して一連の流速分布図を取得しました。計算の条件は、非定常・非圧縮

性・LES-Smagorinsky 乱流モデル, 空気の流速は20m/s, レイノルズ数は 2.5×10^6 で, 0.01 秒刻みで 2 秒間の流速分布図を生成しましたので, 1 ケースで 200 枚, 総計では 3000 枚となりました。

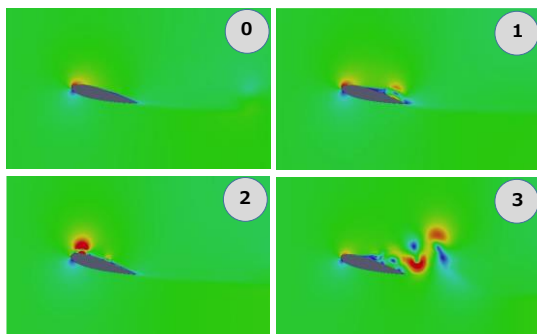


Fig. 1. Velocity map example and supervised labels (0: no vortex, 1: vortex at trailing edge, 2: vortex at leading edge, 3: after peeling).

得られた流速分布図の画像ファイルに対し, Fig.1 に示すように剥離(渦の発生)の状態に応じて四つのラベル(教師データに相当)を 3000 枚分付けるのは伊藤(隆)が(目視)行いました。TensorFlow⁽⁶⁾の畳み込みの隠れ層の数は先行事例⁽⁴⁾の経験を元に 3 とし, Intel i7(5820K)の CPU に加速用の GPU として Nvidia GeForce(GTX750Ti)を加えた PC 上で DL を実行しました。画像データセットの分割は, 翼型あたり 1000 枚を判定用 200 枚, 学習用 800 枚(訓練用 640 枚, テスト用 160 枚)の標準的な比率です。DL の活性化関数は ReLU, 画像の読み込みは 56×56 ピクセル, 過学習防止のドロップアウト率は 0.03 で段階可変としました。

TensorFlow⁽⁶⁾の DL 終了後に流速分布図の判定精度を視るのとは別に, 剥離の前段階を自動検出させることを試みました。状態としては, 「ラベル 0 とラベル 1 の遷移領域」となるので, 0 と 1 の確度が共に 30%以上の時を「剥離前段階」とし見做すことにしました。Fig. 2 に当該部分の Python スクリプトを示しますが, ラベルは” 0.1” として返ってきます。

```
for i in range(len(test_image)):
    pred_nd = logits.eval(feed_dict={
        images_placeholder: [test_image[i]],
        keep_prob: 1.0 })
    pred = pred_nd.tolist()
    if ((pred[0][0] > 0.3) and (pred[0][1] > 0.3) and (pred[0][0] > pred[0][1]):
        print 0.1
    elif ((pred[0][1] > 0.3) and (pred[0][3] > 0.3) and (pred[0][1] > pred[0][3]):
        print 1.3
    elif ((pred[0][0] > 0.3) and (pred[0][2] > 0.3) and (pred[0][0] > pred[0][2]):
        print 0.2
    elif ((pred[0][2] > 0.3) and (pred[0][3] > 0.3) and (pred[0][2] > pred[0][3]):
        print 2.3
    else:
        print np.argmax(pred_nd)
```

Fig. 2. Kernel of Python script to judge transition from label “0” to label “1”.

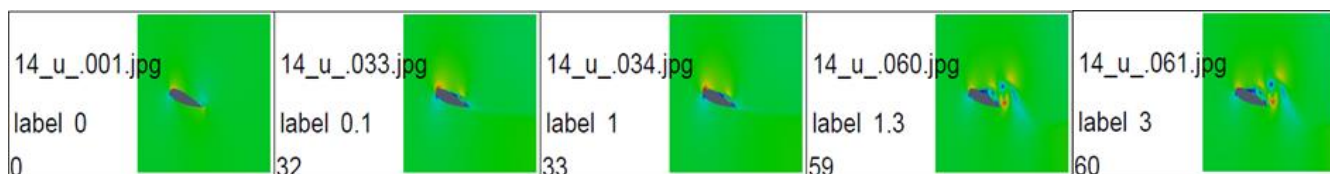


Fig. 3. Velocity map sequences with judged labels for NACA 0018 model in the case of attack angle 14 degree (pattern 2).

結果についてまとめます。DL は, (1)NACA0018 のみ,(2) 全翼型, (3)NACA0025 と 1408 の二つ, の三通りのパターンで行いました。いずれの場合でも, 最終的な判定精度は 95%に達しましたので, 今回は TensorFlow⁽⁶⁾の隠れ層の数が 3 でも十分であると判断されました(画像のローディングサイズにも依存します)。

剥離前の流速分布図のフレームを自動判定させる対象には, NACA 0018 の迎角 14° のケースを選びました。この場合, 36 フレーム目で剥離が発生しますが, TensorFlow⁽⁶⁾による判定の結果はパターン(1), (2), (3)の順に, 34, 33, 40 フレーム目となりました。Fig. 3 にパターン(2)の例を示しますが, jpg 画像のラベルの方が当該フレームとなっていることに注意して下さい(実行スクリプトの都合)。パターン(1)では, 他の迎角で学習が済んでいるので 34 と最も 36 に近くなりました。また, 三つの翼型で学習させた(2)でも「前段階」の 33 で検出されました。一方, 違う翼型で学習させた(3)では剥離後の 40 となりましたが, 誤差的には許容範囲であると思われます。

加藤先生のご助言もあり, TensorFlow⁽⁶⁾によって同定された剥離前のフレームの流速ベクトルの情報を格子点毎にチェックしてみたのですが, 翼上面付近で流れとは逆向きの成分を持つ点が確認されました。つまり, DL は渦の発生を検知していたこととなります。この事実は, ポスト処理として膨大な CFD 数値情報の中から「きちんと調べるべきところ」を可視化画像ベースで DL によって(人を介さずに)絞り込める可能性を示しています。

以上のように, 最初の一里塚としての概念実証は成功裏に終わることが出来ました。この成果を, 望月がマルチスケールシミュレーション関係⁽⁶⁻¹⁰⁾で参画している文科省ポスト「京」重点課題 6 の代表である東京大学工学部の吉村忍先生にもお伝えしたところ評価いただき, GPU として Nvidia GeForce(GTX1080Ti)を 2 枚積む DL 用サーバを望月研究室で利用出来るようになりました(2018 年 3 月)。このサーバは強力で, 畳み込み層 6 層での学習が 200 秒で完了するようになりました(従前の PC で i7 CPU のみの場合に比すと 18 倍の加速)。Keras⁽¹⁷⁾による転移学習⁽¹⁸⁾では事前学習で最適化済みの層数を積んでおくことが前提となりますので, 計算環境の向上は本質的なものでした。

3. 試行の第二段階：転移学習の準備

本研究での転移学習⁽¹⁸⁾の意図はシンプルで, 翼モデルの CFD 結果を転用して車モデルでの学習を効率化出来ないか, というものです。その下準備として, 2017 年度後期に 3 年次生の仮配属で望月研究室に来ていた伊藤(雅)が車種判定の転移学習を行いました⁽¹⁹⁾。内容の要約は以下のとおりです。

インターネットから画像をダウンロードする Python ベースのスクレーパー⁽²⁰⁾を使い、ラベルと車種銘柄{0:BMW, 1:BenZ, 2:Audi, 3: Lexus}の画像を合計で 2000 枚ダウンロードしました。次に、この画像データセットを 3:1 で訓練と検証に分け、Keras⁽¹⁷⁾でラップした TensorFlow⁽⁶⁾で学習させました(PC 上)。また、上記 4 車種とは別に{4:Peugeot}の画像も準備し、5 車種で最初から学習させた場合には判定の確度は 80%になりました。一方、先の 4 車種を事前学習させ、後から Peugeot を追加して転移学習を行った場合には正答率は 89%に向上し、総学習時間は 1/3 で済みました。転移学習の利用により、学習時間の短縮と判定確度の向上が得られたということです。

4. 試行の第三段階：非対称翼の追加と転移学習

3 種の対称的な NACA 翼モデル⁽¹⁴⁾での DL 解析の試みは、既述の学会発表⁽¹⁵⁾と雑誌での報告⁽¹⁶⁾とは別に、ポスト「京」重点課題 6-8 の連携会議で内容を紹介する機会がありました(加藤先生は課題 8 の代表)。その席では、神戸大学の坪倉誠先生からは車系モデルの難しさについて、JAXA の高木亮治先生からは NACA の非対称翼の挙動についてご助言をいただきました。そこで車に進む前に、2018 年度の卒論生として研究室に正式配属となっていた伊藤(雅)が転移学習⁽¹⁸⁾と合わせて非対称翼の扱いに取り組むことになりました(進行中)。Keras-TensorFlow の DL 環境で、サーバは GeForce の 2 枚刺しのものです。

追加した NACA 翼⁽¹⁴⁾は{6306, 6712}の 2 種で、CFD の計算条件は前と同じです。実際に流速図を描いてみると、新しいラベルとして前後同時剥離の“4”を設定しないといけないことが分かりました(Fig. 4 参照)。NACA 6306 と 6712 の翼型についても迎角を変えて各 1000 枚の流速図を取得し、6306 は訓練用に、一方で 6712 の方は予測判定にのみ用いました。DL の畳み込み層の数は 4(全層の数は 12)とし、ドロップアウトも適宜設定しました。なお、「遷移領域」の判定のロジックは、ラベル 4 の追加によって相当に複雑なものとなりました。

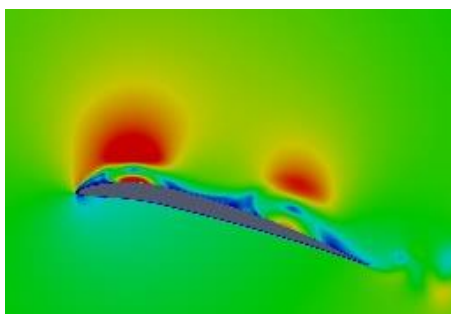


Fig. 4. Velocity map example of NACA 6306 (4: vortices at both leading and trailing edges).

ここでは、NACA0018 と 6306 を使った例を示します。Fig. 5 で上側は両翼を合わせて学習させた場合、下側は 0018 で事前学習させた後で 6306 を加えて転移学習⁽¹⁸⁾させた場合の学習曲線です。最終的な学習の精度は各々 98%と 95%でほぼ変わらず、転移学習が上手く機能していることが分かります。NACA 6712 の判定結果などは発表当日にお示しします。

5. まとめと今後の課題

本稿では、2 次元に簡約化した NACA 翼に対する CFD 計算の結果を DL によって自動解釈・判定させるポスト処理の試みをご紹介しました。今後の方向としては、2 次元の車モデルを組み合わせた転移学習を予定していますが、Fig. 6 のホンダ社の公開モ

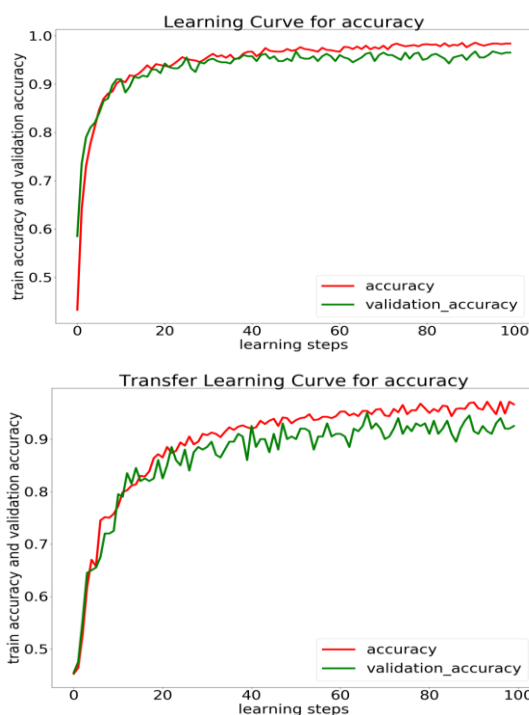


Fig. 5. Learning curves (up: NACA 0018 and 6306, down: NACA 6306 with NACA 0018 prior results).

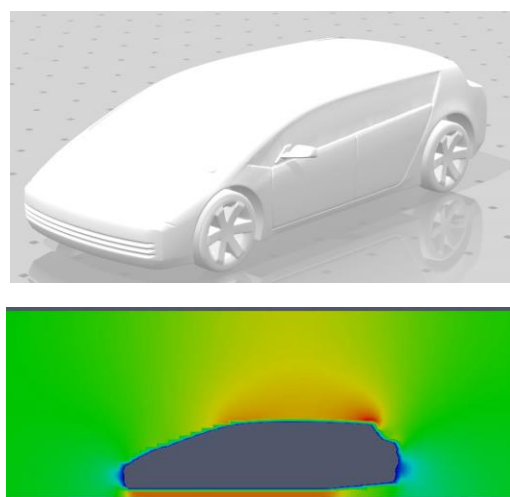


Fig. 6. Honda - Kiwami⁽²¹⁾ (up: model view, down: velocity map example of 2-dimensionalized model).

デル Kiwami⁽²¹⁾の例のように剥離パターンが翼の場合とは異なりますので、ラベルを人が付けるのは困難です。そのため、「何らかの数値指標」によってラベル付を自動化する必要があり、現在試行錯誤を続けているところです。自動ラベル付が確立出来れば、将来的には 3 次元モデルの扱いにも目処が付くと思われます。もちろん、3 次元のボリュームデータの畳み込み手法には別途検討が必要ではあります。

DL によるポスト処理の先進事例としては、米 DOE 系研究所と Nvidia が共同研究で最近発表した全世界の気象シミュレーションに関する論文⁽²²⁾があります。この研究は、V100 GPU を 27360 枚積む強力な DL ネットワークを構成し、サイクロンなどの「特異気象」に対応する「セグメンテーション」を自動検知させるもので、膨大な数値計算データを可視化と DL との組み合わせで処理する方向性が明確に示されています。一方で、DL によってコス

トのかかる実際の CFD 計算を回避させようとする事例⁽²³⁻²⁶⁾も出てきており、当該研究領域の今後の発展が期待されます。

6. 謝辞

現在、本研究は文科省ポスト「京」プロジェクト重点課題 6 の支援を得て進められています。本研究に対して励ましとご支援を頂いている東京大学生産技術研究所の加藤千幸先生、東京大学工学部の吉村忍先生、東京大学名誉教授の小林敏雄先生、(株)ヴァイナスの藤川泰彦社長に深謝します。技術面では、JAXA の高木亮治先生には非対称翼について、神戸大学大学院システム情報学研究所の坪倉誠先生には車体モデルについて、また九州大学情報基盤研究開発センターの小野謙二先生には高性能計算と DL の関係について、夫々ご教示・ご議論をいただきました。

参考文献

- (1) LeCun, Y., Bengio, Y., Hinton, G., “Deep learning, *Nature*”, 521 (2015), pp. 436-444.
- (2) 望月, 奥沢, “フラグメント分子軌道 (FMO) 計算の結果の自動解析の試み”, *計算工学*, 22 (2017), pp. 3539-3542.
- (3) 望月, 坂倉, 秋永, 他, “FMO プログラム ABINIT-MP の開発状況と機械学習との連携”, *J. Comp. Chem. Jpn.*, 16 (2017), pp. 119-122, 2017.
- (4) Saitou, S., Iijima, J., Fujimoto, M., et al., “Application of TensorFlow to recognition of visualized results of fragment molecular orbital (FMO) calculations”, *ChemBio-Inform. J.*, 18 (2018), pp. 58-69.
- (5) <<https://www.tensorflow.org/>>.
- (6) Okuwaki, K., Mochizuki, Y., Doi, Y., et al. “Fragment Molecular Orbital-based Parameterization Procedure for Mesoscopic Structure Prediction of Polymeric Materials”, *J. Phys. Chem. B*, 122 (2018) pp. 338-347.
- (7) Doi, H., Okuwaki, K., Mochizuki, Y., et al., “Dissipative particle dynamics (DPD) simulations with fragment molecular orbital (FMO) based effective parameters for 1-Palmitoyl-2-oleoyl phosphatidyl choline (POPC) membrane”, *Chem. Phys. Lett.*, 684 (2017) pp. 427-432.
- (8) Okuwaki, K., Mochizuki, Y., Doi, H., et al., “Theoretical Analyses on Water Cluster Structures in Polymer Electrolyte Membrane by Using Dissipative Particle Dynamics Simulations with Fragment Molecular Orbital Based Effective Parameters”, *RSC Adv.*, 8 (2018) pp. 34582-34595.
- (9) 土居, 奥脇, 望月, 他, “散逸粒子動力学におけるシリカー脂質膜界面付近の水の取扱い”, *J. Comp. Chem. Jpn.*, 16 (2017) pp. 28-31.
- (10) 奥脇, 土居, 望月, “フラグメント分子軌道(FMO)法を用いた散逸粒子動力学シミュレーションのための有効相互作用パラメータ算出の自動化フレームワーク”, *J. Comp. Chem. Jpn.*, 17 (2018) pp. 102-109.
- (11) <<http://www.ciss.iis.u-tokyo.ac.jp/project/>>.
- (12) <<https://www.openfoam.com/>>.
- (13) <<https://engys.com/products/helyx-os>>.
- (14) <http://www.cenav.org/wing_conv/>.
- (15) 伊藤, 齊藤, 土居, 他, “二次元翼周り流れの CFD 解析結果の TensorFlow による自動解釈の試み”, G-03-02, 計算工学講演会, 名古屋, (2018/6/7).
- (16) 小杉, 近藤, 秋永, 他, “AI を活用した流体解析シミュレーション技術の開発”, *機械設計*, 2018 年 7 月号, 62 (2018) pp. 42-46.
- (17) <<https://keras.io/ja/>>.
- (18) Weiss, K., Khoshgoftaar, T. M., Wang, D.D., “A survey of transfer learning”, *J. Big Data*, 3:9 (2016) pp. 1-40.
- (19) 伊藤, 伊藤, 齊藤, 他, “Keras を使った転移学習の応用事例”, IP04, 日本コンピュータ化学会 2018 年春季年会, 東京, (2018/6/8).
- (20) <<https://pypi.org/project/beautifulsoup4/>>.
- (21) <http://www.honda-3d.com/no_flash.html>.
- (22) Kurth, T., Treichler, S., Romero, J., et al., “Exascale Deep Learning for Climate Analytics”, arXiv: 1810.01993, (2018).
- (23) “Data Driven Smart Proxy for CFD: Application of Big Data Analytics & Machine Learning in Computational Fluid Dynamics, Part One: Proof of Concept”, NETL Technical Report Series, NETL-PUB-21574, <<https://www.osti.gov/biblio/1417305>>.
- (24) Ladicky, L., Jeong, S., Solenthaler, B., et al., “Data-driven Fluid Simulations using Regression Forests”, <https://www.inf.ethz.ch/personal/ladicky/fluid_sigasia15.pdf>.
- (25) Guo, X., Li, W., Lorio, F., “Convolutional Neural Networks for Steady Flow Approximation”, <<https://autodeskresearch.com/sites/default/files/ADSK-KDD2016.pdf>>.
- (26) Hennigh, O., “Lat-Net: Compressing Lattice Boltzmann Flow Simulations using Deep Neural Networks”, arXiv: 1705.09036, (2017).