A MACHINE LEARNING FRAMEWORK FOR PREDICTION OF DISCRETISATION ERRORS ON COARSE CFD GRIDS

M. Sardar[1], A. Skillen[1], S. Draycott[1], M. J. Zimoń[2], A. Revell[1]

[1]Department of Mechanical, Aerospace and Civil Engineering, University of Manchester, Manchester M13 9PL, United Kingdom
[2]IBM Research Europe, Daresbury Laboratory, WA4 4AD, United Kingdom

1 INTRODUCTION

Heat transfer processes can be complex. In simple cases, prediction of thermal phenomena is often reliant on empirical models. For more complex scenarios, computationally intensive numerical modelling and simulation techniques are required, such as Computational Fluid Dynamics (CFD). While the computational requirements of CFD may be reduced at the expense of lower-fidelity Coarse Grid (CG) results, the discretisation error will, in general, be non-zero. This study will investigate novel data-driven methods which leverage the computational efficiency of CG solutions without adversely affecting accuracy. We will evaluate selected Neural Network (NN) architectures in their capacity to correct low-fidelity CG data by learning from high-fidelity grid-converged data, incorporating corrective terms for discretisation error. Selected neural networks will be employed to predict the error in a given variable for a coarse grid simulation.

2 RECENT DEVELOPMENTS

Hanna et al. [1] proposed a simple neural network architecture with dimensionless flow variables in the feature vector to learn the error in a specified quantity of interest. The poor extrapolative capabilities of their model highlighted a key limitation of hybrid CFD-NN approaches – Neural Networks are highly sensitive to hyperparameters. We anticipate a systematic approach to network design will provide improved predictive capacity. Margenberg et al. [2] successfully incorporated the learned model to a layer of the solver in order to tackle transient simulations. Bao et al. [3] demonstrated the capabilities of their CFD-NN corrective method. For their particular case study, they showed a strong agreement between the model predicted values and high-fidelity ground truth. Similarly, Kochkov et al. [4] showcased a corrective approach with strong agreement with high resolution DNS data. Their chosen Convolutional Neural Network (CNN) architectures will be studied in more detail in our ongoing work in this area.

3 METHODOLOGY

A simple test case has been identified: incompressible 2D transport of a passive scalar (Equation 1). A neural network was employed to predict the discretisation error in a given variable for a CG simulation of this equation. In Equation 1, T is a given scalar, $\mathbf{u}$ is the velocity vector, and α is the diffusion constant. Source terms were set to zero.

$$\frac{\partial T}{\partial t} + \nabla \cdot (T\mathbf{u}) - \nabla \cdot (\alpha \nabla T) = 0 \tag{1}$$

OpenFOAM v8's `scalarTransportFoam` solver was used for the simulations. The 2D domain size was 1m in width and height. Mesh sizes were as follows:

- **Coarse**: 20 cells $\times$ 20 cells
- **Fine**: 100 cells $\times$ 100 cells

The velocity magnitude was specified as constant across the domain, and input angle of velocity was

*Corresponding author: mohammed.sardar@postgrad.manchester.ac.uk

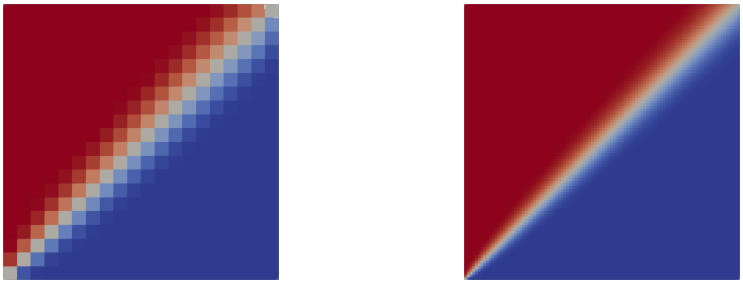

Figure 1: Coarse mesh (left) and Fine mesh (right) simulations, with $\alpha = 0.001$.

constant at $\theta = 45^{\circ}$. A preliminary set of experiments was used to determine the interpolative capability of the trained models. This was achieved by evaluating the model performance at intermediate values of α between those used for training. In our future work, we will also determine the extrapolative capability by extending the range of α. The Python library, PyTorch[1], was used to build the machine learning models. A library was developed to automate OpenFOAM simulations and convert the output data to PyTorch tensors, with code available at: https://github.com/HamzaSardar/TorchFoam. For element-wise comparison between the meshes, high fidelity data was down-sampled where cell-centres overlap with the CG solution.

The leading terms in the Taylor expansion of scalar T, and the local Péclet number at each cell, were considered for input to the feature vector: $\frac{\partial T}{\partial x_i}$, $\frac{\partial^2 T}{\partial x_i^2}$, Pe_{cell}. These features were selected to provide information on the physical diffusivity to the network, and the derivatives are linked to the numerical calculation of the passive scalar via a Taylor series expansion. Thus, we anticipated a relationship between the derivatives and the discretisation error in the solution. The hyperparameters to be explored in this work include the size and number of hidden layers, activation functions, optimisation algorithms, the effects of regularisation, and the learning rate. The initial setup was a fully connected neural network using Tanh activation and a single hidden layer with 10 neurons, going to a single output—the error in T. The starting optimiser was Adam, from Kingma and Ba [5]. The initial learning rate was set to 0.01 which would be adapted by the optimisation algorithm as the training progressed. A shallow initial network was selected to gain a strong fundamental understanding of the learning mechanisms in NNs before expanding the current work by exploring the potential for deeper networks.

4 RESULTS

Our results are presented in Figure 2. It is worth emphasizing that these results are a preliminary effort, representing a work in progress. As we advance this project, we expect improvement in the already promising results presented herein, with models capable of capturing increasing levels of non-linearity. The scenarios considered are summarised below.

- Training: $\alpha \in 0.001, 0.005, 0.01, 0.05$
- Interpolative Evaluation: $\alpha \in 0.0025, 0.0075, 0.025$

Model evaluation on the training flows by comparing the predicted error to the ground truth gives a correlation coefficient of 0.982, indicating a strong correlation between the actual and predicted error. Evaluation on the evaluative flows gives a correlation coefficient of 0.929 – a good indication the model is performing well on scenarios the NN has not previously seen. During testing, the network was consistently trained to a training and validation loss on the order of 10^{-5}, suggesting that our network had not overfitted to the training data. The network performs marginally worse on the evaluation set, but the correlation coefficient is sufficiently high to imply a strong performance on the evaluative dataset, with

[1] https://pytorch.org/

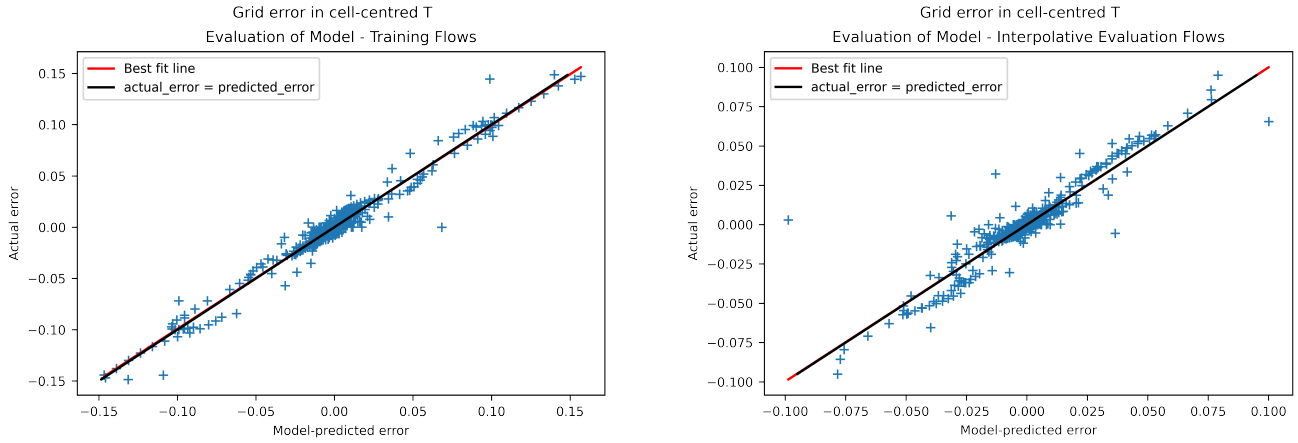

Figure 2: Model evaluation on training flows (left), model evaluation on evaluative flows (right).

the network predicting the actual error with a high degree of accuracy. The overall performance is an indicator that the trained network has appropriately addressed the 'bias-variance trade-off' – a modest error on the training set is accepted in exchange for improved ability to generalise to unseen data. It should be noted that we do not seek a perfect correlation on the training set; this would imply overfitting to the training set, and thus poorer generalisation.

5 CONCLUSIONS AND PROPOSED WORK

We have successfully demonstrated that a simple neural network can learn to predict the discretisation error in a given variable from characteristic dimensionless groups and variable derivatives. Our trained model performs well on the benchmark case of incompressible 2D transport of a scalar, attaining a strong correlation coefficient against the ground truth of 0.929 in the evaluative scenarios. We can deduce that our network has generalised well, learning a meaningful relationship between inputs and outputs. Further work will extend this methodology to explore:

- The use of this network in a CFD code in inference mode.
- Physics-constrained loss functions and regularisation to embed governing physics in the error prediction.
- Alternative network architectures, including convolutional layers to enhance translational invariance and exploit spatially coherent flow features.
- Application of developed models to more complex multi-physics scenarios, involving turbulent flows and heat transfer, as well as network sensitivity to mesh quality and resolution. We will investigate extending this method to Nuclear Thermal Hydraulics (NTH) applications.

REFERENCES

[1] B. N. Hanna, N. T. Dinh, R. W. Youngblood, & I. A. Bolotnov. Machine-learning based error prediction approach for coarse-grid Computational Fluid Dynamics (CG-CFD). *Progress in Nuclear Energy*, **118** (2020) 103140.

[2] N. Margenberg, D. Hartmann, C. Lessig, & T. Richter. A neural network multigrid solver for the Navier-Stokes equations. *Journal of Computational Physics*, p. 36.

[3] H. Bao, N. Dinh, L. Lin, R. Youngblood, J. Lane, & H. Zhang. Using deep learning to explore local physical similarity for global-scale bridging in thermal-hydraulic simulation. *Annals of Nuclear Energy*, **147** (2020) 107684.

[4] D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, & S. Hoyer. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences of the United States of America*, **118** (2021) 1–13.

[5] D. P. Kingma & J. L. Ba. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15.