27/28th International Conference
ENGINEERING MECHANICS 2022
Milovy, Czech Republic, May 9 –12, 2022
Paper **#98**, pp. 41–44, doi: 10.21495/51-2-41

FLOW FIELD PREDICTION IN A BLADE CASCADE USING A CONVOLUTION NEURAL NETWORK

Bublík O.*, **Heidler V.****, **Pecka A.*****, **Vimmr J.**†

Abstract: *This paper deals with fast flow field prediction in a blade cascade using a machine-learning architecture called a convolutional neural network. Specifically, our work focuses on the U-Net architecture. With the following study, our goal is to parameterize the neural network architecture, in dependence on Reynolds and Mach numbers. The hyper network in our model is a simple fully-connected feedforward neural network generating weights, which fit the U-net architecture with the specific Mach and Reynolds number. The concept of the hyper network-based parametrization is tested on the problem of a compressible fluid flow through the blade cascade.*

Keywords: U-net, Hypernetwork, Convolution neural network, Blade cascade, Compressible fluid flow.

1. Introduction

Convolution neural network (CNN) is a class of neural networks developed mainly for image processing. The common use of CNN lies in objects classification or image segmentation. The pioneering use of the CNN for the steady fluid flow simulation was published by Guo (2016), where the input image, which contains the boundary information, was transformed to the velocity field. The use of CNN for the unsteady fluid flow simulation was firstly described by Hennig (2017). In both papers, the U-net architecture of CNN was used, see Fig. 3.

This work aims to use U-net based neural network model for the flow field prediction through blade cascade and establishes the lift and drag forces acting on the blade. The high performance and differentiability of a neural network yield a very promising alternative to classical CFD methods for tough problems such as flow control or shape optimization. This means in tasks where a pressure field calculation is required for many geometry variations.

To provide a parameter-dependent neural network model predicting the flow field for different Reynolds and Mach numbers, the concept of a hyper network is employed. The idea of the hyper network is, that when we modify the Reynolds or the Mach number, the hyper network modifies the main network in such a way that it produces flow fields that correspond to that Reynolds and Mach number. In our model, the hyper network is a simple fully-connected feedforward neural network with one hidden layer, see Fig. 4. Hypernetworks can therefore be thought of as weight generators.

The neural network architecture was implemented using Python programming language with the help of Keras (Chollet, 2015) and TensorFlow (Tensorflow, 2015) libraries.

* Ing. Ondřej Bublík, Ph.D.: Faculty of Applied Sciences, University of West Bohemia, Technická 8, 30100 Pilsen; CZ, obublik@kme.zcu.cz

** Ing Václav Heidler, PhD.: Faculty of Applied Sciences, University of West Bohemia, Technická 8, 30100 Pilsen; CZ, vheidler@kme.zcu.cz

*** Mgr. Aleš Pecka: Faculty of Applied Sciences, University of West Bohemia, Technická 8, 30100 Pilsen; CZ, pecka@kme.zcu.cz

† Assoc. Prof. Jan Vimmr, PhD.: Faculty of Applied Sciences, University of West Bohemia, Technická 8, 30100 Pilsen; CZ, jvimmr@kme.zcu.cz

2. Geometry of blade cascade and training set

The geometry of the blade cascade consists of a blade profile whose body is formed by a bezier curve with six control points, see Fig 1. The computational domain covered by a computational mesh is shown in Fig 2. The inlet and outlet boundary is considered on the left and right vertical sides of a domain. On the top and the bottom sides of the domain, the periodic boundary conditions are considered. The training set of 50 random profiles was generated. Each of the profile was trained for all combinations of chosen Mach numbers, Ma = 0.2, 0.5, 0.9 and Reynolds numbers, Re = 100, 1000, 10000. The numerical computation of the specimens was performed with open-source CFD software FlowPro (Bublik, 2000).

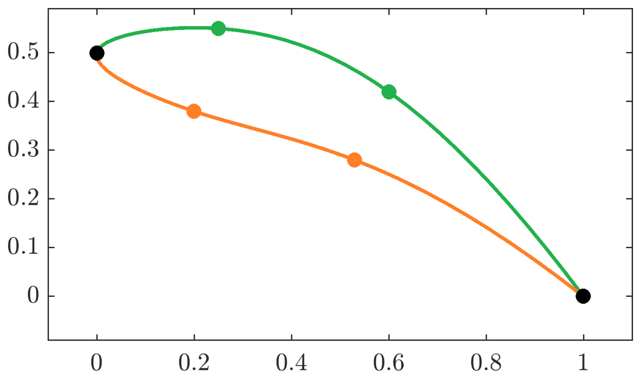

Fig. 1: Blade profile.

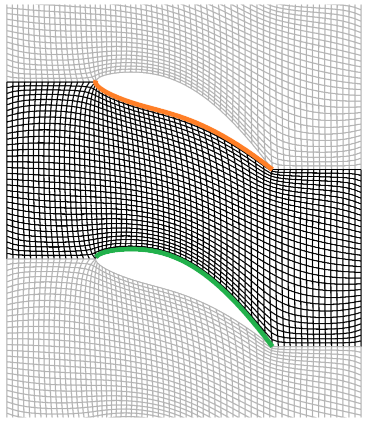

Fig. 2: Computational mesh.

The evaluation of the lift and drag forces acting on the blade is performed using the relations

$$L = \oint_{\Gamma} p n_y \,, \qquad D = \oint_{\Gamma} p n_x \,, \tag{1}$$

where L and D are lift and drag forces, p is a pressure field, $\mathbf{n} = [n_x, n_y]^T$ is a unit normal vector and Γ is a profile surface.

3. Neural network architectures

For the prediction of the flow field, the U-net architecture of CNN was employed. The x and y coordinates of the structured mesh with 64 x 32 points is considered as U-net input, see Figs. 2 and 3 (black colored mesh in the middle). At the output of the U-net, the flow field included density, pressure and velocity is generated. The U-net based model was trained for 50 blade profiles and for each of the Reynolds and Mach numbers combinations. The resulting weights were then used for training the hypernetwork.

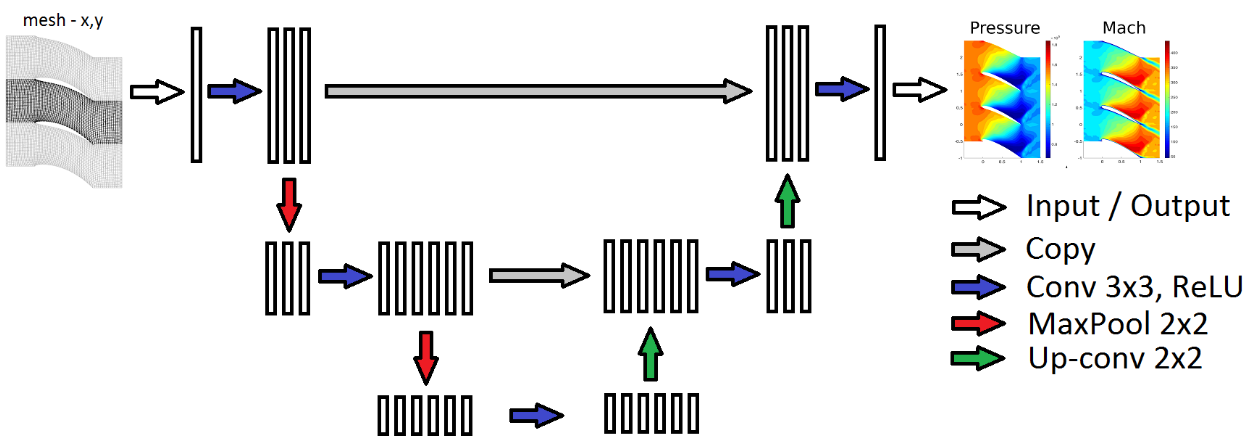

Fig. 3: U-net architecture of convolution neural network predicting flow field.

The hypernetwork consists of a dense neural network, see Fig. 4. At the input, the Reynolds and Mach numbers are supplied. The weights of the U-net are then generated at the output of the hypernetwork.

Tab. 1: Average error across all Mach and Reynolds numbers.

test blade	drag error [%]	lift error [%]
1	5,99	5,87
2	1,31	0,75
3	4,62	5,02
4	2,45	3,76
5	5,05	3,73

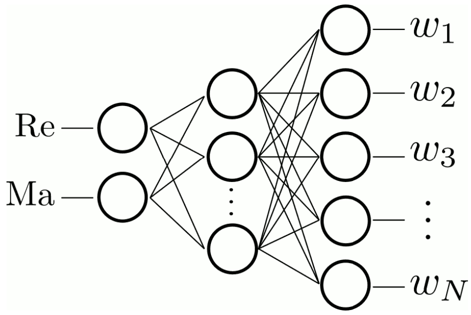

Fig. 4: Hypernetwork architecture. The input Reynolds and Mach numbers are transformed onto the U-net weights.

4. Numerical results

The concept of the flow field parametrization has been evaluated on the set of 5 testing blade profiles, which were not included in the training set.

Firstly, the weights of the U-net CNN were generated for the chosen Reynolds and Mach numbers using the hypernetwork. The U-net based model was used afterwards for the flow field predictions.

Figure 5 shows a comparison of predicted flow field for Mach number 0.9 and Reynolds number 100 for the first test profile. Further, Fig. 6 illustrates the computed and predicted flow field for Mach number 0.5 and Reynolds number 1000 around the second test profile. The Tab. 1 contains the average relative error for each of the test profiles.

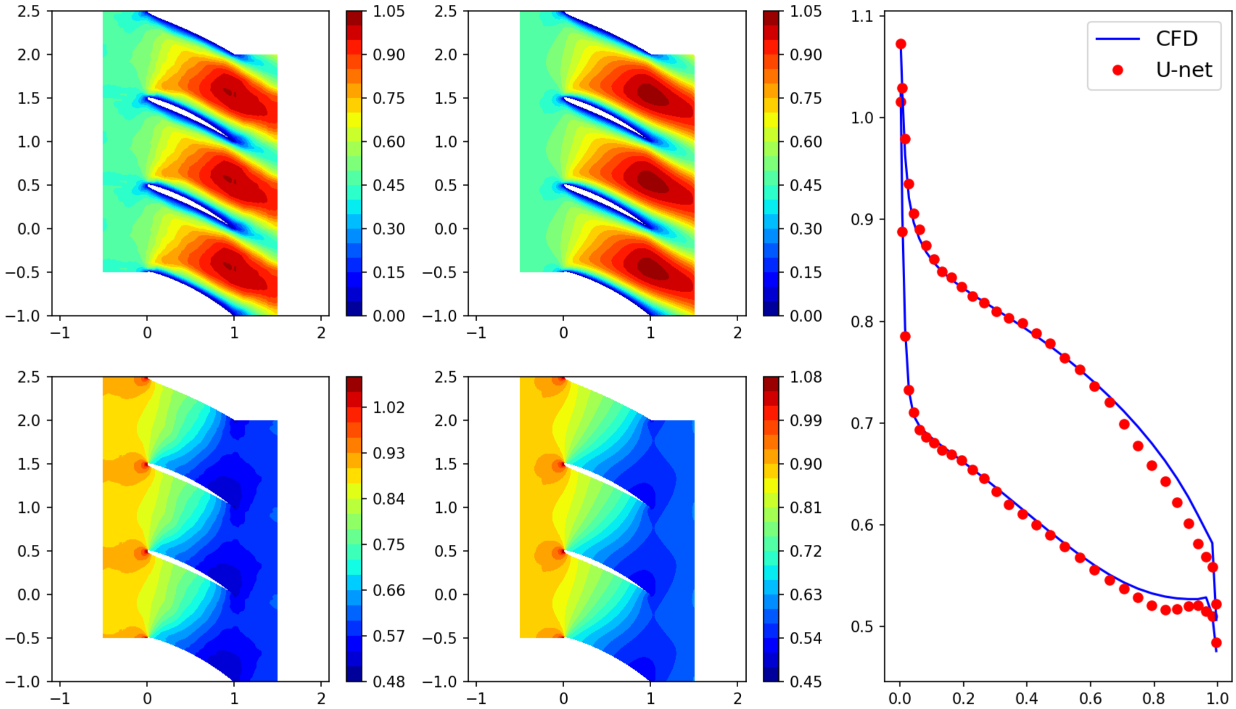

Fig. 5: Comparison of predicted (left) and computed (middle) flow field for Mach number 0.9 and Reynolds number 100, for the test profile 1. Pressure plot along the profile (right).

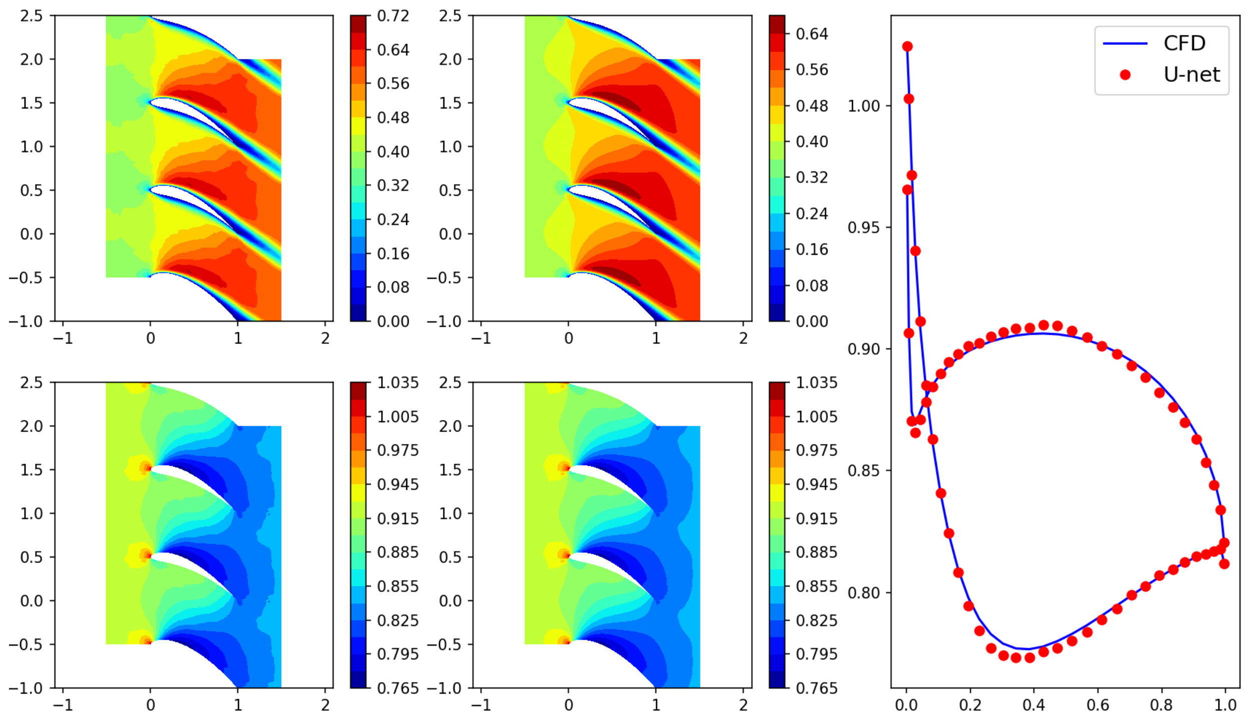

Fig. 6: Comparison of predicted (left) and computed (middle) flow field for Mach number 0.5 and Reynolds number 1000, for the test profile 2. Pressure plot along the profile (right).

5. Conclusions

Within this study, the concept of flow field parametrization through the hypernetwork was introduced. The hypernetwork was trained for all of the combinations of three Reynold numbers (100, 1000, 10000) and three Mach numbers (0.2, 0.5, 0.9). The U-net based model with weights generated by the hypernetwork was evaluated for five test profiles. The Tab. 1 contains average errors between computed and predicted values. All of the average errors are less than six percent. The results indicate that the concept of the flow field parametrization by the hypernetwork could be a very promising alternative to classical CFD methods for tough problems such as flow control or shape optimization. The future studies will be focused on a fully parametrized flow field.

Acknowledgements

This research is supported by project ”Fast flow-field prediction using deep neural networks for solving fluid-structure interaction problems” GA21-31457S of the Grant Agency of the Czech Republic.

References

Abadi, M., et al. (2015) *TensorFlow: Large-scale machine learning on heterogeneous systems.* Software available from tensorflow.org.

Bublík, O., Pecka, A. and Vimmr, J. (2017), FlowPro — multipurpose CFD software written in Java. In Adámek, V., Jonášová, A., Plánička, S and Zajíček, M., eds, *Computational Mechanics*, University of West Bohemia, pp. 13–14.

Chollet, F. and others (2015) *Keras*, https://keras.io

Guo, X., Li, W., Iorio, F. (2016), Convolutional neural networks for steady flow approximation, In Krishnapuram, B., et al., eds., *Proc. the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, New York, NY, pp. 481–490.

Hennigh, O., (2017), *Lat-Net: Compressing Lattice Boltzmann Flow Simulations using Deep Neural Networks*, arXiv:1705.09036v1.