



Fast Aerodynamics Prediction of Wedge Tail Airfoils Using Multi-head Perceptron Network

Md. Moynul Hasan¹ · Md. Mashiur Rahaman¹ · N. M. Golam Zakaria¹

Received: 16 July 2023 / Accepted: 26 December 2023 / Published online: 31 January 2024
© King Fahd University of Petroleum & Minerals 2024

Abstract

Wedge tail airfoils refer to the addition of a wedge section to the tail of an airfoil. This modification has demonstrated greater efficiency in increasing the lift forces compared to the base airfoil design. Such airfoils are particularly important in scenarios where increased lift forces are required, such as in the fast and efficient maneuvering of marine and aerospace vehicles. This study aimed to predict the flow fields and aerodynamic coefficients of wedge tail airfoils using a multi-head perceptron (MHP) network and classical machine learning (ML) algorithms, including k-nearest neighbors, decision tree, and random forest. These predictions were based on airfoil sections, x - y grid coordinates, Mach number, and angle of attack, eliminating the need to solve the Navier–Stokes (NS) equations. The database required for training the MHP network and the ML models were generated using the Ansys solver, which solved the NS equations. The results of the models were compared, and it was observed that the MHP network consistently outperformed the others in terms of prediction accuracy. Additionally, the prediction process demonstrated a significant speed improvement of 125 times compared to conventional computational fluid dynamics techniques. Once the training was completed, the flow fields and aerodynamic coefficients for a wedge tail airfoil could be obtained within seconds.

Keywords Wedge tail airfoil · Aerodynamics · Multi-head perceptron · Machine learning

1 Introduction

Airfoils play a critical role in various engineering applications [1, 2], particularly in the aerospace and marine fields. In aerospace, airfoils such as ailerons, elevators, and flaps are essential control surfaces on aircraft wings, enabling the adjustment of lift and drag forces and facilitating directional movement [3–5]. Similarly, in marine engineering, airfoils serve as control surfaces on rudders and fins, allowing for precise maneuvering of ships and submarines, with the ship rudder being a prominent example [6–9]. The growing implementation of the Energy Efficiency Design Index (EEDI) to

reduce carbon dioxide emissions and promote environmental sustainability has emphasized the importance of maneuverability at low speeds in the marine industry. In this context, marine rudders play a crucial role [10]. Developing high-lift rudders, such as those incorporating wedge tail airfoil profiles, has proven beneficial for maneuvering in shallow waters and at low speeds [11]. These rudders generate high-lift coefficients but also introduce additional drag [12]. Previous studies have shown that incorporating wedge tails into NACA airfoil sections can lead to improved lift coefficients [10, 13].

The analysis of flow fields and aerodynamic coefficients of wedge tail airfoils holds significant importance due to its practical implications. Traditionally, such analyses have relied on wind tunnel tests and computational fluid dynamics (CFD) techniques. Wind tunnel tests, which require prior experience, are known for their high costs and time-consuming nature, typically conducted in the final stages of airfoil design and analysis. On the other hand, advancements in hardware have led to faster processing times in CFD, enabling computational experiments for airfoil design and analysis. However, in cases where extensive iterations of flow

✉ Md. Moynul Hasan
moynul0210@gmail.com

Md. Mashiur Rahaman
mashiurrahaman@name.buet.ac.bd

N. M. Golam Zakaria
gzakaria@name.buet.ac.bd

¹ Department of Naval Architecture and Marine Engineering,
Bangladesh University of Engineering and Technology,
Dhaka 1000, Bangladesh



solutions are necessary, using Navier–Stokes (NS) solvers can significantly extend the overall design time [14, 15]. Furthermore, expertise in the field is required to employ CFD techniques effectively. Nevertheless, data-driven approaches such as machine learning and deep learning have gained popularity due to their ability to learn from complex data and provide accurate predictions [16]. These approaches offer advantages over traditional methods by reducing the reliance on wind tunnel tests and mitigating the computational burdens associated with CFD techniques. Previous studies, as shown in Table 1, have effectively utilized data-driven approaches to predict flow fields and aerodynamic coefficients of airfoils.

However, the focus was primarily on standard airfoils, neglecting investigation into specialized designs like wedge tail airfoils. This study aimed to expedite the analysis of wedge tail airfoils by predicting their flow fields and aerodynamic coefficients through machine learning. Notably, prior studies heavily relied on various neural network architectures. However, it is essential to analyze the effectiveness of classical machine learning algorithms in these cases, as they are easier to implement than neural networks. Therefore, this study utilizes classical machine learning algorithms, namely k-nearest neighbors (KNN), decision tree (DT), and random forest (RF), to predict flow fields, namely pressure (P), velocity magnitude (U), velocity component in the x-direction (U_x), and y-direction (U_y), as well as aerodynamic coefficients, specifically lift coefficient (C_L) and drag coefficient (C_D). These predictions are then compared to those made by the multi-head perceptron (MHP) network. The algorithms receive inputs including airfoil sections (A), x – y grid coordinates (X , Y), and flow conditions such as Mach number (M) and angle of attack (α). This rich input enables the models to estimate the flow fields and aerodynamic coefficients accurately. The training process focuses on making point-by-point predictions, with particular attention given to achieving high accuracy in the boundary regions of the airfoils. To evaluate the performance and accuracy of the trained models, they are tested on new airfoil cases that were not included in the training phase.

2 Airfoil Sections

The airfoil sections used in this study were based on NACA 0012, NACA 0015, NACA 0018, and NACA 0021 airfoil profiles. In order to create the airfoil sections, the shape of the airfoil profiles was modified by incorporating a concave shape at the $0.95c$ position, with a tail wedge thickness of $0.1c$, where c represents the chord length of the airfoil. An illustration of the modified airfoil sections is showed in Fig. 1.

3 Dataset Generation

The dataset used in this study was generated through 2D RANS-CFD simulations using ANSYS for wedge tail airfoils. The simulations covered a range of Mach numbers ($0.06 \leq M \leq 0.3$) with a step size of 0.06 and angle of attack ($-5^\circ \leq \alpha \leq 5^\circ$) with a step size of 1° . To focus on the performance of the wedge tail, the dataset generation specifically targeted the trailing section of the airfoils, as shown in Fig. 2, only the flow field of the mesh around the wedge tail of the airfoils is selected as the training and test data for the model. Data from regions outside of this specified area do not contribute to the training and prediction of the model. A total of 55 cases were created for each airfoil, resulting in a cumulative count of 220 cases across all four airfoils. The dataset includes various features such as airfoil sections (A), x – y grid coordinates (X , Y), Mach number (M), angle of attack (α), pressure (P), velocity magnitude (U), velocity component in the X direction (U_x), Y direction (U_y), lift coefficient (C_L), and drag coefficient (C_D). The features of the airfoil sections (A) initially consisted of the names of the corresponding airfoils. However, these names cannot serve as inputs for machine learning models, including the multi-head perceptron (MHP) network. In order to facilitate the training of both machine learning models and the MHP network, the airfoil sections (A) feature was converted to numerical values using numerical encoding. Subsequently, the dataset was divided into training and test sets. The distribution of the test dataset is shown in Table 2. The histograms of the training dataset are shown in Fig. 3, revealing mostly uneven value distributions. Notably, the pressure (P) is distributed across a broader range of values compared to the other features (U , U_x , U_y , C_L , C_D).

The correlation matrix of the entire dataset is shown in Fig. 4, revealing that airfoil sections (A) exhibit a very weak positive correlation with pressure (P) and the velocity component in the X-direction (U_x), indicating minimal influence. Additionally, there is a very weak negative correlation with the velocity magnitude (U). Furthermore, there is no correlation with the velocity component in the Y-direction (U_y) and lift coefficient (C_L). Notably, airfoil sections (A) demonstrate a moderate negative correlation with the drag coefficient (C_D), emphasizing the potential for significant drag reduction. Concerning grid coordinate parameters, such as horizontal position (X), very weak positive correlations are observed with P , U , U_x , and U_y , with no correlations with C_L and C_D . Vertical position (Y) shows a very weak positive correlation with P and a very weak negative correlation with U , U_x , and U_y , with no apparent correlation with C_L and C_D . The Mach Number (M) exhibits a moderate negative correlation with P and a moderate positive correlation with U , accompanied by a very weak positive correlation on U_x . No correlation is observed with U_y , and



Table 1 Summary of studies reviewed for airfoil aerodynamics predictions

References	Approach	Key findings
Sekar et al. [15]	Two-stage deep learning method for rapid flow field prediction over airfoils. The first stage involves parameterizing the airfoil using a convolutional neural network (CNN) to capture its geometric features. The second stage utilizes a multilayer perceptron (MLP) network for flow field prediction, considering inputs like airfoil parameters, x - y grid coordinates, and flow conditions	The MLP network predicts point-by-point velocity and pressure distributions for airfoils, using CNN-derived airfoil parameters, x - y grid coordinates, and flow conditions like Reynolds number and angle of attack. It shows promise in accurately predicting diverse airfoil flow fields, adapting well to varying geometries. The CNN-based parameterization method is general, flexible, and applicable to 3D general geometries, improving the generalization performance with an increase in the number of airfoil samples. Trained on these inputs, the MLP accurately predict the flow field near the boundary, which is important for airfoils
Duru et al. [17]	Neural networks combined with RANS simulations predict transonic flow field around airfoils, evaluating flow predictions and aerodynamic coefficient accuracy. It further explores estimating shear stress and computing moment coefficients for steady flow fields	The neural network model utilizes RANS simulations as input to predict transonic flow fields around airfoils. It successfully captures features such as shock waves and flow separation, while also computing aerodynamic coefficients. Though slightly less accurate, this approach shows potential in replacing time-consuming CFD simulations for airfoil design optimization
Du et al. [18]	Introduces DPCNN, a CNN-based framework for airfoil design and performance prediction, integrating the Bezier-GAN model for efficient parameterization and generating smooth profiles. The surrogate model includes physical field and aerodynamic performance prediction networks, enhancing accuracy	The model predicts physical field distributions and aerodynamic performance metrics using airfoil and operating parameters within the DPCNN framework, ensuring precise airfoil design and rapid surrogate modeling. The Bezier-GAN model generates numerous smooth airfoil profiles, enriching the training dataset and accurately capturing local physical field characteristics near the airfoil surface. This enables the deep learning model to swiftly predict the complete physical field within milliseconds, providing real-time insights
Kasmaiee et al. [19]	Optimized wind turbine airfoil blowing jet performance during dynamic stall via neural networks and genetic algorithms. It utilized URANS simulations with the k - ω SST turbulence model and employed two neural networks to link design variables to mean lift and drag coefficients, aiming to lower computational costs	The design variables for blowing jet include location, opening length, velocity, and jet angle, optimizing the average lift-to-drag ratio (L/D) over an oscillation period. Optimization improved mean aerodynamic performance by reducing drag and increasing lift coefficients. The optimal jet parameter reached 11.727 or 4.717 times the uncontrolled case, with the best location identified at approximately 2–5% of the chord, near the leading edge vortex. Jet angles between 55° and 70° notably enhanced aerodynamic performance
Wu et al. [20]	Introduces daGAN, a data augmented Generative Adversarial Network, offering precise flow field prediction with limited data. It involves pre-training with a conditional GAN (cGAN) to understand data distribution, followed by fine-tuning using a unique two-generator adversarial architecture	daGAN uses numerical simulation data of airfoils and flow conditions, particularly when working with limited training data. Its primary objective is predicting highly detailed flow fields for airfoils based on this input. Through the utilization of sparse airfoil and flow condition data in numerical simulations, daGAN demonstrates exceptional proficiency in rapidly and accurately evaluating these complex flow fields without an extensive need for training data
Thuerey et al. [21]	Assesses accuracy of deep learning models in inferring RANS solutions. Using a modern U-net architecture, it evaluates multiple trained neural networks for computing pressure and velocity distributions	The U-net architecture predicts pressure and velocity distributions in RANS solutions, leveraging airfoil shape and velocity components for freestream conditions. It shows that variations in data size and neural network weights influence accuracy, achieving less than 3% mean relative errors for pressure and velocity across diverse airfoil shapes



Table 1 continued

References	Approach	Key findings
Chen et al. [22]	Introduces FLOWGAN, a conditional generative adversarial network that directly maps given boundary conditions and geometry shapes to corresponding flow fields. It aims to predict flow fields accurately in various conditions without the need for re-training. FLOWGAN combines cGAN and the U-Net architecture to achieve this one-to-one mapping and adaptability to different flow conditions based on experimental observations	The FLOWGAN model utilizes observations of flow fields under various conditions, including boundary conditions and geometry shapes, as its input. It predicts corresponding flow fields tailored to these specific input conditions. It reduces prediction errors by 2.27% compared to leading DL-based methods, showing superior generalization without extensive re-training. It outperforms counterparts like Endec and Unet in accurately predicting entire flow fields. Moreover, FLOWGAN excels in learning boundary information and predicting velocity quantities near airfoil surfaces
Zuo et al. [23]	A data-driven approach based on convolutional neural network (CNN) and multi-head perceptron (MHP) to predict the incompressible laminar steady sparse flow field around airfoils. CNN is used to extract the geometry parameters of the airfoil from input grayscale images	Both MLP and MHP predict airfoil pressure and velocity distributions using CNN-extracted airfoil geometry parameters from grayscale images alongside Reynolds number, angle of attack, and flow field coordinates as inputs. The study compares their respective outcomes, demonstrating that the proposed MHP network architecture can predict airfoil flow fields within seconds. Results indicate that the MHP outperforms the MLP in predicting sparse flow fields, yielding better prediction outcomes
Jiang et al. [24]	Proposed TransCFD, a Transformer-based decoding architecture for flow field prediction, which learns an end-to-end mapping between the aerodynamic shape and the flow fields	The model predicts pressure coefficient, X velocity, and Y velocity based on 2D airfoil geometry input, achieving a mean absolute error (MAE) of less than 1% and a remarkable three-order increase in speed compared to traditional CFD methods. Additionally, the approach simplifies input requirements compared to many existing methods and exhibits strong generalization capabilities
Zhao et al. [25]	Utilized machine learning (ML) alongside CFD to optimize airfoil parameters for the Mars helicopter. Multiple ML algorithms such as Ada-Boost, SVM-Gaussian, SVM-Linear, ANN (20–20, 30–30, 40–40) were assessed for their predictive capabilities, facilitating the optimization process	Aerodynamic characteristics (C_l/C_d) and ($C_l^{3/2}/c_d$) of rotor system of the Mars helicopter was predicted by considering geometric parameters like relative camber of airfoils (m), maximum camber position (ρ), relative thickness (t), and angle of attack (α). The SVM-G regression model exhibits high accuracy for predicting C_l/C_d , and an ANN 40–40 model is used for $C_l^{3/2}/c_d$ based on MSE and R^2 scores. Prediction errors fall within 3% for C_l/C_d and within 5% for $C_l^{3/2}/c_d$ compared to simulation results
Ribeiro et al. [26]	A reduced order model (ROM) is developed to predict unsteady flow on the NLR 7301 airfoil using neural networks trained on high-fidelity CFD data. The approach involves dimensionality reduction, incorporates flight physics, and validates the model with unsteady flow data from DLR TAU-Code [27]	Predictions were made for coefficients of pressure (C_p) and friction (C_f), drag (C_D), and lift (C_L) using neural networks. The study used unsteady Schroeder multi-sine signals for angle of attack and its first and second time-derivatives as inputs for the NLR 7301 aerodynamic airfoil. Evaluating three unsteady reduced order model (ROM) variants for Mach 0.5 and 0.7 conditions revealed different performances. At Mach 0.5, ROMs displayed errors around 1–2% for C_p and approximately 1% for C_f . However, at Mach 0.7, proper-orthogonal decomposition (POD) based ROM struggled near shock waves, exhibiting higher C_p/C_f errors of up to 3–4%. Autoencoder (AE) based ROMs showed C_p errors below 2% except for high-frequency angles. Integral Coefficients (Mach 0.5) indicated mean absolute percentage errors (MAPE) around 1%, slightly improving with AE and flight-physics-based regularization. For Mach 0.7, AE based models achieved MAPE of 1–5%, further enhanced with flight-physics-based regularization



Table 1 continued

References	Approach	Key findings
Zhang et al. [28]	Deploys DNNs to predict turbulent flows around a NACA 4412 airfoil using LES generated training data. A convolutional autoencoder (CAE) reduces dimensions while a multi-head attention-strengthened long short-term memory (MH-LSTM) network predicts latent vector evolution, enabling reconstruction of high-dimensional flow dynamics from the reduced representation	The model processes high-dimensional flow dynamics from LES of a NACA 4412 airfoil as input, employing a CAE for nonlinear reduction. Using MH-LSTM, it predicts latent vector evolution and reconstructs the flow dynamics from the reduced form. The study introduces AirNet, combining a convolutional autoencoder (CAE) and multi-head LSTM to predict turbulent structures over the airfoil, informed by LES and experimental data. It uses POD and CAE to reproduce future flow fields, integrating physical quantities and employing attention mechanisms for enhanced training. The MH-LSTM approach is compared against LSTM for predicting turbulent flow evolution
Song et al. [29]	Employed pix2pix method, utilizing conditional generative adversarial networks (cGANs) for image-to-image translation, alongside a DNN approach. It effectively predicted the airfoil flow field using fully implicit, high-resolution scheme-based compressible CFD codes with genetic algorithms	The pix2pix and a DNN predict airfoil flow fields and aerodynamic performance across various wind turbine blade configurations. Pix2pix, using conditional generative adversarial networks (cGANs), accurately predicted vortical flow fields in thick airfoils. Meanwhile, the DNN, leveraging ReLU and Leaky ReLU activation functions, outperformed traditional computational fluid dynamics (CFD) methods. This approach holds promise for enhancing wind turbine blade designs, particularly for larger blades in high-speed offshore environments
Wang et al. [30]	Introduces BiFlowAN, a deep transfer learning framework for predicting airfoil flow fields with limited data. Leveraging a generative adversarial network (GAN), it effectively predicts flow fields in data-rich conditions. Transfer learning extends this capability to BiFlowAN-TL, improving generalization with smaller datasets	The model standardizes a parameterized airfoil, predicting its flow field under varying conditions. The study introduces BiFlowAN, a transfer learning framework based on GANs, accurately predicting data-rich flow fields. Transfer learning extends this capability to BiFlowAN-TL, improving generalization for small-scale datasets and enhancing prediction accuracy. The methods showcased the potential for aerodynamic evaluation with limited data, emphasizing detailed flow field considerations for refined airfoil designs and improved generalization across diverse flow conditions
Kasmaee et al. [31]	Employed a genetic algorithm to optimize the configuration of a pure suction jet actuator on an oscillating airfoil. Neural networks based on multilayer perceptrons (MLP) were used to train the aerodynamic coefficients as functions of the control parameters and reduce the number of simulations	The model utilizes the location, velocity, opening length, and suction jet angle relative to the airfoil surface to predict the mean performance coefficient, which represents the ratio of the average lift to the average drag during an oscillation period. The optimized jet configuration, positioned near the leading edge vortex at 3–6% of the chord, significantly improved aerodynamic performance. If increased the average lift coefficient from 0.58 to 0.92 and decreased the average drag coefficient from 0.23 to 0.02. Additionally, the optimized jet prevented the growth of the dynamic stall vortex, suppressing the counterclockwise trailing edge vortex



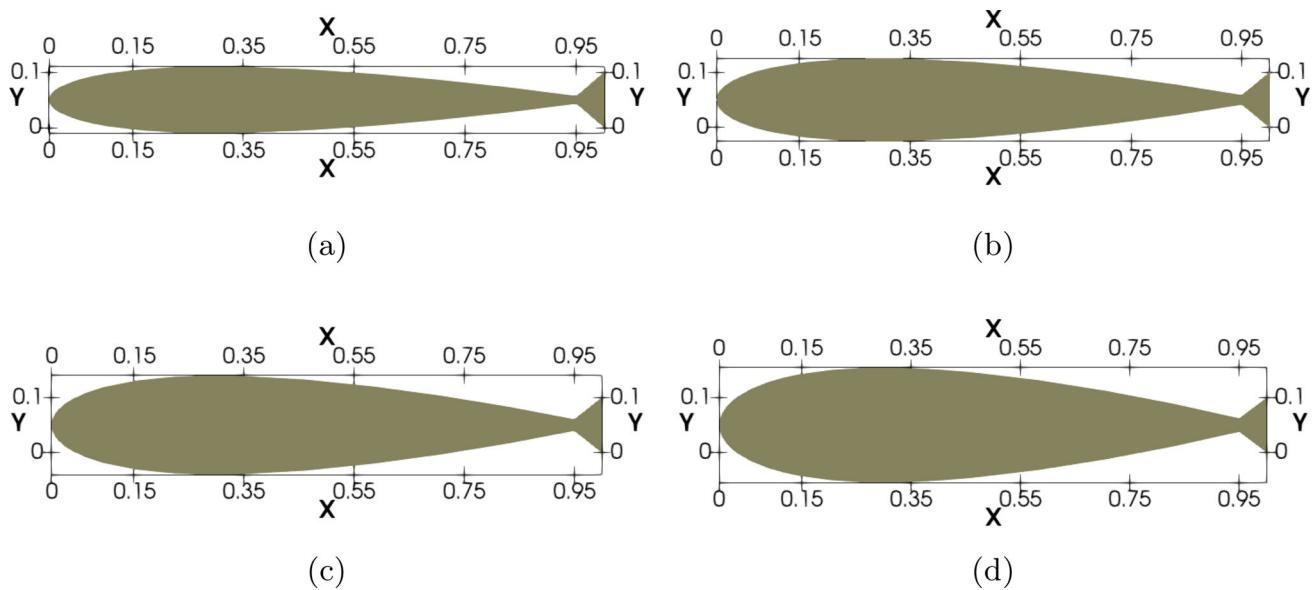


Fig. 1 Geometry of the **a** NACA 0012—wedge at $0.1c$, **b** NACA 0015—wedge at $0.1c$, **c** NACA 0018—wedge at $0.1c$ and **d** NACA 0021—wedge at $0.1c$ airfoil sections

Fig. 2 Airfoil trailing edge portion used in generating the dataset



Table 2 Sample distribution of the test dataset

Airfoil sections	Mach number (M)	Angle of attack (α)	
		Flow field prediction	Aerodynamic coefficients prediction
NACA 0012—wedge at $0.1c$	0.06	-5	$[-5, 5]$ ($\Delta \alpha = 1$)
NACA 0015—wedge at $0.1c$	0.12	-3	$[-5, 5]$ ($\Delta \alpha = 1$)
NACA 0018—wedge at $0.1c$	0.18	3	$[-5, 5]$ ($\Delta \alpha = 1$)
NACA 0021—wedge at $0.1c$	0.24	5	$[-5, 5]$ ($\Delta \alpha = 1$)

there is a very weak negative correlation with C_L . A fairly strong negative correlation is evident with C_D , highlighting significant drag reduction at higher Mach Numbers. In contrast, the angle of attack (α) shows no correlation with P , U , and U_x , with a very weak negative correlation on U_y and C_D . However, it demonstrates a perfect positive correlation with C_L , suggesting a significant increase in lift at high angles of attack. Since experimental data for the wedge tail airfoils was unavailable, the simulation results were validated using NACA 0012, NACA 0015, NACA 0018, and NACA 0021 airfoils. A detailed discussion of the RANS-CFD analysis is provided in the subsequent sections.

4 RANS-CFD Analysis

To investigate the flow fields and aerodynamic characteristics of the wedge tail airfoils, simulations were conducted using the Reynolds-Averaged Navier–Stokes (RANS) method in a two-dimensional domain. The simulations covered a range of Mach numbers ($0.06 \leq M \leq 0.3$) and angles of attack ($-5^\circ \leq \alpha \leq 5^\circ$). The commercial CFD software ANSYS Fluent was utilized to perform the simulations, and high-quality meshes were generated using ANSYS Mechanical. To validate the accuracy of the RANS method, 2D NACA 0012, NACA 0015, NACA 0018, and NACA 0021 airfoil sections with a chord length (c) of 0.1524 m were employed.

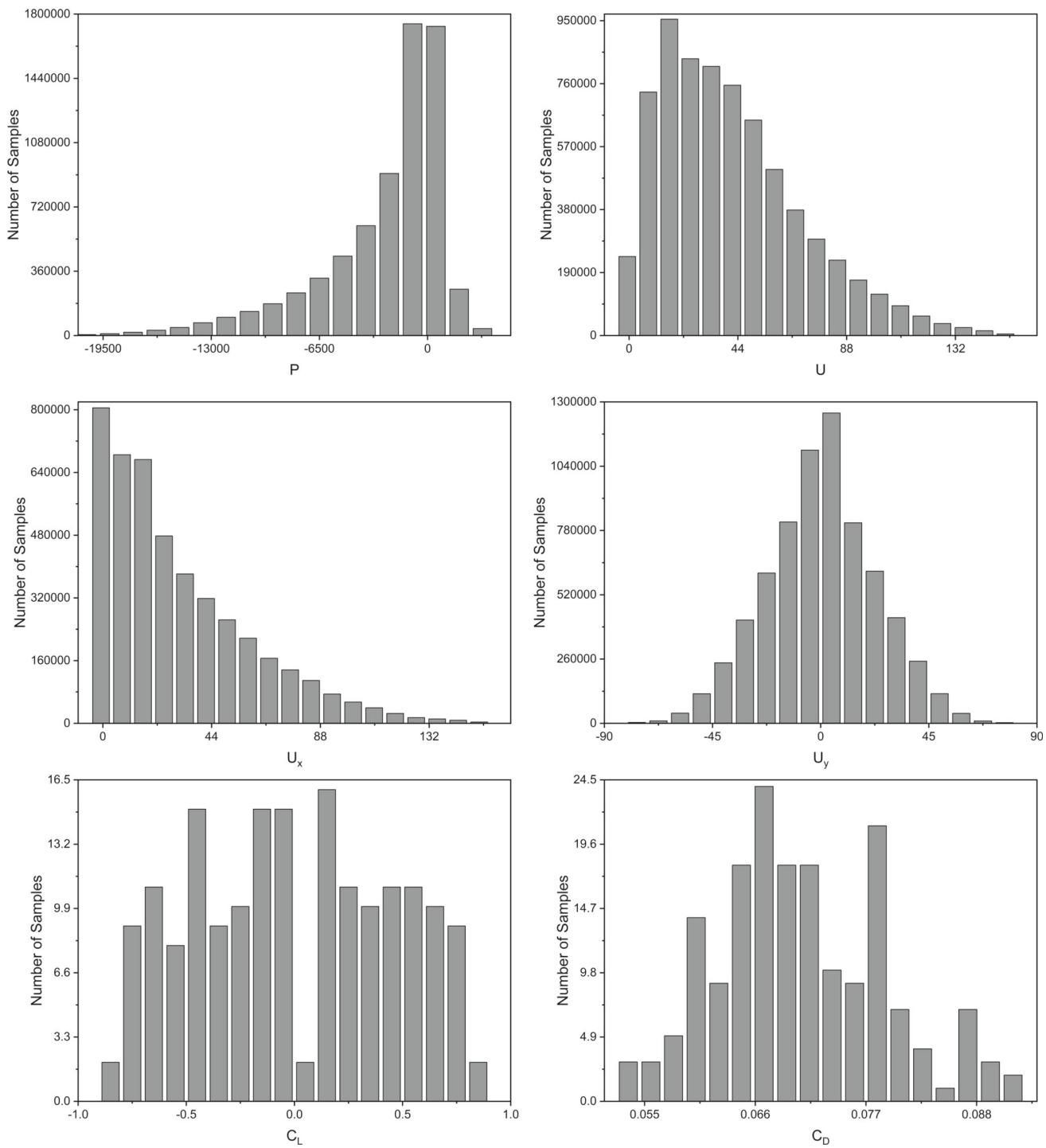


Fig. 3 Histograms of the training dataset

This selection of airfoil sections aligns with the experimental setup conducted by Sheldahl et al. [32].

4.1 Governing Equations

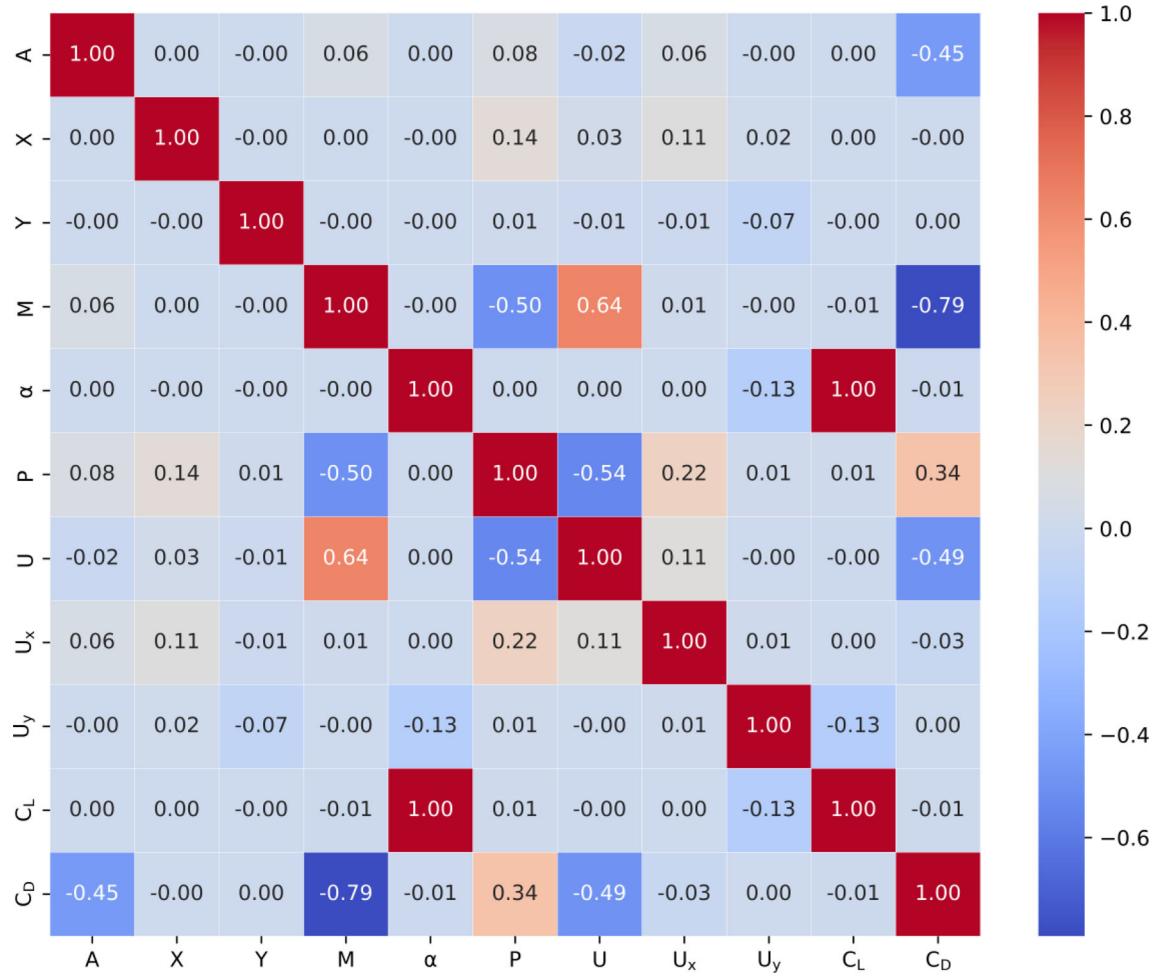
The equations governing fluid flow problems are derived from physical laws and can be expressed mathematically. In

particular, the equations for mass and momentum conservation are essential for describing fluid flow in any given case. These equations can be represented as follows [33]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = S_m, \quad (1)$$

$$\frac{\partial}{\partial t}(\rho \vec{u}) + \nabla \cdot (\rho \vec{u} \vec{u}) = -\nabla p + \nabla \cdot (\bar{\tau}) + \rho \vec{g} + \vec{F}, \quad (2)$$



**Fig. 4** Correlation matrix of the dataset

where $\bar{\tau}$ is the stress tensor, which can be written as

$$\bar{\tau} = \mu \left[\left(\nabla \vec{u} + \nabla \vec{u}^T \right) - \frac{2}{3} \right] \nabla \cdot \vec{u} I. \quad (3)$$

For steady and non-compressible two-dimensional flows, the continuity and momentum equations for viscous flow in the x and y directions are:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (4)$$

$$\rho \frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \rho f_x, \quad (5)$$

$$\rho \frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \rho f_y. \quad (6)$$

In this study, the Spalart–Allmaras turbulence model [34] has been utilized to account for turbulence effects in the simulation. This model is a one-equation turbulence model. The model is effective at predicting stalled flows in boundary layers with adverse pressure gradients [35]. The

SA model consists of one equation expressed as follows [36]:

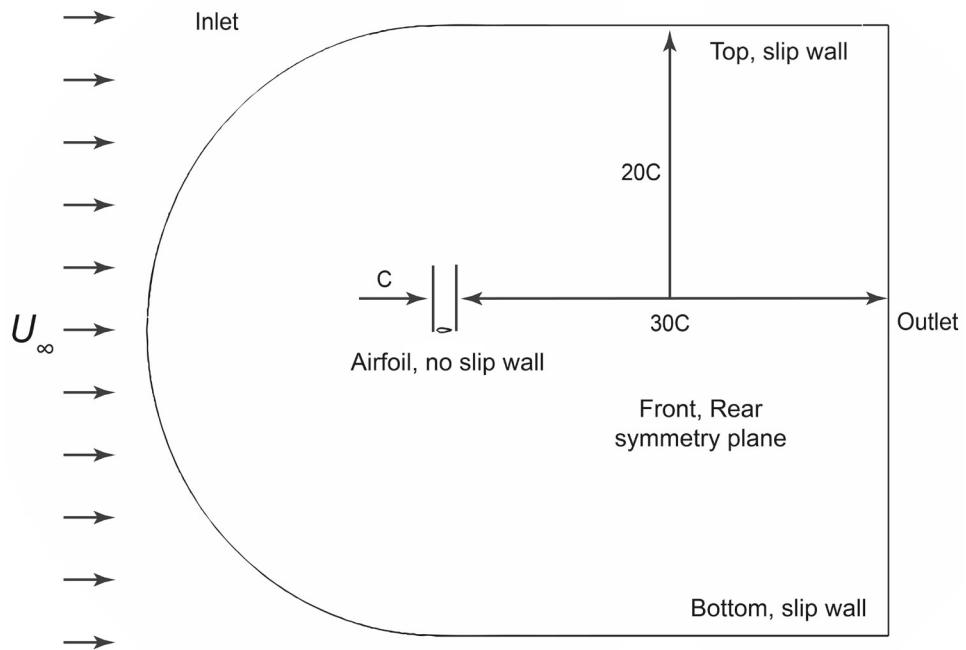
$$\begin{aligned} & \frac{\partial(\rho \tilde{v})}{\partial t} + \operatorname{div}(\rho \tilde{v} U_i) \\ &= \frac{1}{\sigma_v} \operatorname{div} \left[(\mu + \rho \tilde{v}) \nabla \tilde{v} + \rho C_{b2} \frac{\partial \tilde{v}}{\partial x_i} \frac{\partial \tilde{v}}{\partial x_i} \right] \\ &+ \rho C_{b1} \tilde{v} \tilde{\Omega} - \rho C_{w1} \left(\frac{\tilde{v}}{\kappa y} \right)^2 f_w, \end{aligned} \quad (7)$$

$$\tilde{\Omega} = \Omega_{ij} + \frac{\nabla}{(\kappa y)^2} f_{v2}, \quad (8)$$

where \tilde{v} , $\tilde{\Omega}$, Ω_{ij} , and μ represent the kinematic eddy viscosity, local mean vorticity, and mean vorticity tensor, respectively. Furthermore, the wall-damping functions are denoted as $f_{v2} = f_{v2}(\tilde{v}/v)$ and f_w . The values of σ_v , C_{b1} , C_{b2} , and k are 0.67, 0.1355, 0.622, and 0.4187, respectively.



Fig. 5 Simulation of fluid flow around an airfoil in a C-type domain



4.2 Boundary Conditions

In this study, the computational domain featured a C-shaped configuration. The left side of the domain represented the inlet, where a uniform velocity was imposed. The top and bottom walls were subjected to slip boundary conditions, while the right side of the domain served as the outlet, with the flow controlled by atmospheric pressure. The front and rear domains were defined by symmetry boundary conditions. The airfoil profile wall was assigned a no-slip condition with a zero velocity ($u_p = 0$). Simulations were performed considering an incompressible flow with a density of 1.225 kg/m^3 and dynamic viscosity of $1.7894 \times 10^{-5} \text{ kg/(ms)}$. The Mach numbers (M) considered ranged from 0.06 to 0.3. The geometry of the airfoil domain and the corresponding boundary conditions are shown in Fig. 5.

4.3 Mesh Generation

The accuracy of the numerical solution in CFD simulations is heavily influenced by the size of the computational mesh. While increasing the number of mesh nodes can enhance accuracy, it also comes with the drawback of requiring more computational resources and time. Consequently, the initial step in CFD simulations involves examining the effect of the mesh size on the solution outcomes. In this study, the impact of the number of mesh elements on the lift coefficient (C_L) and drag coefficient (C_D) was investigated for the NACA 0012, NACA 0015, NACA 0018, and NACA 0021 airfoils. The simulations were conducted at an angle of attack (α) of 1° , with a corresponding Reynolds number (Re) of

0.8×10^5 . The results of this analysis are presented in Table 3. It was observed that a C-type grid consisting of 146,400 quadrilateral elements provided a mesh-independent solution. The mesh resolution was increased to achieve higher computational precision in specific regions, such as around the airfoils. The near-wall cells on the upper and lower surfaces of the airfoil were adjusted to achieve a desired y^+ value of 1, following boundary layer theory. The Pointwise® y^+ calculator was employed for this purpose. According to boundary layer theory, the inner parts of the boundary layer should be resolved with a size corresponding to the y^+ value. Figure 6 illustrates the mesh around an airfoil and the associated near-body meshes.

5 Multi-head Perceptron (MHP) Network

This study employed the multi-head perceptron (MHP) network instead of the multi-layer perceptron (MLP) network due to the significant limitations of MLP in processing sparse data-characterized by an uneven distribution of data points. This uneven distribution can lead to suboptimal performance when using standard neural network architectures like MLP. The inefficiencies arise from difficulties in generalization and parameter optimization in areas where data is sparse. In cases with multiple output targets, MLP updates network parameters in each preceding layer to enhance predictive accuracy. Unfortunately, the presence of sparse data can lead to unsatisfactory model fitting, necessitating a more tailored approach. Zuo et al. [23] demonstrated the superiority of MHP over MLP in predicting flow field patterns around air-



Table 3 Mesh independence study for a different number of elements on the surface of airfoils with a Reynolds number (Re) of 0.8×10^5

Airfoils	Test case	No. of elements	Angle of attack, α 1°	C_L	% Deviation from experimental		C_D	% Deviation from experimental	% Deviation from experimental medium
					% Deviation from medium	% Deviation from experimental			
NACA 0012	Coarse	75,600	0.10236	-3.84	0.11	-6.95	0.01885	8.17	0.0134
	Medium	146,400	0.10645	0.00		-3.23	0.01743	0.00	30.04
	Fine	291,200	0.10805	1.50		-1.77	0.01796	3.05	34.01
	Coarse	75,600	0.09852	-6.12	0.11	-10.44	0.02052	6.14	0.0148
	Medium	146,400	0.10494	0.00		-4.60	0.01933	0.00	38.65
	Fine	291,200	0.10582	0.84		-3.80	0.01989	2.90	30.62
	Coarse	75,600	0.09225	4.20	0.0889	3.77	0.02241	8.86	34.42
	Medium	146,400	0.08853	0.00		-0.41	0.02059	0.00	37.48
	Fine	291,200	0.08834	-0.22		-0.63	0.02052	-0.33	26.30
NACA 0018	Coarse	75600	0.08332	-6.89	0.0921	-9.53	0.02454	9.75	25.88
	Medium	146400	0.08948	0.00		-2.84	0.02236	0.00	37.87
	Fine	291,200	0.09010	0.69		-2.17	0.02290	2.41	25.63
NACA 0021	Coarse	75600	0.08332	-6.89	0.0921	-9.53	0.02454	9.75	28.65
	Medium	146400	0.08948	0.00		-2.84	0.02236	0.00	
	Fine	291,200	0.09010	0.69		-2.17	0.02290	2.41	



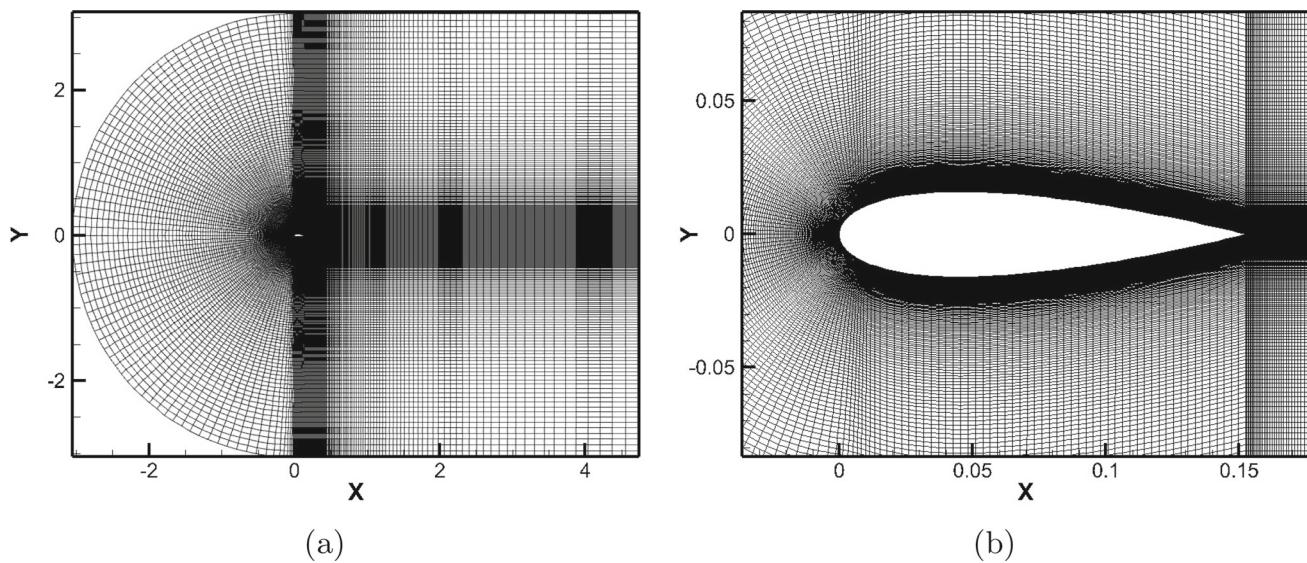


Fig. 6 **a** Mesh around airfoil and **b** detail close to the airfoil

foils from sparse data. MHP excels in scenarios with multiple output targets, optimizing predictive accuracy without sacrificing computational efficiency. The multi-head architecture of MHP allows it to capture diverse patterns within the data, mitigating the shortcomings of traditional neural networks in sparse data scenarios, making it a promising alternative to MLP. Since this study addresses an uneven distribution of data points, the adaptability and performance of MHP in such settings make it a compelling choice. The MHP network architecture utilized for flow fields and aerodynamic coefficients prediction is shown in Fig. 7. The prediction functions for flow fields and aerodynamic coefficients of the MHP are defined in Eqs. 9 and 10, respectively.

$$\begin{cases} f_{\text{MHP-P}}(A, X, Y, M, \alpha) = (P), \\ f_{\text{MHP-U}}(A, X, Y, M, \alpha) = (U), \\ f_{\text{MHP-U}_x}(A, X, Y, M, \alpha) = (U_x), \\ f_{\text{MHP-U}_y}(A, X, Y, M, \alpha) = (U_y). \end{cases} \quad (9)$$

$$\begin{cases} f_{\text{MHP-C}_L}(A, M, \alpha) = (C_L), \\ f_{\text{MHP-C}_D}(A, M, \alpha) = (C_D). \end{cases} \quad (10)$$

In the MHP prediction model, the left side of the function represents the model itself, while the right side represents the resulting predictions. To evaluate the performance of the MHP network and other machine learning (ML) models, metrics including the R^2 score, mean absolute error (MAE), mean squared error (MSE) and, root mean squared error (RMSE) are employed, as defined in Eqs. 11, 12, 13, and 14, respectively. These metrics serve as evaluative measures

for monitoring and assessing the accuracy of the MHP networks and other ML models.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (11)$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (12)$$

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (13)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (14)$$

Here, n represents the total number of samples. y_i denotes the actual value for the i th sample, while \hat{y}_i represents the predicted value for the same sample. Additionally, \bar{y} corresponds to the mean of the actual values across all samples. Standardization is applied to all features, ensuring consistent scaling across the dataset. The data is then split into training and validation sets using a 90–10% ratio. The weights and biases of the multi-head perceptron (MHP) networks are trained using the ADAM optimizer [37]. The learning rate is set to 1×10^{-3} with a batch size of 128. In the MHP network, the Rectified Linear Unit (ReLU) activation function [38] is applied in all layers, except for the output layer, where a linear activation function is utilized. The network configuration for predicting flow fields and aerodynamic coefficients



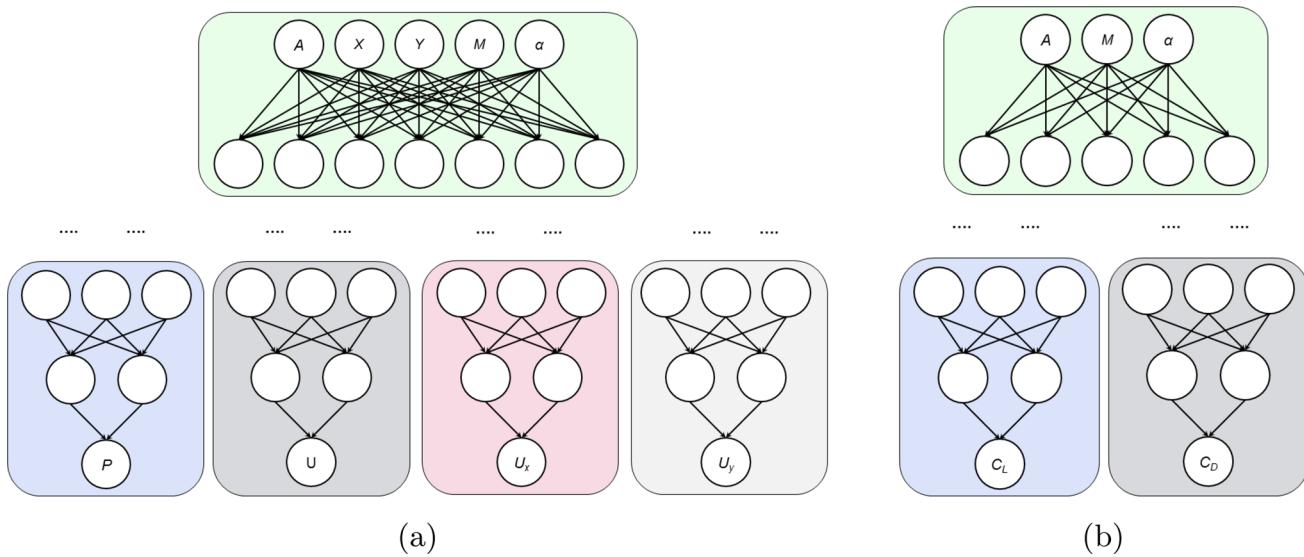


Fig. 7 Multi-head perceptron (MHP) network for **a** flow fields and **b** aerodynamic coefficients prediction

Table 4 MHP network with hyperparameters

Layer type	MHP	
	Flow fields prediction	Aerodynamic coefficients prediction
Input	1×5	1×3
Hidden	10×256	10×256
Output	1×1	1×1

is determined through a systematic trial-and-error process, revealing an optimal structure with 10 hidden layers and 256 nodes, as shown in Table 4. The training process for the flow field prediction network spans 500 epochs, while for the aerodynamic coefficient prediction network, the model is trained for 50 epochs. This choice is motivated by the limited number of data points, as further training epochs led to observed overfitting. During these epochs, the loss functions exhibit a convergence to a steady state, indicating satisfactory model performance.

6 Machine Learning (ML) Models

Three machine learning (ML) models were developed using the scikit-learn library in Python [39] to predict flow fields (P , U , U_x , U_y) and aerodynamic coefficients (C_L , C_D) based on airfoil sections (A), x - y grid coordinates (X , Y), Mach number (M), and angle of attack (α). The objective was to select appropriate ML algorithms for fitting the data in this multivariate supervised regression learning problem [40]. The chosen ML models for this study are as follows: decision tree regressor (DTR), k-nearest neighbors regressor (KNNR), and random forest regressor (RFR) implemented using the scikit-learn library.

Decision Trees (DTs) are highly adaptable algorithms for classification and regression tasks. They exhibit versatility by accommodating multiple outputs and possessing a hierarchical structure resembling a tree. In this structure, nodes represent features, branches symbolize decisions, and leaves signify outputs [41]. For this study, the Decision Tree Regressor (DTR) implementation provided by the scikit-learn library was employed to train, validate, and test the simulated data of the wedge tail airfoils.

K-nearest neighbors (KNN) is a straightforward algorithm for classifying new data based on similarity measures with the input data. This algorithm is commonly employed in statistical estimation and pattern recognition tasks. It selects a predetermined number of training objects closest to a new point and uses them to predict its corresponding label. The KNNR algorithm, a modified variant of KNN, calculates the average of the numerical targets instead of directly assigning a class label [41].

Random Forest (RF) is an ensemble learning technique that builds multiple decision trees during the training process and generates the mode of the classes (for classification tasks), or the mean prediction (for regression tasks) based on the individual trees outputs [41]. In this study, the RFR estimator provided by the scikit-learn library was utilized for training and testing the simulated datasets of the wedge

Table 5 R^2 score for training and validation cases

Datasets	Models	R^2					
		P	U	U_x	U_y	C_L	C_D
Training	MHP	0.9999	0.9996	0.9999	0.9997	0.9975	0.9937
	DT	0.9906	0.9906	0.9906	0.9906	0.8774	0.8774
	KNN	0.9945	0.9945	0.9945	0.9945	0.8890	0.8890
	RF	0.9908	0.9908	0.9908	0.9908	0.9028	0.9028
Validation	MHP	0.9996	0.9992	0.9997	0.9993	0.9968	0.9934

Table 6 MAE for training and validation cases

Datasets	Models	MAE					
		P	U	U_x	U_y	C_L	C_D
Training	MHP	24.8540	0.2638	0.3314	0.1737	0.0154	0.0008
	DT	18.7704	18.7704	18.7704	18.7704	0.0090	0.0090
	KNN	16.0552	16.0552	16.0552	16.0552	0.0246	0.0246
	RF	18.0777	18.0777	18.0777	18.0777	0.0073	0.0073
Validation	MHP	18.7543	0.2926	0.3359	0.2712	0.0199	0.0024

Table 7 MSE for training and validation cases

Datasets	Models	MSE					
		P	U	U_x	U_y	C_L	C_D
Training	MHP	1131.0167	0.2034	0.1530	0.0852	0.0005	8.07×10^{-7}
	DT	3031.3197	3031.3197	3031.3197	3031.3197	0.0002	0.0002
	KNN	3169.7090	3169.7090	3169.7090	3169.7090	0.0014	0.0014
	RF	2874.4347	2874.4347	2874.4347	2874.4347	0.0001	0.0001
Validation	MHP	768.3647	0.2312	0.2206	0.1446	0.0006	8.48×10^{-6}

tail airfoils. The selection of these machine learning (ML) models was based on their ease of implementation using scikit-learn and their established track record of effectively fitting real-world data. Each model was trained individually using the same training data and evaluated using identical testing data. The performance of these models will be discussed in detail in the subsequent “Results and discussion” section.

7 Results and Discussion

A set of four previously unseen wedge tail airfoil cases are used to test the trained models. The trained multi-head perceptron (MHP) model, decision tree (DT), k-nearest neighbors (KNN), and random forest (RF) are employed to make predictions for these test cases. The R^2 scores, MAE, MSE, and RMSE obtained for the training and validation cases are shown in Tables 5, 6, 7, and 8, respectively. Table 5 for R^2 score shows the exceptional performance of the MHP across all features in the training set, achieving R^2 scores of 0.9999 for P, 0.9996 for U, 0.9999 for U_x , 0.9997 for U_y , 0.9975 for C_L , and 0.9937 for C_D . Comparative analysis

Table 8 RMSE for training and validation cases

Datasets	Models	RMSE					
		P	U	U_x	U_y	C_L	C_D
Training	MHP	33.6306	0.4510	0.3912	0.2919	0.0226	8.98×10^{-4}
	DT	55.0574	55.0574	55.0574	55.0574	0.0123	0.0123
	KNN	56.3002	56.3002	56.3002	56.3002	0.0371	0.0371
	RF	53.6138	53.6138	53.6138	53.6138	0.0109	0.0109
Validation	MHP	27.7194	0.4808	0.4697	0.3803	0.0249	0.0029



Table 9 R^2 score of different models for test airfoil cases in flow fields prediction

Airfoil sections	Algorithms	R^2			
		P	U	U_x	U_y
NACA 0012—wedge at 0.1c	MHP	0.98283	0.99763	0.99969	0.99940
	DT	0.99692	0.99027	0.99564	0.99149
	KNN	0.99729	0.99495	0.99876	0.99413
	RF	0.99720	0.99191	0.99659	0.99208
NACA 0015—wedge at 0.1c	MHP	0.99977	0.99931	0.99988	0.99968
	DT	0.99787	0.98929	0.99681	0.99043
	KNN	0.99981	0.99924	0.99981	0.99946
	RF	0.99815	0.99059	0.99711	0.99157
NACA 0018—wedge at 0.1c	MHP	0.99947	0.99891	0.99944	0.99853
	DT	0.99708	0.98508	0.99532	0.98676
	KNN	0.99914	0.99807	0.99952	0.99837
	RF	0.99726	0.98572	0.99547	0.98788
NACA 0021—wedge at 0.1c	MHP	0.99974	0.99918	0.99983	0.99921
	DT	0.99572	0.97114	0.99213	0.98166
	KNN	0.99461	0.98124	0.99402	0.98330
	RF	0.99603	0.97190	0.99217	0.98244

Table 10 MAE score of different models for test airfoil cases in flow fields prediction

Airfoil sections	Algorithms	MAE			
		P	U	U_x	U_y
NACA 0012—wedge at 0.1c	MHP	13.3142	0.1862	0.1583	0.1252
	DT	9.6791	0.3973	0.4347	0.4607
	KNN	10.3795	0.3289	0.3349	0.3975
	RF	9.2165	0.3787	0.4195	0.4463
NACA 0015—wedge at 0.1c	MHP	10.0281	0.1843	0.2097	0.1698
	DT	34.0623	0.9548	1.0590	1.0570
	KNN	10.1936	0.2539	0.2769	0.2186
	RF	31.3142	0.8922	1.0009	0.9742
NACA 0018—wedge at 0.1c	MHP	30.0942	0.4421	0.6658	0.5677
	DT	83.4777	1.8034	2.0848	1.7346
	KNN	40.5998	0.6345	0.6200	0.5719
	RF	80.4069	1.7506	2.0395	1.6537
NACA 0021—wedge at 0.1c	MHP	31.7838	0.5200	0.4320	0.5189
	DT	154.5756	2.9258	3.1924	2.5426
	KNN	184.4831	2.5350	2.8783	2.3049
	RF	145.8821	2.9639	3.2448	2.5388

with other ML models, including DT, KNN, and RF, underscores the superior predictive capability of the MHP. In the validation set, the MHP model maintains robust performance with R^2 scores of 0.9996, 0.9992, 0.9997, 0.9993, 0.9968, and 0.9934 for P , U , U_x , U_y , C_L , and C_D , respectively. In Table 6 for MAE, it is observed that the MHP exhibits notable discrepancies in predicting P for the training set, yielding an MAE of 24.8540, which is higher compared to other ML models achieving lower MAE values ranging from 16.0552 to 18.7704. This indicates that, in the training phase,

the performance of the MHP in predicting P falls short compared to the other ML models. However, it is important to note that the MHP excels in predicting U , U_x , U_y , C_L , and C_D , as evidenced by its lower MAE values of 0.2638, 0.3314, 0.1737, 0.0154, and 0.0008, respectively, when compared to the ML models. Moving on to the validation set, considering P , the MAE value of 18.7543 indicates a comparatively higher level of prediction error of the MHP, contrasting the commendable accuracies demonstrated for U , U_x , U_y , C_L ,



Table 11 MSE score of different models for test airfoil cases in flow fields prediction

Airfoil sections	Algorithms	MSE			
		\bar{P}	U	U_x	U_y
NACA 0012—wedge at 0.1c	MHP	1139.7300	0.0925	0.0458	0.0293
	DT	189.2337	0.3787	0.6711	0.4059
	KNN	167.1017	0.1959	0.1903	0.2865
	RF	171.7654	0.3161	0.5254	0.3765
NACA 0015—wedge at 0.1c	MHP	187.7227	0.1090	0.0783	0.0546
	DT	1738.0237	1.6335	1.9992	1.6952
	KNN	152.6746	0.1171	0.1200	0.0935
	RF	1503.9660	1.4386	1.8062	1.4886
NACA 0018—wedge at 0.1c	MHP	1840.5396	0.3817	0.7698	0.5143
	DT	10291.7779	5.2537	6.6011	4.6056
	KNN	2977.6495	0.6601	0.6638	0.5721
	RF	9657.3632	5.0205	6.3884	4.2019
NACA 0021—wedge at 0.1c	MHP	2177.2105	0.4927	0.3603	0.4642
	DT	36238.8291	17.5722	17.7895	11.0144
	KNN	47413.4264	11.1920	13.4042	9.5338
	RF	33679.1599	16.9466	17.6920	3.2478

Table 12 RMSE score of different models for test airfoil cases in flow fields prediction

Airfoil sections	Algorithms	RMSE			
		P	U	U_x	U_y
NACA 0012—wedge at 0.1c	MHP	33.7599	0.3041	0.2139	0.1713
	DT	13.7562	0.6154	0.8192	0.6371
	KNN	12.9268	0.4426	0.4363	0.5352
	RF	13.1059	0.5623	0.7248	0.6136
NACA 0015—wedge at 0.1c	MHP	13.7012	0.3301	0.2798	0.2336
	DT	41.6896	1.2781	1.4139	1.3020
	KNN	12.3562	0.3422	0.3465	0.3057
	RF	38.7810	1.1994	1.3440	1.2201
NACA 0018—wedge at 0.1c	MHP	42.9015	0.6178	0.8774	0.7171
	DT	101.4484	2.2921	2.5693	2.1461
	KNN	54.5678	0.8124	0.8147	0.7564
	RF	98.2719	2.2407	2.5275	2.0499
NACA 0021—wedge at 0.1c	MHP	46.6606	0.7019	0.6003	0.6813
	DT	190.3650	4.1919	4.2178	3.3188
	KNN	217.7462	3.3455	3.6612	3.0877
	RF	183.5188	4.1166	4.2062	10.5483

Table 13 R^2 score, MAE, MSE, and RMSE for different models in aerodynamic coefficients prediction

Models	C_L				C_D			
	R^2	MAE	MSE	RMSE	R^2	MAE	MSE	RMSE
MHP	0.99928	0.0082	0.0001	0.0123	0.99364	0.0008	9.91×10^{-7}	0.0010
DT	0.99875	0.0128	0.0003	0.0162	0.75616	0.0052	3.80×10^{-5}	0.0062
KNN	0.98706	0.0449	0.0027	0.0521	0.79102	0.0042	3.26×10^{-5}	0.0057
RF	0.99902	0.0105	0.0002	0.0144	0.80659	0.0041	3.01×10^{-5}	0.0055



and C_D , with MAE values of 0.2926, 0.3359, 0.2712, 0.0199, and 0.0024, respectively.

From Table 7 for MSE, MHP emerges as a standout performer than ML models in the training set, with MSE values for P , U , U_x , U_y , C_L , and C_D of 1131.0167, 0.2034, 0.1530, 0.0852, 0.0005, and 8.07×10^{-7} , respectively. While the MSE for P is higher, it is lower compared to other ML models. In comparison, models such as DT, KNN, RF exhibit higher MSE values during the training phase, suggesting relatively lower predictive accuracy. The MSE values for these models range from 2874.4347 to 3169.7090, further highlighting the superior performance of the MHP model. Moving on to the validation set, the MHP exhibits substantial predictive accuracy. However, when considering P , the MSE value of 768.3647 indicates a comparatively higher level of prediction error. In contrast to the commendable accuracy demonstrated for other parameters such as U , U_x , U_y , C_L , and C_D with MSE values of 0.2312, 0.2206, 0.1446, 0.0006, and 8.48×10^{-6} , respectively.

In Table 8 for RMSE, among the models evaluated, the MHP performs better in training set with RMSE values for P , U , U_x , U_y , C_L , and C_D of 33.6306, 0.4510, 0.3912, 0.2919, 0.0226, and 8.98×10^{-4} , respectively. Although the RMSE of P is higher, it is lower compared to other ML models. Contrastingly, models such as DT, KNN, and RF exhibit higher RMSE values during the training phase, indicative of comparatively lower predictive accuracy. The RMSE values for these models range from 53.6138 to 56.3002, underscoring the superior performance of the MHP model. Moving on to the validation set, the MHP exhibits substantial predictive accuracy. However, when considering P , the RMSE value of 27.7194 indicates a comparatively higher level of prediction error. In contrast to the commendable accuracy demonstrated for other parameters such as U , U_x , U_y , C_L , and C_D with RMSE values of 0.4808, 0.4697, 0.3803, 0.0249, and 0.0029, respectively.

The performance of different models varies across the tested airfoil cases. In Table 9 showing the R^2 scores, notable performances were observed in predicting pressure (P) for the NACA 0012—wedge at 0.1c and NACA 0015—wedge at 0.1c with KNN achieving impressive R^2 scores of 0.99729 and 0.99981, respectively. Additionally, for the NACA 0018—wedge at 0.1c, KNN exhibited a strong performance in predicting U_x with an R^2 score of 0.99952. However, aside from these instances, the MHP model displayed superior predictive capability for other flow fields. Transitioning to Table 10 showcasing Mean Absolute Error (MAE), RF emerged as the better predictor for pressure (P) in the NACA 0012—wedge at 0.1c case, demonstrating an MAE of 9.2165. Similarly, KNN excelled in predicting U_x for NACA 0018—wedge at 0.1c with an MAE of 0.6200. Yet, apart from these cases, MHP exhibited superior predictive accuracy for most flow fields, albeit with relatively

higher errors in predicting pressure (P). Moving to Table 11 and considering Mean Squared Error (MSE), KNN showcased superior prediction for pressure (P) in NACA 0012—wedge at 0.1c and NACA 0015—wedge at 0.1c with the lowest MSE values of 167.1017 and 152.6746, respectively. Moreover, KNN performed notably well in predicting U_x for NACA 0018—wedge at 0.1c with an MSE of 0.6638. However, aside from these instances, MHP demonstrated better prediction for the majority of flow fields, despite relatively higher errors in predicting pressure (P). Referring to Table 12 containing Root Mean Squared Error (RMSE) values, KNN showcased superior prediction for pressure (P) in NACA 0012—wedge at 0.1c and NACA 0015—wedge at 0.1c with the lowest RMSE values of 12.9268 and 12.3562, respectively. Similarly, for NACA 0018—wedge at 0.1c, KNN demonstrated strong performance in predicting U_x with an RMSE of 0.8147. Nonetheless, outside these instances, MHP consistently displayed better predictive accuracy for various flow fields, despite somewhat higher errors in predicting pressure (P). Moving on to Table 13 showing R^2 score, MAE, MSE and RMSE for aerodynamic coefficients prediction, MHP performs better in terms of higher R^2 score and lower MAE, MAE and RMSE for predicting C_L and C_D .

Based on the excellent performance of the MHP network, it is selected for further comparison with computational fluid dynamics (CFD) results. Figure 8 visually compares the CFD and MHP results for the NACA 0015—wedge at 0.1c airfoil. The comparison reveals significantly lower residual errors between the two, and the predicted contour patterns of the flow fields show good agreement with the CFD results. Figure 8c shows that the residual error range between MHP- P and CFD- P is –150 to 67. Histograms of the residual errors, shown in Fig. 9, provide a better understanding of the errors. In Fig. 9a, about 2800 error data are distributed around 0 for P , taking threshold of 0.5 that means going –0.5 and 0.5 around 0. The high residual errors in P can be attributed to the sparsity of the training data for this parameter. However, the residual errors for other flow fields (U , U_x , U_y) are predominantly centered around 0. Figure 8f shows that the residual error range between MHP- U and CFD- U is –1.8 to 66.8. In Fig. 9b there are about 34,000 error data distributed in the numerical range of 0, with 0.5 threshold. Figure 8i shows that the residual error range between MHP- U_x and CFD- U_x is –3.0 to 1.8. In Fig. 9c there are about 34,500 error data distributed in the numerical range of 0 again with threshold 0.5. Figure 8l shows that the residual error range between MHP- U_y and CFD- U_y is –2.0 to 4.5. In Fig. 9d there are about 35,500 error data distributed in the numerical range of 0 with threshold 0.5. Remarkably, MHP outperforms all other ML models in predicting the lift coefficient (C_L) and drag coefficient (C_D). For the lift coefficient (C_L) the other three ML models are close to the performance of MLP network, but for the drag coefficient (C_D) the ML models are



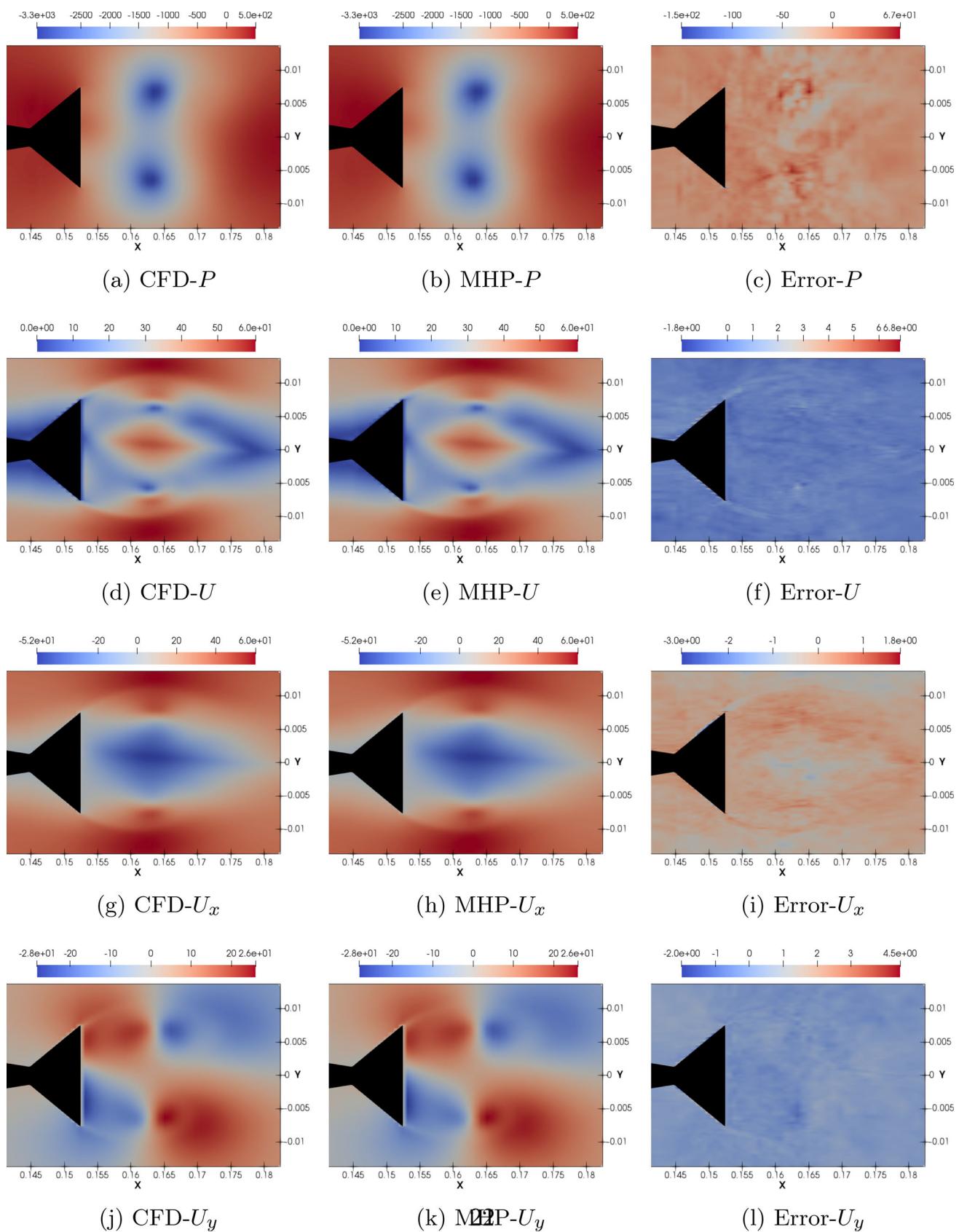


Fig. 8 Comparison of MHP and CFD results for the test case of a NACA 0015—wedge at $0.1c$ airfoil, operating at a Mach number (M) of 0.12 and an angle of attack (α) of -3°



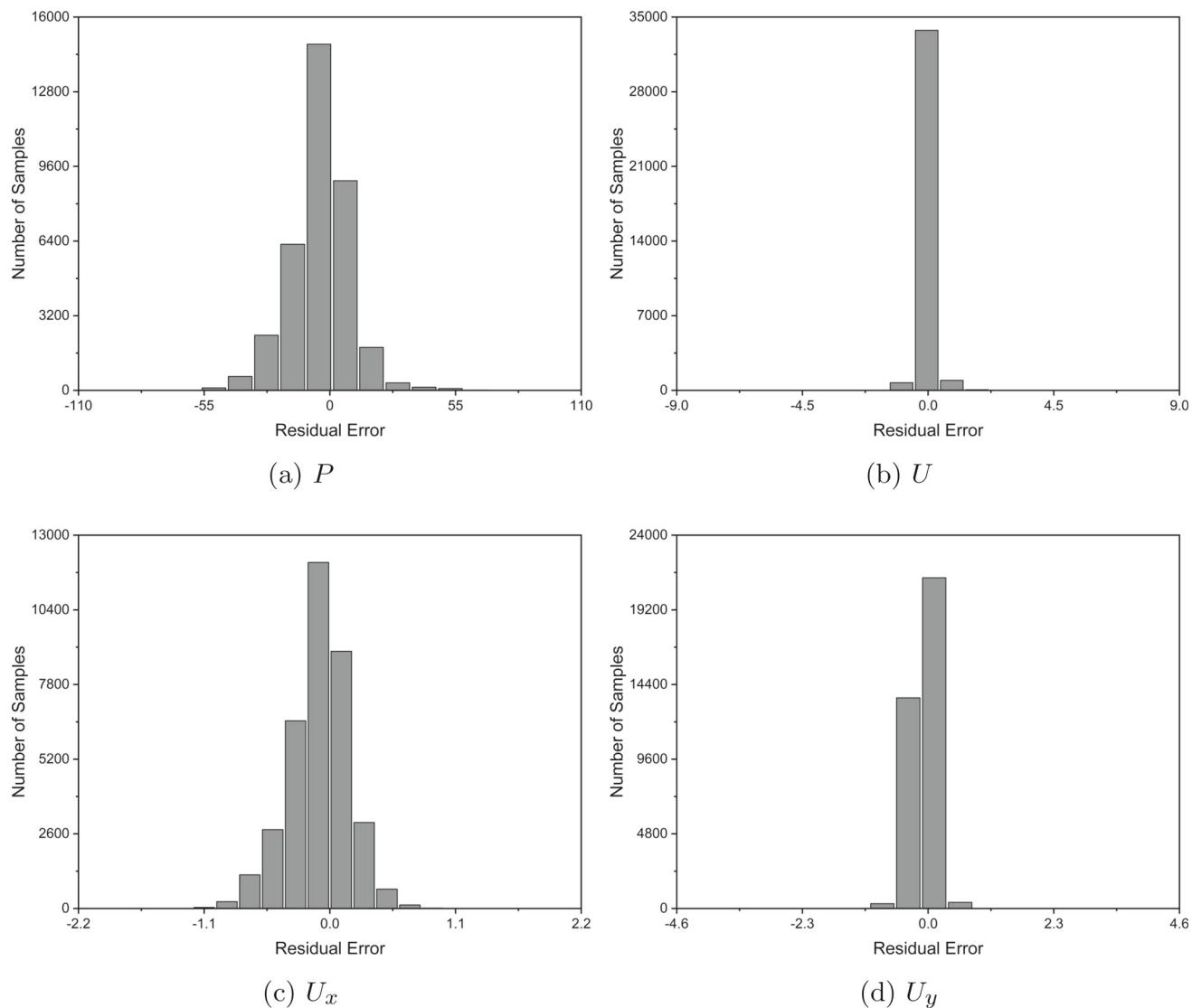


Fig. 9 Residual error histograms between CFD and MHP for the test case of a NACA 0015—wedge at $0.1c$ airfoil, operating at a Mach number (M) of 0.12 and an angle of attack (α) of -3°

way behind, this is due the fact of the magnitude of the C_D values. on the other hand C_L values are straight forwardly linear, that is why the ML models did not face issue while generalizing. Hence, MHP is selected for comparison with CFD results. Figure 10 illustrates the comparison between MHP and CFD results for the prediction of C_L and C_D for NACA 0015—wedge at $0.1c$ airfoil. MHP demonstrates satisfactory performance in predicting C_L , but there is room for improvement in predicting C_D . The other three test airfoil cases are provided in the “Appendix” section. where MHP did better in predicting C_L , but for the C_D the MHP faced some challenges, due to the magnitude of the C_D values.

Further comparisons between CFD and MHP results for the other three test airfoil cases are provided in the “Appendix” section. These comparisons also reveal signif-

icantly lower residual errors between the CFD and MHP network, and the predicted contour patterns of the flow fields show good agreement with the CFD results. Histograms of the residual errors are included to facilitate understanding of these errors. Although the MHP network encountered challenges in predicting the pressure field (P), most residual error values are centered around 0. On the other hand, the residual errors for other flow fields (U , U_x , U_y) are predominantly centered around 0 for the other three test airfoil cases.

From the presented results, it is observed that the trained model shows a good performance in the prediction of the flow fields and aerodynamic coefficients. The computational times needed by the network training and prediction are shown in Table 14, along with the time required for CFD simulations. Compared to the conventional CFD methods, the current

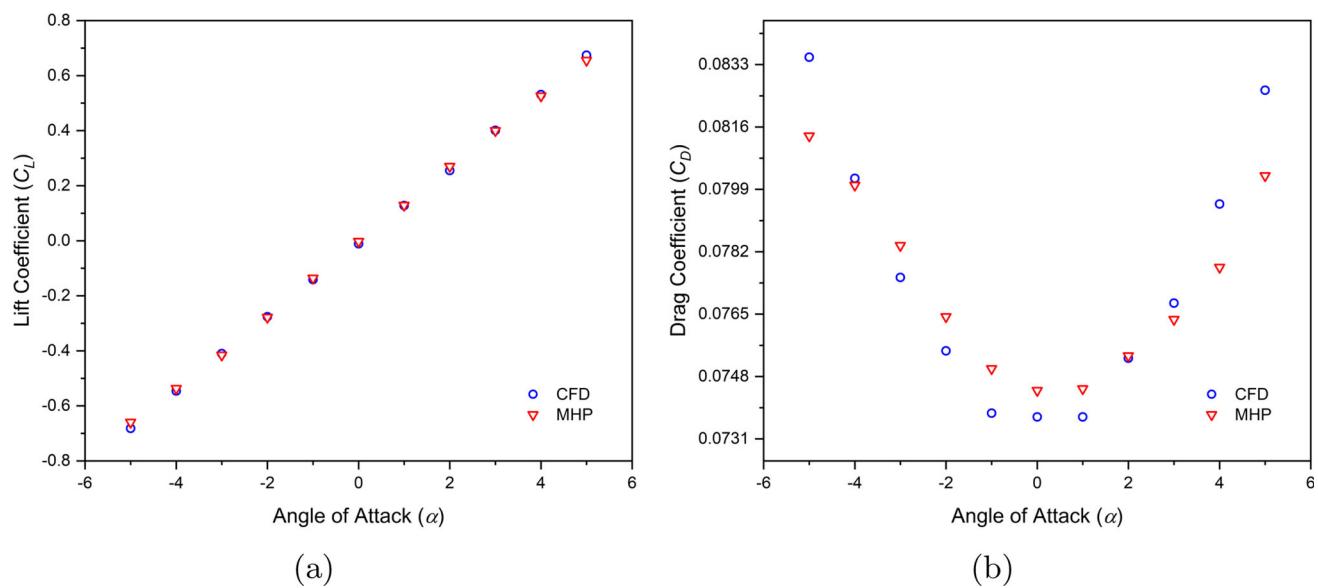


Fig. 10 Comparison between CFD and MHP prediction of **a** lift coefficient (C_L) and **b** drag coefficient (C_D) with variation of angle of attack (α) at a Mach number (M) of 0.12 for the NACA 0015—wedge at 0.1c airfoil section

Table 14 Comparison of wall time requirement

Operation	Wall time (Intel Core i7 @ 2.90GHz)
Training: prediction network (MHP)	40 h
Prediction time per airfoil case	1.39×10^{-3} h (5 s)
Ansys simulation time per case (average)	0.17 h (625 s)
Prediction speed up	125 times

approach is much faster and takes only a few seconds to predict the flow fields near the given airfoil and aerodynamic coefficients once trained, which is 125 times faster than the conventional CFD approach.

8 Conclusions

The objective of this study was to speed up the prediction of flow fields, including pressure (P), velocity magnitude (U), velocity components in the x -direction (U_x) and y -direction (U_y), as well as specific aerodynamic coefficients such as lift coefficient (C_L) and drag coefficient (C_D) for wedge tail airfoils. Predictions were made using an MHP network and classical machine learning (ML) algorithms, namely DT, KNN, and RF. Input parameters consisted of airfoil sections (A), x – y grid coordinates (X , Y), Mach number (M), and angle of attack (α). The evaluation and comparison of the MHP network and ML models based on the R^2 score, MAE, MSE and RMSE indicated that the MHP network outperformed the ML models in predicting flow fields

and aerodynamic coefficients. In addition, a comparison was made between the MHP network and CFD methods, revealing a good agreement between the two approaches. This favorable outcome can be attributed to using a point-by-point prediction approach, known for its improved accuracy in the nearby airfoil region, instead of an image-to-image prediction method.

Consequently, it can be inferred that the proposed method successfully facilitates the mapping of input features, such as airfoil sections, coordinates, Mach number, and angle of attack, to the desired output features of flow fields (P , U , U_x , U_y) and aerodynamic coefficients (C_L , C_D) for wedge tail airfoils. Since this study represents the first investigation of wedge tail airfoils, it is a foundational reference for future research endeavors involving the prediction of flow fields and aerodynamic coefficients specific to this airfoil configuration. Subsequent work will explore applying a convolutional neural network (CNN) to parameterize wedge-tail airfoils based on input airfoil images while also considering the extension of the study to incorporate three-dimensional wedge-tail airfoils.

Appendix A NACA 0012 wedge at 0.1c

See Figs. 11, 12 and 13.

Appendix B NACA 0018 wedge at 0.1c

See Figs. 14, 15 and 16.



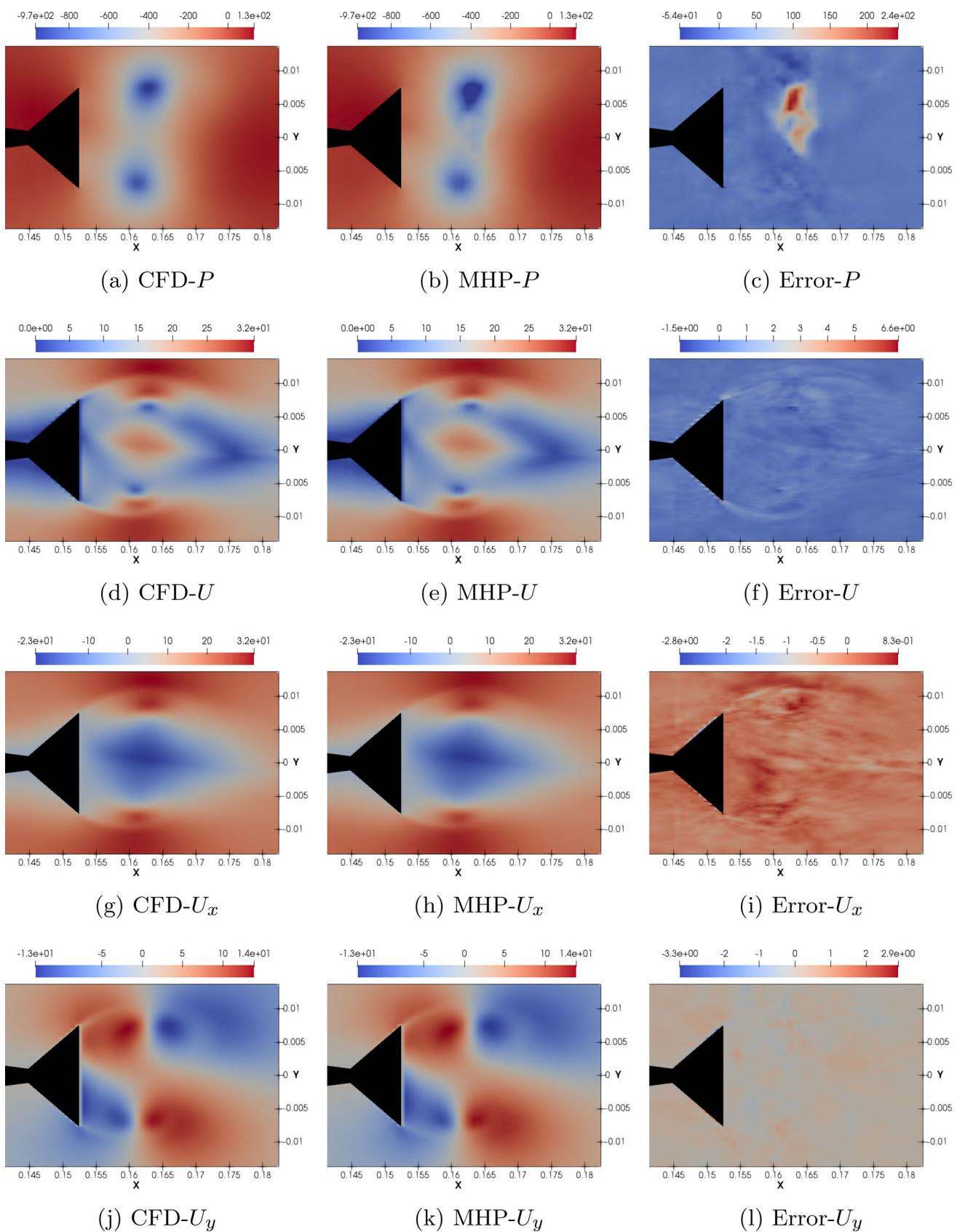


Fig. 11 Comparison of MHP and CFD results for the test case of a NACA 0012—wedge at $0.1c$ airfoil, operating at a Mach number (M) of 0.06 and an angle of attack (α) of -5°



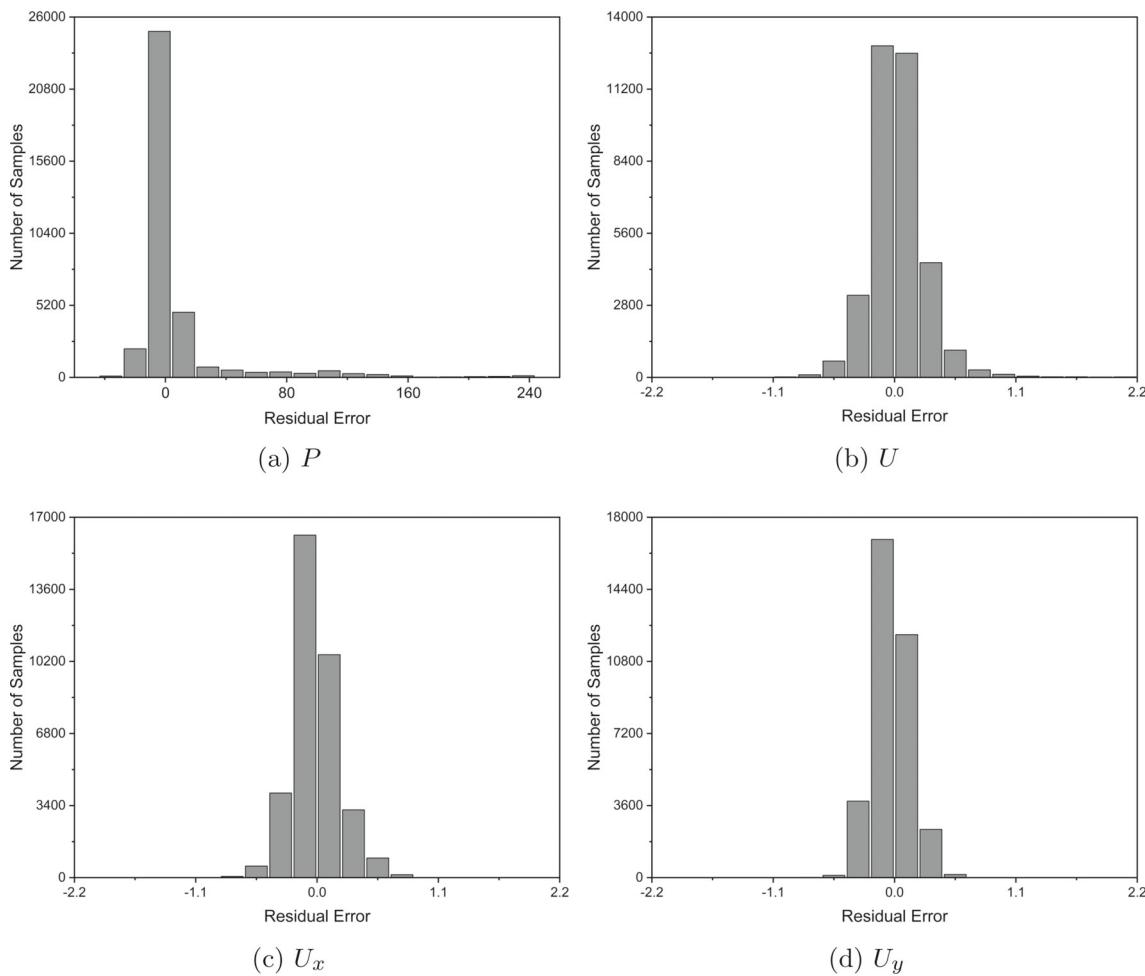


Fig. 12 Residual error histograms between CFD and MHP for the test case of a NACA 0012—wedge at $0.1c$ airfoil, operating at a Mach number (M) of 0.06 and an angle of attack (α) of -5°

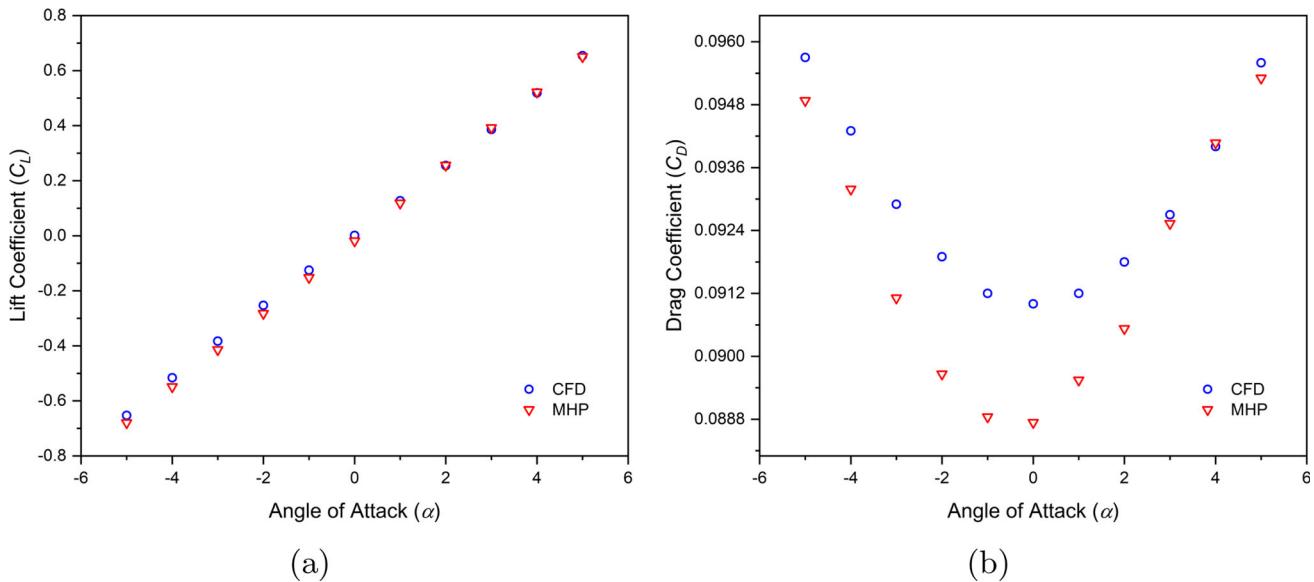


Fig. 13 Comparison between CFD and MHP prediction of **a** lift coefficient (C_L) and **b** drag coefficient (C_D) with variation of angle of attack (α) at a Mach number (M) of 0.06 for a NACA 0012—wedge at $0.1c$ airfoil



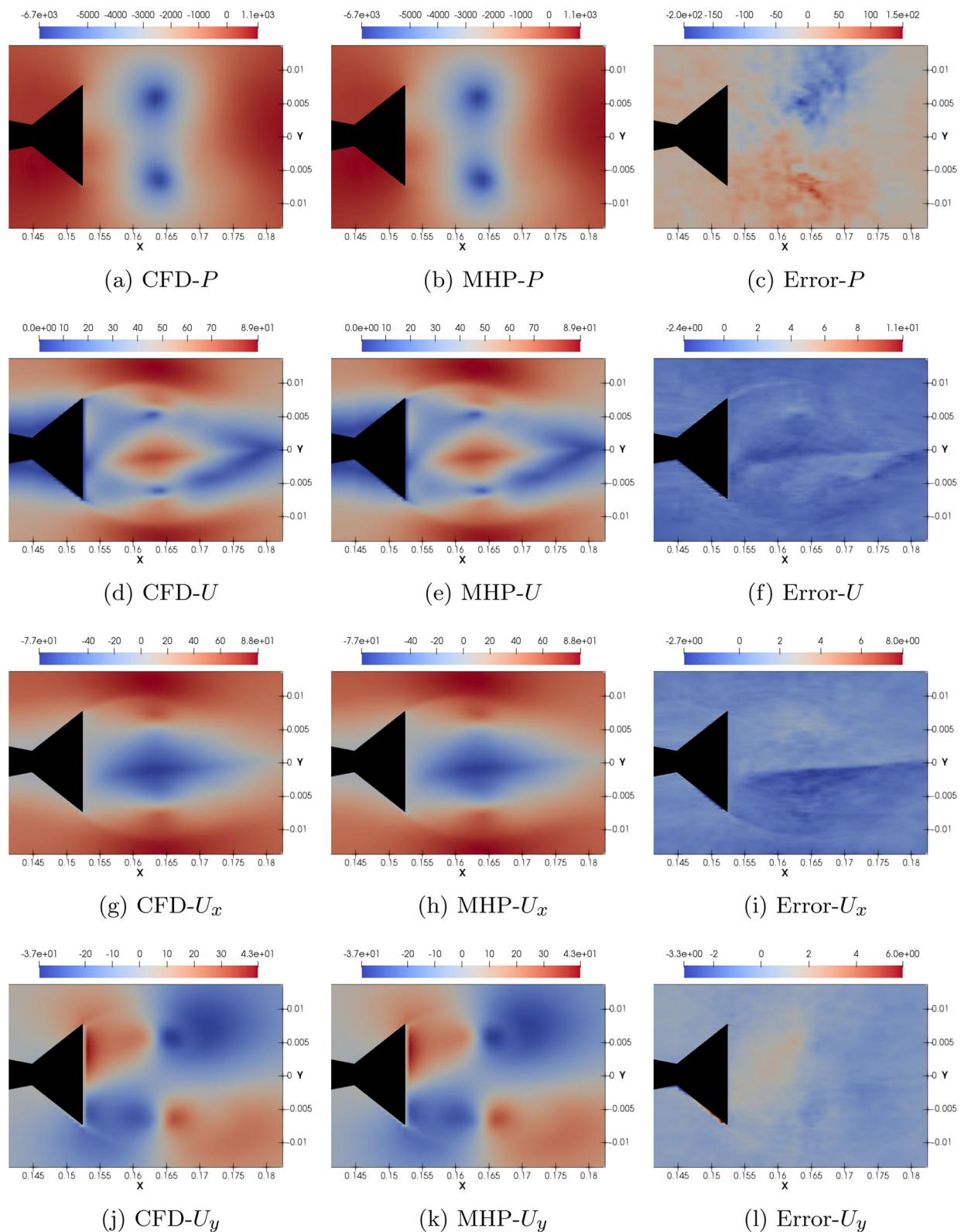


Fig. 14 Comparison of MHP and CFD results for the test case of a NACA 0018—wedge at $0.1c$ airfoil, operating at a Mach number (M) of 0.18 and an angle of attack (α) of 3°

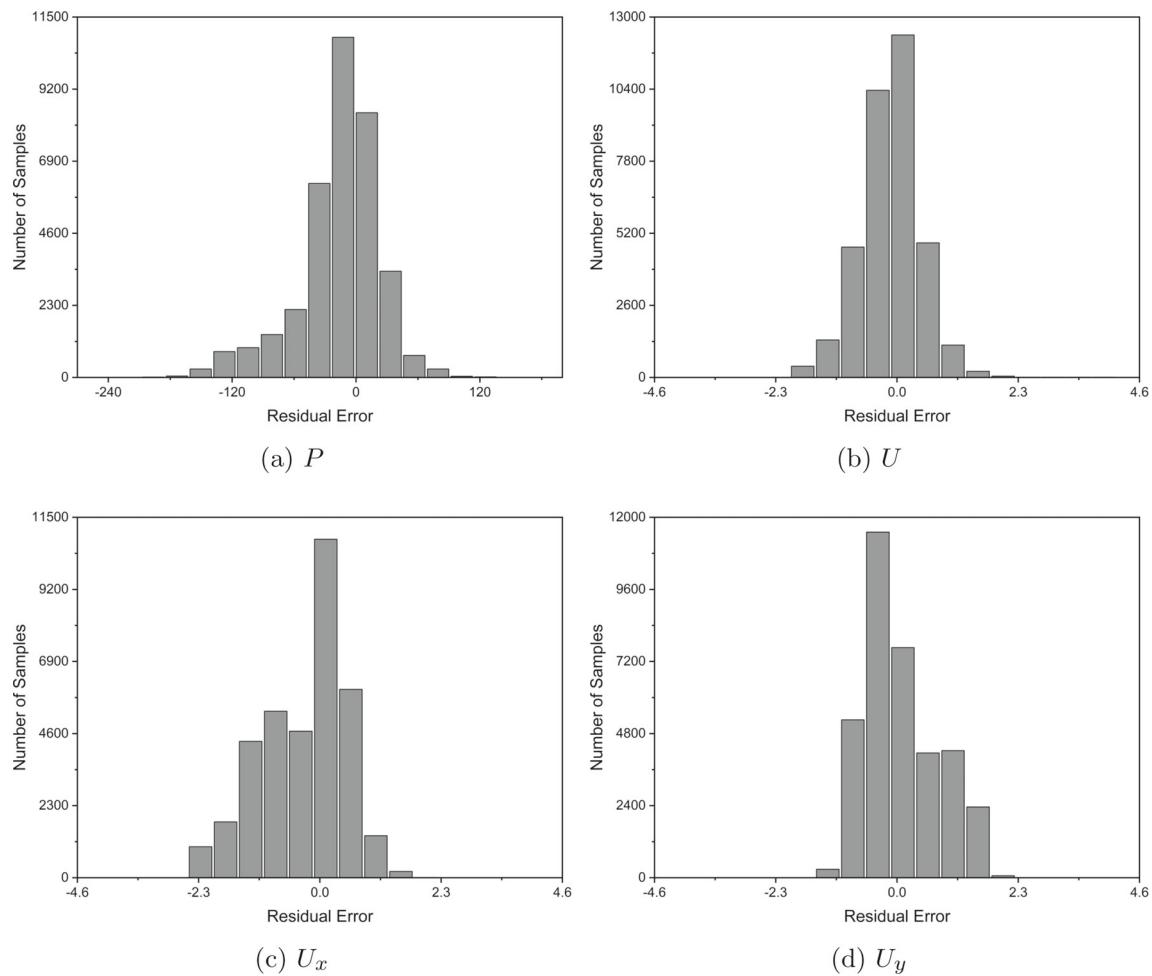


Fig. 15 Residual error histograms between CFD and MHP for the test case of a NACA 0018—wedge at 0.1c airfoil, operating at a Mach number (M) of 0.18 and an angle of attack (α) of 3°

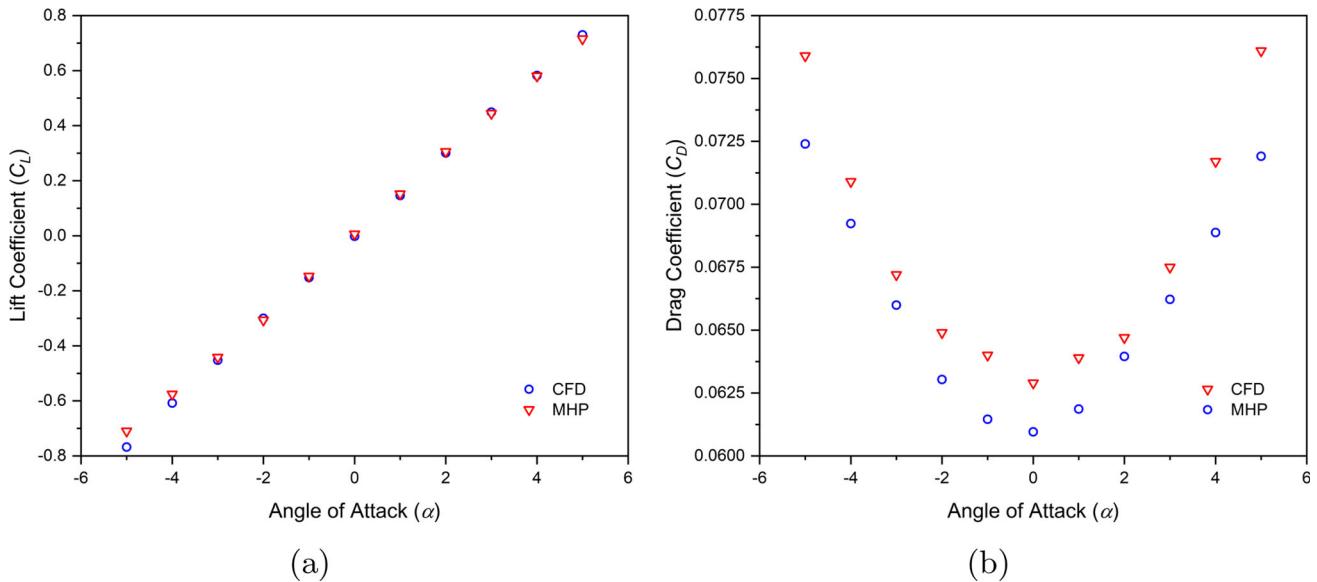


Fig. 16 Comparison between CFD and MHP prediction of **a** lift coefficient (C_L) and **b** drag coefficient (C_D) with variation of angle of attack (α) at a Mach number (M) of 0.18 for a NACA 0018—wedge at 0.1c airfoil section



Appendix C NACA 0021 wedge at 0.1c

See Figs. 17, 18 and 19.

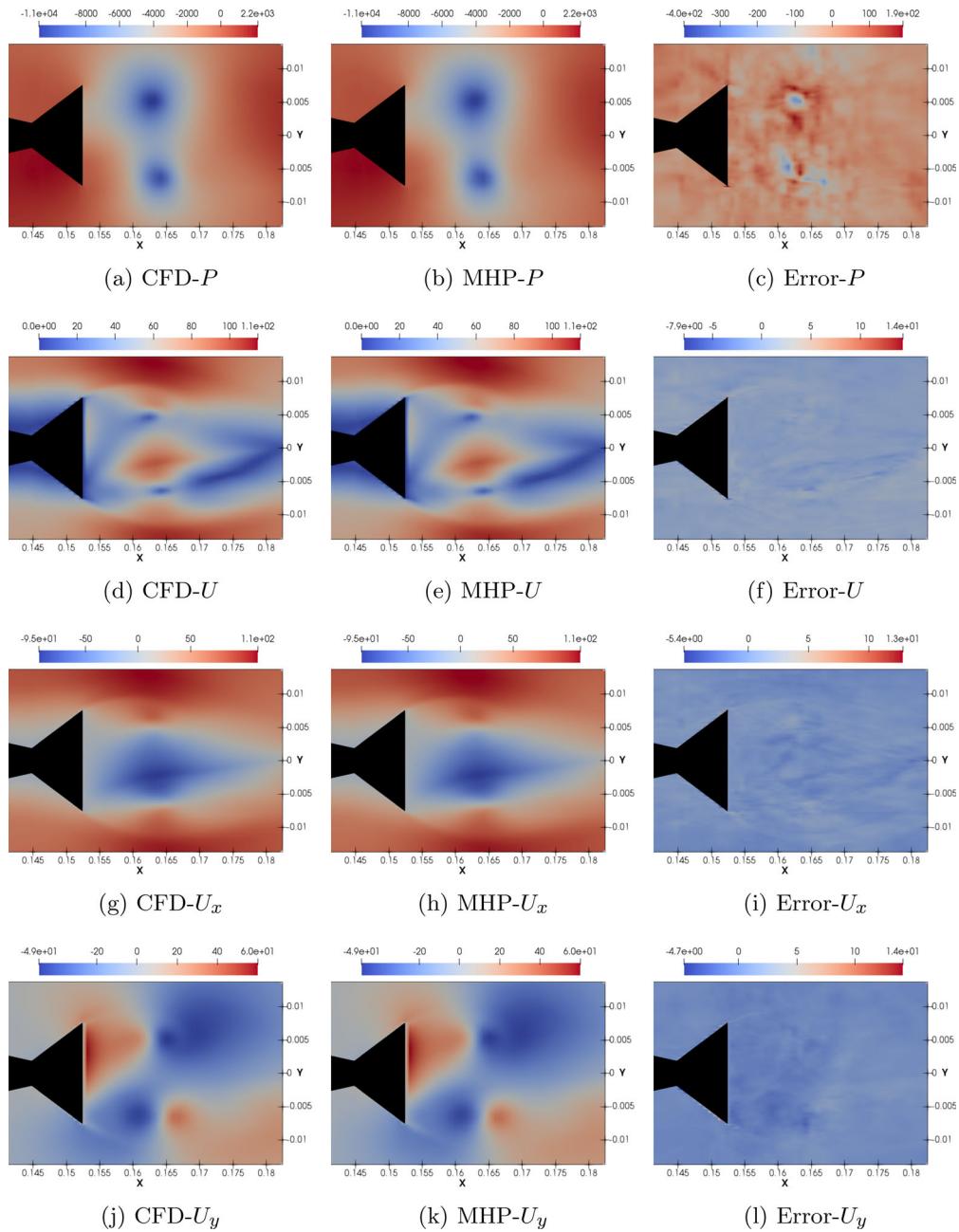


Fig. 17 Comparison of MHP and CFD results for the test case of a NACA 0021—wedge at 0.1c airfoil, operating at a Mach number (M) of 0.24 and an angle of attack (α) of 5°



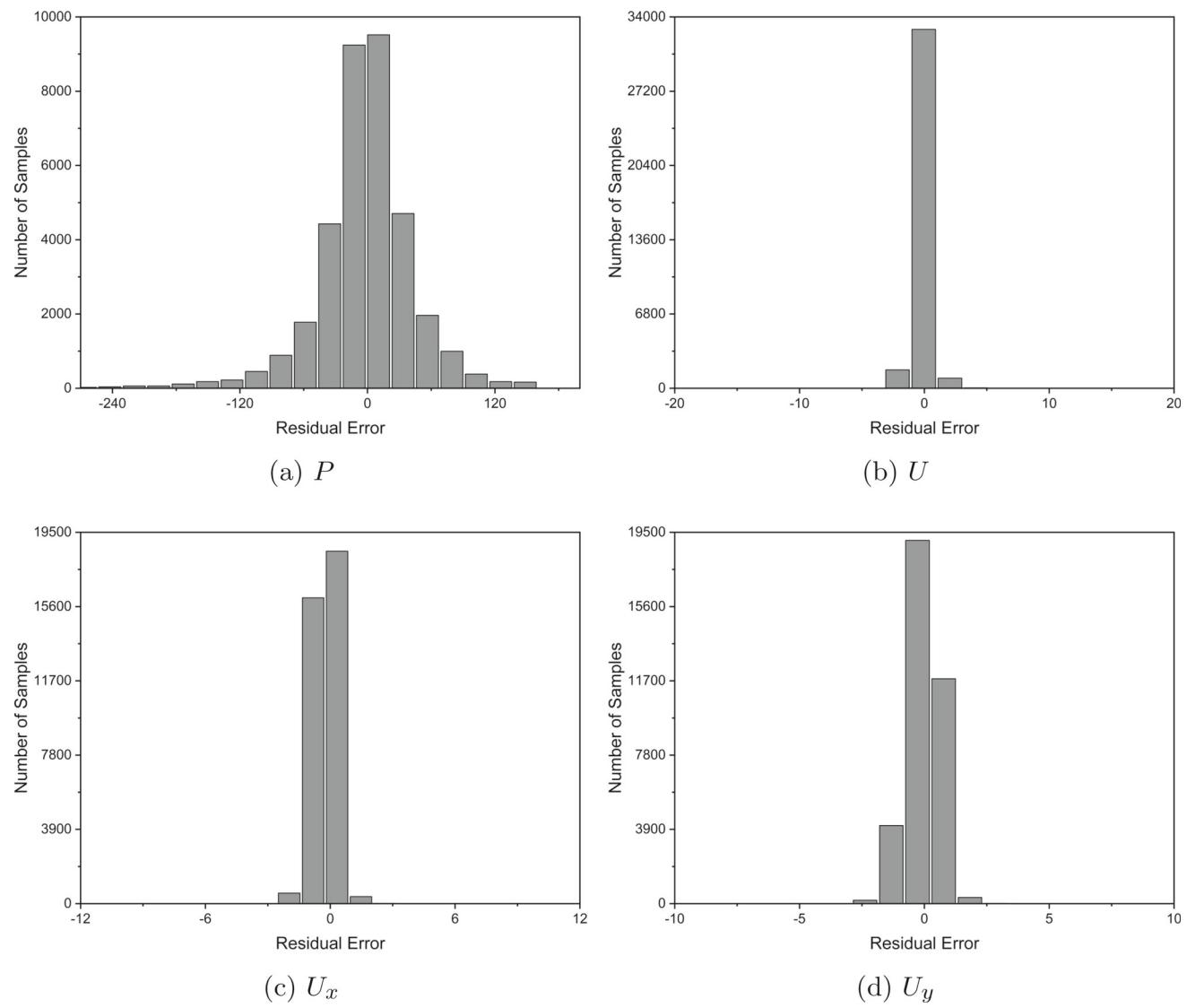


Fig. 18 Residual error histograms between CFD and MHP for the test case of a NACA 0021—wedge at $0.1c$ airfoil, operating at a Mach number (M) of 0.24 and an angle of attack (α) of 5°



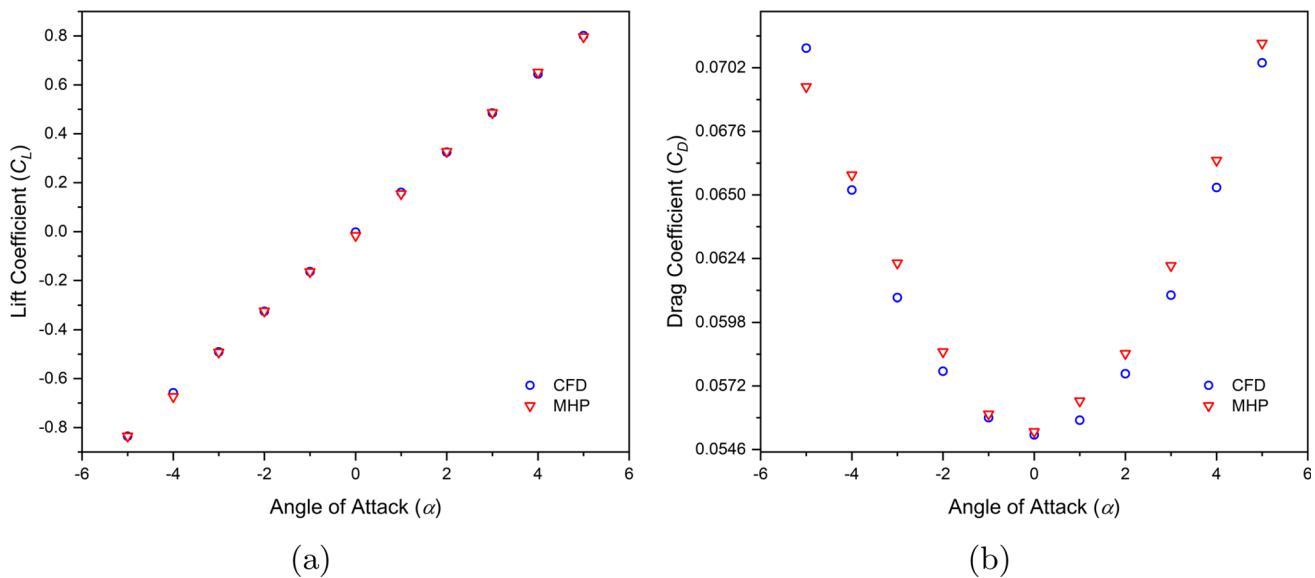


Fig. 19 Comparison between CFD and MHP prediction of **a** lift coefficient (C_L) and **b** drag coefficient (C_D) with variation of angle of attack (α) at a Mach number (M) of 0.24 for a NACA 0021—wedge at 0.1c airfoil section

References

- Song, C.; Wu, G.; Zhu, W.; Zhang, X.: Study on aerodynamic characteristics of Darrieus vertical axis wind turbines with different airfoil maximum thicknesses through computational fluid dynamics. *Arab. J. Sci. Eng.* **45**, 689–698 (2020). <https://doi.org/10.1007/s13369-019-04127-8>
- Esmaili, A.; Sousa, J.: Flow-driven piezoelectric energy harvester on a full-span wing for micro-aerial-vehicle (MAV) application. *Arab. J. Sci. Eng.* **45**, 5713–5728 (2020). <https://doi.org/10.1007/s13369-020-04526-2>
- Khan, M.A.; Padhy, C.: Aerodynamic characterization of bio-mimicked pleated dragonfly aerofoil. *Int. J. Aviat. Aeronaut. Aerosp.* **8**(2), 3 (2021). <https://doi.org/10.15394/ijaaa.2021.1562>
- Vigneswaran, C.; Kumar, G.C. V.: Aerodynamic performance analysis of co-flow jet airfoil. *Int. J. Aviat. Aeronaut. Aerosp.* **8**(1), 10 (2021). <https://doi.org/10.15394/ijaaa.2021.1555>
- Tuncer, I.H.; Platzer, M.F.: Computational study of flapping airfoil aerodynamics. *J. Aircr.* **37**(3), 514–520 (2000). <https://doi.org/10.2514/2.2628>
- Guo, Cy.; Zhang, Z.-t.; Cao, X.-x.; Wu, T.-c.; Su, Y.-m.: Numerical and experimental studies of hydrodynamic performance of bionic leading-edge tubercle airfoil. *J. Hydrodyn.* **31**(6), 1240–1249 (2019). <https://doi.org/10.1007/s42241-019-0068-3>
- Karim, M.M.; Prasad, B.; Rahman, N.: Numerical simulation of free surface water wave for the flow around NACA 0015 hydrofoil using the volume of fluid (VOF) method. *Ocean Eng.* **78**, 89–94 (2014). <https://doi.org/10.1016/j.oceaneng.2013.12.013>
- Sener, M.Z.; Aksu, E.: The numerical investigation of the rotation speed and Reynolds number variations of a NACA 0012 airfoil. *Ocean Eng.* **249**, 110899 (2022). <https://doi.org/10.1016/j.oceaneng.2022.110899>
- Chapter 8—manoeuvring. In: Molland, A.F. (ed.) *The Maritime Engineering Reference Book*, pp. 578–635. Butterworth-Heinemann, Oxford (2008). <https://doi.org/10.1016/B978-0-7506-8987-8.00008-1>
- Nguyen, T.; Ikeda, Y.: Hydrodynamic characteristics of high lift rudders with wedge tails. *Proc. AP Hydro* **2016**, 238–245 (2016)
- Liu, J.; Hekkenberg, R.: Hydrodynamic characteristics of twin-rudders at small attack angles. In: IMDC 2015: Proceedings of the 12th International Marine Design Conference, Tokyo, Japan, 11–14 May 2015. Citeseer (2015)
- Thieme, H.: Design of Ship Rudders (zur formgebung von schiffsrudern). Technical report, David Taylor Model Basin Washington, DC (1965)
- Tasif, T.H.; Karim, M.M.: Effect of fish wedge on the hydrodynamic characteristics of a marine rudder. *Procedia Eng.* **194**, 136–143 (2017). <https://doi.org/10.1016/j.proeng.2017.08.127>
- Guo, X.; Li, W.; Iorio, F.: Convolutional neural networks for steady flow approximation. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 481–490 (2016). <https://doi.org/10.1145/2939672.2939738>
- Sekar, V.; Jiang, Q.; Shu, C.; Khoo, B.C.: Fast flow field prediction over airfoils using deep learning approach. *Phys. Fluids* **31**(5), 057103 (2019). <https://doi.org/10.1063/1.5094943>
- Liu, W.; Wang, Z.; Liu, X.; Zeng, N.; Liu, Y.; Alsaadi, F.E.: A survey of deep neural network architectures and their applications. *Neurocomputing* **234**, 11–26 (2017). <https://doi.org/10.1016/j.neucom.2016.12.038>
- Duru, C.; Alemdar, H.; Baran, O.U.: A deep learning approach for the transonic flow field predictions around airfoils. *Comput. Fluids* **236**, 105312 (2022). <https://doi.org/10.1016/j.compfluid.2022.105312>
- Du, Q.; Liu, T.; Yang, L.; Li, L.; Zhang, D.; Xie, Y.: Airfoil design and surrogate modeling for performance prediction based on deep learning method. *Phys. Fluids* **34**(1), 015111 (2022). <https://doi.org/10.1063/5.0075784>



19. Kasmaiee, S.; Tadjfar, M.; Kasmaiee, S.: Optimization of blowing jet performance on wind turbine airfoil under dynamic stall conditions using active machine learning and computational intelligence. *Arab. J. Sci. Eng.* (2023). <https://doi.org/10.1007/s13369-023-07892-9>
20. Haizhou, W.; Xuejun, L.; Wei, A.; Hongqiang, L.: A generative deep learning framework for airfoil flow field prediction with sparse data. *Chin. J. Aeronaut.* **35**(1), 470–484 (2022). <https://doi.org/10.1016/j.cja.2021.02.012>
21. Thuerer, N.; Weißenow, K.; Prantl, L.; Hu, X.: Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows. *AIAA J.* **58**(1), 25–36 (2020). <https://doi.org/10.2514/1.J058291>
22. Chen, D.; Gao, X.; Xu, C.; Chen, S.; Fang, J.; Wang, Z.; Wang, Z.: Flowgan: a conditional generative adversarial network for flow prediction in various conditions. In: 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), pp. 315–322. IEEE (2020). <https://doi.org/10.1109/ICTAI50040.2020.00057>
23. Zuo, K.; Bu, S.; Zhang, W.; Hu, J.; Ye, Z.; Yuan, X.: Fast sparse flow field prediction around airfoils via multi-head perceptron based deep learning architecture. *Aerospace Sci. Technol.* **130**, 107942 (2022). <https://doi.org/10.1016/j.ast.2022.107942>
24. Jiang, J.; Li, G.; Jiang, Y.; Zhang, L.; Deng, X.: TransCFD: a transformer-based decoder for flow field prediction. *Eng. Appl. Artif. Intell.* **123**, 106340 (2023). <https://doi.org/10.1016/j.engappai.2023.106340>
25. Zhao, P.; Gao, X.; Zhao, B.; Liu, H.; Wu, J.; Deng, Z.: Machine learning assisted prediction of airfoil lift-to-drag characteristics for mars helicopter. *Aerospace* **10**(7), 614 (2023). <https://doi.org/10.3390/aerospace10070614>
26. Ribeiro, M.D.; Stradtner, M.; Bekemeyer, P.: Unsteady reduced order model with neural networks and flight-physics-based regularization for aerodynamic applications. *Comput. Fluids* (2023). <https://doi.org/10.1016/j.compfluid.2023.105949>
27. Schwamborn, D.; Gerhold, T.; Heinrich, R.: The DLR TAU-code: recent applications in research and industry. In: ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics, Egmond aan Zee, The Netherlands, September 5–8, 2006. Delft University of Technology; European Community on Computational Methods... (2006). <https://elib.dlr.de/22421/>
28. Zhang, B.: Airfoil-based convolutional autoencoder and long short-term memory neural network for predicting coherent structures evolution around an airfoil. *Comput. Fluids* **258**, 105883 (2023). <https://doi.org/10.1016/j.compfluid.2023.105883>
29. Song, H.-S.; Mugabi, J.; Jeong, J.-H.: Pix2pix and deep neural network-based deep learning technology for predicting vortical flow fields and aerodynamic performance of airfoils. *Appl. Sci.* **13**(2), 1019 (2023). <https://doi.org/10.3390/app13021019>
30. Wang, Z.; Liu, X.; Yu, J.; Wu, H.; Lyu, H.: A general deep transfer learning framework for predicting the flow field of airfoils with small data. *Comput. Fluids* **251**, 105738 (2023). <https://doi.org/10.1016/j.compfluid.2022.105738>
31. Kasmaiee, S.; Tadjfar, M.; Kasmaiee, S.: Machine learning-based optimization of a pitching airfoil performance in dynamic stall conditions using a suction controller. *Phys. Fluids* (2023). <https://doi.org/10.1063/5.0164437>
32. Sheldahl, R.E.; Klimas, P.C.: Aerodynamic characteristics of seven symmetrical airfoil sections through 180-degree angle of attack for use in aerodynamic analysis of vertical axis wind turbines. Technical Report, Sandia National Labs. Albuquerque, NM (USA) (1981). <https://doi.org/10.2172/6548367>
33. Douvi, C.E.; Tsavalos, I.A.; Margaris, P.D.: Evaluation of the turbulence models for the simulation of the flow over a national advisory committee for aeronautics (NACA) 0012 airfoil. *J. Mech. Eng. Res.* **4**(3), 100–111 (2012). <https://doi.org/10.5897/JMER11.074>
34. Spalart, P.; Allmaras, S.: A one-equation turbulence model for aerodynamic flows. In: 30th Aerospace Sciences Meeting and Exhibit, p. 439 (1992). <https://doi.org/10.2514/6.1992-439>
35. Versteeg, H.K.; Malalasekera, W.: An Introduction to Computational Fluid Dynamics: The Finite Method. Pearson education, London (2007)
36. Suvanjumrat, C.: Comparison of turbulence models for flow past naca0015 airfoil using openfoam. *Eng. J.* **21**, 207–221 (2017). <https://doi.org/10.4186/ej.2017.21.3.207>
37. Kingma, D.P.; Ba, J.: Adam: A Method for Stochastic Optimization (2014). [arXiv:1412.6980](https://doi.org/10.48550/arXiv.1412.6980). <https://doi.org/10.48550/arXiv.1412.6980>
38. Agarap, A.F.: Deep Learning Using Rectified Linear Units (RELU) (2018). [arXiv:1803.08375](https://doi.org/10.48550/arXiv.1803.08375); <https://doi.org/10.48550/arXiv.1803.08375>
39. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E.: Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* **12**(85), 2825–2830 (2011)
40. Izenman, A.J.: Modern Multivariate Statistical Techniques, vol. 1. Springer, Berlin (2008). <https://doi.org/10.1007/978-0-387-78189-1>
41. Tran, M.-K.; Panchal, S.; Chauhan, V.; Brahmabhatt, N.; Mevawalla, A.; Fraser, R.; Fowler, M.: Python-based scikit-learn machine learning models for thermal and electrical performance prediction of high-capacity lithium-ion battery. *Int. J. Energy Res.* **46**(2), 786–794 (2022). <https://doi.org/10.1002/er.7202>

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

