Aerospace Science and Technology 168 (2026) 111102

Contents lists available at ScienceDirect

Aerospace Science and Technology

journal homepage: www.elsevier.com/locate/aescte

Original article

AI-Driven surrogate modeling for iced tailplane aerodynamic prediction

Ciro Cuozzo [a,b], Vincenzo Cusati [a], Agostino De Marco [a], Salvatore Corcione [a,*], James H. Page [b], Alessandro Zanon [b]

[a] University of Naples Federico II, Industrial Engineering Department, via Claudio, 21, Napoli, 80125, Italy
[b] AIT Austrian Institute of Technology GmbH, Giefinggasse 4, Vienna, 1210, Austria

ARTICLE INFO

Communicated by Jérôme Morio

Keywords:
Ice accretion
Deep learning
Convolutional neural networks
Aerodynamics

ABSTRACT

Aircraft tailplane icing poses a significant challenge to flight safety and aerodynamic efficiency, particularly under holding or approach conditions. While high-fidelity numerical tools exist for simulating ice accretion and its aerodynamic effects, their computational cost is prohibitive for integration into design workflows such as multidisciplinary design optimization (MDO). This work introduces a hybrid surrogate modeling framework that combines physics-based modeling with machine learning techniques to enable rapid aerodynamic assessment of iced configurations. An in-house dataset of 2D iced airfoils is generated using a numerical icing code and RANS-based CFD simulations. Neural networks, namely Multi-Layer Perceptrons (MLPs) for iced airfoil geometry prediction and Convolutional Neural Networks (CNNs) with Signed Distance Fields (SDFs) for lift prediction, are trained to replace expensive CFD computations. These models are then integrated into a quasi-3D nonlinear Vortex Lattice Method (VLM), allowing spanwise interpolation and global aerodynamic prediction for clean and iced tailplane geometries. The proposed framework achieves a three to four order-of-magnitude reduction in computational cost while preserving key aerodynamic trends with acceptable accuracy. A comparative study against full RANS simulations demonstrates good agreement, indicating the method's suitability for use in early phase MDO of ice tolerant tailplanes.

1. Introduction

Inflight icing remains a critical weather-related hazard to aviation. Between 1989 and early 1997, the National Transportation Safety Board (NTSB) identified in-flight icing as a contributing or causal factor in approximately 11 % of all weather-related general aviation accidents [1]. From 1982 to 2000, 583 icing-related incidents resulted in over 800 fatalities [2], with two-thirds of fatal cases linked to ice accretion on the airframe during flight [3].

Ice accumulation on the wing disrupts the aerodynamics, promoting earlier boundary layer transition and separation at lower angles of attack. This degradation typically results in a reduced lift curve slope, reduced maximum absolute lift coefficient, and increased drag. Ice accreting near the leading edge shifts the center of pressure forward, though the accompanying loss in lift often precludes a corresponding increase in nose-up pitching moment [4,5]. Notably, in the presence of deflected ailerons, icing can diminish the differential lift between wings, potentially impairing roll control or even leading to its complete loss [6].

While large aircraft wings are generally equipped with de-icing or anti-icing systems, these systems are seldom implemented on tailplanes due to constraints related to weight, complexity, and cost. A common mitigation strategy involves oversizing the tailplane to ensure tolerance to aerodynamic degradation from ice. However, this approach inevitably increases structural weight and parasitic drag, thereby raising fuel consumption across all phases of flight. CS-25.143 [7] mandate that aircraft must remain controllable in all flight phases even under severe icing. Moreover, prior to activating ice protection systems, the aircraft must be capable of executing a 1.5 g pull-up maneuver under worst-case icing scenarios [7]. These requirements highlight the need to accurately assess aerodynamic degradation at high angles of attack caused by ice accretion.

Numerical simulation of ice accretion typically involves four sequential calculations: (1) flowfield computation, (2) droplet collection efficiency estimation, (3) runback and ice growth, and (4) remeshing.

Early tools like LEWICE [8] employed full potential methods for rapid, low-cost analyses, while its successor LEWICE-E [9] introduced Euler-based solvers for improved physical fidelity, albeit with greater computational demands. Later advancements such as CANICE2D-NS [10], FENSAP-ICE [11], and I^2CE [12] coupled RANS or Eulerian solvers with thermodynamic icing models, often requiring multistep or predictor-corrector strategies to track evolving ice shapes. While these approaches enhanced model accuracy, they also increased simulation

* Corresponding author.
E-mail address: salvatore.corcione@unina.it (S. Corcione).

https://doi.org/10.1016/j.ast.2025.111102
Received 9 July 2025; Received in revised form 12 September 2025; Accepted 14 October 2025
Available online 25 October 2025

time and meshing complexity. More recent solvers, including the NSMB-based framework by Peña et al.[13], and the OpenFOAM-based solver by Li and Paoli [14], leverage fully coupled 3D solvers, mesh morphing, and Eulerian droplet models. Similarly, LEWICE3D [15] and the two-phase flow approach of Cao et al. [16] integrate aerodynamic and thermodynamic solvers within unified environments.

Despite their sophistication, these high-fidelity tools remain impractical for integration into iterative design loops such as those in Multidisciplinary Design Optimization (MDO), due to prohibitive computational cost, extensive preprocessing requirements, and slow turnaround time. A recent study by Page et al. [17] demonstrated the computational burden of simulating a 45-minute in-flight icing event under FAR 25 Appendix C using FENSAP-ICE. Their high-fidelity 3D simulation on a jet transport aircraft tailplane required one week for ice accretion modeling and an additional week for the aerodynamic evaluation. Such intensive demands highlight the impracticality of embedding traditional CFD approaches within MDO loops, where rapid iteration is essential.

To address this bottleneck, considerable attention has shifted toward data-driven alternatives that eliminate the need for full CFD simulations at each design point. Machine learning models, surrogate techniques, and reduced-order frameworks offer promising avenues for estimating ice growth and its aerodynamic impact efficiently. These methods leverage precomputed data to deliver rapid predictions, enabling real-time evaluations during optimization. Notably, Corcione et al. [18,19] employed radial basis function (RBF)-based surrogates to model both aeroelastic effects and aerodynamic behavior of advanced tailplane configurations, exemplifying the integration of such techniques into modern design workflows.

Recent advances in data-driven approaches have shown promising potential in modeling ice accretion, offering a practical alternative to high-fidelity CFD. Jung et al. [20] applied Proper Orthogonal Decomposition (POD) to extract dominant modes from complex datasets, enabling efficient prediction of ice shapes and collection efficiency. Li et al. [21] proposed a data-driven classification of icing severity based on Extreme Gradient Boosting (XGBoost), identifying flight conditions prone to hazardous ice formation. Strijhak et al. [22] used neural networks to predict ice shapes across different airfoils, and Massegur et al. [23] introduced a local POD and convolutional autoencoder approach to reconstruct ice contours on a NACA 0012 profile.

In the context of aerodynamic prediction, machine learning has increasingly been adopted to reduce reliance on high-fidelity CFD simulations. Balla et al. [24] employed Multi-Layer Perceptrons (MLPs) to predict aerodynamic coefficients for parametrized 2D airfoils and 3D wings, demonstrating the feasibility of using lightweight neural architectures for fast aerodynamic evaluation. Liu, Li, and Lu [25] explored the use of Convolutional Neural Networks (CNNs), incorporating binarized representations of airfoil geometries combined with rotation to account for varying angles of attack. Bhatnagar et al. [26], Zhang et al. [27], and Duru et al. [28] further advanced this approach by using Signed Distance Fields (SDFs) to represent geometry, providing richer spatial information than binary masks. Guo et al. [29] systematically compared binary and SDF-based representations for flowfield prediction around airfoils, finding that SDF encodings resulted in significantly lower predictive errors. This improvement was attributed to the ability of SDFs to capture global geometric features more effectively, reducing the need for excessively deep or complex network architectures.

Despite these developments, data-driven applications to the aerodynamics of iced lifting surfaces remain limited. Corcione et al. [30] developed a reduced-order model for iced tailplane aerodynamics, enabling fast prediction of key parameters such as lift curve slope and high-lift performance to support ice-tolerant tail design. Massegur et al. [23] analyzed ice-induced high-lift performance degradation, while Zhao et al. [31] employed Deep Operator Networks to map iced shapes to aerodynamic coefficients.

While this paper focuses on 3D aerodynamic evaluation, it builds on 2D high-fidelity data integrated through a quasi-3D methodology. Previous efforts in this direction include the modified Vortex Lattice Method (VLM) by Della Vecchia et al. [32], and the conceptual framework of Fujiwara and Bragg [33], which combined 2D solvers and empirical icing codes. Although effective, such methods typically fall short of RANS-level fidelity in modeling the degraded performance of iced sections.

This study proposes the formulation of a computational framework that substantially reduces the cost associated with the prediction of aerodynamic performance degradation under icing conditions, while preserving a level of accuracy compatible with both preliminary design and certification-oriented analyses. In contrast to conventional methodologies, which are often constrained to either prohibitively expensive three-dimensional CFD simulations or overly simplified two-dimensional analyses, the present approach exploits high-fidelity two-dimensional aerodynamic data to train machine-learning surrogates subsequently embedded within a quasi-three-dimensional solver. This enables the efficient reproduction of key three-dimensional effects-such as the spanwise variation of lift-within a computationally affordable 2.5D formulation, thus making the methodology suitable for integration into multidisciplinary design optimization environments. The proposed framework demonstrates a reduction in computational cost of several orders of magnitude compared to full RANS-based icing workflows, while maintaining discrepancies with high-fidelity references within acceptable bounds for engineering applications. This integration enables a robust yet computationally efficient simulation of iced tailplane behavior, with direct implications for tailplane sizing, control authority assessment, and compliance with FAR 25 Appendix C requirements.

2. Methodology

This study presents a predictive framework for estimating the three-dimensional lift curve of a tailplane subjected to ice accretion. The framework integrates a nonlinear Vortex Lattice Method (VLM) enhanced by high-fidelity two-dimensional data. This hybrid approach exploits the computational efficiency of the VLM while incorporating detailed aerodynamic characteristics derived from two-dimensional viscous simulations, thereby aiming to strike a balance between computational cost and predictive accuracy. The workflow is summarized in Fig. 1.

The analysis process begins by defining the geometry and flight conditions through a set of macro-level parameters, which include both design and flow field variables. The baseline airfoil adopted in this study is the symmetrical supercritical NASA SC(2)-0010 airfoil [34] parametrized with its leading-edge-radius-to-chord ratio (R_{LE}/c), and thickness-to-chord ratio (t/c). The wing planform is characterized by its root chord ($c_{\mathrm{r,w}}$), taper ratio (λ_{w}), aspect ratio ($Æ_{\mathrm{w}}$), wingspan (b_{w}), and leading-edge sweep angle ($\Lambda_{\mathrm{LE,w}}$). The flow conditions considered in the analysis are the Mach number (M_{ice}), icing angle of attack (α_{ice}),median volume diameter (MVD), liquid water content (LWC), and ambient temperature (T). The macro-level parameters of the lifting surface are formally collected in the following array:

$$\boldsymbol{x} = [x_1, \ldots, x_{12}] = \big[\, R_{\mathrm{LE}}/c,\ t/c,\ c_{\mathrm{r,w}},\ \lambda_{\mathrm{w}},\ Æ_{\mathrm{w}},\ b_{\mathrm{w}},\ \Lambda_{\mathrm{LE,w}},\ \alpha_{\mathrm{ice}},\ M_{\mathrm{ice}},\ MVD,\ LWC,\ T \,\big] \tag{1}$$

Once the design vector $\boldsymbol{x}$ is defined, the lifting surface is discretized into several user-defined cross sections, each associated with an airfoil geometry. For a spanwise discretization $\mathcal{Y}_{\mathrm{w}} = \{y_k : y_1 = 0,\ y_2,\ \ldots,\ y_{N_{\mathrm{sec}}} = \frac{1}{2}b_{\mathrm{w}}\}$ over N_{sec} sections chosen along the semi-wing, the 3D geometry is represented by a collection $\mathcal{C}_{\mathrm{w}} = \{c(y_k) : c_{\mathrm{root}} = c(0),\ c(y_2),\ \ldots,\ c(\frac{1}{2}b_{\mathrm{w}}) = \lambda_{\mathrm{w}} c_{\mathrm{root}}\}$ of chords. The ice accretion process is then simulated using a trained neural network, described in Section 2.1. This model maps the local flow parameters and global atmospheric conditions to a corresponding ice shape for each section. The output is a set of iced airfoil geometries distributed along the span of the lifting surface. These iced airfoils are subsequently analyzed by a second neural network, detailed in Section 2.2, which estimates their two-dimensional

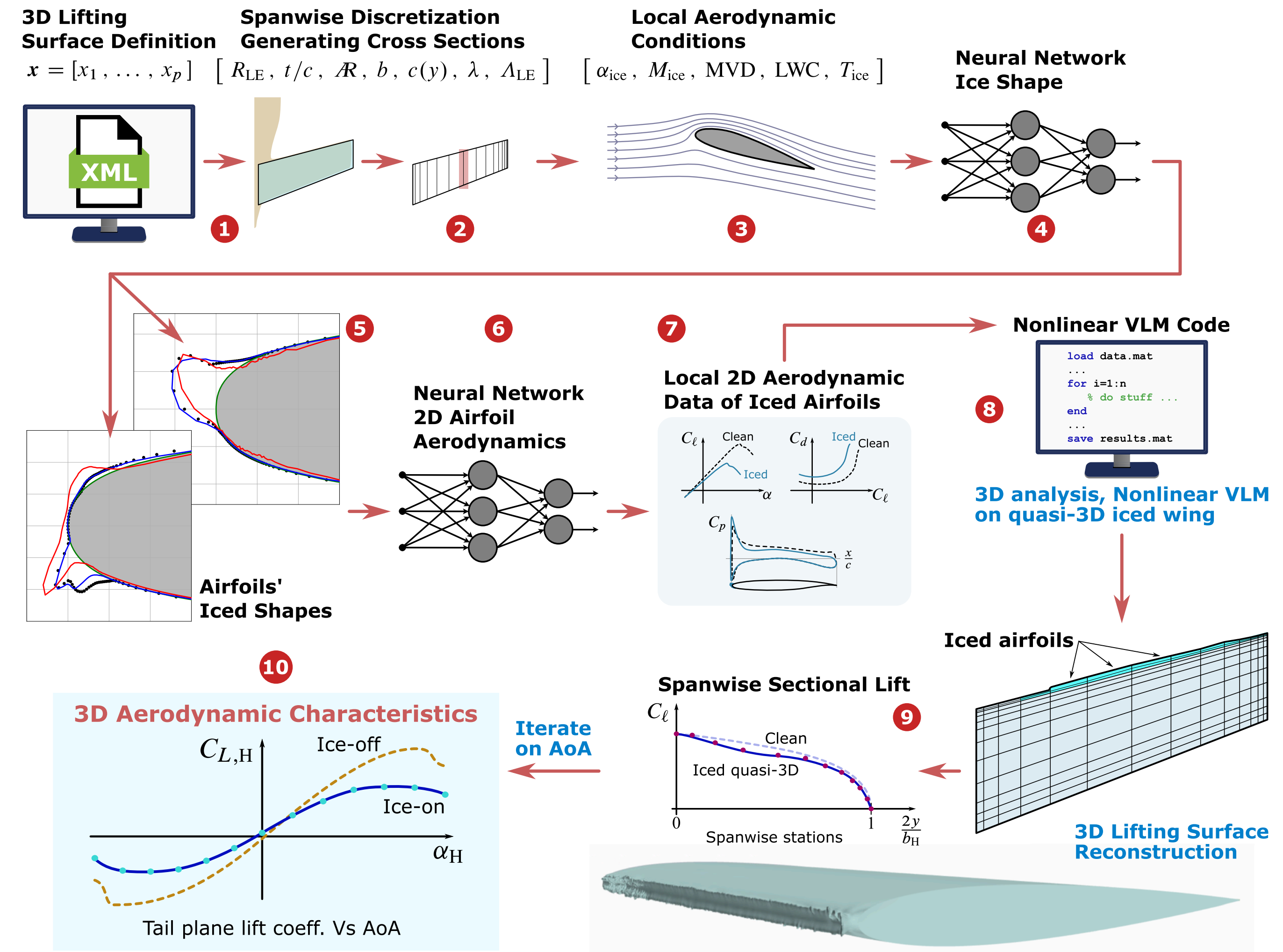


Fig. 1. Workflow for quasi-3D, iced-wing aerodynamic performance estimation. The method uses data-driven surrogate models for 2D ice accretion and 2D iced, viscous airfoil aerodynamics predictions; the latter are the inputs to a nonlinear VLM for the estimation of lifting surface aerodynamics.

Table 1
Design space for 2D icing simulations including relevant physical effects.

Variable	Description	Lower bound	Upper bound
Airfoil Geometry			
R_{LE}	Leading-edge radius	$0.8\,R_{LE,ref}$	$1.2\,R_{LE,ref}$
t/c	Thickness-to-chord ratio	8 %	15 %
c	Chord length	0.5 m	3.5 m
Flight Conditions			
M	Mach number	0.2	0.4
MVD	Median Volume Diameter of droplets	15 µm	40 µm
T	Free-stream temperature	−30 °C	0 °C
α_{ice}	Angle of attack during icing	−4°	4°
Atmospheric Reference Conditions (ISA, fixed)			
h	Altitude	5000 m	
p	Static pressure	54 000 Pa	
ρ_{air}	Air density (ISA + ideal gas)	$0.736\,\mathrm{kg\,m^{-3}}$	

aerodynamic performance over a range of angles of attack. This results in a collection of 2D aerodynamic datasets that include the lift curves of clean and iced airfoils along the span.

The final step of the process involves the computation of three-dimensional aerodynamic characteristics. The 2D data obtained from the iced sections are input to a quasi-3D aerodynamic solver based on a VLM [32], which accounts for the spanwise distribution of aerodynamic properties and returns the overall lift curve and load distribution of the iced wing.

The input space used to define the simulations incorporates both aerodynamic design parameters and environmental conditions relevant to icing. In particular, ice accretion is driven by free-stream

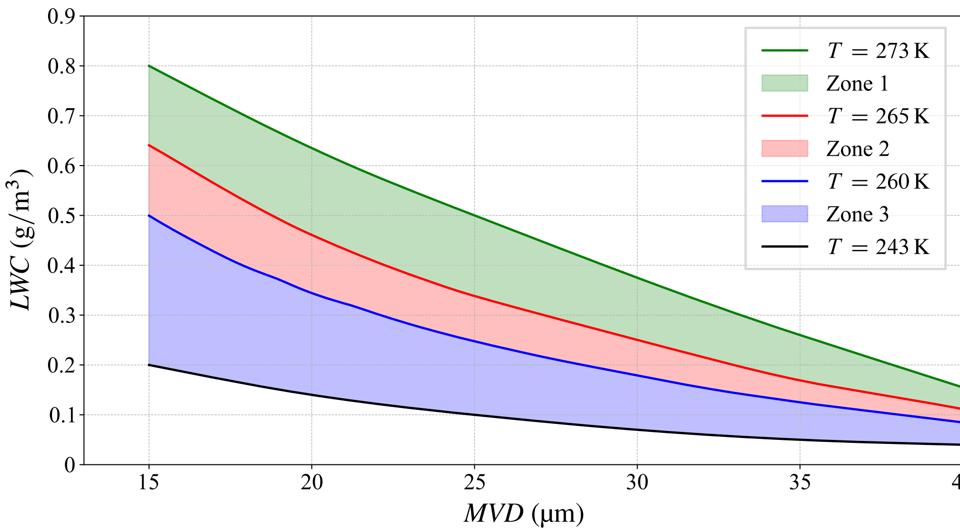


Fig. 2. Appendix C icing envelope division into three regions for three neural networks: zone 1 corresponds to glaze icing, zone 2 to mixed icing, and zone 3 to rime icing.

thermodynamic conditions, with the most influential variables being MVD, LWC, and T. These parameters govern both the collection efficiency and freezing behavior of supercooled droplets on the surface.

Droplet size determines the inertial response of particles near the stagnation region: smaller droplets (typically ≤ 15 µm) tend to follow streamlines, reducing their likelihood of impacting the surface. The runback dynamics, convective heat transfer and phase change thermodynamics decide the ice morphology. Rime ice generally forms under colder, low-LWC conditions, where droplets freeze upon impact. In contrast, glaze ice is typically observed when T is closer to freezing and LWC is high, resulting in liquid film flow and delayed freezing. These phenomena significantly affect the shape and location of accreted ice, with significant implications for the aerodynamic degradation observed in both 2D and 3D performance.

To assess the aerodynamic impact of ice accretion under conservative and operationally relevant conditions, this study adopts a 45-minute holding scenario, consistent with the continuous maximum icing envelope defined in Appendix C of 14 CFR Part 25 [35]. This scenario is recognized as one of the most critical for stability and control evaluations. In such conditions, LWC is parameterized as a function of T, MVD, and fixed exposure time of 45 min.

As a result, the design vector used to characterize each simulation becomes:

$$\boldsymbol{x} = \Big[\, \overbrace{R_{\mathrm{LE}}/c,\ t/c,\ c_{\mathrm{r,w}},\ \lambda_{\mathrm{w}},\ AR_{\mathrm{w}},\ b_{\mathrm{w}},\ \Lambda_{\mathrm{LE,w}},}^{\text{Wing Geometry}}\ \overbrace{\alpha_{\mathrm{ice}},\ M_{\mathrm{ice}},\ MVD,\ T}^{\text{Flow Conditions}} \,\Big] \tag{2}$$

To reduce computational complexity while preserving representativeness, atmospheric conditions are fixed according to the International Standard Atmosphere (ISA) at a typical holding altitude of 5000 m. Static pressure is set to 54 000 Pa, air density is calculated using the ideal gas law, water and ice density are assumed constant at 997 kg m^{-3} and 917 kg m^{-3}, respectively.

The design space thus reflects a physically informed range of variables spanning both geometry and flight conditions, enabling the simulation of ice accretion across various supercritical, symmetrical airfoil configurations. Table 1 summarises the parameter bounds adopted in the 2D icing calculations.

2.1. Data-Driven prediction of 2D ice shapes using neural networks

Accurate prediction of iced airfoil geometries is a prerequisite for evaluating their aerodynamic performance, which serves as input to the 3D aerodynamic solver. To avoid computationally expensive CFD-based accretion simulations in the final toolchain, this work employs machine learning techniques to estimate 2D ice shapes over a broad design space. Due to the complex physics of ice accretion, a wide variety of ice geometries may form under different environmental and operational conditions. To capture this variability, an empirical classification of icing regimes was developed based on the primary influencing parameters: T, LWC, and M_{ice}. The design space was divided into three distinct zones as shown in Fig. 2: zone 1 is dominated by glaze ice formations, zone 3 by rime ice, and zone 2 represents transitional or mixed ice shapes with increased sensitivity to Mach number. Three neural networks were trained, one for each zone.

The dataset used for training was generated using a proprietary in-house ice accretion code that mimics the typical numerical sequence used in icing analysis. The tool includes: a 2D inviscid flow solver, a Lagrangian particle tracking algorithm for droplet trajectories, a heat transfer model using the Makkonen correlation [36], and the Messinger equations [37] for computing local energy balance. Starting from the stagnation point, the solver iteratively computes local equilibrium surface temperatures and freezing fractions along the airfoil surface. Any unfrozen water is allowed to convect downstream, enabling the representation of complex geometries including glaze features. Surface roughness is estimated following Shin et al. [38]. The validation and benchmarking of this icing code is reported in [30]. Although the tool supports both ice-on and ice-off aerodynamic evaluations using inviscid flow assumptions, viscous computations are preferred for post-processing aerodynamic performance. A total of 2100 simulations were performed using Latin Hypercube Sampling (LHS) over the design space defined in Table 1, randomly split into training (75 %), validation (15 %), and test (10 %) sets. To further augment the dataset without performing additional simulations, geometric symmetry was leveraged. Specifically, for cases where the clean airfoil is symmetric, new samples were generated by mirroring the iced airfoil geometry, that is, by inverting both the icing angle of attack and the z coordinates, according to Eq. 3.

$$z(\alpha_{\mathrm{ice}}) = -z(-\alpha_{\mathrm{ice}}) \tag{3}$$

Two neural network architectures were explored to predict iced airfoil shapes: a Multi-Layer Perceptron (MLP), denoted $\mathrm{MLP}_{\mathrm{ice}}$, and a Convolutional Neural Network (CNN), denoted $\mathrm{CNN}_{\mathrm{ice}}$. In the $\mathrm{MLP}_{\mathrm{ice}}$ model, each input sample is a vector that contains seven scalar input variables. The $\mathrm{MLP}_{\mathrm{ice}}$ model is not described in detail, as it represents a well-established and widely adopted architecture. In contrast,

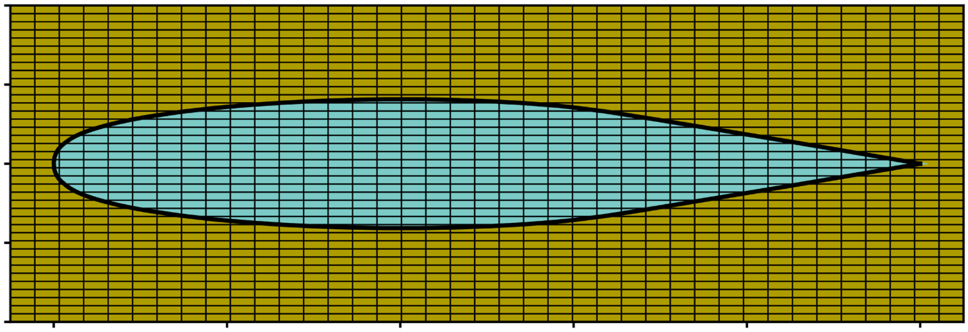


Fig. 3. A CNN_{ice} input sample, shown with reduced grid resolution for clarity.

Table 2
Summary of hyperparameter configurations for MLP_{ice}. Best validation and test-set performance highlighted in green.

Set	FC_1	FC_2	FC_3	FC_4	FC_5	$L_{1,\text{train}}$ $(\times 10^{-3})$	$L_{1,\text{val}}$ $(\times 10^{-3})$	$L_{1,\text{test}}$ $(\times 10^{-3})$
1	512	512	512	1024	–	1.51	1.61	1.64
2	512	512	512	1024	1024	1.24	1.51	1.48
3	1024	1024	1024	2048	2048	0.71	1.06	1.09
4	2048	2048	2048	4096	4096	0.59	1.28	1.24

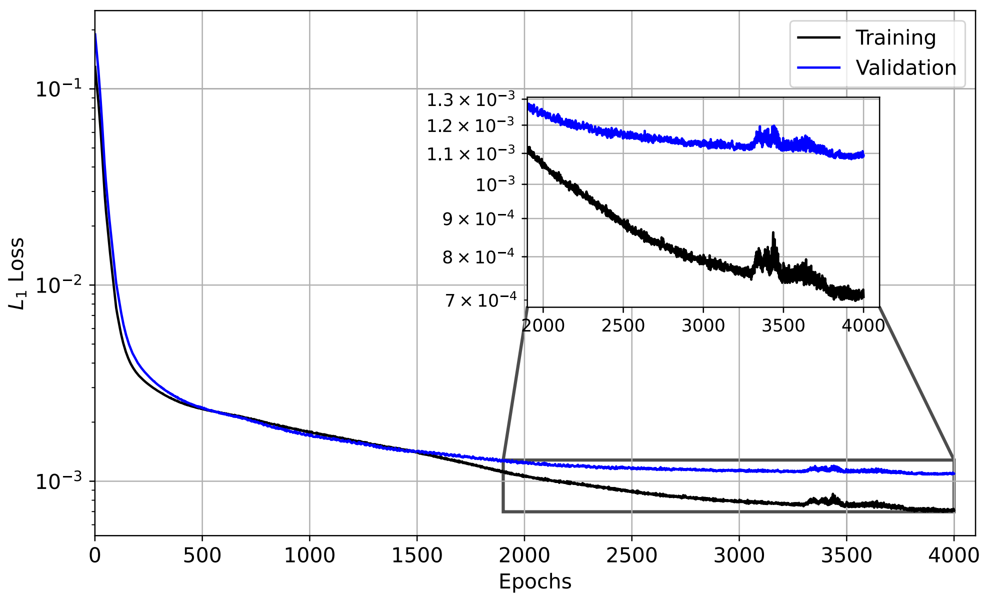


Fig. 4. Convergence of the training and testing loss functions for the MLP_{ice} model, with a zoom on the final 2000 epochs.

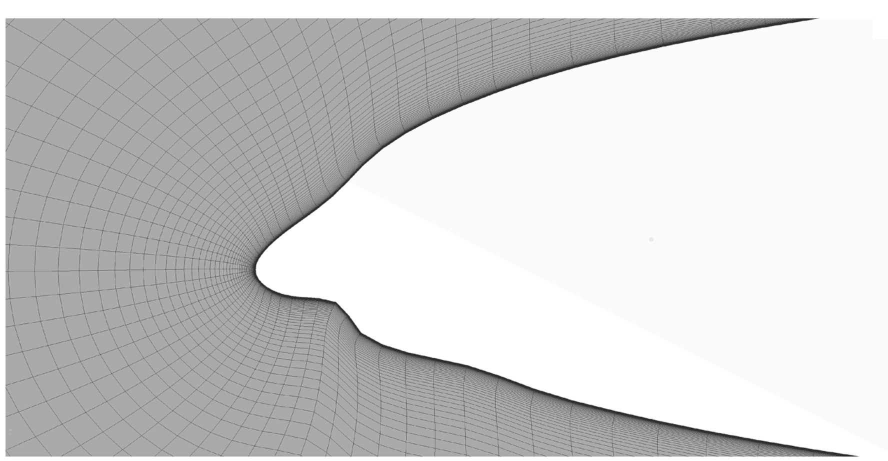


Fig. 5. Example of a structured O-grid generated using Construct2D for an iced airfoil. The mesh is refined near the leading edge and along the boundary layer to capture steep gradients.

Table 3
CFD setup used in SU2 for 2D iced airfoil simulations.

Parameter	Description
Solver type	Compressible RANS - Pressure based
Turbulence model	Spalart-Allmaras
Mach number	0.35 (holding condition)
Reynolds number	Based on chord; evaluated using ISA conditions at 5000 m
Altitude	5000 m (ISA standard atmosphere)
Angle of attack range	−14 deg to 14 deg
Grid topology	Structured O-grid (Construct2D)
Grid resolution	Typical: 512 × 256 (points × layers), clustered near walls
Grid format	Plot3D converted to SU2 using Python script
Non-dimensionalization	Not applied; grid scaled by chord length
CFL number	2.0
Convergence criterion	Residual threshold: 10^{-3}
Automation	mesh generation and SU2 execution scripted in Python

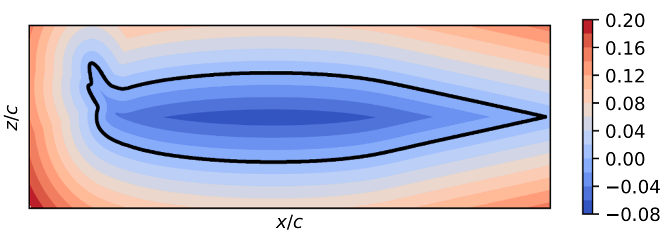


Fig. 6. Example of an SDF of an iced airfoil. The zero-level set (black) represents the airfoil boundary.

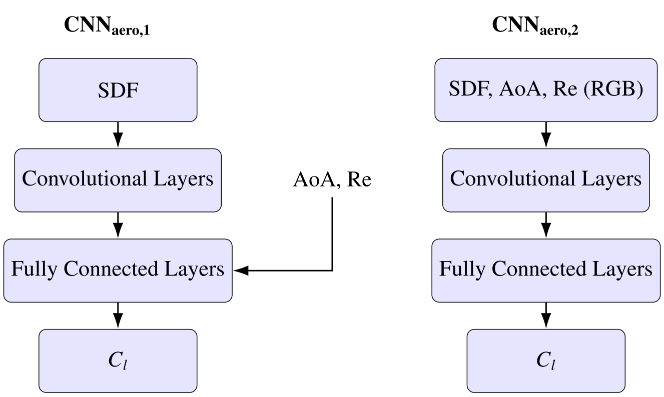


Fig. 7. Schematic representation of CNN architectures for lift prediction.

Table 4
Hyperparameter study results for $\mathrm{CNN}_{\mathrm{aero},1}$. Best validation and test-set performance highlighted in green.

Set	$N_{\mathrm{ch},1}$	$N_{\mathrm{ch},2}$	$N_{\mathrm{ch},3}$	FC_1	FC_2	$L_{1,\mathrm{train}}\,(\times 10^{-2})$	$L_{1,\mathrm{val}}\,(\times 10^{-2})$	$L_{1,\mathrm{test}}\,(\times 10^{-2})$
1	4	–	–	16	–	8.4	9.2	9.1
2	4	8	–	16	–	4.9	7.5	7.8
3	8	8	–	16	32	2.9	4.2	4.3
4	8	8	8	32	32	1.8	5.8	6.0

the $\mathrm{CNN}_{\mathrm{ice}}$ model encodes each sample as an artificial image in which pixel intensities represent encoded input variables within a structured 2D grid. The discretization for CNN input was carried out on a 150 × 150 uniform grid, covering the spatial domain $-0.05 < x/c < 1.05$ and $-0.08 < z/c < 0.08$. Pixel intensities $I(x, y)$ were defined using an indicator function $\mathbb{I}_D(x, y)$, as shown in Eq. 4. A visual example of a $\mathrm{CNN}_{\mathrm{ice}}$ input sample is shown in Fig. 3. The input tensor is processed through a sequence of convolutional layers, after which the resulting feature maps are flattened into a one-dimensional vector. This vector is then passed through fully connected layers up to the final output. The $\mathrm{CNN}_{\mathrm{ice}}$ architecture is not reported, as it is identical to that shown in Fig. 9, except for the absence of angle of attack and Reynolds number as additional inputs. Specifically, $\mathrm{CNN}_{\mathrm{ice}}$ employs two convolutional layers with 8 and 16 output channels, respectively followed by two fully connected layers with 32 and 64 neurons. A kernel size of 3, a stride of 1, and no padding were applied in the convolutional layers.

$$I(x, y) = \mathbb{I}_D(x, y) \cdot (T, MVD, \alpha_{\mathrm{ice}}) + (1 - \mathbb{I}_D(x, y)) \cdot (c, M, 0) \tag{4}$$

To improve training stability and avoid inactive neurons, the Leaky ReLU activation function was adopted in hidden layers, with a slope coefficient $\alpha = 0.05$ [39]. The Adam optimizer [40] was used with a learning rate of 10^{-4}, a batch size of 16, and a training duration of 4000 epochs. This learning rate was selected after tests with values in the range $10^{-6} - 10^{-2}$, which indicated that 10^{-4} provided stable convergence with minimal loss oscillations while keeping a reasonable training time. An L_1 loss function was used to reduce the sensitivity to outliers. Although dropout regularization was initially tested, it did not significantly improve performance. Lasso (L_1) regularization [41] was incorporated into the training process, with a penalty coefficient λ set to 10^{-3}. Although both architectures were initially tested, hyperparameter tuning was performed only for $\mathrm{MLP}_{\mathrm{ice}}$ due to its higher accuracy. Table 2 summarizes the performance of four tested configurations. The third configuration provided the best trade-off between accuracy and generalization, minimizing test loss with low overfitting.

Fig. 4 presents the convergence of the L_1 loss function for the $\mathrm{MLP}_{\mathrm{ice}}$ model using the selected set of hyperparameters. Both training and validation losses decrease rapidly during the initial epochs, followed by a more gradual reduction. As highlighted in the magnified region, the validation loss reaches a plateau, whereas the training loss continues to decline. This divergence suggests the onset of overfitting, where the model achieves improved performance on the training data without a corresponding gain on validation data. Consequently, the training process was not extended beyond 4000 epochs to avoid further overfitting.

2.2. CFD-Based Dataset generation and machine learning framework for lift prediction

This section introduces the methodology used to predict the lift curve of iced airfoils, whose geometries are computed as described in Section 2.1. As with the ice shape prediction, neural networks are employed to estimate aerodynamic performance, thereby avoiding direct CFD simulations in the final 3D workflow. For all lift predictions, the Mach number is fixed at 0.35, representative of typical holding conditions. This choice is motivated by the relatively minor influence of Mach number on subsonic lift, particularly when compared to the dominant effects of geometric deformation due to ice accretion. Should alternative flight conditions be required, well-established compressibility corrections such as the Prandtl-Glauert or Laitone transformations can be

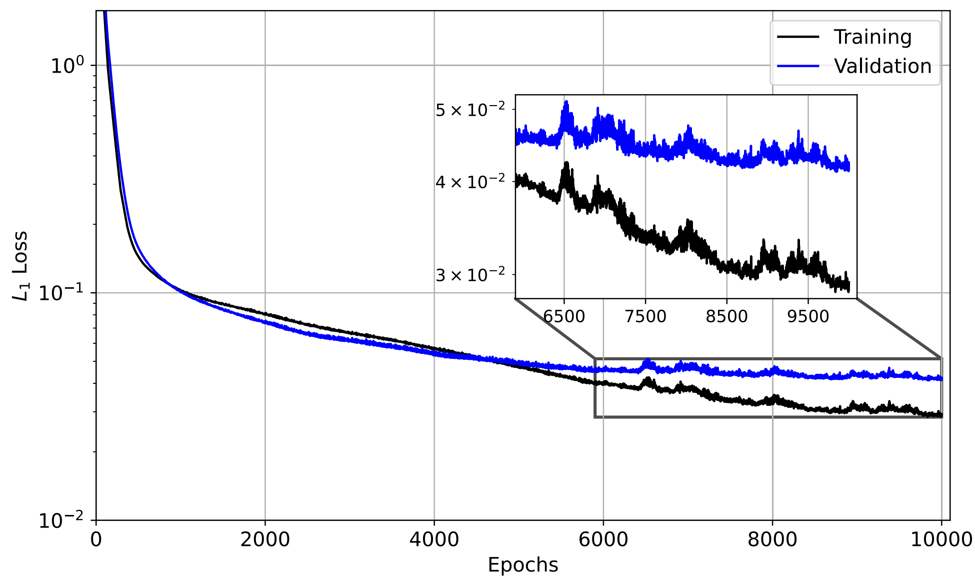


Fig. 8. Convergence of the training and testing loss functions for the $\mathrm{CNN}_{\mathrm{aero},1}$ model, with a zoom on the final 4000 epochs.

Table 5
Comparison of neural network architectures for 2D lift prediction.

Model	Geometry Input	$\alpha + Re$ Injection	Observations
$\mathrm{MLP}_{\mathrm{aero}}$	7 scalars	Scalars in input layer	Limited accuracy
$\mathrm{CNN}_{\mathrm{aero},1}$	SDF (grayscale)	Post-convolution $(1 \times 1$ scalars$)$	**Best generalization observed**
$\mathrm{CNN}_{\mathrm{aero},2}$	SDF $+ \alpha + Re$ (RGB)	Input layer $(150 \times 150$ uniform tensors$)$	High model capacity

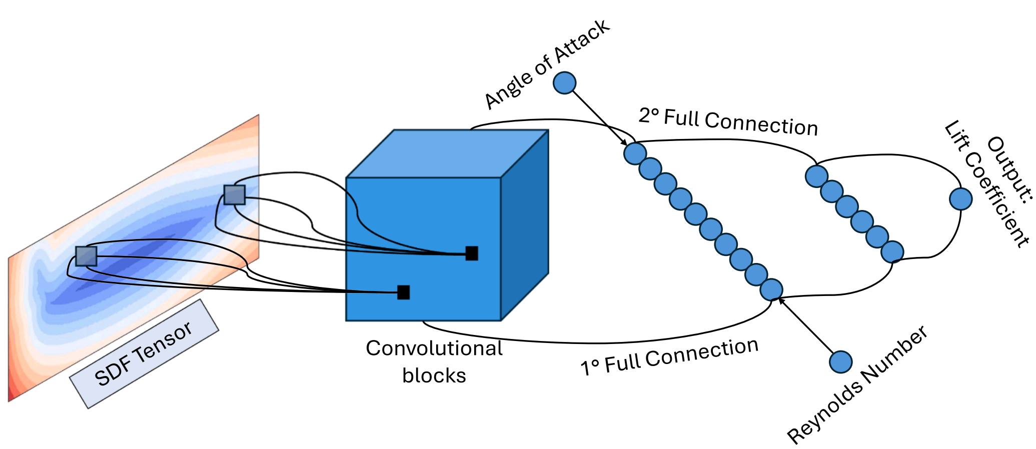


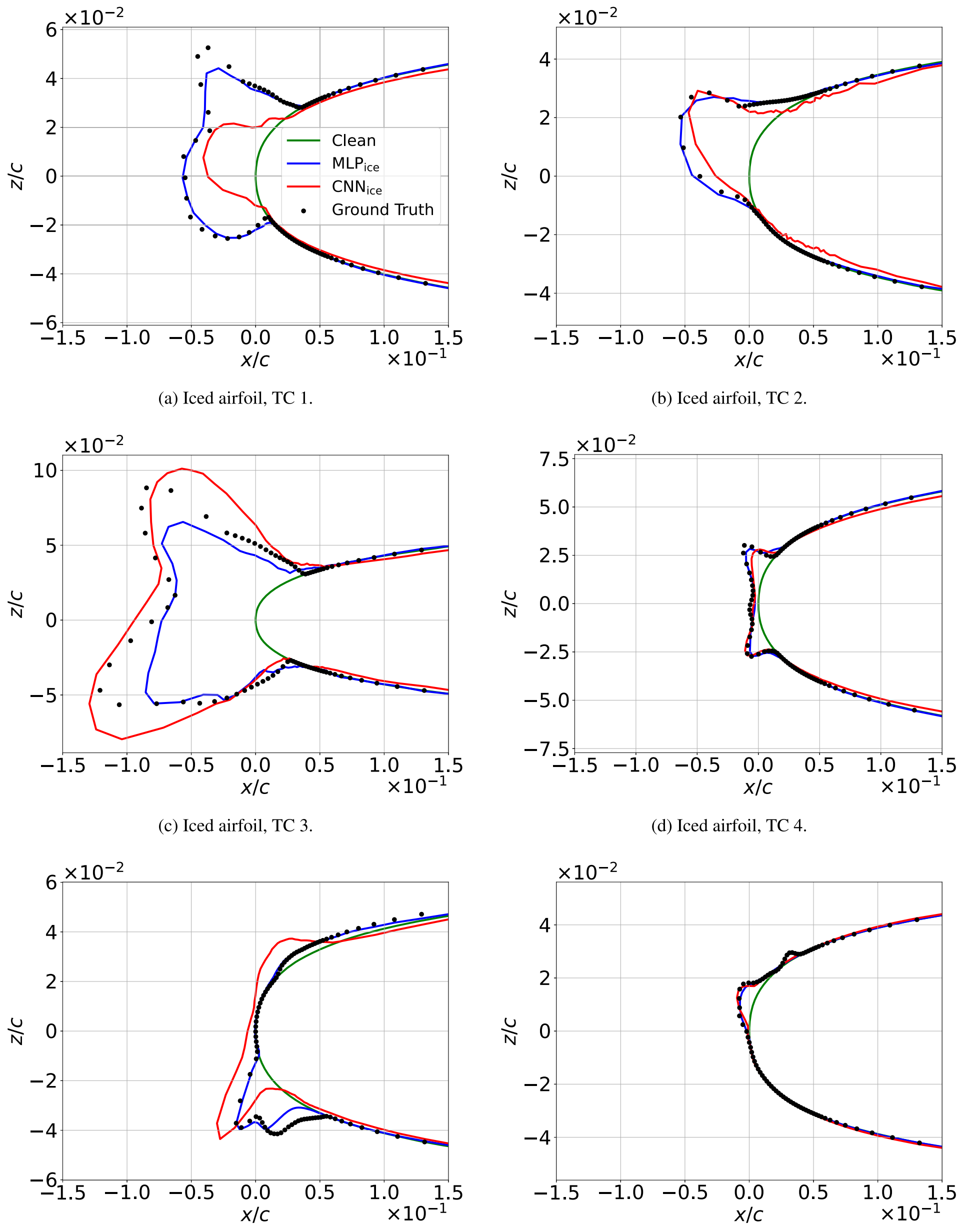
Fig. 9. $\mathrm{CNN}_{\mathrm{aero},1}$ architecture, metadata merged post-convolution.

Table 6
Input variables of 6 test samples for 2D ice prediction validation. Design space zone refers to the Appendix C division reported in Fig. 2.

Case	Mach	T (°C)	MVD (μm)	c (m)	$R_{\mathrm{LE}}/R_{\mathrm{LE,ref}}$	t/c (%)	α_{ice} (deg)	LWC $(\mathrm{g\,m^{-3}})$	Design space zone
TC1	0.37	−10.8	23.1	1.31	0.91	11.1	−1.54	0.33	Mixed (zone 2)
TC2	0.35	−14.6	34.4	0.54	1.05	9.3	−2.52	0.13	Rime (zone 3)
TC3	0.22	−7.8	22.0	0.56	0.81	12.1	2.23	0.42	Glaze (zone 1)
TC4	0.28	−3.8	19.5	2.66	0.93	14.0	0.34	0.57	Glaze (zone 1)
TC5	0.25	−1.7	29.3	1.42	1.10	10.9	−1.93	0.37	Glaze (zone 1)
TC6	0.38	−21.7	22.2	3.23	1.14	10.2	−3.71	0.17	Rime (zone 3)

applied with sufficient accuracy. The dataset used to train the neural networks was generated using the SU2 suite (Stanford University Unstructured) [42], a widely adopted open-source CFD solver. Since SU2 lacks a built-in grid generator, Construct2D [43], an open-source Fortran-based code, was used to create structured C- and O-type meshes around two-dimensional geometries. This tool allows fine control over key meshing parameters, including boundary layer resolution, y^+, and far-field domain size. As Construct2D outputs meshes in Plot3D format,[1] a Python script was used to convert these files into SU2-compatible format. To

[1] See NASA Ames' documentation on Plot3D format.



(a) Iced airfoil, TC 1. (b) Iced airfoil, TC 2.

(c) Iced airfoil, TC 3. (d) Iced airfoil, TC 4.

(e) Iced airfoil, TC 5. (f) Iced airfoil, TC 6.

Fig. 10. Performance evaluation of 2D ice shape prediction for 6 airfoils. "Ground truth" indicates the results obtained with the ice accretion code described in Section 2.1. All airfoils are scaled such that the corresponding clean airfoil has unitary chord.

Table 7
Input variables of 6 test samples for 2D lift prediction validation.

Case	Mach	T (°C)	MVD (μm)	c (m)	$R_{LE}/R_{LE,ref}$	t/c (%)	α_{ice} (deg)
TC1	0.27	−7.8	33.5	1.53	1.13	8.4	0.86
TC2	0.25	−29.1	16.3	0.90	1.18	11.7	−1.30
TC3	0.21	−15.1	25.7	2.78	1.17	10.1	3.88
TC4	0.36	−4.0	27.2	1.74	0.97	9.1	2.4
TC5	0.36	−18.1	22.2	1.16	0.89	11.4	1.05
TC6	0.34	−11.7	31.4	3.35	0.81	14.9	1.71

Table 8
Error statistics (maximum, mean, and median absolute errors) for 2D lift prediction models evaluated on the test set in the pre-stall range. Percentages in parentheses denote relative errors with respect to the ground-truth C_l.

Model	Max Absolute Error	Mean Absolute Error	Median Absolute Error
MLP_{aero}	0.82 (> 100 %)	0.119 (15.6 %)	0.067
$CNN_{aero,1}$	**0.27 (38.3 %)**	**0.043 (6.7 %)**	**0.021**
$CNN_{aero,2}$	0.33 (60.9 %)	0.050 (7.5 %)	0.024

ensure dimensional consistency in compressible RANS simulations (which are not non-dimensionalized with Reynolds number), mesh scaling by the airfoil chord length was applied using the `SU2_DEF` executable. Both Construct2D and SU2 support automation via scripting, which was essential to efficiently generate a large and diverse dataset. An example of a structured grid over an iced airfoil is shown in Fig. 5.

The dataset used to train the neural networks for lift curve prediction was constructed from CFD simulations performed on 50 iced airfoils. Each configuration was evaluated at angles of attack ranging from −14° to 14° in 2° increments, excluding ±2°. This choice yielded a total of 650 simulations, of which 80 % were allocated for training and 20 % reserved for validation. The final performance of the models was assessed

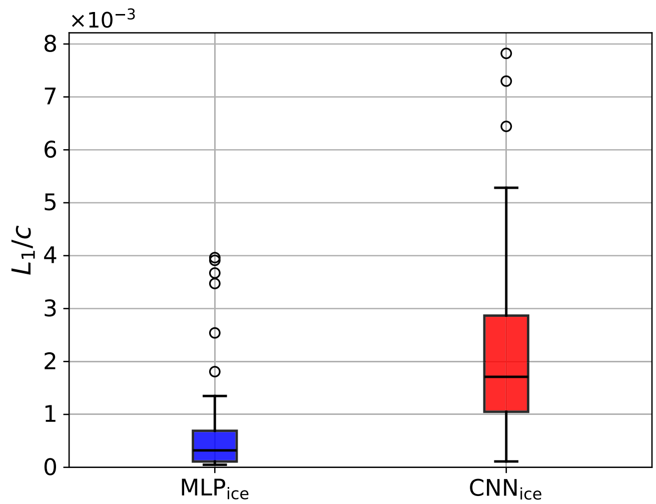


Fig. 11. Box plots of the L_1 norm between model predictions and ground truth, normalized by the corresponding clean airfoil chord, for $\mathrm{MLP}_{\mathrm{ice}}$ and $\mathrm{CNN}_{\mathrm{ice}}$.

Table 9
Geometric parameters and flight conditions used in the 3D simulation. t/c and R_{LE}/c are constant along the lifting surface span.

Parameter	Description	Unit	Value
Geometry			
$Æ_{\mathrm{w}}$	Wing Aspect Ratio	–	6
S_{w}	Wing area	m^2	25
$\Lambda_{LE,\mathrm{w}}$	Leading Edge sweep angle	deg	15
λ_{w}	Taper ratio	–	0.6
t/c	Thickness to chord ratio	–	0.10
R_{LE}/c	Leading Edge Radius to chord ratio	–	0.01428
Flight conditions			
T	Freestream temperature	°C	−8
MVD	Droplets Median Volume Diameter	µm	20
LWC	Droplets Liquid Water Content	$\mathrm{g/m}^3$	0.459
M	Mach number	–	0.35
α_{ice}	Freestream Icing Angle of Attack	deg	3

on an independent test set comprising 16 previously unseen airfoils, as described in Section 3.2. The solver settings used in these computations, representative of a loitering flight condition, are summarized in Table 3. To expand the dataset without additional CFD computations, geometric symmetry was exploited. Specifically, the lift curve of an airfoil mirrored about the x-axis can be obtained by leveraging the antisymmetry of the lift coefficient with respect to the angle of attack.

The use of surrogate models significantly reduces the computational burden typically associated with aerodynamic performance evaluation. On average, each compressible RANS simulation required approximately 15–30 minutes to converge on a standard workstation (8-core CPU), depending on the complexity of the iced geometry and angle of attack. In contrast, inference with the trained neural network takes around 1 s per sample.

Considering the generation of a full lift curve with 13 angles of attack per airfoil, the CFD-based process requires approximately 5–6 hours of wall-clock time per airfoil. When extrapolated to a design optimization loop or uncertainty quantification scenario involving thousands of evaluations, the computational savings enabled by the surrogate model become substantial, often exceeding a factor of 10^4 in terms of wall-clock time.

2.2.1. Selection of the most suitable neural network architecture for iced airfoil lift prediction

Three neural network architectures were evaluated for predicting the lift coefficient of iced airfoils: a Multi-Layer Perceptron ($\mathrm{MLP}_{\mathrm{aero}}$), and two convolutional neural networks, denoted $\mathrm{CNN}_{\mathrm{aero},1}$ and $\mathrm{CNN}_{\mathrm{aero},2}$, respectively. The architecture of $\mathrm{MLP}_{\mathrm{aero}}$ is analogous to the one described in Section 2.1, with the only difference being the output format: here, the network outputs a scalar lift coefficient rather than a set of 2D airfoil coordinates. Due to its architectural similarity and lower performance, $\mathrm{MLP}_{\mathrm{aero}}$ is not analyzed further in this section.

Results (Section 3.1) showed that the CNN model used for ice shape prediction underperformed relative to its MLP counterpart. This motivated the development of alternative CNN-based representations for lift prediction. In particular, the goal was to enhance the geometric input's informativeness by transitioning from binary masks to Signed Distance Functions (SDF), which are known to preserve more spatial and topological information [44].

Let $\Omega \subset \mathbb{R}^n$ be a region (e.g., the interior of the airfoil), and let $\partial\Omega$ denote its boundary. The SDF $\phi : \mathbb{R}^n \to \mathbb{R}$ is defined as

$$\phi(\boldsymbol{x}) = \begin{cases} +\,\mathrm{dist}(\boldsymbol{x}, \partial\Omega), & \boldsymbol{x} \notin \Omega \quad \text{(outside)} \\ 0 & \boldsymbol{x} \in \partial\Omega \quad \text{(on the boundary)} \\ -\,\mathrm{dist}(\boldsymbol{x}, \partial\Omega), & \boldsymbol{x} \in \Omega \quad \text{(inside)} \end{cases}$$

where

$$\mathrm{dist}(\boldsymbol{x}, \partial\Omega) = \min_{\boldsymbol{y} \in \partial\Omega} \|\boldsymbol{x} - \boldsymbol{y}\|$$

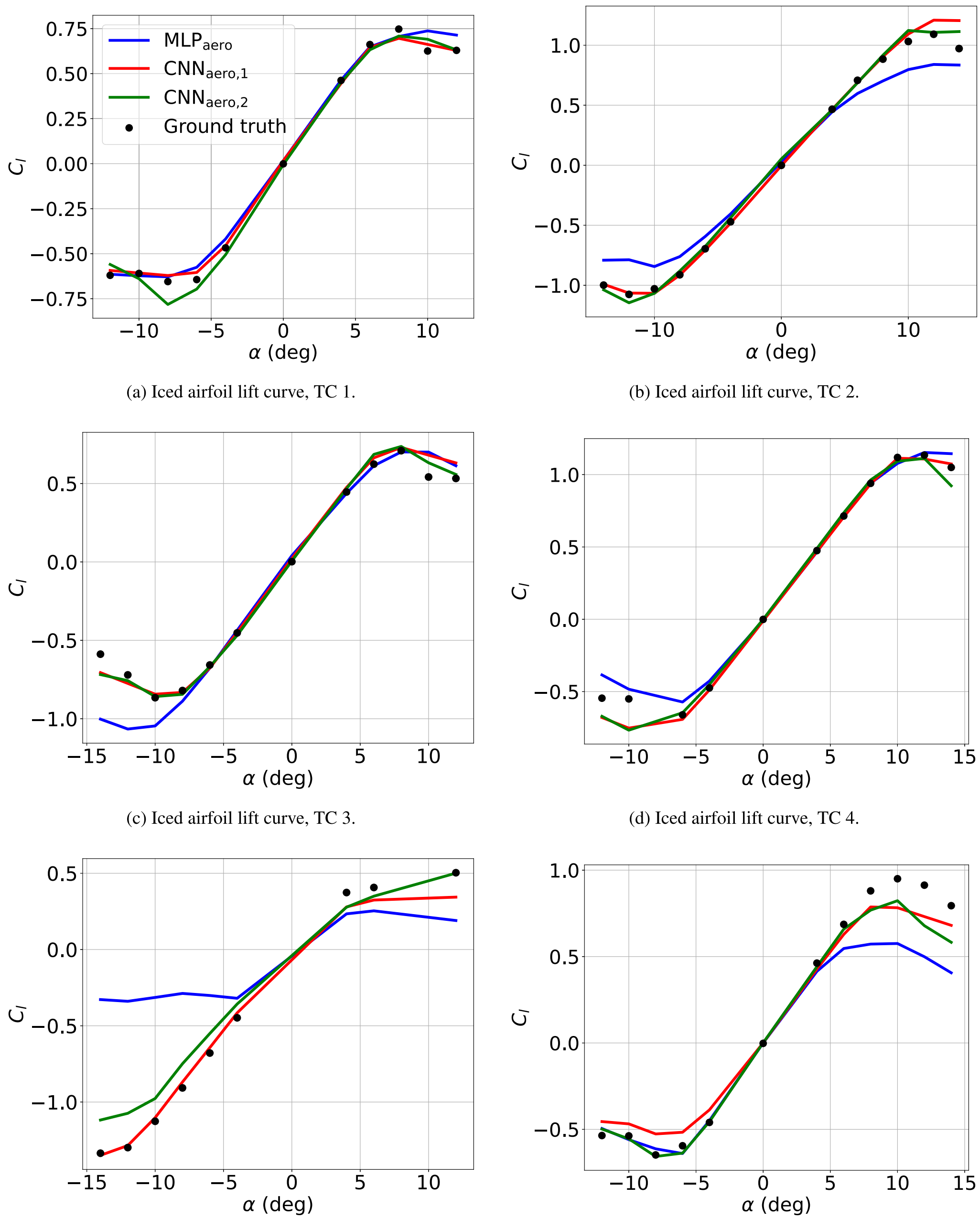
To determine whether a point $\boldsymbol{x} \in \mathbb{R}^2$ lies inside the domain Ω, the Winding Number algorithm [45] is used. The domain is discretized over a 150×150 grid spanning $-0.15 < x/c < 1.01$ and $-0.12 < z/c < 0.12$, which is sufficient to fully encapsulate all iced geometries in the dataset, and the SDF is computed for each of the grid points. An example SDF calculated on an iced airfoil is shown in Fig. 6.

In addition to geometry, both CNN architectures receive the Reynolds number and angle of attack as inputs. In $\mathrm{CNN}_{\mathrm{aero},1}$, these two quantities are provided as scalars and concatenated with the flattened convolutional output — thus increasing its size by two units — and the resulting vector is processed through fully connected layers up to the final regression head. In contrast, $\mathrm{CNN}_{\mathrm{aero},2}$ embeds these quantities directly into the image input as constant-value tensors, each matching the shape of the SDF (150×150). As a result, the input to $\mathrm{CNN}_{\mathrm{aero},2}$ can be interpreted as an RGB image, where the red channel encodes SDF values, the green channel represents the angle of attack, and the blue channel corresponds to the Reynolds number. The CNN architectures are schematized in Fig. 7.

All models were trained using the Adam optimizer [40] with a learning rate of 10^{-4}, L_1 loss function, a batch size of 8, and a training duration of 10,000 epochs. Dropout was evaluated but found to have negligible impact. Lasso (L_1) regularization was applied with a penalty term $\lambda = 10^{-5}$ to prevent overfitting and encourage sparsity. A hyperparameter study was conducted on the CNN models, exploring different configurations of convolutional channel depths ($N_{\mathrm{ch,i}}$) and fully connected layer widths (FC_i), the main quantity of interest is the loss function evaluated on the validation set $\mathcal{L}_{\mathrm{val}}$. Table 4 summarizes the results obtained for $\mathrm{CNN}_{\mathrm{aero},1}$. As expected, performance improved with model capacity up to a certain number of network parameters, beyond which further increases led to overfitting. This trend mirrors observations made in the ice prediction network. The convergence of the training and testing loss functions with this hyperparameter set is shown in Fig. 8. Similarly to $\mathrm{MLP}_{\mathrm{ice}}$, although the training loss continues to decrease, training was not extended to additional epochs due to minimal gains in the validation loss function.

A comprehensive validation of model performance is presented in Section 3.2. Table 5 provides a comparative summary of the three neural network architectures explored for predicting the lift coefficient of iced airfoils. The models differ primarily in how the airfoil geometry and the physical metadata-angle of attack (α) and Reynolds number (Re)-are represented and processed.

The MLP operates on scalar inputs and offers limited modeling capacity for geometric complexity. In contrast, both CNN variants



(a) Iced airfoil lift curve, TC 1.

(b) Iced airfoil lift curve, TC 2.

(c) Iced airfoil lift curve, TC 3.

(d) Iced airfoil lift curve, TC 4.

(e) Iced airfoil lift curve, TC 5.

(f) Iced airfoil lift curve, TC 6.

Fig. 12. Performance evaluation of 2D Lift Predictor for 6 airfoils. "Ground truth" indicates the results obtained with SU2.

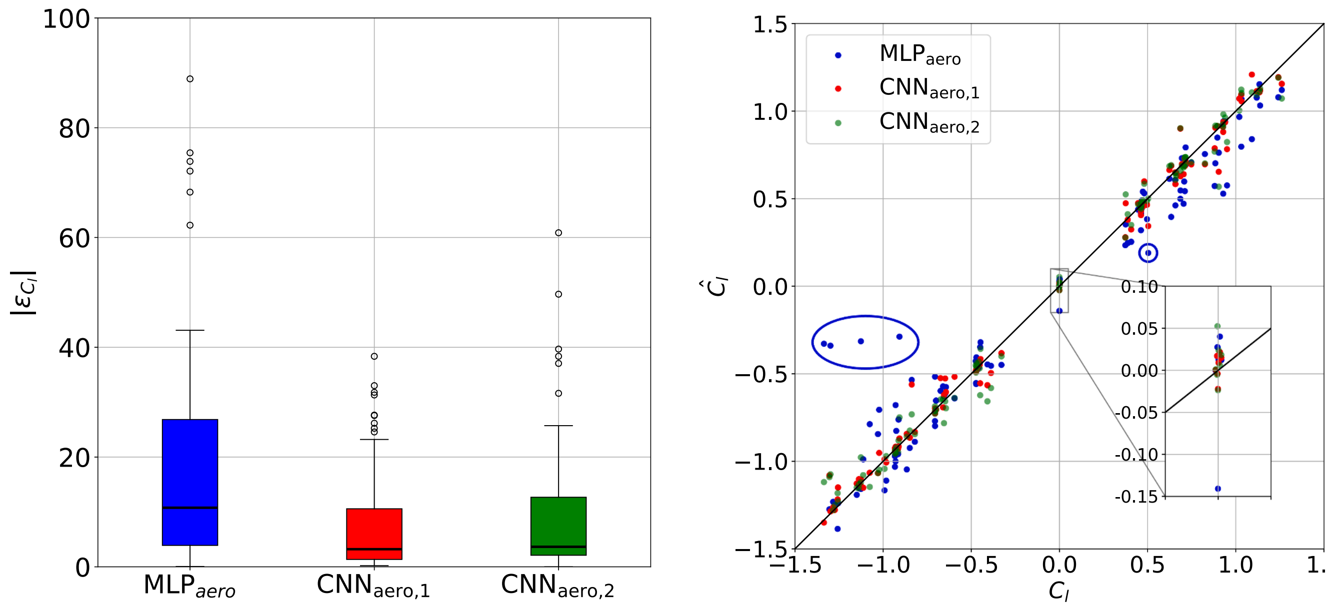
leverage image-based representation. CNN$_{aero,1}$ processes the airfoil geometry through an SDF, with α and Re appended after convolutional feature extraction, while CNN$_{aero,2}$ embeds all three inputs directly into the image channels, simulating an RGB structure. This structural difference impacts learning performance and generalization, as further discussed in Section 3.2. Preliminary training results indicated that CNN$_{aero,1}$ produced the most stable and generalizable predictions. Although CNN$_{aero,2}$ also achieved good accuracy, its performance was inferior to that of CNN$_{aero,1}$. The final architecture employed for the aerodynamic predictions, the CNN$_{aero,1}$, is reported in Fig. 9, where the structural arrangement of convolutional and dense layers, along with the integration of physical inputs, is clearly illustrated.

The final neural network model, selected for lift prediction, maps the iced airfoil geometry and flow condition parameters to the corresponding two-dimensional lift coefficient. The model input consists of a combination of geometric and aerodynamic features: the SDF, paired with α and Re, forming the complete input to the neural network. The output is the corresponding lift coefficient C_l, thereby learning the functional mapping:

$$C_l = \mathcal{F}(\mathrm{SDF}, \alpha, Re) \tag{5}$$

3. Validation and benchmarking of the neural networks' predictive capability

This section validates the proposed hybrid methodology for aerodynamic performance prediction of iced airfoils. The evaluation focuses on the individual components of the toolchain, namely ice shape prediction and lift curve estimation, both of which rely on machine learning surrogates trained on numerical simulations. Each model is benchmarked against reference results withheld during dataset generation. The objective is to determine the most robust surrogate model for integration



(a) Box Plot of C_l percent error $|\varepsilon_{C_l}| = 100 \cdot |(\hat{C}_l - C_l)/C_l|$.

(b) Regression Plot.

Fig. 13. Statistical evaluation of 2D lift coefficient predictions on test set, pre-stall range. For clarity, the box plot is trimmed at 100 % relative error; MLPaero data points with $\varepsilon C_l > 100$ are not shown in the box plot but are encircled in blue in the regression plot. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

into the quasi-3D solver, with selection based on predictive accuracy, robustness across scenarios, and ease of deployment. Six test cases are considered for each phase of the evaluation, denoted as TC_i for the i-th test case.

3.1. 2D iced airfoil shape prediction

In this section, the benchmarking of the neural network for iced airfoil shape prediction is presented. A selection of 6 samples from the benchmarking process is reported in Fig. 10, which refers to the input variables defined in Table 6. Both MLP_{ice} and CNN_{ice} accurately reconstruct the iced airfoil in test cases TC2, TC4, and TC6. In TC1, MLP_{ice} achieves a satisfactory reconstruction, whereas CNN_{ice} fails to capture the overall icing phenomena effectively. In TC5, both models successfully predict ice accretion on the lower part of the leading edge; however, CNN_{ice} overestimates the ice extension on the upper surface. In TC3, CNN_{ice} slightly outperforms MLP_{ice}, which underestimates the ice extent but still provides a qualitatively accurate reconstruction.

Overall, MLP_{ice} consistently achieves higher accuracy across a range of icing scenarios. This performance advantage is quantitatively supported by a statistical analysis conducted on test data. As illustrated in Fig. 11, the box plot of the L_1 norm between model predictions and ground truth (normalized by the chord length of the clean airfoil) shows that MLP_{ice} yields lower median and interquartile values compared to CNN_{ice}.

From a deployment standpoint, the simpler MLP_{ice} architecture also offers reduced inference time and lower memory overhead, making it a more practical choice for integration with the 3D VLM framework. However, future work may revisit CNN-based encodings in combination with Signed Distance Functions, which have proven effective in aerodynamic prediction.

3.2. 2D iced airfoil lift prediction

As illustrated in Fig. 12, which presents the lift curves obtained using the three models tested on the input samples defined in Table 7, in TC1 and TC4, all three models exhibit strong performance, accurately predicting the lift curve. In TC2, MLP_{aero} does not capture the correct slope and stall behavior, whereas both CNN models successfully predict these aerodynamic characteristics. In TC3, all models accurately predict the lift curve slope; however, the CNNs correctly capture both negative and positive stall, while MLP_{aero} accurately predicts only the positive stall. In TC5, MLP_{aero} fails to predict the lift curve altogether, whereas the CNN models correctly capture the slope. Additionally, $CNN_{aero,1}$ provides a more accurate prediction of the negative stall, while $CNN_{aero,2}$ performs better in predicting the positive stall. In TC6, all three models accurately predict the lift curve slope and capture the negative stall. However, while MLP_{aero} and $CNN_{aero,2}$ slightly outperform $CNN_{aero,1}$ in this regard, MLP_{aero} fails to predict the positive stall, whereas the CNN models provide a better prediction. However, it is important to note that, in post-stall conditions, the predictions of all three models become less reliable. This limitation suggests that a larger dataset at high angles of attack or modifications to the models' architectures may be necessary to capture the more complex physics occurring in this regime.

The CNN-based architectures clearly outperform the MLP baseline in modeling the nonlinear aerodynamic response of iced airfoils. In particular, $CNN_{aero,1}$ provides the most robust predictions, balancing model expressivity with generalization. Unlike $CNN_{aero,2}$, which embeds physical metadata directly into the input tensor, $CNN_{aero,1}$ decouples geometric encoding from physical parameters, a design that appears to improve learning stability and reduce overfitting, especially at high angles of attack.

A statistical evaluation was carried out to identify the most suitable model for integration into the VLM framework. Box plots and regression plots are utilized to assess model performance across a broad dataset made up of 224 simulations carried out using SU2, restricted to samples with lift coefficients obtained in pre-stall threshold. To avoid singularities, data points corresponding to zero angle of attack were excluded from the box plots and instead highlighted in the regression plots.

The results, presented in Table 8 and Fig. 13, reveal that the distributions of lift coefficient percentage error for all three models exhibit positive skew, indicating that most prediction errors are relatively small, while a few larger errors extend the tail toward higher values, thereby

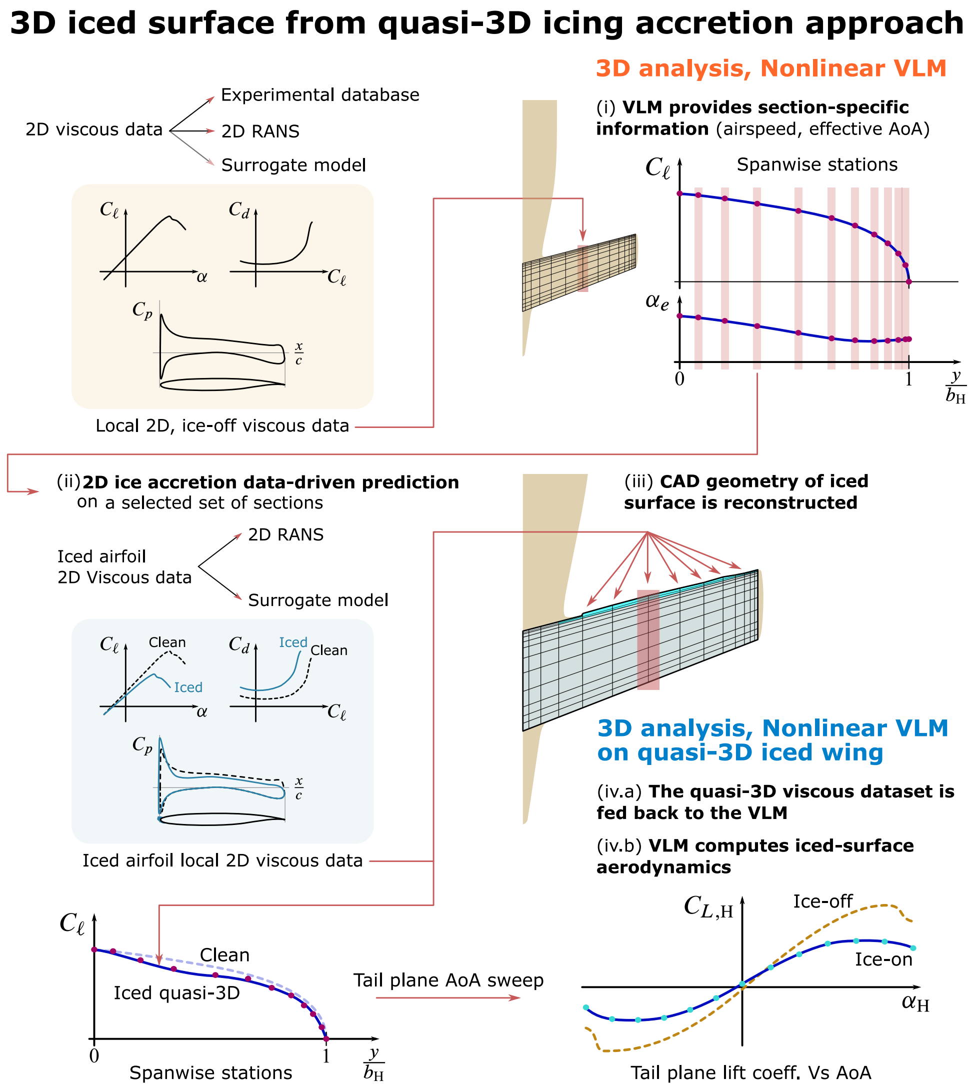


Fig. 14. Schematic representation of the nonlinear VLM framework [30].

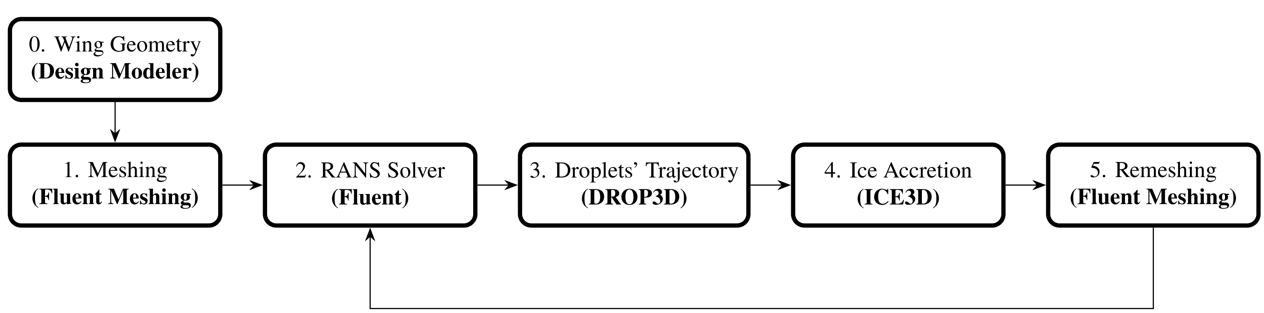


Fig. 15. Workflow for high-fidelity ice accretion simulation.

the mean is higher than the median. Among them, the MLP-based model (MLP_{aero}) demonstrates a higher median prediction value compared to both CNNs, which share similar interquartile ranges. Notably, $CNN_{aero,1}$ achieves slightly lower median, quartile values and exhibits fewer outliers than $CNN_{aero,2}$. Based on this analysis, $CNN_{aero,1}$ was selected for subsequent integration into the VLM-based aerodynamic solver.

Although post-stall prediction remains challenging for all models due to limited data and turbulent separation effects, the selected $CNN_{aero,1}$ model provides reliable lift predictions up to $C_{l,\max}$, making it the most suitable candidate for integration into the VLM pipeline.

3.3. 3D aerodynamic behavior via nonlinear VLM

To extend the aerodynamic evaluation to 3D lifting surfaces, a quasi-three-dimensional (quasi-3D) methodology based on a nonlinear Vortex Lattice Method (VLM) was adopted. This approach enables the estimation of the aerodynamic forces on horizontal tailplanes – both clean and iced – by coupling 2D high-fidelity section data with a spanwise lifting-line formulation that accounts for nonlinearity in the sectional lift response. The implemented framework is based on the architecture previously proposed in [30], and schematically illustrated in Fig. 14.

The lifting surface (i.e., the tailplane) is discretized into multiple spanwise sections, each modeled with its own nonlinear lift curve, including stall and post-stall behavior. These curves are provided by the surrogate models described in Section 2.2, which are trained on a combination of clean and iced 2D simulations. The aerodynamic load distribution is computed iteratively by solving the induced downwash equations through a Newton-type relaxation method, as detailed in [18]. At each iteration, the local effective angle of attack is updated by accounting for both geometric incidence and downwash effects, and the corresponding sectional lift is evaluated through the nonlinear airfoil surrogate. The spanwise coupling is performed under the assumptions of incompressible, inviscid, and irrotational flow, following a Biot-Savart formulation with Horseshoe vortices. This formulation has two main advantages: (i) it allows for the inclusion of complex aerodynamic degradation (e.g., due to ice accretion) without requiring full 3D CFD simulations; and (ii) it retains the physical interaction between sections through induced velocities, which is essential for capturing redistribution effects typical of non-elliptic lift distributions. Notably, the method was shown to predict with reasonable accuracy the aerodynamic coefficients of iced tailplanes when compared to RANS simulations [30], with typical errors on C_L and C_m below 10 % for a range of angles of attack. The nonlinear VLM was previously validated in Appendix A of [18] by applying it to a reference aircraft configuration featuring a forward-swept tailplane.

4. Application of the surrogate framework to full-scale 3D wing simulation

To assess the consistency of the proposed surrogate-based framework with higher-fidelity tools, a comparative analysis is carried out against a RANS simulation performed in Fluent Icing [46] on a three-dimensional wing geometry. Rather than serving as a strict validation, this comparison highlights the agreement and discrepancies between two modeling approaches of differing fidelity. The simulation parameters, summarized in Table 9, correspond to a representative loiter flight condition with hazard of glaze ice formation. T and MVD have been arbitrarily chosen to simulate a glaze ice condition, while LWC is calculated by Appendix C of 14 CFR Part 25.

After the icing simulation, aerodynamic simulations are conducted over a range of negative angles of attack. These conditions are representative of the horizontal tailplane's typical operational regime, where downforce is generated to ensure longitudinal trim. In practice, the horizontal stabilizer is set at a negative incidence angle (e.g., lower than $-10\,\mathrm{deg}$ for aircraft such as the A320), which leads to flow conditions equivalent to operating at a negative angle of attack relative to the freestream. For this reason, aerodynamic simulations at negative angles are physically consistent and particularly relevant for evaluating the tail's ability to provide sufficient stability and control during approach, roundout, and pitch-up maneuvers.

4.1. 3D CFD setup

The process begins by defining the wing CAD model and serves as the basis for all subsequent simulations. From this geometry, the baseline mesh is created and provided to Fluent Icing for the multi-shot calculation process. The calculation begins with the solution of the flowfield using Fluent. Subsequently, droplet trajectories are computed with DROP3D, an Eulerian solver for droplet distribution. This is followed by the ice accretion calculation in ICE3D, where the results from the flowfield and droplet calculations are used to model the buildup of ice on the wing surface. The final step in each sequence is remeshing, performed through meshing journals, which update the mesh to accommodate changes in surface geometry due to ice formation. Each RANS-Droplets-Ice Accretion-Remeshing cycle constitutes one shot. The output of the remeshing step, a new grid, is then used to compute the updated flowfield. This process is repeated iteratively until the user-defined number of shots is reached. A schematic representation of this process is shown in Fig. 15, and the iced wing shape obtained at the last shot is reported in Fig. 18.

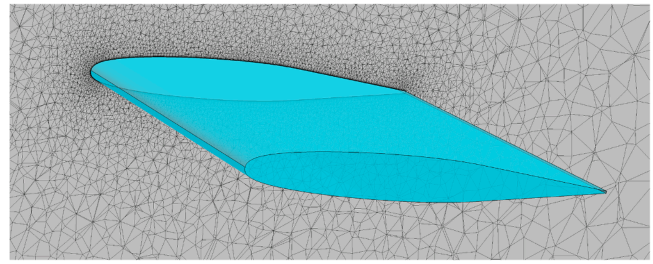


Fig. 16. Volumetric mesh visualized through a planar cut at $y = 2\,\mathrm{m}$, illustrating the unstructured tetrahedral topology and the refined near-wall region.

The CAD model was generated using Ansys DesignModeler [47]. The global coordinate system is defined with its origin located at the leading edge of the wing root, the x-axis extends from the leading edge to the trailing edge, the y-axis spans from the root towards the wingtip, and the z-axis is oriented to ensure a right-handed coordinate system. Exploiting the geometric symmetry about the xz-plane and assuming zero sideslip angle, only half of the wing was modeled, applying a symmetry boundary condition at $y = 0$. The computational domain consists of a bullet-shaped volume, where the upstream region is a hemisphere with radius $2.5b_H/2$, followed by a half-cylindrical section of length $10b_H/2$, with b_H denoting the span of the tail. Unstructured tetrahedral meshes with prism layers were generated using Fluent Meshing[48] to ensure accurate resolution of the boundary layer. The first cell centroid in the wall-normal direction is located within the viscous sublayer ($y^+ < 5$), allowing for the direct integration of the RANS equations without relying on wall function approximations. The baseline mesh consists of approximately 16 million elements. However, as the simulation progresses, the number of elements increases significantly due to adaptive mesh refinement, which is highly sensitive to the geometry of the iced surface, particularly to the local wing curvature. In the current case study, the mesh reaches approximately 50 million elements at the final shot. A visualization of the baseline grid used in the simulations is provided in Fig. 16.

The setup used in the high-fidelity simulations is summarized in Table 10. The maximum number of iterations for the airflow solver was set to 200. However, convergence was consistently achieved before reaching this limit, as the solution satisfied the convergence criterion of all residuals falling below 5×10^{-4}. For the droplet solver, a fixed iteration count of 140 was adopted without enforcing an explicit convergence criterion. This value was empirically determined to be sufficiently high to ensure that the computed collection efficiency reached a steady-state value. For ICE3D, the adaptive timestep provided good accuracy and fast computations. For the calculation of the lift curve of the iced wing, the a_1 coefficient of the SST $k-\omega$ model has been set to 0.35, and pressure, momentum and energy under-relaxation factors of 0.5, 0.1, 0.5 have been set based on the best practice settings proposed in [49].

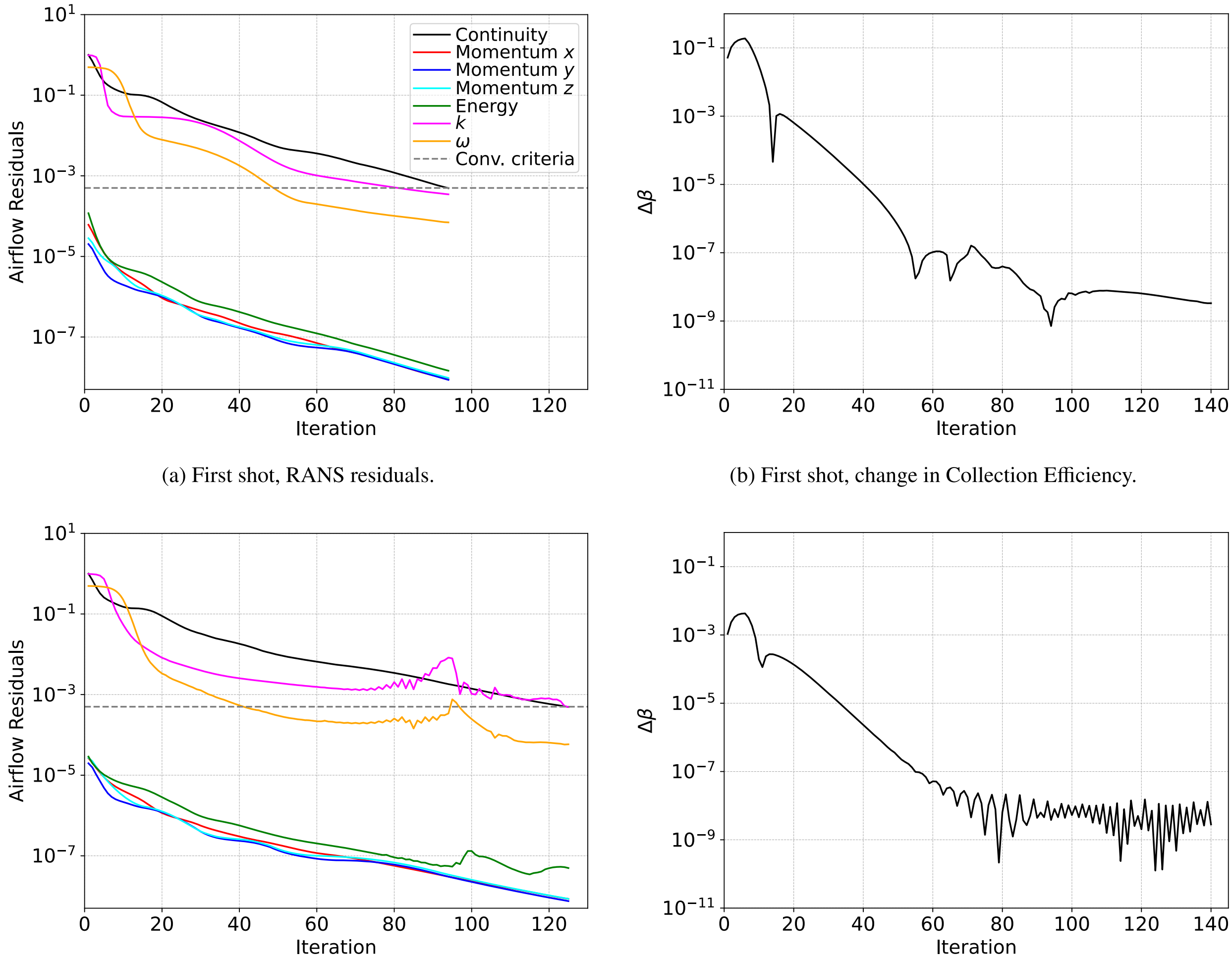
Representative examples of the simulation convergence are presented in Fig. 17, which illustrates the RANS residuals and the collection efficiency histories for the first and last simulation shots. Although the convergence behavior of the final shot exhibits increased stiffness, characterized by mild oscillations and a less monotonic decline in residuals, this does not compromise the overall convergence. In both cases, all RANS residuals fall below the prescribed threshold within the allocated maximum number of iterations. Furthermore, the collection efficiency reaches a steady-state condition, with relative changes between successive iterations on the order between 10^{-9} and 10^{-8}, confirming the numerical stability of the solution.

4.2. Assessing surrogate model fidelity using RANS reference data

The lift curves for both clean and iced wing configurations, along with the predictions obtained using the present surrogate method, are

Table 10
Setup used in fluent for iced wing CFD simulations. Flight conditions parameters are reported in Table 9.

Parameter	Description
Solver type	Compressible RANS - Pressure based
Density model	Ideal gas
Turbulence model	SST $k-\omega$
Operating pressure	54,000 Pa
Gradient discretization	Green-Gauss Node Based
Primitive variables discretization	Second order
SST $k-\omega$ a_1 coefficient	0.35
$p-v$ coupling	Coupled
Fluent max number of iterations	200
Residuals convergence criteria	5×10^{-4}
DROP3D number of iterations	140
ICE3D timestep	Adaptive



(a) First shot, RANS residuals.

(b) First shot, change in Collection Efficiency.

(c) Last shot, RANS residuals.

(d) Last shot, change in Collection Efficiency.

Fig. 17. Convergence of flowfield and collection efficiency of the first and last shot. $\Delta\beta$ is defined as $\beta_i - \beta_{i-1}$, where β is the collection efficiency and i is the iteration index.

illustrated in Fig. 20. For consistency and readability, all lift and angle of attack values are described in terms of their absolute magnitudes. Hence, increasing (i.e., more negative) angles of attack and lift coefficients are referred to as “higher absolute α” and “greater absolute C_L,” respectively.

The iced configuration exhibits a markedly altered C_L-α response. The lift curve no longer presents a well-defined stall point. Instead, the lift curve of the iced wing shows a progressive reduction in lift slope without a clear reversal in gradient. This behavior is consistent with experimental and numerical findings in the literature [17,49], and reflects the disruption of the conventional stall mechanism due to ice accretion. In particular, ice-induced shape irregularities suppress the development of coherent flow separation patterns, thus delaying or altering the stall onset.

The discrepancy between high-fidelity CFD predictions and those of the 2.5D surrogate model is primarily due to the formation of localized, ice-induced vortices at the leading edge. These vortices entrain high-momentum freestream air into the separated boundary layer, thereby promoting early reattachment and locally enhancing lift. An illustration of this mechanism is provided in Fig. 19: the flow separates

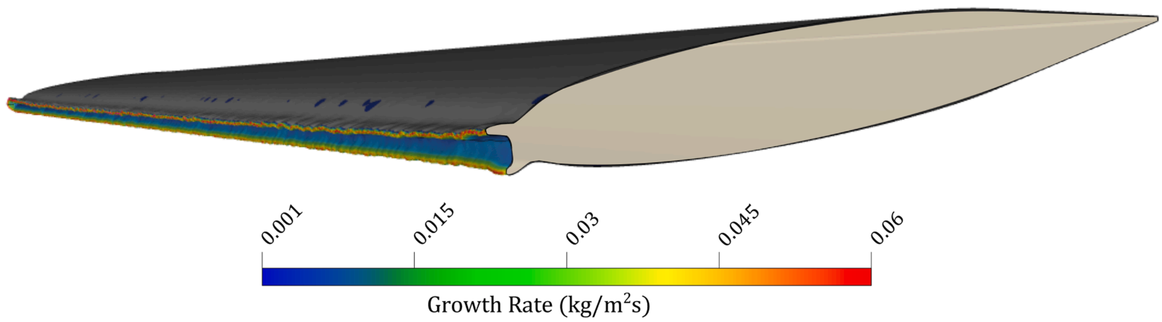


Fig. 18. Iced wing shape (wing root airfoil in foreground) with ice growth rate contour, as obtained at the last shot.

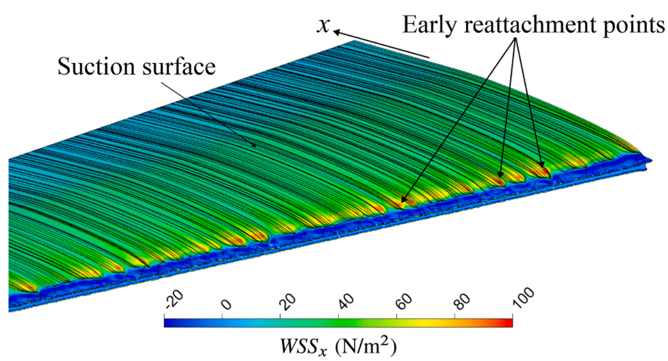


Fig. 19. Iced wing at $\alpha = -3\,\mathrm{deg}$, suction surface WSS_x contour with WSS surface streamlines.

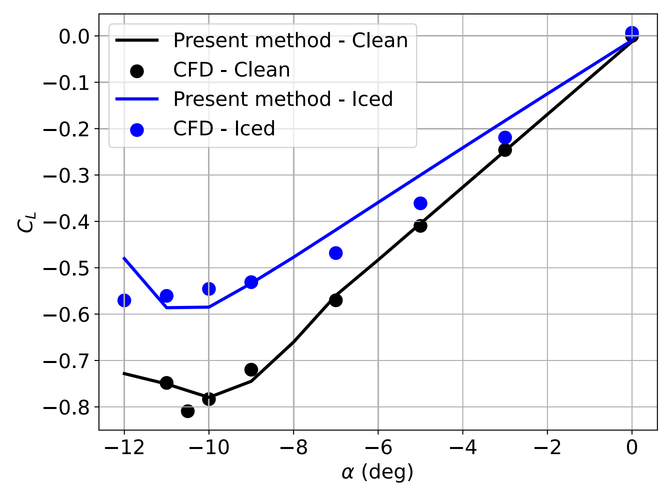


Fig. 20. Comparison of lift curves obtained using the present 2.5D method and high-fidelity CFD simulations.

immediately at the iced leading edges, as indicated by the negative component WSS_x of the wall shear stress along an x direction positively oriented towards the wing's trailing edge; the flow then reattaches shortly downstream. Localized regions of elevated WSS_x, shown in orange and red, correspond to these early reattachment points. As described by Page et al. [17], the interaction between these vortices and the surrounding flow generates localized zones of favorable pressure gradient, which mitigate some of the adverse aerodynamic effects of ice accretion and contribute to lift preservation at increasing absolute α.

Despite its reduced fidelity, the surrogate 2.5D method provides a reasonable approximation of the iced configuration's aerodynamic characteristics, although a slight underprediction in lift gradient is observed. The discrepancy in lift slope, quantified as approximately 14 %, reflects the surrogate's inability to capture complex 3D flow phenomena associated with iced geometries – particularly those related to leading-edge vortex dynamics.

The suitability of the present 2.5D approach in preliminary design is primarily justified by the substantial reduction in computational cost it offers compared to full three-dimensional CFD simulations. By capturing key aerodynamic trends with acceptable accuracy, the framework facilitates early-stage assessment of icing effects without incurring the prohibitive computational overhead of high-fidelity methods. A quantitative comparison of computational time of the two approaches is presented in Table 11. Although CFD runtimes are known to vary significantly with solver settings and convergence criteria, the surrogate approach demonstrates a computational time reduction of approximately three to four orders of magnitude in this case. This gain in efficiency underscores the practical advantage of the proposed methodology for iterative design and optimization tasks.

It is worth noting that the prediction of two-dimensional ice shapes is nearly instantaneous, as the input data are directly fed into the neural network. In contrast, the prediction of two-dimensional lift curves involves a preprocessing step in which the raw airfoil coordinates (both iced and clean) are converted into SDFs, resulting in a comparatively higher overall inference time for this component of the framework.

Table 11
Comparison of execution times between the proposed method and full 3D CFD icing simulations. CFD computations were performed on a workstation equipped with two Intel Xeon E5-2670 CPUs (2.6 GHz, 8 cores each) and 128 GB of RAM. The proposed method was implemented in Python 3.11.7 and executed on the same hardware without the need for parallelization.

Present method	2D iced shapes < 1 s	2D lift curves ≈ 20 s	2.5D execution ≈ 30 s	Total time ≈ 50 s
CFD analysis	Clean wing lift curve ≈ 12 h	Iced wing shape ≈ 48 h	Iced wing lift curve ≈ 36 h	Total time ≈ 96 h

5. Conclusions and further developments

This study presents a low-computational-cost framework for predicting the aerodynamic performance of clean and iced wings by coupling a nonlinear Vortex Lattice Method with neural network-based surrogate models. The methodology has been applied to wings adopting a symmetrical supercritical airfoil, parametrized by its leading-edge radius and thickness-to-chord ratio. The key innovation lies in replacing direct high-fidelity simulations with a hybrid approach that leverages machine learning to supply two-dimensional aerodynamic data to a VLM-based aerodynamic prediction tool, enabling rapid yet reasonably accurate performance predictions.

The proposed method integrates two sequential neural networks: a Multi-Layer Perceptron that predicts iced airfoil shapes trained on inviscid data, followed by a Convolutional Neural Network trained on high fidelity data and utilizing SDF geometry representations to predict the associated lift curves. This setup enables the computation of the lift curve of un-iced and iced wings in two minutes per configuration on a standard workstation, making it highly suitable for integration within multidisciplinary design optimization loops.

Validation against high-fidelity three-dimensional numerical simulations using ANSYS Fluent demonstrates that the framework achieves strong agreement for the clean wing case, with prediction errors below 6 % across all angles of attack. For the iced wing, the lift curve slope is underestimated by approximately 14 %, and although the stall behavior is not captured by the model, the predicted lift coefficients at high angles of attack remain in reasonable agreement with high fidelity numerical results. These outcomes underscore the capability of the proposed method to capture key aerodynamic trends induced by ice accretion while maintaining computational efficiency. Future developments will aim to enhance model accuracy in terms of lift curve prediction.

The current limitation to symmetrical supercritical airfoils will be addressed by expanding the training dataset to include cambered airfoils with a broader range of geometrical parameters. In addition, the framework will be extended to include the prediction of the drag and moment coefficients, offering a more complete aerodynamic assessment. In conclusion, this study presents a computationally efficient and flexible surrogate modeling framework that integrates data-driven approaches with classical aerodynamic methods, offering a promising tool for rapid aerodynamic analysis in early design phases where the computational cost of predictive tools is a critical constraint.

6. Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT (developed by OpenAI) to improve language and readability. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

CRediT authorship contribution statement

Ciro Cuozzo: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Vincenzo Cusati:** Writing – review & editing, Writing – original draft, Data curation, Conceptualization; **Agostino De Marco:** Writing – review & editing, Visualization, Supervision, Investigation, Formal analysis, Conceptualization; **Salvatore Corcione:** Supervision, Software, Resources, Methodology, Investigation, Formal analysis, Conceptualization; **James H. Page:** Writing – review & editing, Validation; **Alessandro Zanon:** Writing – review & editing, Supervision.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] G. Kulesa, Weather and aviation: how does weather affect the safety and operations of airports and aviation, and how does FAA work to manage weather-related effects?, 2003. https://api.semanticscholar.org/CorpusID:108023423.

[2] M. Filip, P. Kandráč, A statistical review of aviation airframe icing accidents, 2012. https://api.semanticscholar.org/CorpusID:114618246.

[3] Icing, 2024, Accessed: 2024-10-28, https://www.aopa.org/training-and-safety/air-safety-institute/accident-analysis/weather/icing.

[4] M.B. Bragg, Aircraft aerodynamic effects due to large droplet ice accretions, in: 34th Aerospace Sciences Meeting and Exhibit, 1996. https://doi.org/10.2514/6.1996-932

[5] S. Lee, M.B. Bragg, Investigation of factors affecting iced-airfoil aerodynamics, J. Aircraft 40 (3) (2003). https://doi.org/10.2514/2.3123

[6] A. Reehorst, J. Chung, M. Potapczuk, Y. Choo, Study of icing effects on performance and controllability of an accident aircraft, J. Aircraft 37 (2) (2000). https://doi.org/10.2514/2.2588

[7] European Union Aviation Safety Agency, CS-25.143 – Controllability and Manoeuvrability, 2023, (https://www.easa.europa.eu/en/document-library/easy-access-rules/online-publications/easy-access-rules-large-aeroplanes-cs-25?page=7#_DxCrossRefBm1730187425). Easy Access Rules for Large Aeroplanes (CS-25), last updated 30 January 2023.

[8] G.A. Ruff, B. Berkowitz, Users manual for the NASA Lewis ice accretion prediction code (LEWICE), 1990. https://api.semanticscholar.org/CorpusID:107371134.

[9] M.G. Potapczuk, LEWICE/E: an Euler based ice accretion code, 1992. https://api.semanticscholar.org/CorpusID:122071816.

[10] K. Hasanzadeh, E. Laurendeau, I. Paraschivoiu, Quasi-steady convergence of multistep Navier–Stokes icing simulations, J. Aircraft 50 (4) (2013) 1261–1274. https://doi.org/10.2514/1.C032197

[11] H. Beaugendre, F. Morency, W.G. Habashi, Development of a second generation in-flight icing simulation code, J. Fluids Eng. Trans. ASME 128 (2006) 378–387. https://api.semanticscholar.org/CorpusID:121217816.

[12] G. Mingione, V. Brandi, Ice accretion prediction on multielement airfoils, J. Aircraft 35 (1998) 240–246. https://api.semanticscholar.org/CorpusID:109773652.

[13] D. Pena, Y. Hoarau, E. Laurendeau, Development of a three-dimensional icing simulation code in the NSMB flow solver, Int. J. Eng. Syst. Model. Simul. 8 (2016) 86. https://doi.org/10.1504/IJESMS.2016.075544

[14] S. Li, R. Paoli, Modeling of ice accretion over aircraft wings using a compressible openFOAM solver, Int. J. Aerospace Eng. (2019). https://doi.org/10.1155/2019/4864927

[15] Y. Han, J. Palacios, Validation of a LEWICE-based icing code with coupled heat transfer prediction and aerodynamics performance determination, 2017. https://doi.org/10.2514/6.2017-3416

[16] Y. Cao, C. Ma, Q. Zhang, J. Sheridan, Numerical simulation of ice accretions on an aircraft wing, Aerosp. Sci. Technol. 23 (1) (2012) 296–304. 35th ERF: Progress in Rotorcraft Research, https://doi.org/10.1016/j.ast.2011.08.004

[17] J. Page, I. Ozcer, A. Zanon, M. De Gennaro, IMPACT: numerical study of aerodynamics of an iced forward-swept tail with leading edge extension, in: Proceedings of the International Conference on Icing of Aircraft, Engines, and Structures, SAE International, 2023, p. 10. https://doi.org/10.4271/2023-01-1371

[18] S. Corcione, V. Cusati, V. Memmolo, F. Nicolosi, R.L. Sandin, Impact at aircraft level of elastic efficiency of a forward-swept tailplane, Aerospace Sci. Technol. (2023). https://api.semanticscholar.org/CorpusID:259451396.

[19] S. Corcione, M. Mandorino, V. Cusati, Beyond conventional: an integrated aerostructural optimization approach for innovative tailplane configurations, Aerospace Sci. Technol. (2024). https://api.semanticscholar.org/CorpusID:270085567.

[20] S. Jung, R. Myong, T. Cho, Efficient prediction of ice shapes in CFD simulation of in-flight icing using a POD-based reduced order model, in: Proceedings of the SAE Technical Paper, 2011-38-0032, SAE International, 2011. https://doi.org/10.4271/2011-38-0032

[21] S. Li, J. Qin, M. He, R. Paoli, Fast evaluation of aircraft icing severity using machine learning based on XGBoost, Aerospace 7 (4) (2020) 36. https://doi.org/10.3390/aerospace7040036

[22] S. Strijhak, D. Ryazanov, K. Koshelev, A. Ivanov, Neural network prediction for ice shapes on airfoils using icefoam simulations, Aerospace 9 (2) (2022) 96. https://doi.org/10.3390/aerospace9020096

[23] D. Massegur, D. Clifford, A.D. Ronch, R. Lombardi, M. Panzeri, Low-dimensional models for aerofoil icing predictions, Aerospace 10 (5) (2023) 444. https://doi.org/10.3390/aerospace10050444

[24] K. Balla, R. Sevilla, O. Hassan, K. Morgan, An application of neural networks to the prediction of aerodynamic coefficients of aerofoils and wings, Appl. Math. Model. 96 (2021) 456–479. https://doi.org/10.1016/j.apm.2021.03.019

[25] H. Liu, Z. Li, F. Lu, An airfoil aerodynamic parameters calculation method based on convolutional neural network https://static1.squarespace.com/static/5dae869b8ea34a7aeca81c58/t/5daebe2ba6a0fa2543f8ca3f/1571733042316/Report_Airfoiler_Haolin+Liu_Zi+Li_Fei+Lu.pdf.

[26] S. Bhatnagar, Y. Afshar, S. Pan, et al., Prediction of aerodynamic flow fields using convolutional neural networks, Comput. Mech. 64 (2019) 525–545. https://doi.org/10.1007/s00466-019-01740-0

[27] Y. Zhang, W.-J. Sung, D. Mavris, Application of convolutional neural network to predict airfoil lift coefficient, AIAA, 2018. https://doi.org/10.2514/6.2018-1903

[28] C. Duru, H. Alemdar, Ö.U. Baran, CNNFOIL: Convolutional encoder decoder modeling for pressure fields around airfoils, Neural Comput. Appl. 33 (2021) 6835–6849. https://doi.org/10.1007/s00521-020-05461-x

[29] X. Guo, W. Li, F. Iorio, Convolutional neural networks for steady flow approximation, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, New York, NY, USA, 2016, pp. 481–490. https://doi.org/10.1145/2939672.2939738

[30] S. Corcione, A. De Marco, V. Cusati, A data-driven methodology to predict ice-induced aerodynamic degradation applied to aircraft tailplane design, Chin. J. Aeronaut. (2025) 103476. https://doi.org/10.1016/j.cja.2025.103476

[31] T. Zhao, W. Qian, J. Lin, H. Chen, H. Ao, G. Chen, L. He, Learning mappings from iced airfoils to aerodynamic coefficients using a deep operator network, J. Aerospace Eng. 36 (5) (2023) 04023035. https://doi.org/10.1061/JAEEEZ.ASENG-4508

[32] P. Della Vecchia, F. Nicolosi, M. Ruocco, An improved high-lift aerodynamic prediction method for transport aircraft, CEAS Aeronaut. J. 10 (2019) 795–804. https://doi.org/10.1007/s13272-018-0349-5

[33] G. Fujiwara, M. Bragg, 3D computational icing method for aircraft conceptual design, in: Proceedings of the 2017 AIAA Aviation Forum, 2017. https://doi.org/10.2514/6.2017-4375

[34] C.D. Harris, NASA Supercritical Airfoils: A Matrix of Family-Related Airfoils, Technical Paper NASA TP-2969, NASA Langley Research Center, Hampton, VA, 1990. L-16625.

[35] U.S. Government, Appendix C to Part 25 – Atmospheric Icing Conditions, 2024, (Electronic Code of Federal Regulations). Last amended on February 16, 2024, up to date as of March 20, 2024, https://www.ecfr.gov/current/title-14/chapter-I/subchapter-C/part-25/appendix-C-to-part-25.

[36] L. Makkonen, Heat transfer and icing of a rough cylinder, Cold Reg. Sci. Technol. 10 (1985) 105–116. https://doi.org/10.1016/0165-232X(85)90022-9

[37] B.L. Messinger, Equilibrium temperature of an unheated icing surface as a function of air speed, J. Aeronaut. Sci. 20 (1) (1953) 29–42. https://doi.org/10.2514/8.2520

[38] J. Shin, B. Berkowitz, H.H. Chen, T. Cebeci, Prediction of ice shapes and their effect on airfoil performance, 1991. https://api.semanticscholar.org/CorpusID:119372032.

[39] B. Xu, N. Wang, T. Chen, M. Li, Empirical Evaluation of Rectified Activations in Convolutional Network, 2015, https://arxiv.org/abs/1505.00853.

[40] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, 2017, https://arxiv.org/abs/1412.6980. 1412.6980

[41] R. Tibshirani, Regression shrinkage and selection via the lasso, J. Roy. Statist. Soc. B (Methodol.) 58 (1) (2018) 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

[42] T.D. Economon, F. Palacios, S.R. Copeland, T.W. Lukaczyk, J.J. Alonso, SU2: an open-source suite for multiphysics simulation and design, AIAA J. (2015). Published Online: 29 Dec 2015, https://doi.org/10.2514/1.J053813.

[43] Construct2D, 2018, Accessed: 2024-11-13, https://sourceforge.net/projects/construct2d/.

[44] P. Lavoie, E. Radenac, G. Blanchard, E. Laurendeau, P. Villedieu, An immersed boundary method for multi-step ice accretion using a level-set, AIAA Aviation 2021 Forum (2021). https://doi.org/10.2514/6.2021-2630

[45] K. Hormann, A. Agathos, The point in polygon problem for arbitrary polygons, Comput. Geom. 20 (3) (2001) 131–144. https://doi.org/10.1016/S0925-7721(01)00012-8

[46] I. Ansys ®, Fluent Icing, 2024. Release 2024R2.

[47] I. Ansys ®, Design Modeler, 2024. Release 2024R2.

[48] I. Ansys ®, Fluent Meshing, 2024. Release 2024R2.

[49] I. Ozcer, A. Pueyo, F. Menter, S. Hafid, et al., Numerical study of iced swept-Wing performance degradation using RANS, in: SAE Technical Paper, 2023-01-1402, SAE International, 2023. https://doi.org/10.4271/2023-01-1402