

MULTI-SCALE LINEAR FEATURE MODULATION DEEP NEURAL OPERATOR FOR RAPID PRESSURE COEFFICIENT PREDICTION

Wei-Feng Zhang[1], Huan Zhao[2], She-Kun Wang[3]

[1-3]School of Aeronautics, Northwestern Polytechnical University, Xi'an Shaanxi, 710072, China

Abstract

Surrogate models based on deep neural operators are emerging as a highly efficient numerical solution paradigm. However, vanilla deep neural operator shows low sensitivity to local variations in aircraft geometry in practical modeling scenarios. To tackle this challenge, this study proposes a Multi-Scale Linear Feature Modulation Deep Neural Operator (MSFiLM-DeepONet), aiming to establish a high-precision end-to-end mapping from high-dimensional geometric shape parameters to the wing surface pressure coefficient distribution. The core innovation of the model is that we replace the plain MLP trunk network with a Transformer-style encoder, which enables the trunk network to extract spatial correlations of the input geometry. This design reduces the computational complexity of the attention mechanism while improving the sensitivity of the deep operator network to geometric shape variations. Meanwhile, the introduced linear feature modulation mechanism generates two sets of modulation parameters from the global geometric parameters received by the branch network, to dynamically adjust the outputs of the trunk network. This allows the model to dynamically enhance sensitivity to high-gradient regions such as shock zones and separation regions according to specific wing geometries, achieving precise conditional control over flow field physical quantities via geometric parameters. Experimental validation on the transonic pressure coefficient prediction task for parameterized deformed M6 wings with high-dimensional geometric parameters verifies the superior performance of the proposed model. Trained with only 480 high-fidelity samples, the model achieves an average relative L2 error of 6.1% in pressure coefficient distribution across 200 unseen test geometries, and simultaneously exhibits excellent generalization and robustness. Furthermore, we successfully applied this model to reduce the drag coefficient of the M6 wing by 25.2 counts.

Keywords: transonic flow; surrogate model; Feature-wise Linear Modulation; Deep Neural Operator; aerodynamic shape optimization

1. Introduction

Strongly nonlinear phenomena such as shock–boundary layer interactions in transonic flow fields pose substantial challenges to the accurate prediction of flow field quantities. Particularly in aerodynamic design optimization scenarios, minor perturbations to aircraft geometry can induce pronounced alterations in global flow structures, further compounding the difficulty of the problem.

In recent years, data-driven neural network methods have become a prominent surrogate modeling approach in the field of flow prediction[1,2]. Leveraging their strong capacity to fit complex nonlinear mappings between inputs and outputs, neural networks offer a viable pathway for developing fast solvers that can replace computationally expensive computational fluid dynamics (CFD) numerical simulations. Nevertheless, most existing neural network-based flow modeling methods[3,4] remain restricted to weakly nonlinear flow regimes such as subsonic flows. More critically, conventional neural network architectures generally lack discretization invariance: their predictive performance is constrained by the discrete grid topology used for training, and they cannot directly perform interpolative prediction of physical quantities at arbitrary spatial coordinates within the computational domain. This limitation severely restricts their generalization capability in engineering scenarios such

https://doi.org/10.71945/icas2026_0473

as mesh-free and variable-resolution simulations.

As an emerging class of data-driven modeling frameworks, neural operators inherently possess discretization invariance across both spatial and temporal dimensions, exhibiting stronger generalization performance than conventional neural networks. Currently, mainstream neural operator architectures include the Deep Operator Network[5], the Fourier Neural Operator[6], and others[7]. These methods have been validated as efficient surrogate modeling tools for a wide range of scientific and engineering problems where traditional numerical solvers incur prohibitive computational costs, and their application has gradually expanded into the field of aerodynamics.

While existing neural operators have demonstrated promising application potential in conventional aerodynamic problems[8,9], systematic investigations into transonic flows remain relatively scarce. In particular, for complex configurations with variable geometries, the prediction accuracy and geometric generalization capability of such models for strongly nonlinear flow structures — including shock–boundary layer interactions — have not yet been thoroughly validated. Accordingly, there is an urgent need to thoroughly explore the applicability of neural operators to variable-geometry transonic flow problems, and to develop a generalized operator-based modeling framework to address such strongly nonlinear, high-sensitivity aerodynamic prediction tasks.

In summary, this paper proposes a multi-scale linear feature modulated deep neural operator. It can accurately predict the pressure coefficient distribution over the wing surface in high-dimensional design spaces under transonic flow conditions. High-fidelity lift and drag coefficients are further obtained via integral calculation, which well meets the requirements for high-dimensional refined aerodynamic design of wings.

2. Data Generation

This study employs an open-source CFD solver SU2[10] to conduct numerical simulations of turbulent flow over the ONERA M6 wing[11]. The ONERA M6 wing is frequently used to test the reliability of computational fluid dynamics (CFD) numerical simulation methods and optimization algorithms in the transonic regime. The numerical simulations are based on the Reynolds-Averaged Navier-Stokes (RANS) equations, coupled with the Spalart-Allmaras (SA) turbulence model. The inflow conditions are uniformly set to a Mach number of 0.8395, an angle of attack of $3.06°$, and a Reynolds number of 11.72×10^{6}. Wing shape parameterization is achieved using the Free-Form Deformation (FFD) method, as illustrated in Figure 1. Eight airfoil sections are selected for wing shape control, with each section configured with 12 independent control points, thereby constructing a design space with a dimensionality of $D=96$. All control points are only allowed to move in the z-direction to drive differential deformation of the wing geometry. To adequately cover the feature distribution within the design space, the Latin Hypercube Sampling (LHS)[12] method is used to generate $5D=480$ sample sets, corresponding to 480 distinct wing configurations.

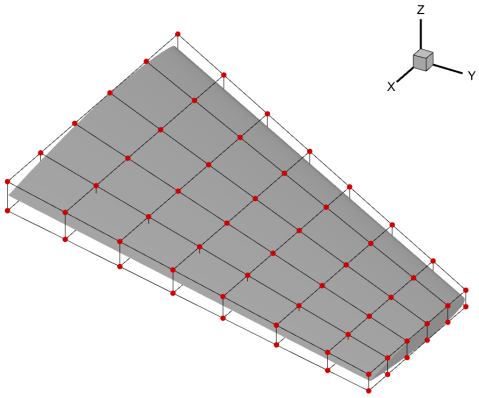
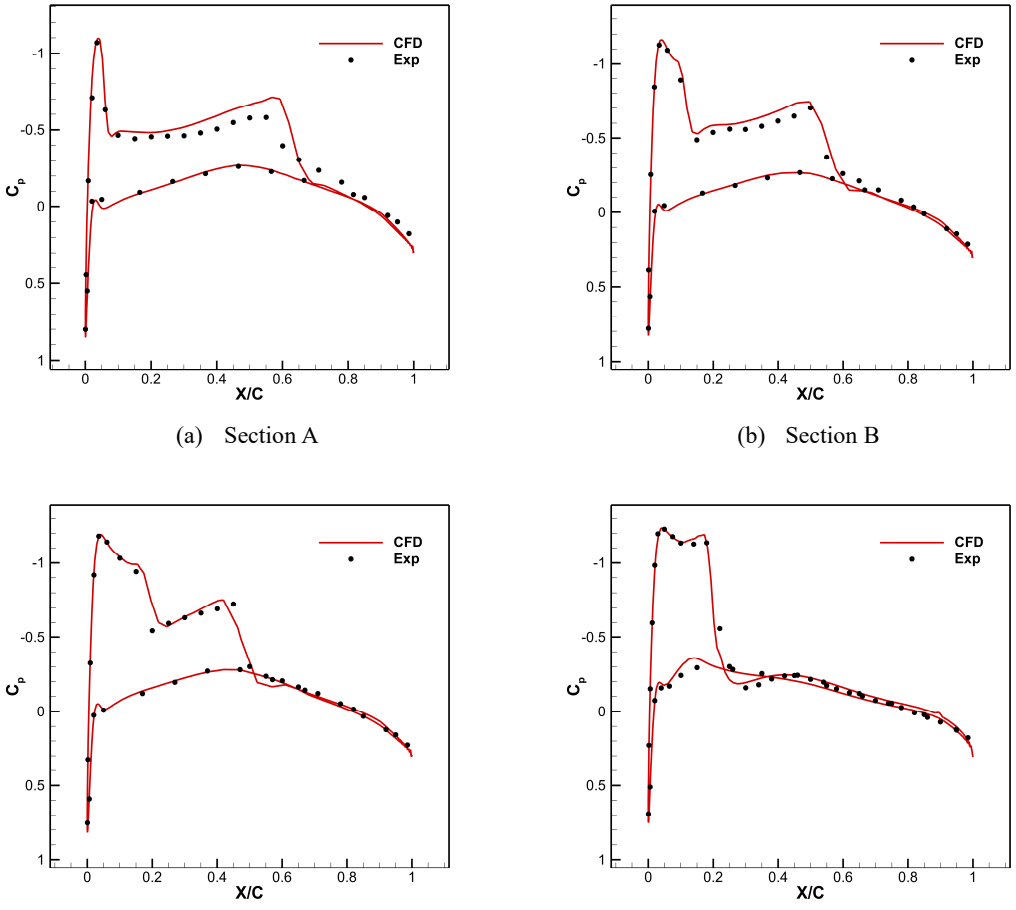


Figure 1 FFD control box of the ONERA M6 wing

Figure 2 presents a comparison between the numerical calculation and experimental results of the pressure distribution at different spanwise positions of the M6 wing. It can be seen that the pressure distribution obtained by CFD calculation agrees well with the experimental results.



(a) Section A (b) Section B

(c) Section C (d) Section D

Figure 2 Comparison of the computed pressure distribution of the M6 wing and the wind tunnel test results

3. Methodology and problem set up

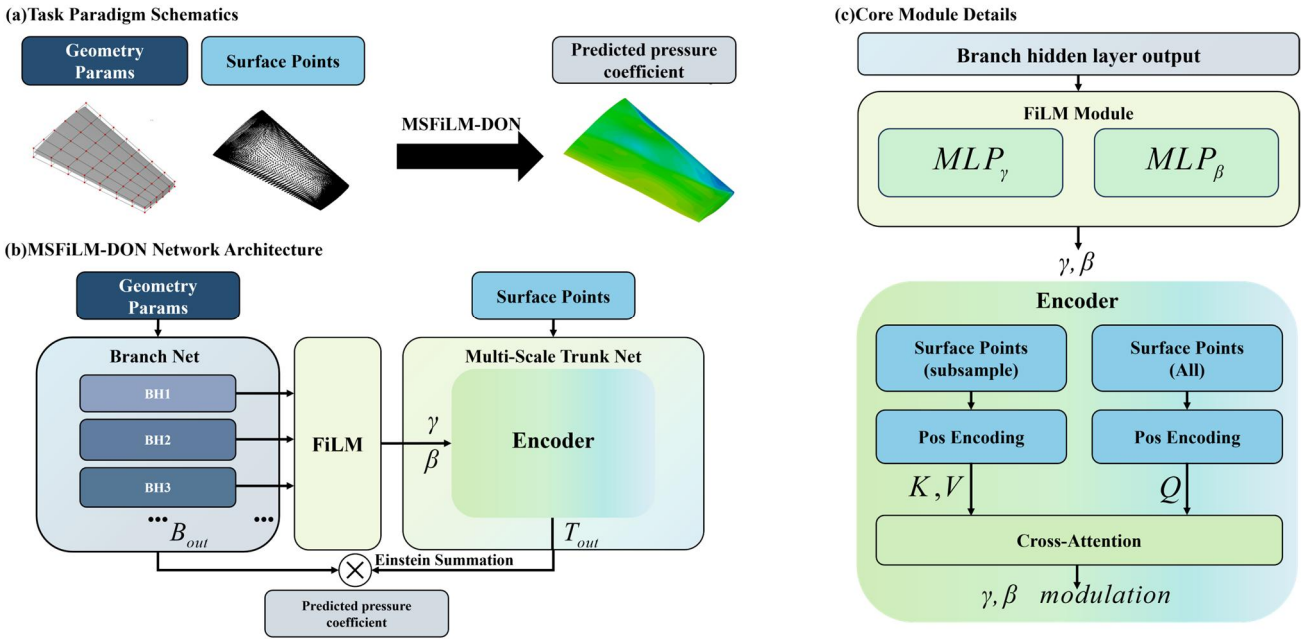


Figure 3 Architecture for MSFiLM-DeepONet

We consider the problem of surface pressure coefficient prediction for transonic ONERA M6 wings under free-form deformation (FFD) parameterization. Let $\Omega \subset \mathbb{R}^3$ denote the continuous point set of the wing surface geometry, and $\boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^{N_m}$ be an N_m -dimensional parameter vector that characterizes the geometric shape. Let the discretized surface sensor coordinates be $G = \{\mathbf{x}_i\}_{i=1}^N \subset \mathbb{R}^3$, with the corresponding pressure coefficient field denoted as $\mathbf{p} = (p_1, p_2, \ldots, p_N)^\top$, where $p_i = C_p(\mathbf{x}_i; \boldsymbol{\mu})$ .We define a nonlinear operator $\mathcal{G}: \mathcal{P} \to C(\Omega)$, which maps the geometric parameter vector to the surface pressure distribution function:

$$\mathcal{G}(\boldsymbol{\mu})(\mathbf{x}) = C_p(\mathbf{x}; \boldsymbol{\mu}), \quad \forall \boldsymbol{\mu} \in \mathcal{P}, \forall \mathbf{x} \in \Omega \tag{1}$$

The objective is to learn a surrogate model $\mathcal{G}_\theta$ parameterized by trainable weights θ, such that:

$$\mathcal{G}_\theta(\boldsymbol{\mu})(\mathbf{x}_i) \approx C_p(\mathbf{x}_i; \boldsymbol{\mu}), \quad \forall \boldsymbol{\mu} \in \mathcal{P}, \forall i = 1, \ldots, N \tag{2}$$

Like the vanilla DeepONet, the MSFiLM-DeepONet shown in Figure 3 is composed of a branch network and a trunk network. The most critical distinction lies in the trunk network architecture. Instead of using a conventional plain MLP, the trunk network adopts an encoder design that reorganizes the representation of the full point cloud by performing cross-attention between the complete point set and latent geometric structures, and the branch network performs feature modulation on the trunk network via the FiLM layer. In what follows, we omit the batch dimension of samples and present the rigorous forward propagation formulation of the network model, along with a detailed account of the function of each component.

First, we define the branch network. For $L \in \mathbb{N}, \ell = 1, \ldots, L_b - 1$ we define the weight matrix $\mathbf{W}_\ell^b \in \mathbb{R}^{n_{\ell-1}^b \times n_\ell^b}$ and the bias term $\mathbf{b}_\ell^b \in \mathbb{R}^{1 \times n_\ell^b}$. we thus obtain the forward propagation formula:

$$\begin{aligned}
&\mathbf{B}_0 = \boldsymbol{\mu} \\
&\mathbf{B}_1 = \sigma_b\left(\mathbf{W}_1^b \mathbf{B}_0 + \mathbf{b}_1^b, a_1^b\right) \\
&\mathbf{B}_\ell = \sigma_b\left(\mathbf{W}_\ell^b \mathbf{F}_{\ell-1} + \mathbf{b}_\ell^b, a_\ell^b\right) \qquad \ell = 2, \ldots, L_b - 1 \\
&\mathbf{F}_1 = \mathbf{B}_1 \\
&\mathbf{F}_\ell = \mathbf{B}_\ell + \mathbf{F}_{\ell-1} \qquad \ell = 2, \ldots, L_b - 1
\end{aligned} \tag{3}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{N_m}$, $n_0^b = N_m$, $n_\ell^b = D_\ell^b (\ell \geq 1)$, N_m is the dimension of the input Free-Form Deformation parameters of the branch network, L_b is the number of layers of the branch network, $\mathbf{B}_\ell$ is the output

of the ℓ layer, $\mathbf{F}_\ell$ is the recursive result obtained by summing the corresponding term $\mathbf{B}_\ell$ and the preceding term $\mathbf{F}_{\ell-1}$. D_ℓ^b is the width of the ℓ layer, σ_b is an adaptive activation function called Rowdy[13], $a_\ell^b \in \mathbb{R}^{N_k}$, N_k is the number of learnable parameters about the activation function.

To address the limitations of conventional fully-connected trunk networks—namely point-wise independent computation, failure to model long-range spatial dependencies, and insufficient sensitivity to local geometric deformations—the proposed trunk network employs a sparse latent representation encoder-attention architecture. The core design principle is as follows: a small set of sparse latent points obtained via random subsampling serve as the global geometric representation, which compresses and aggregates the global geometric information of the full point cloud into a compact latent form. The global features are then mapped back to an arbitrarily discretized surface point set via cross-attention. This design reduces the computational complexity of the attention mechanism from $O(N^2)$ to $O(N \cdot N_{\text{latent}})$ while retaining complete global spatial modeling capability. Furthermore, it inherently supports point cloud inputs with arbitrary discretization patterns, which satisfies the discretization invariance requirement of operator learning.

First, rotary positional encoding[14] is applied to map point cloud coordinates G into high-dimensional features. N_{latent} points are then uniformly sampled from the full features $\tilde{G}$ to form the latent geometric representation. We subsequently execute cross-attention, where full geometric features serve as queries and the latent representation acts as keys and values, to aggregate latent geometric information into full geometric features.

$$\tilde{\mathbf{G}} = \mathrm{RoPE}(\mathbf{G}), \tilde{\mathbf{G}} \in \mathbb{R}^{N_p \times D^t} \tag{4}$$

$$\mathbf{E} = \mathrm{UniformSample}\left(\tilde{\mathbf{G}}, \mathbf{K}\right), \mathbf{E} \in \mathbb{R}^{N_{latent} \times D^t} \tag{5}$$

$$\mathbf{E}_G = \mathrm{CrossAttn}\left(\mathbf{Q} = \tilde{\mathbf{G}}, \mathbf{KV} = \mathbf{E}\right), \mathbf{E}_G \in \mathbb{R}^{N_p \times D^t} \tag{6}$$

Next, we focus on how the feature-wise linear modulation mechanism works. Inspired by the ideas of feature-wise linear modulation and Fusion DeepONet[9], we design a FiLM[15] layer to replace the simple element-wise multiplication operation in Fusion DeepONet. This design can reduce the number of network parameters while achieving a stronger feature fusion effect.

For the branch residual features $\mathbf{F}_\ell \in \mathbb{R}^{D_\ell^b}$ and the trunk features $\mathbf{E}_G \in \mathbb{R}^{N_p \times D^t}$, we first pass them through two multilayer perceptron to generate parameter matrices $\boldsymbol{\gamma}, \boldsymbol{\beta}$. Only the output dimension is constrained, requiring the width of the output layer to be consistent with that of the trunk network.

$$\begin{aligned} \boldsymbol{\gamma}(\mathbf{F}_{L_b-1}) &= MLP_\gamma(\mathbf{F}_{L_b-1}) \\ \boldsymbol{\beta}(\mathbf{F}_{L_b-1}) &= MLP_\beta(\mathbf{F}_{L_b-1}) \\ FiLM(\mathbf{E}_G, \mathbf{F}_{L_b-1} \mid \boldsymbol{\gamma}, \boldsymbol{\beta}) &= \boldsymbol{\gamma}(\mathbf{F}_{L_b-1}) \odot \mathbf{E}_G + \boldsymbol{\beta}(\mathbf{F}_{L_b-1}) \end{aligned} \tag{7}$$

where the output of MLP_γ is $\boldsymbol{\gamma} \in \mathbb{R}^{D^t}$, the output of MLP_β is $\boldsymbol{\beta} \in \mathbb{R}^{D^t}$, and $\odot$ denotes element-wise multiplication. Since the tensor dimensions of $\boldsymbol{\gamma}, \boldsymbol{\beta}, \mathbf{E}_G$ are not uniform, it is necessary to expand $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ along the penultimate tensor dimension before performing element-wise multiplication on the last dimension.

Finally, fusion processing is performed on the outputs. The trunk output is $\mathbf{T}_{out} \in \mathbb{R}^{N_p \times D^t}$, the final branch output is $\mathbf{B}_{out} \in \mathbb{R}^{D_{L_b}}$, and the global bias is $\boldsymbol{\beta}_G \in \mathbb{R}^{N_p}$, such that the final prediction is:

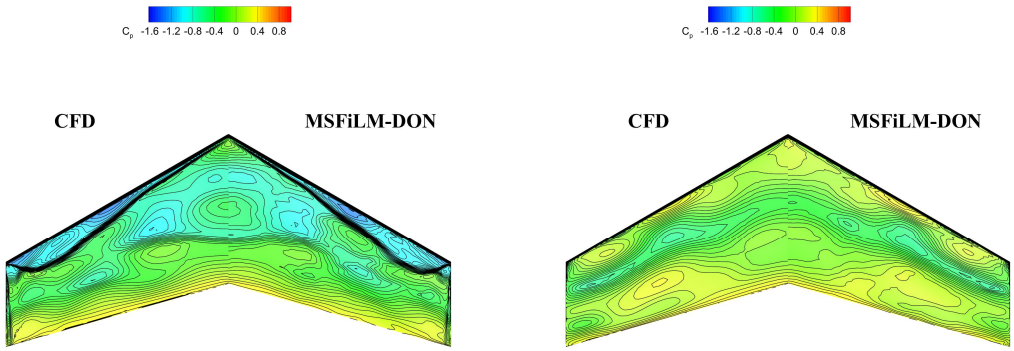
$$\begin{aligned} \mathbf{B}_{out} &= \mathbf{W}_{L_b}^b \mathbf{F}_{L_{b-1}} + \mathbf{b}_{L_b}^b \\ \mathbf{T}_{out} &= FiLM(\mathbf{E}_G, \mathbf{F}_{L_b-1} \mid \boldsymbol{\gamma}, \boldsymbol{\beta}) \\ \mathcal{G}(\boldsymbol{\mu})(\mathbf{x}) &= \mathbf{B}_{out} \otimes \mathbf{T}_{out} + \boldsymbol{\beta}_G \end{aligned} \tag{8}$$

where the final output of the branch network is $\mathbf{B}_{out}$, the trunk output is $\mathbf{T}_{out}$, and in accordance with

conventions, $D_{L_b}^b = D^t = D$.

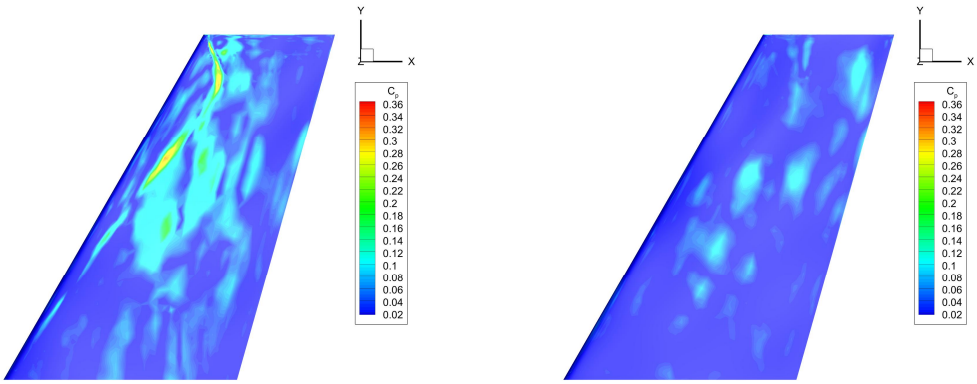
4. Experiments and optimization

Experimental tests show that on the test set consisting of 200 unseen samples, the mean relative L2 error of the pressure coefficient reaches 6.2%. In terms of computational efficiency, the entire process including field prediction, file reading and writing, as well as integrated aerodynamic force calculation takes only 0.4 seconds per case on an NVIDIA GeForce RTX 5070 Ti GPU. Since the convergence time of CFD simulations varies with the complexity of different configurations, the average time of 1836.1 seconds is adopted here, corresponding to a speedup of nearly 5000 times, One sample from the test set is selected for illustration as shown in Figure 4.



(a) Upper surface: Prediction vs. Ground Truth (b) Lower surface: Prediction vs. Ground Truth

Figure 4 Comparison of pressure coefficient between CFD and the trained MSFiLM-DeepONet model on a single test sample



(a) Absolute error of pressure coefficient on upper surface

(b) Absolute error of pressure coefficient on lower surface

Figure 5 Distribution of prediction errors on upper and lower surfaces

Leveraging MSFiLM-DeepONet's capability to directly yield solutions of wing surface pressure coefficients, we can further perform integration to obtain the pressure drag coefficient and lift coefficient, which are then adopted in the subsequent optimization workflow. The optimization problem is formulated as follows:

$$\begin{aligned} &Min \;\; C_D \\ &s.t. \;\; C_L \geq C_{L0} \\ &\quad\;\; Thick_{Optimum} \geq Thick_{Baseline} \end{aligned} \tag{9}$$

The model is trained offline on a dataset of 480 samples. Upon completion of training, it can rapidly

predict flow fields for any FFD-deformed[16] wing geometry within the predefined design space. Notably, high-fidelity CFD labels for training samples are generated using unstructured meshes, while structured meshes are utilized for flow field prediction and integration in the optimization evaluation phase. The results verify that MSFiLM-DeepONet exhibits excellent mesh independence and strong generalization capability. It maintains high prediction accuracy even when evaluated on meshes with distinct topological structures, which substantially ensures the flexibility and robustness of the subsequent optimization procedure.

For transonic flow around the ONERA M6 wing, total drag is predominantly contributed by shock-induced wave drag. Frictional drag takes up only a small fraction and varies marginally. Therefore, pressure drag is approximated as total drag for calculation and optimization in this study. Specifically, the lift coefficient C_L and pressure drag coefficient C_D of a given configuration can be rapidly calculated by integrating the wall pressure coefficient C_p field output by the neural operator over the wing surface.

Combined with the aforementioned flow field integration-based performance evaluation method, we develop an efficient optimization workflow that requires neither additional data sampling nor model retraining. Different from the conventional iterative optimization paradigm for surrogate models, the neural operator only needs one round of offline training. The pre-trained model can be called throughout optimization iterations to generate flow field solutions, and the objective function as well as constraint values are then obtained via integration. No additional high-fidelity CFD simulations or model updates are required during the entire optimization process, which thoroughly eliminates the computational overhead caused by repeated data supplementation and retraining in traditional approaches.

Furthermore, taking advantage of the automatic differentiation capability of neural networks to compute gradients, we adopt the Sequential Least Squares Programming[17] (SLSQP) algorithm for single-objective constrained optimization. The optimization goal is to minimize the total drag coefficient, subject to the constraints that the maximum thickness of the wing remains unchanged and the lift coefficient is no lower than the baseline value.

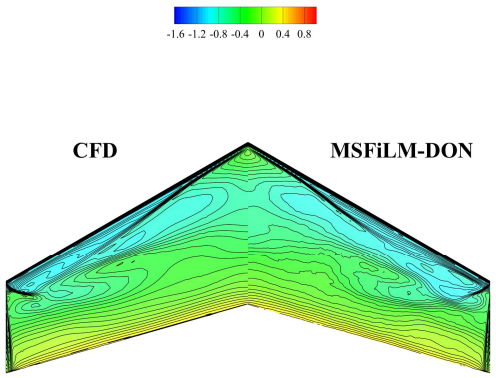
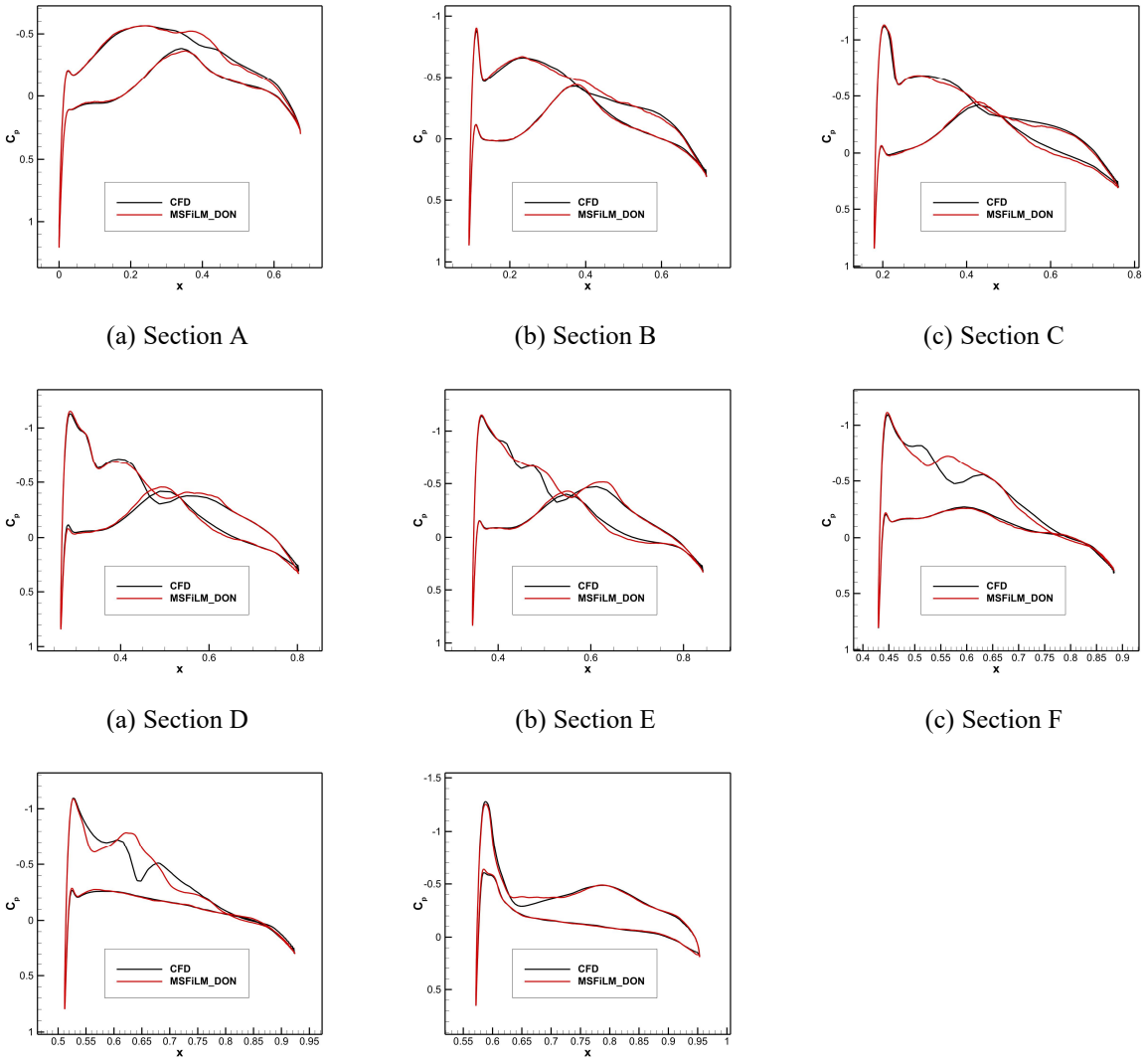


Figure 6 Comparison of MSFiLM-DON predictions of optimized wing and CFD calculation results



(a) Section A (b) Section B (c) Section C

(a) Section D (b) Section E (c) Section F

(a) Section G (b) Section H

Figure 7 Comparison between CFD results and model predictions for selected cross-sections

The optimization results demonstrate that the proposed method obtains the optimal aerodynamic configuration with merely dozens of model evaluations, without additional data sampling or model retraining. The drag coefficient is reduced substantially from the baseline value of 170.8 cts to 145.6 cts, corresponding to a total reduction of 25.2 cts

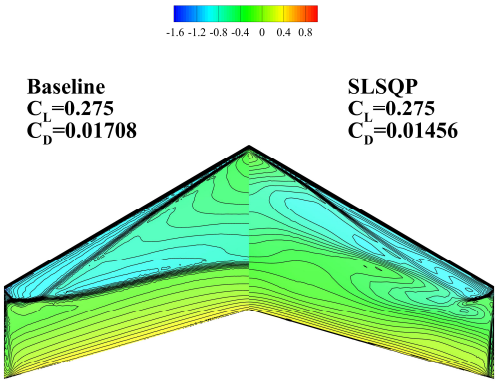
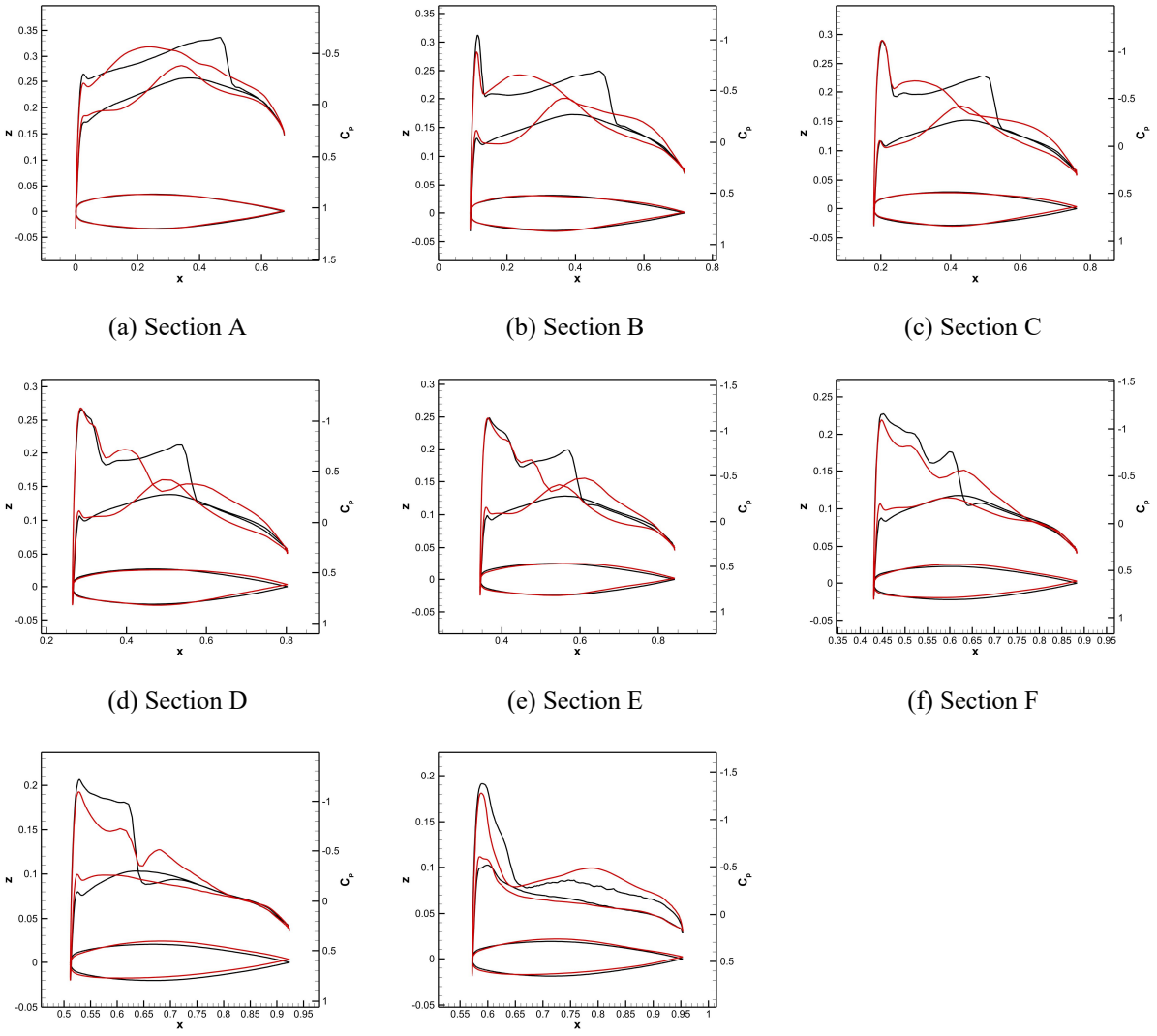


Figure 8 Pressure contours before and after optimization



(a) Section A (b) Section B (c) Section C

(d) Section D (e) Section E (f) Section F

(g) Section G (h) Section H

Figure 9 Comparison of optimized sectional shapes and pressure distributions

Flow field comparisons reveal that the optimized wing features full elimination of shock waves at multiple sections and attenuated residual shocks under transonic conditions, which greatly cuts down pressure drag. Benefiting from the capability of neural operators to generate complete flow field data in high-dimensional design spaces, the optimization algorithm can finely tune geometric parameters to control flow structures and remove unfavorable flow characteristics. It realizes high-precision aerodynamic design and significantly boosts optimization efficiency.

5. Conclusion

This study addresses the challenges in transonic wing flow prediction, including difficult-to-capture highly nonlinear shock structures, low sensitivity of conventional deep neural operators to local geometric deformations. A Multi-Scale Linear Feature Modulation Deep Neural Operator (MSFiLM-DeepONet) is proposed for rapid prediction of wing surface pressure coefficients. A high-accuracy, well-generalized and computationally efficient aerodynamic surrogate model is developed, and its effectiveness is validated through aerodynamic optimization cases.

This work presents a systematic improvement to the classic DeepONet. The conventional fully-connected MLP trunk network is replaced with a sparse latent attention encoder integrated with three-dimensional rotary positional encoding. Global geometric information is aggregated via sparse latent

points, and feature reconstruction is implemented using cross-attention. This design reduces the computational complexity of attention operations, substantially cutting computational cost while endowing the trunk network with the capability to perceive subtle local geometric variations. Furthermore, a Feature-wise Linear Modulation (FiLM) mechanism is introduced. The branch network dynamically modulates trunk features using global geometric and flow condition parameters, which greatly improves the model's accuracy in characterizing highly nonlinear transonic flows over various wing configurations.

Experimental results demonstrate that the MSFiLM-DeepONet trained on merely 480 high-fidelity samples achieves an average relative L2 error of 6.1% for pressure coefficient predictions on a test set consisting of 200 unseen wing geometries. The total time consumption for single-case prediction and performance evaluation is only 0.4 seconds, yielding a speedup of nearly 5000 times compared with traditional CFD simulations. The proposed model achieves a favorable trade-off between prediction accuracy and inference efficiency, and exhibits excellent mesh independence and generalization capability across meshes with different topological structures. Leveraging the developed neural operator, an efficient aerodynamic optimization framework is constructed. Constrained optimization is performed using the Sequential Least Squares Programming (SLSQP) algorithm. Under the constraints of fixed maximum wing thickness and maintained lift performance, the drag coefficient is effectively reduced. Shock structures on the transonic wing surface are eliminated or attenuated, enabling high-dimensional high-fidelity aerodynamic shape optimization.

Overall, the proposed MSFiLM-DeepONet breaks the performance limitations of traditional neural operators in predicting highly nonlinear transonic flows. With superior accuracy, strong generalization and ultra-high inference speed, it provides an efficient and reliable technical solution for rapid aerodynamic design and iterative optimization of aircraft configurations.

6. Contact Author Email Address

Huan Zhao: huanzhao@nwpu.edu.cn

7. Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.

8. Acknowledgement

This work was sponsored by the National Natural Science Foundation of China (NSFC) under grant No. 52572420 and the foundation of National Key Laboratory of Aircraft Configuration Design under grant No. JBGS-202502. These supports are gratefully acknowledged.

References

[1] Anhichem, M., Timme, S., Castagna, J., Peace, A. J., and Maina, M., "Bayesian Machine Learning for Predicting Wing Pressure Distributions at Transonic Flow Conditions," *AIAA Journal*, Vol. 63, No. 10, 2025, pp. 4126–4142. https://doi.org/10.2514/1.J064617

[2] Zuo, K., Ye, Z., Zhu, L., Yuan, X., and Zhang, W., "QLingNet: An Efficient and Flexible Modeling Framework for Subsonic Airfoils," No. Compendex, 2024. https://doi.org/10.48550/arXiv.2405.08248

[3] "Learning the solution operator of two-dimensional incompressible Navier-Stokes equations using physics-aware convolutional neural networks," *Journal of Computational Physics*, Vol. 535, 2025, p. 114027. https://doi.org/10.1016/j.jcp.2025.114027

[4] Thuerey, N., Weißenow, K., Prantl, L., and Hu, X., "Deep Learning Methods for Reynolds-Averaged Navier–Stokes Simulations of Airfoil Flows," *AIAA Journal*, Vol. 58, No. 1, 2020, pp. 25–36. https://doi.org/10.2514/1.J058291

[5] Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E., "Learning Nonlinear Operators via DeepONet Based on the Universal Approximation Theorem of Operators," *Nature Machine Intelligence*, Vol. 3, No. 3, 2021, pp. 218–229. https://doi.org/10.1038/s42256-021-00302-5

[6] Li, Z., Huang, D. Z., Liu, B., and Anandkumar, A., "Fourier Neural Operator with Learned Deformations for PDEs on General Geometries," *Journal of Machine Learning Research*, Vol. 24, No. Compendex, 2023.

[7] Jin, P., Meng, S., and Lu, L., "Mionet: Learning Multiple-Input Operators Via Tensor Product," *Siam Journal on Scientific Computing*, Vol. 44, No. 6, 2022, pp. A3490–A3514. https://doi.org/10.1137/22M1477751

[8] Shukla, K., Oommen, V., Peyvan, A., Penwarden, M., Plewacki, N., Bravo, L., Ghoshal, A., Kirby, R. M., and Karniadakis, G. E., "Deep Neural Operators as Accurate Surrogates for Shape Optimization," *Engineering Applications of Artificial Intelligence*, Vol. 129, 2024, p. 107615. https://doi.org/10.1016/j.engappai.2023.107615

[9] Peyvan, A., Kumar, V., and Karniadakis, G. E., "Fusion-DeepONet: A Data-Efficient Neural Operator for Geometry-Dependent Hypersonic and Supersonic Flows," *Journal of Computational Physics*, Vol. 544, 2026, p. 114432. https://doi.org/10.1016/j.jcp.2025.114432

[10] Ilie, M., and Havenar, J., "Aeroelasticity of Supersonic ONERA M6 Wing; Parametric Studies Using a Fully-Coupled Computational Approach," presented at the AIAA AVIATION 2023 Forum, 2023. https://doi.org/10.2514/6.2023-4089

[11] "AGARD Reports," NATO Science and Technology Organization. Retrieved 15 May 2026. https://www.sto.nato.int/publications/agard-reports/

[12] "Optimal Latin-Hypercube Designs for Computer Experiments," *Journal of Statistical Planning and Inference*, Vol. 39, No. 1, 1994, pp. 95–111. https://doi.org/10.1016/0378-3758(94)90115-5

[13] Jagtap, A. D., Shin, Y., Kawaguchi, K., and Karniadakis, G. E., "Deep Kronecker Neural Networks: A General Framework for Neural Networks with Adaptive Activation Functions," *Neurocomputing*, Vol. 468, 2022, pp. 165–180. https://doi.org/10.1016/j.neucom.2021.10.036

[14] Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y., "RoFormer: Enhanced Transformer with Rotary Position Embedding," *Neurocomputing*, Vol. 568, 2024, p. 127063. https://doi.org/10.1016/j.neucom.2023.127063

[15] Perez, E., Strub, F., De Vries, H., Dumoulin, V., and Courville, A., "FiLM: Visual Reasoning with a General Conditioning Layer," 2018.

[16] Sederberg, T., and Parry, S., "Free-Form Deformation of Solid Geometric Models," *International Conference on Computer Graphics and Interactive Techniques*, 1986. Retrieved 30 April 2026. https://www.semanticscholar.org/paper/Free-form-deformation-of-solid-geometric-models-Sederberg-Parry/9b86d8af6c5cce934044e482c89f786861392bd4

[17] Ma, Y., Gao, X., Liu, C., and Li, J., "Improved SQP and SLSQP Algorithms for Feasible Path-Based Process Optimisation," *Computers & Chemical Engineering*, Vol. 188, 2024, p. 108751. https://doi.org/10.1016/j.compchemeng.2024.108751