*Journal of Zhejiang University-SCIENCE A in press*
www.jzus.zju.edu.cn; www.springer.com/journal/11582
E-mail: jzus_a@zju.edu.cn

**Research Article**
https://doi.org/10.1631/jzus.A2600117

# A study on feature engineering for pressure field modeling of general aircraft under multi-condition scenarios

Hao ZHANG, Yang SHEN, Hao WANG✉, Wei HUANG✉, Yaobin NIU, Shuangxi LIU, Chaoyang LIU

*National Key Laboratory of Advanced Propulsion Technology, National University of Defense Technology, Changsha 410073, China*

**Abstract:** Aerodynamic modeling of three-dimensional aircraft under varying flight conditions is essential for aircraft optimization design. This study formulates the task as condition-guided regression of the surface pressure field using a point-cloud-based deep surrogate model, in which surface mesh points, geometric attributes, and flight-condition parameters are jointly represented. The dataset covers Mach number from 3.0 to 10.0 and angle of attack from -10° to 10°. Within this framework, the key challenge is how to fuse condition and geometry features and how to aggregate local geometry while preserving generalization across configurations and data scales. To address this issue, we construct task-specific datasets, design representative static and dynamic fusion modules, and perform systematic ablation experiments on geometric aggregation backbones. The obtained results lead to two practical guidelines. First, condition information should be introduced through progressive static fusion, whereas late concatenation and attention-based reweighting require caution; modulation inspired by deep operator networks is more promising when sufficient data are available. Second, a hybrid aggregation backbone that uses dynamic aggregation in dense point-cloud stages and static aggregation in sparse stages offers a better balance between accuracy and computational cost.

**Key words:** Surrogate model; Point cloud; Pressure field; Feature fusion; Feature aggregation



## 1 Introduction

Three-dimensional (3D) aerodynamic modeling is essential for aerodynamic shape optimization within multidisciplinary design optimization (Martins and Lambe, 2013). Although computational fluid dynamics (CFD) provides high-fidelity aerodynamic evaluations, it is computationally expensive and often requires substantial expert effort. Because aerodynamic optimization usually involves hundreds or even thousands of repeated evaluations, efficient surrogate models have become an important tool for accelerating the design process (Liu et al., 2023)

Conventional surrogate models, such as kriging (Liu et al., 2019), polynomial response surfaces (Mukherjee et al., 2021), support vector regression (Kilic and Nikbay, 2025) and early artificial neural networks (Sun and Wang, 2019), are based on establishing an explicit mapping from geometric parameters to aerodynamic coefficients. In low-dimensional parameter spaces, these approaches exhibit sufficient accuracy in prediction. For instance, Zhang et al. (2016) applied a surrogate-based multi-round optimization strategy for 2D airfoils with 17 design variables, and the number of samples for each round was set in the range of 100-300 to decrease the error of force coefficients.

However, when confronted with 3D aircraft featuring hundreds of geometric design parameters, the performance of these traditional approaches encounters bottlenecks. The reason is that these traditional methods rely on parameter-independent assumptions and local linear approximation, which struggle to handle complex nonlinear coupling effects in high-dimensional spaces (Li et al., 2022). Advanced 3D geometric representation approaches are crucial for enhancing the accuracy and efficiency of aircraft modeling. Driven by the rapid advancement of deep learning, point cloud representation approaches, which have emerged in the field of computer vision (Guo et al., 2020), have

✉ Hao WANG, Wei HUANG, weihuang@nudt.edu.cn
Hao ZHANG, https://orcid.org/0009-0004-5248-8105
Wei HUANG, https://orcid.org/0000-0001-9805-985X

Received Feb. 27, 2026; Revision accepted May 15, 2026;
Crosschecked

demonstrated significant generalization advantages over traditional pixel-based or graph-based approaches.

The pixel-based input approach employs a convolutional neural network for flow feature extraction (Chen and Akasaka, 2021). This approach has difficulty preserving the precision of the original grid points and is accompanied by a large loss of 3D spatial information owing to 2D pixels (Bhatnagar et al., 2019; Duru et al., 2021; Chen et al., 2021). The graph convolutional network can fully leverage the graph-based information, encompassing not only points but also topological relationship of grids (Hines and Bekemeyer, 2023; Massegur and Da Ronch, 2024). However, current graph-based approaches are constrained by their requirement for explicit modeling of neighborhood node topological relationships, resulting in computational overhead that scales exponentially with mesh size, hindering generalization across diverse geometries (Shao et al., 2023).

In contrast, the point-based approach considers that the relationship between points will be obtained by implicit mapping of neural networks, thus ignoring the complicated topological relationships in graph structure. By treating cell vertices as point clouds, the natural alignment between geometric coordinates and physical quantities can be realized. This consistency in data modality not only minimizes information loss caused by interpolation but also establishes a solid foundation for a condition-guided geometric registration task (Kashefi et al., 2021; Xiong et al., 2023; Shen et al., 2023).

The task of condition-guided geometric registration essentially involves effectively fusing condition features and geometric features. Existing research predominantly employs static fusion methods that directly concatenate low-dimensional condition vectors at the stage of input or output (Zhang et al., 2025). However, there remains a lack of exploration into the performance risks associated with concatenating such features at different network depths.

Dynamic fusion approaches, represented by Deep Operator Network (DeepONet) (Shukla et al., 2024) and Transformer (Shih et al., 2025), have demonstrated powerful coupling modeling capabilities in other multimodal data association tasks. However, their application to the specific domain of the aircraft surface flow field remains relatively scarce. In light of this, we aim to introduce dynamic fusion concepts into this problem and evaluate the performance of both static and dynamic fusion strategies. This will assess the suitability of different fusion mechanisms and prevent unnecessary performance degradation due to inappropriate fusion.

Meanwhile, geometric feature aggregation plays an important role in the point-cloud sampling and feature-extraction process. Existing aggregation strategies can be broadly divided into static and dynamic approaches. Static methods, represented by max pooling (Qian et al., 2022), aggregate local features using fixed operations, which may smooth or discard fine geometric details in local regions. For example, static pooling has been reported to blur edge features in point-cloud registration tasks for autonomous driving (Shi et al., 2020). Dynamic methods, exemplified by the Point Transformer (Zhao et al., 2021), use attention-based weighting to adaptively aggregate neighborhood features.

The core of dynamic aggregation is attention, which captures geometric features within local neighborhoods. This dynamic capability enables outstanding performance in multiscale feature modeling tasks such as target detection (Yuan et al., 2021) and semantic segmentation (Ma et al., 2024). However, gains may come at the cost of a surge in computational resources. Therefore, researching the performance of dynamic aggregation techniques in predicting the multi-condition surface pressure field of aircraft can provide valuable recommendations for subsequent network architecture design.

To address the aforementioned feature engineering challenges in this field, this paper builds upon the PointNeXt Regression (PointNeXtReg, PNR) (Shen et al., 2024) framework—in which the Inverted Residual Multilayer Perceptron (InvResMLP) module takes max pooling as the feature aggregation body—to systematically investigate the impact of feature fusion and aggregation mechanisms on aircraft point cloud registration. First, the study proposes fusion strategies for static concatenation at different locations and dynamic strategies inspired by neural operators and transformers. Then, the experiment compares the deployment of the InvResMLP module and Point Transformer (PT)

block, providing design references for network architecture.

The rest of this paper is organized as follows. Section 2 elaborates on the paper's methodology, including the framework of the baseline, the strategies of geometry-condition feature fusion, and the backbone of geometric aggregation. Section 3 details the creation of the dataset and the training strategies. Section 4 presents the results and discussion. Finally, we summarize this work and look forward to the future in Section 5.

## 2 Methodology

### 2.1 The base framework

As shown in Fig. 1, PointNeXtReg contains three modules, namely, Encoder, Decoder, and Regression head. The model inputs a point cloud of size $n$, which is first expanded to a 32D feature using the per-point Multilayer Perceptron (MLP). The data then goes to Encoder, which is synergistically implemented by Set Abstraction and InvResMLP modules to achieve efficient point cloud feature learning. Although both adopt neighborhood aggregation, the purpose is hierarchical abstraction and homologous feature enhancement, respectively.

Set Abstraction is designed to construct a multilevel feature pyramid, with its primary objectives being downsampling and feature abstraction. The process begins by selecting a sparse set of centroids from the input point set using farthest point sampling. Subsequently, each centroid serves as the center of a local neighborhood defined within the original point set via k-nearest neighbors and ball query searches.

In this design, the selection of neighboring points is not intended to refine the feature of any single point. Instead, its purpose is to define a local geometric region for information aggregation. The module then employs max-pooling to consolidate the information from this neighborhood into a single feature vector, which is assigned to the corresponding centroid. This achieves a reduction in spatial resolution while simultaneously elevating the local level of the feature.

In contrast, the InvResMLP module focuses on deep feature enhancement and propagation at a single resolution. Unlike the Set Abstraction, it does not perform downsampling. Instead, it performs a neighborhood query for each point in the input set and uses max-pooling to aggregate neighborhood information to update that point's own feature. This operation is designed to maintain the point cloud resolution while aggregating and distilling the flow features dispersed across individual points through rich local interactions.

The data blocks undergo pointwise length compression at each level, resulting in a latent feature. The latent feature is subsequently restored to the original scale of the point cloud. The feature is rescaled through interpolation based on the coordinates from the higher level via Feature Propagation. It is concatenated with the corresponding level of the data block in Encoder. The feature channels are modified via per-point MLPs.

The compressed global latent feature size is (8, 512). The model aims to infer flow characteristics through global-level geometric features, which are decoded to reconstruct the comprehensive aircraft point cloud. This iterative process generates a pressure field feature map with dimensions ($n$, 32). The feature map is then fed into Regression head, wherein it undergoes further processing by per-point MLPs and a linear layer, ultimately predicting the pressure characteristics.

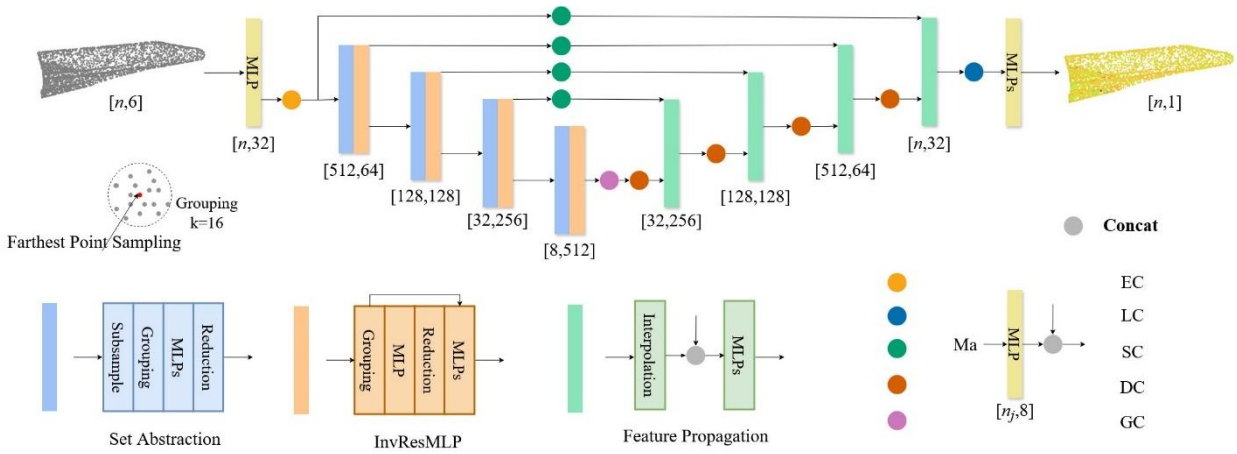


**Fig. 1 The architecture of the PointNeXtReg backbone network and schematic of static feature fusion strategies. The downsampling encoder path is composed of an initial MLP layer, Set Abstraction for downsampling, and InvResMLP blocks for core feature extraction. There are *k* neighborhood query points in grouping. The upsampling decoder path progressively restores the per-point feature using Feature Propagation, complemented by skip connections. The labels EC, LC, SC, DC, and GC denote five different locations for feature fusion, representing static strategies for concatenating condition features with geometric features at various depths of the network.**



## 2.2 Geometry-condition feature fusion

### 2.2.1 Static fusion

Static fusion strategies integrate geometric and flight-condition information through feature concatenation, with their core design variables being the position of the fusion operation within the network. To systematically explore this design space, we design and evaluate five representative static concatenation approaches, namely, early (EC), late (LC), skip (SC), decoder (DC), and global (GC), as shown in Fig. 1. The fusion locations are distributed along the network, covering key nodes from the input layer to the prediction layer. Since the AoA (angle of attack) condition is handled during point cloud preprocessing (described in Section 3.1.4), only the Ma (Mach number) condition is expanded to 8D using the MLP before being concatenated.

PNR-EC injects condition features at the input of Encoder, aiming to enable the network to learn condition-sensitive geometric representations from the initial stages. This achieves the deepest interaction between the two types of information throughout the entire forward propagation pass. Conversely, PNR-LC fuses feature at the final prediction head of the network, assuming that geometry and operating conditions can be separately refined into final representations before joint prediction.

We also investigate multilevel fusion strategies (Perez et al., 2018; Park et al., 2019), which are predicated on the principle that condition information exerts a differential influence on feature maps at various resolutions. Specifically, PNR-SC injects condition features into each skip connection from Encoder to Decoder, providing continuous awareness for the multilevel feature reconstruction process. PNR-DC performs fusion within each decoder's upsampling module. This approach places a greater emphasis on using condition information to directly guide the upsampling and refinement of the feature maps.

Additionally, PNR-GC operates at the network's bottleneck. It concatenates the most abstract global geometric feature map with the condition feature, and the resulting fused feature is then fed into Decoder. The rationale for this strategy is not to interfere with the global geometric feature extraction of Encoder, while the reconstruction task of Decoder is guided by concatenated features. Through a rigorous comparison of the above five strategies, we aim to reveal the performance impact of different static fusion locations.

### 2.2.2 Dynamic fusion

To explore more interactive fusion mechanisms, we further develop and investigate dynamic strategies. Such strategies aim to enable condition features to adaptively modulate geometric features based on content. Our exploration is in the same

location as PNR-GC. Single-point fusion at the network's bottleneck creates an ideal testbed, allowing us to isolate the effects of the dynamic interaction itself and to be free from confounding architectural variables.

We compare two core dynamic interaction paradigms, namely, one inspired by the classic style of Neural Operator, DeepONet (Lu et al., 2021), and the other based on Attention (Vaswani et al., 2017). Referencing the trunk and branch net of DeepONet, we treat the condition feature as an operator that acts upon the geometric feature by the dot product operation. As shown in Fig. 2, four specific approaches, namely, DeepONet-Style (DS), DeepONet-Style and Concatenation (DSC), DeepONet-Style and Attention (DSA), and DeepONet-Style and Residual Attention (DSRA), are designed. Let the global geometric feature output by Encoder be denoted as $F_G$, and let the condition feature processed by the MLP be denoted as $F_C$.

PNR-DS views geometric features as a set of basis vectors, while condition features are processed as coefficients for linear combinations. Through a dot product operation, this generates a novel feature $F_Q$. This mechanism aims to enable condition information to directly reconstruct geometric features via explicit linear transformations. Building upon this, PNR-DSC introduces a concatenation operation to combine $F_Q$ with $F_G$. This design is motivated by information enhancement, aiming to provide richer features for subsequent network layers by preserving the original information flow.

The attention mechanism theoretically makes feature modulation more flexible. PNR-DSA sets up $F_Q$ as Query, while the original $F_G$ serves simultaneously as both Key and Value. The Query probes the Key, computes attention weights, and uses them for a weighted aggregation of $F_G$. The essence of this step is to selectively enhance or suppress specific patterns within global geometry features based on the preliminary operator modulation characteristics.

Finally, PNR-DSRA introduces a residual connection outside the attention module. The $F_G$ is added back to the attention module's output. This ensures that the information stream can steadily pass through the module without loss, even if the attention block fails to learn an effective transformation. By analyzing these dynamic strategies, we aim to evaluate the potential of adaptive feature fusion in such tasks.



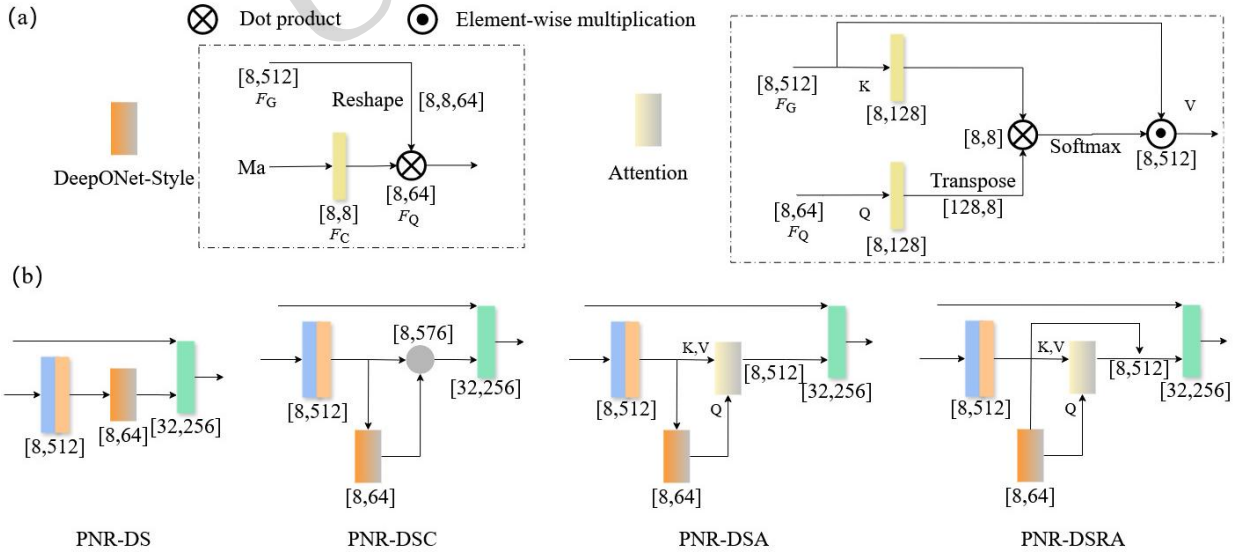


**Fig. 2 (a) Two modules for fusing geometric and condition features: the DeepONet-Style module (left), which modulates geometric features using weights derived from the condition vector; and the attention module (right), which uses the geometry-condition feature outputted by the DeepONet-Style block to query key information from the geometric feature. (b) The four network architectures built upon different fusion strategies: PNR-DS (employing the DeepONet-Style module), PNR-DSC (adding the static concatenation part), PNR-DSA (increasing the attention module), and PNR-DSRA (enhancing the attention module with a residual connection).**

### 2.3 Geometric feature aggregation

We first need to establish a fixed fusion framework as the baseline backbone for future research. To this end, PNR-SC was selected due to its suitability for the research. PNR-SC not only ensures that the condition feature persists across all resolutions within the network but also minimizes interaction with the main Encoder and Decoder structures. This provides a convenient testbed for subsequently implementing aggregation modules at various locations of the trunk. In the following study of feature aggregation, this baseline PNR-SC model will be referred to simply as PNR.

As mentioned in Section 2.1, the InvResMLP module relies on max-pooling to aggregate neighborhood information. This represents a static choice. To explore the potential of dynamic aggregation strategies, we introduce the PT block and design a series of deployment schemes at various locations to systematically evaluate its impact. The PT block employs an attention mechanism to dynamically weight and aggregate features from neighboring points.

As illustrated in Fig. 3, its core principle is that for a central point, the contribution of each neighboring point is computed based on their feature differences and relative spatial relationships. Specifically, feature differences are computed by mapping functions $\varphi$ and $\psi$, while relative position information is encoded through $\delta$. Together, these factors determine the final attention weight $\alpha$. This mechanism enables the aggregation process to perceive local geometric context, theoretically allowing for the effective capture and preservation of critical structural details.

To evaluate the efficacy of the dynamic aggregation, we design four logically progressive models. First, we attempt to replace all InvResMLP modules in Encoder with PT blocks, resulting in the PNR-PT architecture. The purpose of this approach is to examine whether dynamic aggregation outperforms static aggregation in the performance of sparse point clouds during the downsampling stage for such pressure field prediction tasks.

Next, we insert a single PT block immediately after the initial pointwise MLP, yielding the PNR-IPT model. The core motivation is to validate the effectiveness of early dynamic aggregation. This design is used to examine whether modeling local geometric relationships in dense raw point clouds can provide a higher-quality feature foundation, which could then be leveraged by computationally cheaper Encoder. In addition, to compare the performance of the static aggregation strategy at dense point clouds, we added the InvResMLP module at the same location, namely, PNR-IIR.

Finally, to explore the upper bounds of performance, we concatenate the PT block for each preceding module of Regression head, resulting in Point Transformer Regression (PTR). Through comparisons of these four deployment schemes against the baseline PNR model, we aim to systematically reveal the value of dynamic aggregation and to provide valuable practical guidelines for integrating attention mechanisms in similar architectures.

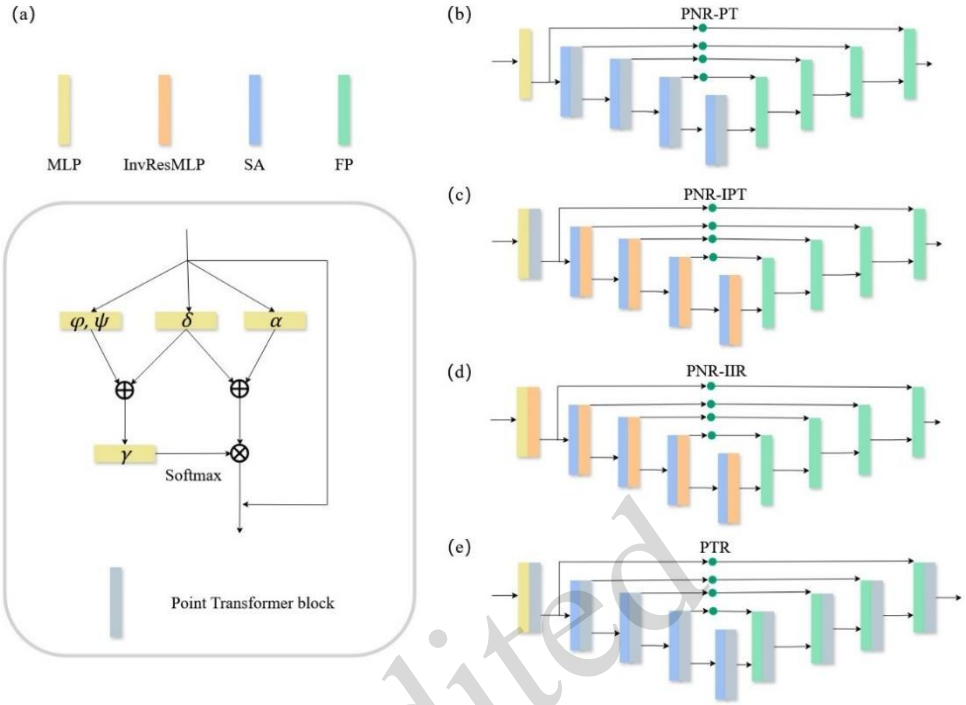


**Fig. 3 (a) Representations of the baseline network modules and PT block architecture. The right presents four distinct network backbones designed to investigate the performance of various combinations of static InvResMLP and dynamic PT modules: (b) PNR-PT: A fully dynamic architecture where all InvResMLPs are replaced by PT blocks; (c) PNR-IPT: An architecture that introduces a PT block after the initial MLP layer; (d) PNR-IIR: An architecture that first uses an InvResMLP block after the initial MLP layer; (e) PTR: Point Transformer Regression architecture shown for comparison.**



## 3 Experimental setup

### 3.1 Dataset generation

#### 3.1.1 Numerical validation

According to the study (Erickso et al., 1990), using the inviscid method is an acceptable for aerodynamic evaluation in high-speed environments. The computational speed is relatively fast, meeting the requirements of substantial samples. In this work, we describe the behavior of inviscid and compressible air in a flow field by the Euler equations. They can be expressed as

$$\begin{cases} \nabla \cdot (\rho \boldsymbol{u}) = 0 \\ \nabla \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u}) + \nabla p = 0, \\ \nabla \cdot (\rho E + p)\boldsymbol{u} = 0 \end{cases} \tag{1}$$

where $\boldsymbol{u}$ is the fluid velocity, $\rho$ denotes the density, $p$ represents the pressure, and $E$ is the total energy per unit mass with the following equation:

$$E = \frac{1}{2}u^2 + \frac{1}{\gamma - 1}\frac{p}{\rho}. \tag{2}$$

We employ the density-based implicit solution method. The air is assumed to be the ideal gas. A second-order spatially accurate upwind scheme, with the advection upstream splitting method (AUSM) method applied to the flux vector, is adopted, and the gradient is calculated using a Gaussian-Green method. To ensure stability, the starting Courant–Friedrichs–Lewy (CFL) number is set to 0.1, and the maximum CFL number is set to 0.6. The computation terminates when the residual drops to 1e-6 or after 10000 iterations.

The simulation method is first validated on the Space Shuttle benchmark geometry with high-fidelity experimental data. Regarding mesh independence, we employed automatic meshing techniques to rapidly generate unstructured meshes, which were then optimized multiple times. This approach ensured that the number of mesh elements is consistent with that used in the research group's previously validated work (Shen et al., 2024) while also guaranteeing the quality of convergence. The entire validation process is shown in Fig. 4. We set the value of the reference

area to 0.01m$^2$. The experimental values and simulation results displayed in Fig. 5 are consistent with the work (Ding et al., 2015), demonstrating the correctness of the numerical simulation.

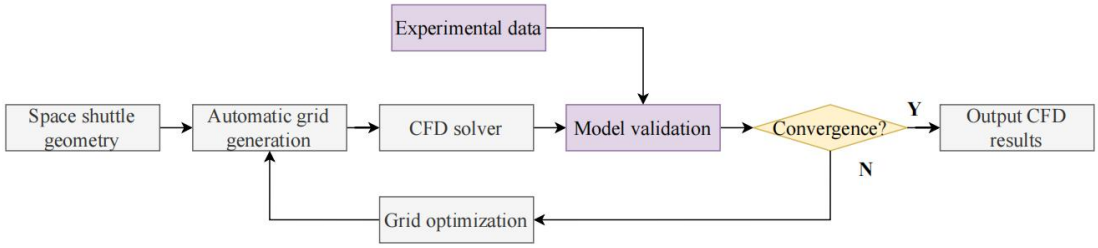


**Fig. 4 The flowchart of numerical validation.**

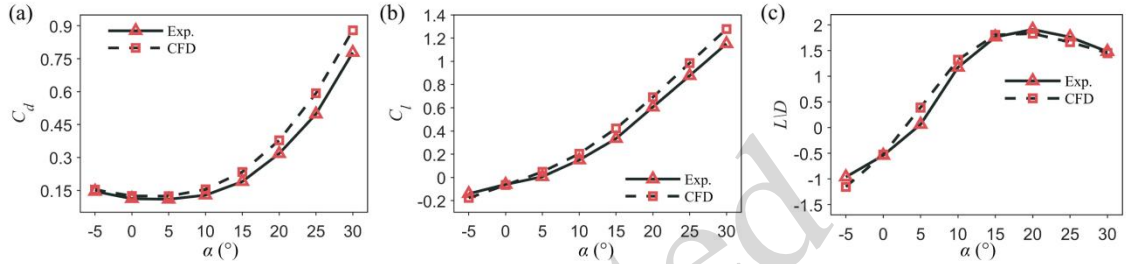


**Fig. 5 The results of numerical validation: (a) drag coefficients, (b) lift coefficients and (c) lift-to-drag ratios.**

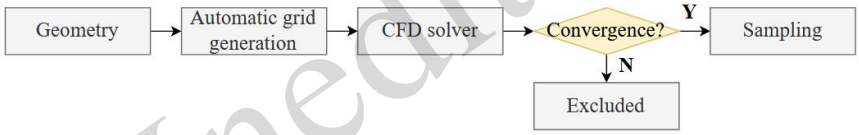


**Fig. 6. The automatic process of numerical simulation.**



### 3.1.2 Sample simulation

To study the generalization performance under multi-condition scenarios, we create a dataset. The shape of the samples is derived from the Space Shuttle variations (Shen et al., 2023). The differences between the shapes are obvious and can be used to study generalization properties. Moreover, to efficiently sample the conditions and avoid duplication, a Latin hypercube sampling method is employed in the range of Ma $\in$ [3.0, 10.0] and AoA $\in$ [−10°, 10°]. The method is a statistical method for generating a near-random sample of parameter values from a multidimensional distribution, which ensures that the set of random numbers generated is a good representation of the real variability of parameters.

Then, based on the accumulated experience to improve the robustness of automated meshing as described in section 3.1.1, we normalize the length of all aircraft to 0.51 m. This operation aligns with the similarity principle of inviscid flow. The pressure far-field radius is set to 0.7 m, ensuring fully developed aerodynamic performance. The samples involve surface meshes exceeding 200,000 points and total volumetric meshes surpassing 3 million points, holding practical significance for 3D pressure field prediction and high-dimensional aerodynamic surrogate modeling. We have completed the simulations using node resources on the BSCC-A6 supercomputing system based on Python. The configuration of one computing node is AMD7452@2.35 GHz, 256 GB of memory, and 64 cores. Fig. 6 outlines the procedure for sample generation by simulation.

### 3.1.3 Ablation setting

Through simulations, a total of 1646 converged samples were obtained. To ensure a fair evaluation, 100 samples were first randomly selected as a fixed test set, which was used exclusively for reporting the final performance of all models. The remaining 1546 samples were used for training and validation. It should be noted that the sample sizes of 352 and 1194 were not arbitrary prescribed ratios but two naturally formed groups of converged samples during the staged CFD campaign, corresponding to an early

low-data regime and a larger-data regime after further data accumulation, respectively.

To systematically investigate model generalization and robustness, three training scenarios were designed: 1) Limited-Data scenario: 352 samples were used for training, with the remaining 1194 samples serving as the validation set. 2) Moderate-Data scenario: A complementary split was adopted, with 1194 samples used for training and 352 samples used for validation. 3) Full-Data scenario: All 1546 samples were used for training. Validation was performed on the same 352 samples used in the Moderate-Data scenario.

It is important to emphasize that our model receives a sparse point cloud as input during training, which is obtained by farthest point sampling with a fixed random number. In the validation phase, model evaluation is performed on the original point cloud. In the Full-Data Scenario, although the geometry corresponding to the validation set exists in the training set, it can demonstrate the reconstruction capability of the high-resolution point cloud.

We also note that a key parameter investigated in the study is the neighborhood size $k$ for grouping feature aggregation (as shown in Fig. 1). Therefore, we configured distinct sizes at the same sampling radius for our two main ablation studies: $k$=4 for the feature fusion analysis and $k$=16 for the geometric aggregation study. Through this nested ablation study, we are able to conduct a more rigorous and comprehensive evaluation of the model architecture while keeping the test set constant.

#### 3.1.4 Sample simulation

To convert the numerical simulation results into point cloud mode, we extract the original surface mesh centroid coordinates ($x_0$, $y_0$, $z_0$) and face normal vectors ($n_x$, $n_y$, $n_z$) as inputs, using pressure coefficients $C_p$ as the outputs for supervision. The surface meshes of the samples typically consist of approximately 200,000 elements. To reduce the computational burden and retain redundant information, a preprocessing stage is applied to the training samples. This stage utilizes farthest point sampling to downsample each sample to 40,000 points, thereby preserving essential geometric characteristics. The AoA is essentially a geometric attitude parameter, while the Mach number is the eigenparameter that determines the physical state of the flow. Therefore, a data-driven strategy consistent with the physical process is adopted, and the AoA information is innovatively integrated into the geometry by multiplying with the points during the downsampling process, as demonstrated in Eq. (3).

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} cos\,(\alpha) & 0 & sin\,(\alpha) \\ 0 & 1 & 0 \\ -sin\,(\alpha) & 0 & cos\,(\alpha) \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \tag{3}$$

Different from directly combining the AoA feature with the geometric feature at the data level, we input the Ma information and normalize it to [0, 1] by min–max scaling. Since the origin of our coordinates is in the middle of the length of the aircraft, we normalized the changed geometric coordinates ($x$, $y$, $z$) to [-1, 1] with the maximum distance from the origin. In deep learning, normalization is a crucial step in data preprocessing. Normalization technology adjusts the scale of input data so that the data has a similar distribution range, improving the model's solution speed and generalization ability.

### 3.2 Training strategies

#### 3.2.1 Downsampling and augmentation

During model training, to address GPU memory constraints while effectively capturing essential geometric features, we employ a combined strategy of downsampling and augmentation for point clouds. For each training epoch and each sample, the 40,000 input points are randomly downsampled to 4096, followed by applying data augmentation to the 4096 points. Data augmentation is a critical technique in deep learning that expands the training dataset's size and diversity by adopting stochastic transformations to raw data, thereby improving model generalization, mitigating overfitting, and enhancing robustness.

In this work, some of the common augmentation operations on point cloud processing are independently applied: 1) Random rotation: A rotation radian is uniformly sampled from [-0.5, 0.5] to rotate the point cloud coordinates and normal vectors around the inflow axis. Since gravity is neglected in the numerical simulations, rotations along the inflow axis preserve physical consistency, while rotations around other axes violate the

governing equations. 2) Random translation: Three translation distances are independently sampled from the uniform interval [-0.2, 0.2], corresponding to the addition operation along the *X*, *Y*, and *Z* axes. 3) Random jittering: Each point's coordinates are perturbed by randomly sampling from a normal distribution $\mathcal{N}(0, 0.001^2)$, introducing local variations while preserving the global structure. These methods collectively enhance the model's adaptability to geometric and positional variations in point cloud data.

3.2.2 Optimization and loss

We accelerate convergence with batch normalization technology. Specifically, considering computational efficiency and gradient stability across different data scales, we set the batch size to 16 in the Limited-Data Scenario. In the Moderate-Data Scenario and Full-Data Scenario, the batch size is uniformly set to 32. To ensure fair experimental comparisons, all other hyperparameters remained consistent across all experiments.

We empirically select AdamW as the optimizer with commonly used hyperparameter settings, including an initial learning rate of 1e-3 and a weight decay of 1e-4. These settings are further examined in section 4.2.3. The model is trained for 5000 epochs. We adopt the mean squared error (MSE) of $C_p$ as the loss function, with the form of Eq. (4), where $n$ is the size of the CFD-specific point cloud. The same MSE is also used as the evaluation metric during the prediction phase.

$$\mathcal{L}_{\mathrm{MSE}} = \frac{1}{n}\sum_{i=1}^{n}(\widehat{C}_{p,i} - C_{p,i})^2 \tag{4}$$

All training experiments were conducted on the computing cluster using an NVIDIA RTX 4090 GPU with 24 GB of VRAM, with training times of approximately 10 h, 34 h, and 42 h for the three data-scale scenarios. These differences mainly reflect the increase in training data size. The final inference tests were performed on a personal workstation with an NVIDIA RTX 3090 GPU with 24 GB of VRAM. Since testing used a full-resolution surface point cloud, the inference time and memory usage were strongly affected by the model architecture, especially in the exploration of feature aggregation.

## 4 Results and discussion

### 4.1 Evaluation of feature fusion

4.1.1 Training dynamics

Fig. 7 illustrates the training and validation loss curves of different feature fusion approaches across three data scenarios, providing preliminary qualitative insights for subsequent quantitative analysis. From the perspective of training loss shown in Fig. 7(a), all models effectively converge to similar levels across different data scenarios without exhibiting phenomena such as gradient vanishing. This demonstrates the optimization feasibility of all proposed architectural designs.

The validation loss curves in Fig. 7(b) reveal two completely different behavior patterns—shape generalization and high-resolution reconstruction—depending on whether the evaluation samples are within the training domain. In the Limited-Data and Moderate-Data scenarios, the validation results demonstrate generalization to unseen shapes. PNR-DSA and PNR-DSRA exhibit generalization failure in these two scenarios, respectively, with their validation losses remaining persistently high. Additionally, PNR-LC shows classic signs of overfitting in the Moderate-Data scenario, where its validation loss consistently rises a period of decline. Beyond these three inappropriate feature engineering approaches, the remaining methods form a high-performance cluster, demonstrating reliable generalization performance.

The validation phase of the Full-Data scenario assesses the ability to reconstruct a full-resolution point cloud with the same geometric shape from sparsely trained point clouds. In this task, all models exhibit similar and effective convergence behavior. At this stage, the core task of the model is to interpolate in a known geometry, rather than inferring an unknown structure. Consequently, the stable behavior reflects the model's ability to memorize specific samples, rather than its generalization capability on unseen data. Finally, all the fusion variants complete inference on the test set within 1 s, indicating that their computational cost differences are negligible.

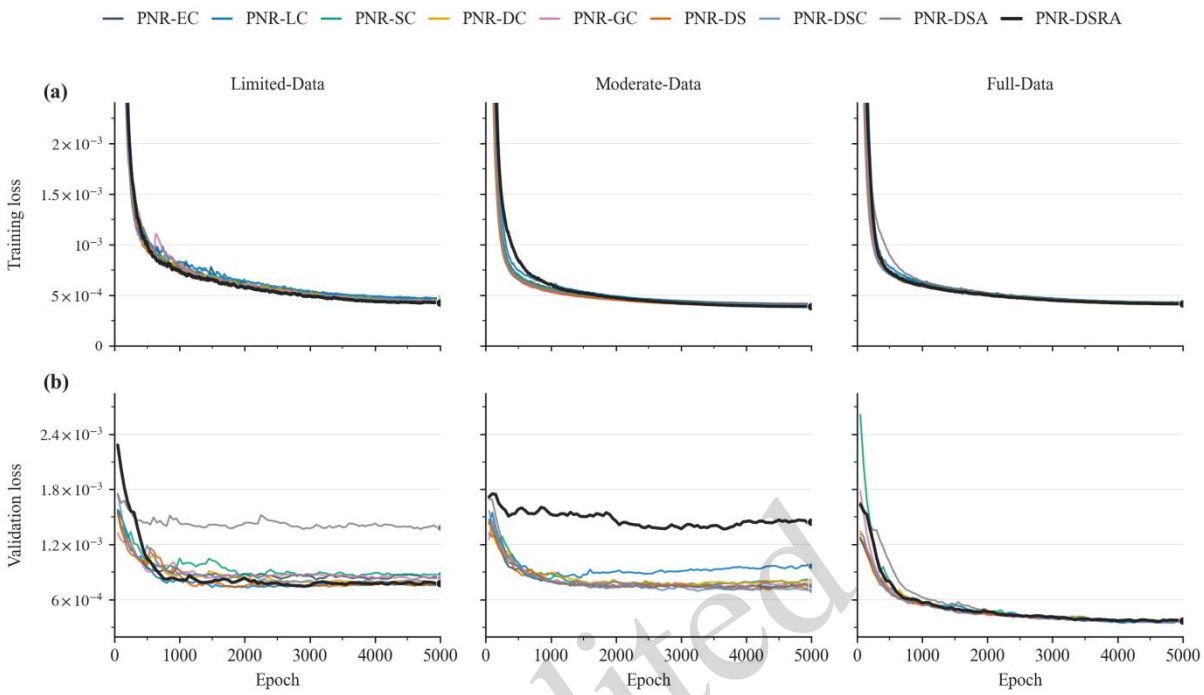


**Fig. 7. Convergence analysis of various fusion strategies for condition and geometric features in the three data scenarios. (a) Training loss and (b) validation loss.**



4.1.2 Testing of static fusion

To quantify the effect of different static fusion strategies on model generalization, the models trained with Seed=0 were first evaluated on 100 unseen test samples using the MSE of $C_p$ prediction. The results are reported in Table 1, where PNR-EC is used as the baseline method. For each data scenario, the best-performing method is highlighted in bold, while the worst-performing method is italicized.

Consistent with the trends observed in the validation curves, PNR-LC achieves the lowest MSE in the Limited-Data scenario, improving over PNR-EC by 6.7%. However, this advantage does not persist as the data scale increases. In the Moderate-Data and Full-Data scenarios, PNR-LC deteriorates by 13.0% and 8.1%, respectively. This suggests that late fusion may be beneficial when training data are scarce, but its capacity to model the complex nonlinear coupling between geometric features and flight-condition features may become insufficient when more training data are available.

Among the remaining methods, PNR-DC shows relatively stable performance in the single-seed comparison, reducing the MSE by 7.8%, 3.4% and 0.2% in the Limited-Data, Moderate-Data and Full-Data scenarios, respectively. This indicates that progressively injecting condition features into the decoder may provide a comparatively robust fusion scheme. In contrast, PNR-SC and PNR-GC exhibit relatively small variations across different data scales but do not show a consistent advantage over the baseline.

It should be noted that the improvements of the static fusion strategies over PNR-EC are generally limited, with most changes being smaller than 10%. Therefore, the results based on a single random initialization are insufficient to determine whether these differences originate from the fusion strategies themselves or from random initialization and training variability. Motivated by this observation, Seeds = 1, 2, 3 and 4 were further introduced, and the MSE changes of each static fusion strategy relative to PNR-EC were compared in a paired manner across three data scenarios and five random seeds, as shown in Fig. 8.

**Table 1 Effect of static fusion strategies on the pressure coefficient MSE over 100 unseen test samples (Seed=0).**

| Scenarios | PNR-EC (Baseline) | PNR-LC | PNR-SC | PNR-DC | PNR-GC |
|---|---|---|---|---|---|
| Limited-Data | 8.37e-04 | 7.81e-04 (+6.7%) | *8.66e-04* (-3.5%) | **7.72e-04** (+7.8%) | 7.83e-04 (+6.5%) |
| Moderate-Data | 5.52e-04 | *6.24e-04* (-13.0%) | 5.49e-04 (+0.5%) | **5.33e-04** (+3.4%) | 5.63e-04 (-2.0%) |
| Full-Data | 5.55e-04 | *6.00e-04* (-8.1%) | 5.72e-04 (-3.1%) | **5.54e-04** (+0.2%) | **5.54e-04** (+0.2%) |

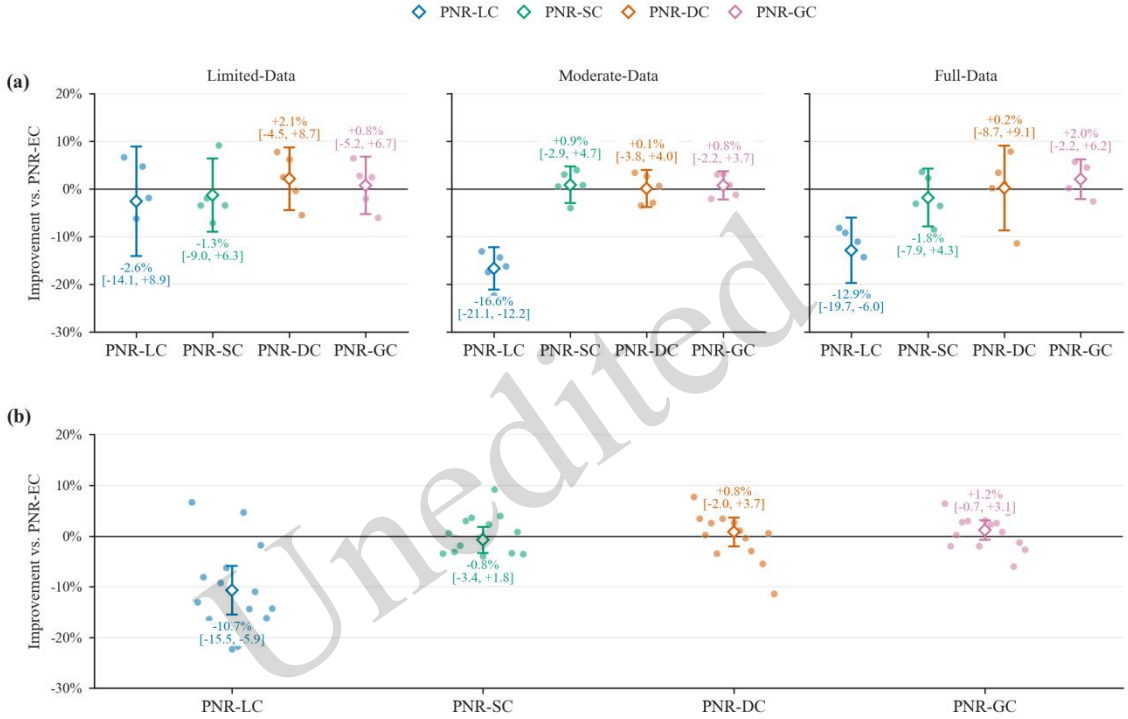


**Fig. 8: Distribution of the relative change in test MSE for static fusion strategies with respect to the baseline PNR-EC. The pressure coefficient MSE is computed over three data scenarios and five random seeds. Positive values denote a reduction in MSE relative to PNR-EC, whereas negative values denote degradation. Dots represent individual seeds, hollow diamonds denote mean values, and vertical lines indicate 95% confidence intervals. (a) Results separated by data scenario and (b) results aggregated over all scenarios and random seeds.**

The multiseed results confirm that the main risk lies in late fusion. PNR-LC exhibits a consistent degradation relative to PNR-EC, with an overall mean change of -10.7% and a 95% confidence interval of [-15.5, -5.9%]. In particular, the mean changes in the Moderate-Data and Full-Data scenarios are -16.6% and -12.9, respectively, and their confidence intervals remain entirely below zero. This indicates that late concatenation is consistently weaker in the present task, likely because it lacks sufficiently deep interaction between geometric and flight-condition features.

In contrast, the mean changes in PNR-SC, PNR-DC, and PNR-GC are all close to zero, with overall values of -0.76%, +0.81% and +1.18%, respectively. Their 95% confidence intervals all cross zero, indicating that their improvements over PNR-EC are not statistically reliable under the current five-seed setting. Therefore, the static-fusion study does not demonstrate that one strategy is universally superior. Instead, it shows that late fusion is a consistently risky design, whereas deeper or multilevel fusion strategies provide more robust alternatives.

### 4.1.3 Testing of dynamic fusion

To provide a more intuitive comparison of different dynamic fusion strategies, Fig. 9 presents the predictive performance of each model relative to the static fusion baseline PNR-GC. Specifically, Fig.

9(a) reports the relative improvement in the pressure coefficient $C_p$, where positive values indicate error reduction compared with PNR-GC and negative values indicate error increase. Fig. 9(b) further shows the error distributions of the global drag and lift coefficients: $C_d$, $C_l$. It should be noted that this figure is based on results from a single random seed and is therefore intended to illustrate performance trends among different methods rather than provide evidence of statistical significance.

As shown in Fig. 9(a), PNR-DS increases the MSE by 7.4% in the Limited-Data scenario, indicating a certain degree of degradation. However, it reduces the MSE by 14.6% and 16.1% in the Moderate-Data and Full-Data scenarios, respectively, achieving the most pronounced improvements in these two settings. This suggests that direct dynamic feature replacement may be unstable when data are limited but can offer stronger performance potential when sufficient data are available. In contrast, PNR-DSC shows relatively small variations: it degrades performance by only 1.0% in the Limited-Data scenario and reduces the MSE by 2.5% and 1.6% in the Moderate-Data and Full-Data scenarios, respectively. This may indicate that concatenation-based dynamic feature enhancement provides a more conservative and stable fusion design.

The attention-based variants, however, exhibit more pronounced instability. PNR-DSA remains nearly comparable to the baseline in the Moderate-Data scenario, with only a 0.2% MSE reduction, but leads to MSE increases of 75.6% and 56.3% in the Limited-Data and Full-Data scenarios, respectively. PNR-DSRA degrades performance by 7.0% in the Limited-Data scenario and 30.1% in the Full-Data scenario and shows the most severe degradation in the Moderate-Data scenario, where the MSE increases by 95.6%. These results indicate that introducing attention mechanisms does not consistently improve the effectiveness of dynamic fusion. Instead, it may increase the sensitivity of the model to the data scale and training conditions.

Fig. 9(b) further demonstrates that this instability is propagated to the prediction of integrated aerodynamic coefficients. In the violin plots, a wider body and longer tails indicate larger error variance and more outlier samples. The error distributions of $C_d$ and $C_l$ for PNR-DS and PNR-DSC are generally more compact, with most errors concentrated around zero. In contrast, the attention-based models show broader error distributions and more pronounced long-tailed behavior in their degraded scenarios, such as PNR-DSA under the Limited-Data scenario and PNR-DSRA under the Moderate-Data scenario. This suggests that the instability of attention-based variants is not limited to local $C_p$ prediction but further affects the reliability of global aerodynamic prediction.

Overall, Fig. 9 suggests that the effectiveness of dynamic fusion strategies is strongly dependent on the data scale. PNR-DS shows greater performance potential when sufficient data are available, whereas PNR-DSC provides a more robust design choice. The attention-based variants exhibit stronger data-dependent degradation and less stable error distributions, suggesting that the learned weighting patterns may not be sufficiently robust. Therefore, the following subsection further diagnoses the failure modes associated with attention mechanisms through latent feature visualization and surface attention/error mapping.

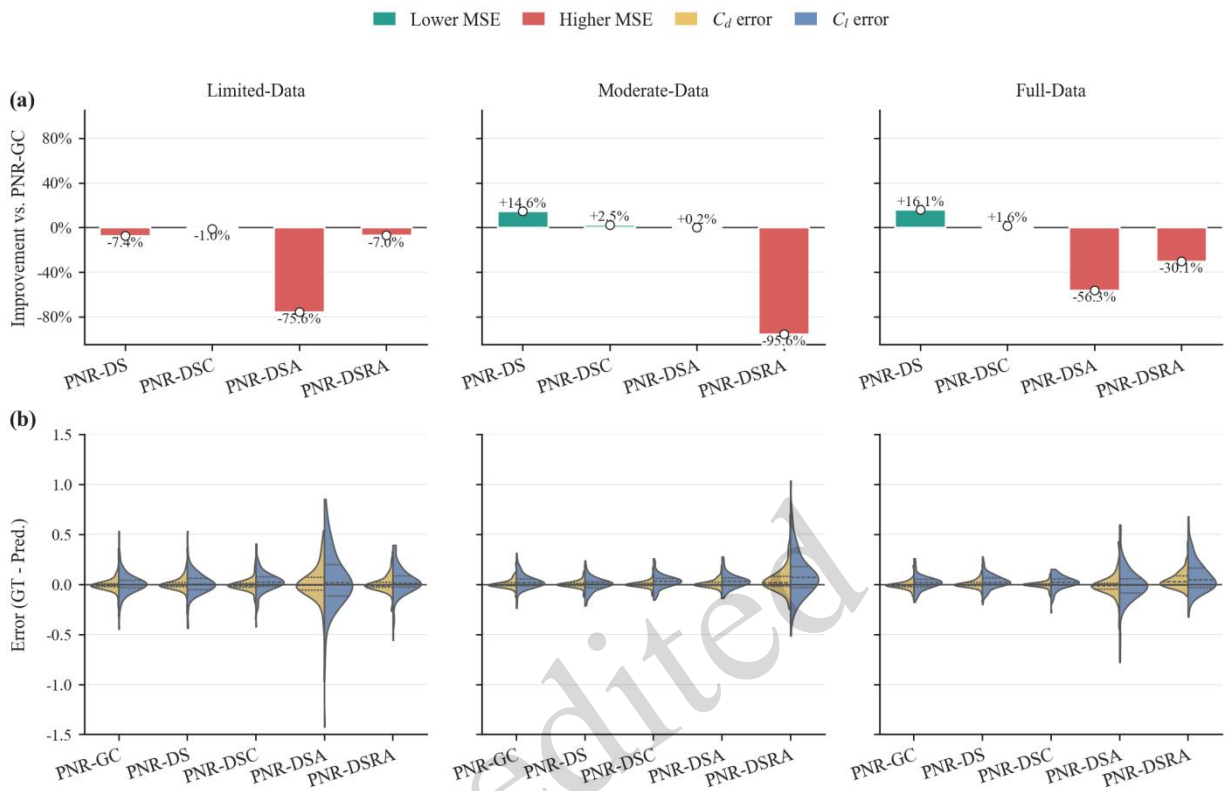


**Fig. 9. Performance comparison of dynamic fusion strategies with respect to the baseline PNR-GC. (a) Percentage change in MSE for four dynamic fusion models under three data scenarios. Red bars indicate performance degradation, and green bars indicate gains. (b) Violin plots of the prediction errors for $C_d$ and $C_l$, where the error is defined as ground truth minus prediction. Narrower distributions centered around zero indicate more accurate and stable integral-coefficient predictions.**



4.1.4 Diagnostic analysis of attention-based dynamic fusion

To make clear that the attention module does not always establish a stable correspondence among the condition-driven Query, the geometry-driven Key, and the surface pressure response, we diagnose PNR-DSA from three complementary levels. First, Fig. 10 examines the samplewise MSE distribution in the condition space to determine whether the error is governed by a simple input variable. Then, Fig. 11 and Fig. 12 analyze the organization and compactness of the Query/Key latent representation using t-SNE and PCA, respectively. Finally, the three best/median/worst surface-attention maps are placed in the Appendix and discussed as a unified evidence set, while the main text retains a fixed-sample cross-scenario comparison to isolate how the training-data regime changes the learned attention organization and the resulting prediction error.

Fig. 10 shows that large-error samples are not strictly confined to a specific conditional interval or shape-index range. This observation weakens an overly simple explanation that model failure is caused only by a condition, a specific shape. The comparison among the three scenarios further indicates that the data scale changes the error landscape of PNR-DSA but does not guarantee monotonic improvement in all regions. This is because the attention module needs to establish matching relations between condition features and geometric features, and these relations are reorganized when the training distribution changes. For this diagnostic case, Moderate-Data may provide a more suitable calibration strength, allowing some edge samples and high-error samples to be corrected. In contrast, Full-Data may introduce more complex geometry-condition combinations, causing the existing matching relations to become oversmoothed, overcompressed, or locally mismatched.

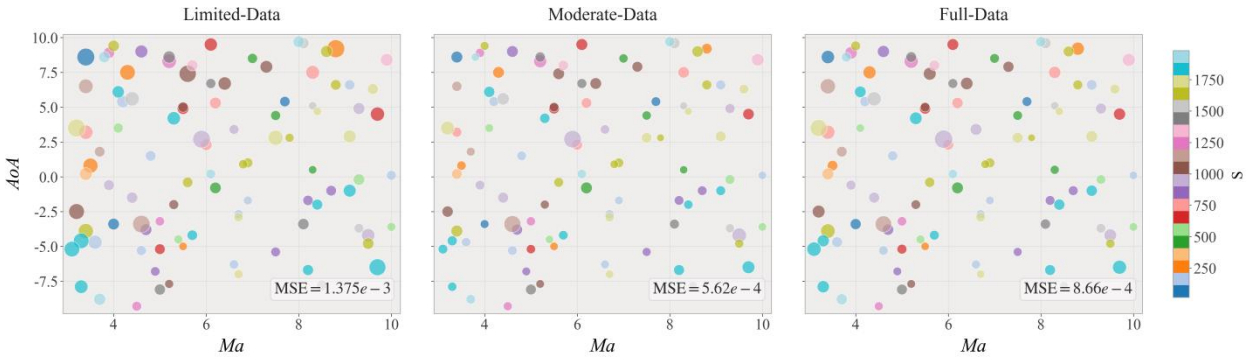


**Fig. 10 Samplewise MSE distribution of PNR-DSA in the Ma-AoA condition space. Each bubble represents one test sample. The bubble area is proportional to the samplewise MSE of the pressure coefficient, and the color denotes the shape index S.**




**Fig. 11 t-SNE visualization of concatenated Query/Key embeddings of PNR-DSA under different training data scenarios. Each row uses the same t-SNE projection, and the four columns differ only in the coloring variable: Ma, AoA, shape index 'S', and 'log10(MSE)'.**

Fig. 11 visualizes the concatenated Query/Key representations using t-SNE. In the Limited-Data scenario, the dimension-reduced representations of the test samples exhibit a diffuse distribution, with all four coloring variables showing substantial mixing. In the Moderate-Data scenario, clearer clusters emerge, and this structure exhibits the most consistent gradient when colored by Ma. In contrast, AoA, 'S', and 'log10(MSE)' remain visibly mixed within the same branches. Therefore, these clusters should not be interpreted as a complete disentanglement of all input variables. Instead, they are more appropriately understood as several relatively stable local attention neighborhoods formed along the Ma-dominated direction. In the Full-Data scenario, the representations exhibit a more continuous band-like distribution, while the Ma gradient remains evident.

The Query/Key representations in the Moderate-Data scenario are neither as poorly organized as those in the Limited-Data scenario nor

compressed into a single continuous ordering as in the Full-Data scenario. Instead, they preserve distinguishable local subgroups. Such local calibration may help the attention module reuse stable matching relations within similar flow conditions while avoiding the forced merging of substantially different AoAs and configurations into a single global direction. This interpretation is consistent with the PCA results in Fig. 12, where the Moderate-Data Query representations are relatively compact, whereas the Key representations still retain moderate flexibility. It also provides a potential explanation for the lower median and mean errors observed in this scenario.

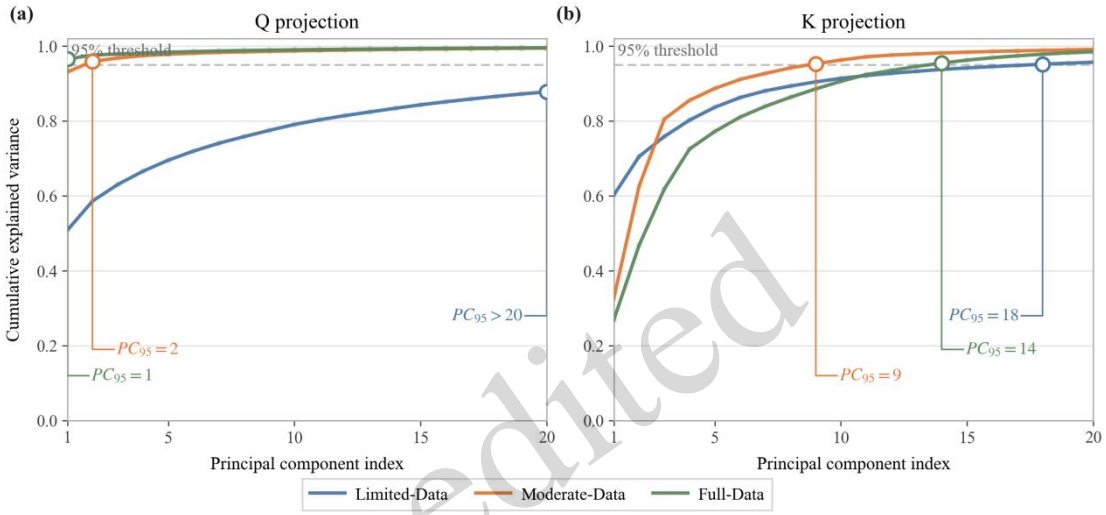


**Fig. 12 Cumulative explained variance of the Query and Key projections in PNR-DSA. (a) Query projection. (b) Key projection. The dashed line denotes the 95% cumulative-variance threshold, and 'PC95' denotes the number of principal components required to reach this threshold.**



The PCA results in Fig. 12 quantify the latent structure differences observed in Fig. 11. For the Query projection, the Limited-Data scenario requires more than 20 principal components to reach the 95% explained-variance threshold. This indicates a highly dispersed condition-driven representation that cannot be summarized by a few stable directions. Such dispersion may make the attention weights more sensitive to sample perturbations and weaken the stable modulation of the geometric feature space by condition embedding.

In contrast, the Query projection in the Moderate-Data scenario reaches the threshold with only 2 principal components, indicating a much more compact condition representation. Meanwhile, its Key projection requires 9 components, preserving sufficient degrees of freedom for geometry. In the Full-Data scenario, the Query projection is further compressed into only one dominant component, while the Key projection still requires 14 components. This suggests a potential imbalance between an overcompressed condition representation and a relatively complex geometric representation. Therefore, the reliability of attention-based fusion depends on a balanced Query/Key organization rather than simply on the training-data size or the compression degree of one side of the representation.

Figs. S1–S3 in the electronic supplementary materials (ESM) provide the surface attention, predicted pressure field, and pointwise residual maps under the three regimes. It should be noted that the surface-attention color scale is normalized independently in each panel. Therefore, color intensity should not be directly compared across different panels. A more reliable interpretation is to compare the relative spatial pattern within each panel and relate it to the colorbar range and samplewise error.

Overall, the visual structure in the attention map does not guarantee reliable pressure-field prediction. In the Limited-Data scenario, the maximum attention weight is close to a uniform distribution, indicating that the model has not learned sufficiently discriminative attention selection. In the

Moderate-Data scenario, some samples show more selective attention partitions and lower overall error. In the Full-Data scenario, some samples exhibit large-scale partitioned or nearly saturated attention, suggesting that a larger training set may introduce geometry-condition modes that are difficult to reconcile through direct attention reweighting. This evidence is consistent with Fig. 10-Fig. 12: the failure of attention does not arise from the absence of visual hotspots but from an unstable correspondence among Query/Key latent organization, surface weights, and the true pressure response.

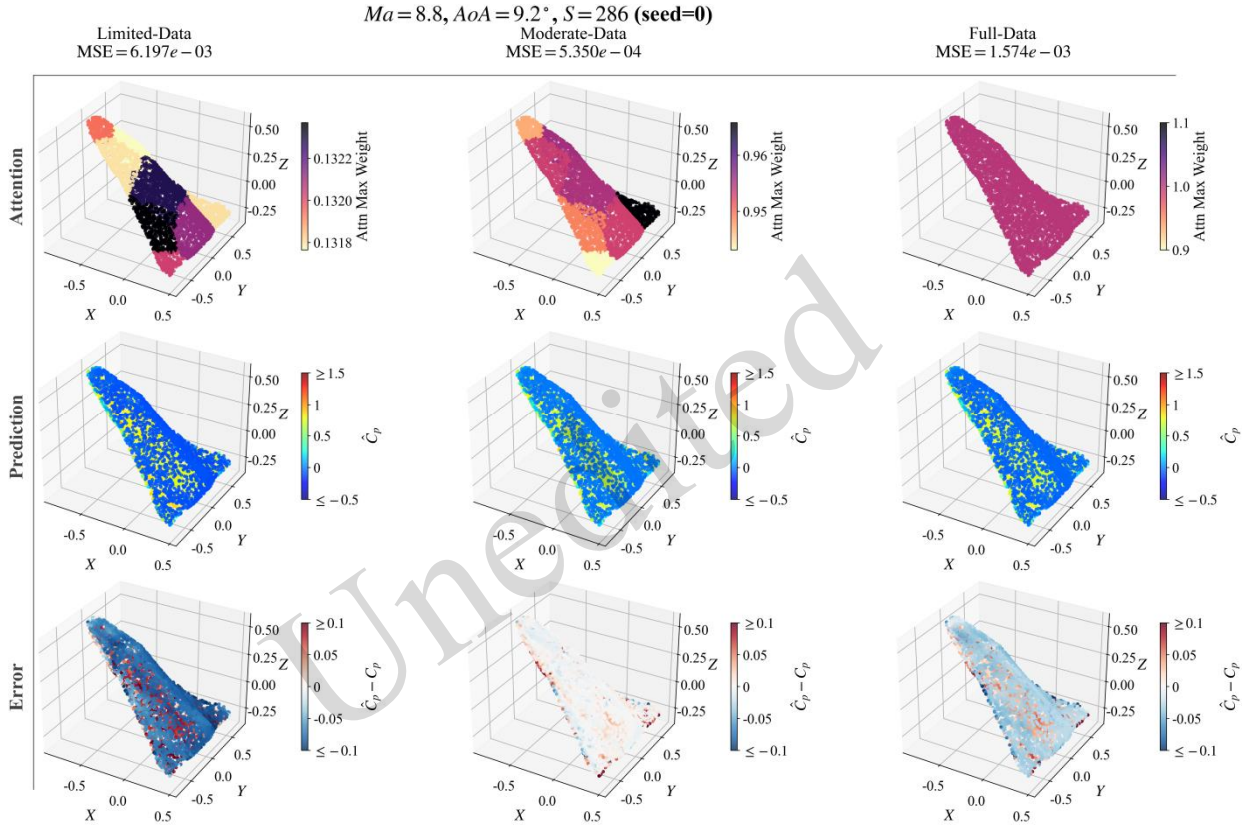




**Fig. 13 Cross-scenario surface visualizations of PNR-DSA for a fixed test configuration. The columns correspond to the Limited-Data, Moderate-Data, and Full-Data scenarios, respectively, while the rows show the surface attention concentration, the predicted pressure coefficient $\widehat{C}_p$, and the pointwise prediction error $\widehat{C}_p$-$C_p$. The shared test configuration is specified by Ma=8.8, AoA=9.2°, and S=286, selected using random seed 0.**

Fig. 13 fixes the same geometry-condition sample and therefore removes the confounding effect of changing inputs. Although all three columns use identical Ma=8.8, AoA=9.2°, and S=286 samples, the learned attention distributions differ substantially across the training scenarios. This confirms that the surface attention map is not an intrinsic property of the geometry alone. It is a learned organization determined by the interaction among the training distribution, the condition embedding, and the geometric representation.

For this sample, the Moderate-Data model gives an MSE of 6.446e-04, which is approximately one order of magnitude lower than the Limited-Data result and approximately 45% of the Full-Data result. The Limited-Data model has an almost uniform maximum-attention pattern and produces stronger positive and negative residuals over the surface. The Full-Data model produces nearly saturated attention weights, indicating an overconfident global partition, but its residual remains larger than that of the Moderate-Data model. By comparison, the Moderate-Data model avoids both near-uniform and fully saturated attention patterns, and its residual field is weaker and more localized.

This fixed-sample comparison reinforces the conclusion from the Appendix maps: attention-based fusion is not intrinsically ineffective, but it is reliable

only when the Query/Key representation, surface attention weights, and pressure-response structure remain stably calibrated. This diagnosis also helps explain why PNR-DS and PNR-DSC are more robust in the quantitative comparison, as they retain the original geometric feature stream and use the conditionally generated dynamic feature as complementary information. In contrast, PNR-DSA directly redistributes geometric information through learned attention weights, making it more susceptible to undercalibrated, overcompressed, or near-saturated attention patterns.

## 4.2 Evaluation of feature aggregation

### 4.2.1 Training dynamics

Fig. 14 presents the training dynamics of the baseline PNR backbone and four logically progressive dynamic aggregation models (PNR-PT, PNR-IPT, PNR-IIR and PTR). As the number of data samples increases, the curves differ more from each other than the curves of fusion strategies in Fig. 7 (except for PNR-LC, PNR-DSA, and PNR-DSRA). This means that the research value of fusion strategies lies in identifying high-risk integration patterns. However, the aggregation mechanism represents a more fundamental architectural design decision, and in-depth analysis of its performance is essential.

When assessing generalization on unseen samples, the performance of the aggregation backbones is highly dependent on the data scale. In the Limited-Data scenario, the curves are largely indistinguishable. However, as the scale is Moderate-Data, a subtle but consistent performance hierarchy emerges: PNR-IPT, PNR-IIR and PTR distinguish themselves by achieving a lower validation loss than both PNR and PNR-PT. This provides a possible explanation that deploying dynamic aggregation modules to process dense point clouds can help improve generalization performance.

The superiority of dynamic aggregation becomes clear in the Full-Data scenario, which evaluates the ability to reconstruct a field at high resolution. In this scenario, the architectures employing the PT block outperform the static baseline, with a validation loss lower than PTR. This indicates that when the task shifts to high-fidelity field reconstruction of known samples, the advantages of the complex dynamic module become apparent, serving as the key to enhance prediction accuracy.

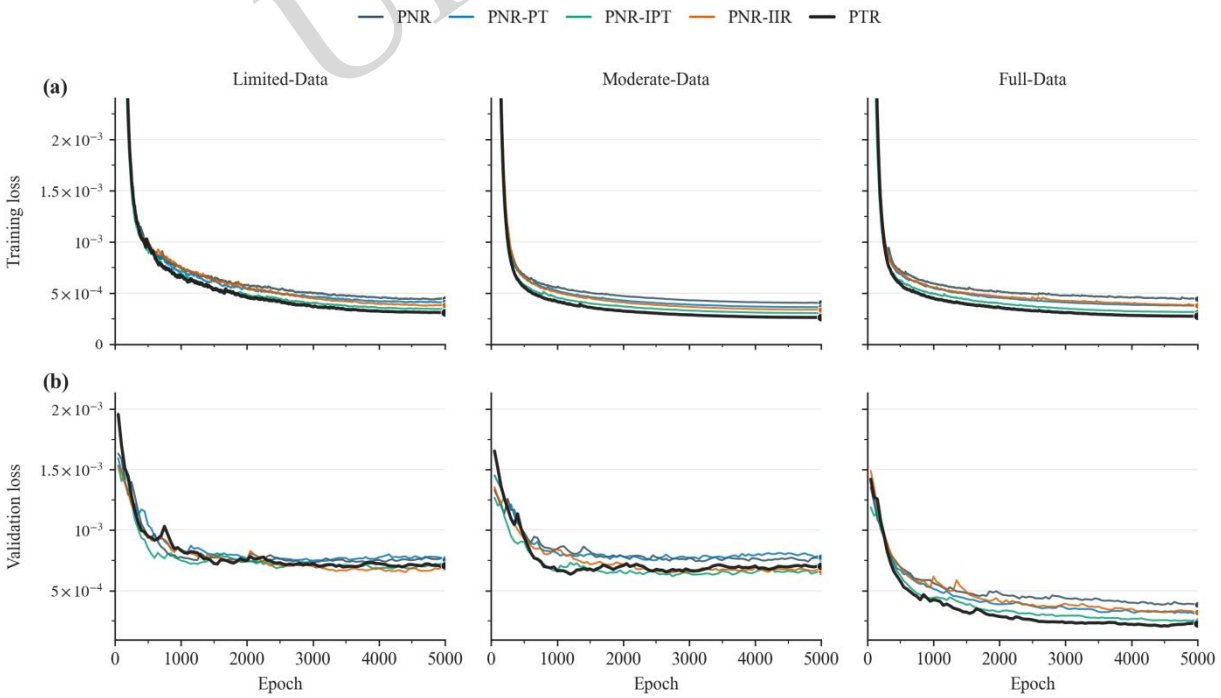


**Fig. 14. Convergence analysis of various aggregation strategies for geometric features in the three data scenarios. (a) Training loss and (b) validation loss.**

### 4.2.2 Testing of geometric aggregation

To quantify the performance of different

aggregation backbones, we evaluate the MSE of the pressure coefficient on 100 unseen samples for PNR-PT, PNR-IPT, PNR-IIR, PTR, and the baseline PNR. Table 2 summarizes the results of the models in various scenarios, with the best and worst performers highlighted in bold and italics, respectively. The last line shows the total time spent on the test.

We find that replacing static layers with the PT block may be a suboptimal design choice. As shown in the table, PNR-PT underperforms the static baseline PNR in three data scenarios, with performance degradations of 23.0%, 3.7% and 4.7%, respectively. This suggests that for such tasks, the theoretical advantages of attention mechanisms may only manifest when processing dense and feature-rich point clouds. Conversely, for the sparse point cloud encountered during the downsampling stages, the simpler static module proves to be more reliable.

PNR-IPT compensates for the loss incurred by PNR-PT relative to the baseline model in the three data scenarios, achieving MSE reductions of 10.9% and 5.1% in the Moderate-Data and Full-Data scenarios, respectively. Although PNR-IIR reduces MSE to a greater extent by 3.9% and 15.2% in the Limited-Data and Moderate-Data scenarios, it suffers a catastrophic failure in the full-data scenario (MSE surged by 57.6%). Therefore, utilizing the dynamic block for local information enhancement in the early network stage can provide a more reliable and higher-quality input for subsequent modules.

PTR introduces the PT block during the upsampling stage and achieves the best performance across all three data scenarios, with MSE reductions of 5.6%, 18.4% and 5.3%, respectively. This indicates that beyond feature extraction in the early stages of the network, enhancing local geometric relation modeling in the decoder can also improve prediction accuracy. However, the accuracy gain of PTR comes with a clear increase in computational cost. During training, PTR requires 14.11 GiB of GPU memory, which is higher than PNR-IPT at 9.09 GiB and the baseline model at 3.25 GiB. During testing, PTR also shows the highest memory usage, requiring 4.09 GiB/sample, compared with 3.91 GiB/sample for PNR-IPT and 1.43 GiB/sample for the baseline model. In addition, PTR requires 63 minutes to predict 100 samples, which is approximately twice the time required by PNR-IPT and approximately 60 times that of the baseline model.

This difference mainly results from the different local feature aggregation strategies. The InvResMLP module aggregates neighborhood features mainly through max-pooling, which is computationally straightforward. In contrast, although the Point Transformer block uses local *k*-nearest neighbors to restrict the attention range, it still needs to compute attention weights, positional encodings, and weighted feature aggregation between each center point and its neighboring points. Therefore, as the number of points increases during the upsampling stage, attention-based aggregation introduces more intermediate features and computational steps, increasing GPU memory usage during both training and testing and substantially extending inference time. Overall, PTR achieves the best prediction accuracy at the cost of higher computational and memory overhead.

**Table 2 Effect of aggregation backbones on the pressure coefficient MSE over the test set.**

| Scenarios | PNR (Baseline) | PNR-PT | PNR-IPT | PNR-IIR | PTR |
|---|---|---|---|---|---|
| Limited-Data | 6.64e-04 | 8.17e-04 (-23.0%) | 6.63e-04 (+0.2%) | 6.38e-04 (+3.9%) | 6.27e-04 (+5.6%) |
| Moderate-Data | 4.88e-04 | 5.06e-04 (-3.7%) | 4.35e-04 (+10.9%) | 4.14e-04 (+15.2%) | 3.98e-04 (+18.4%) |
| Full-Data | 4.72e-04 | 4.94e-04 (-4.7%) | 4.48e-04 (+5.1%) | 7.44e-04 (-57.6%) | 4.47e-04 (+5.3%) |
| Training GPU memory (GiB, batch size = 32) | 3.25 | 4.38 | 9.09 | 8.59 | 14.11 |
| Test GPU memory (GiB/sample) | 1.43 | 1.44 | 3.91 | 1.85 | 4.09 |
| Test time (Minutes,100 samples) | 1 | 1 | 32 | 26 | 63 |

4.2.3 Parameter sensitivity analysis

Additionally, to address the sensitivity of the model to training hyperparameter choices, we further conducted a lightweight parameter sensitivity analysis on the PNR model by varying the learning rate and weight decay. Starting from the default setting, we tested a smaller learning rate of 5e-4, a larger learning rate of 2e-3, and a reduced weight decay of 1e-5. The results in Table 3 show that, across the three data-scale scenarios, the test MSE of the pressure coefficient remains within the same order of magnitude, and most perturbed settings further reduce the MSE. Specifically, the relative MSE improvement over the default setting ranges from 0.6% to 21.5%. This indicates that the performance of the PNR model does not rely on a fragile set of optimizer hyperparameters and that the model maintains good training stability within the tested range of learning-rate and weight-decay perturbations.

Building on this analysis, we further investigate the influence of the neighborhood size $k$ in the grouping operation on aggregation performance. As shown in Table 4, for the PNR model, increasing $k$ from 4 to 16 leads to clear improvements in the pressure coefficient MSE across all three data-scale scenarios, with reductions ranging from 11.0% to 23.3%. Compared with the sensitivity results for the learning rate and weight decay, the neighborhood size $k$ directly controls the range of local geometric feature aggregation and therefore has a more explicit structural and physical meaning. For aircraft geometries with complex shapes, strong local curvature variations, and highly nonuniform flow features, a larger neighborhood receptive field helps the model capture local surface characteristics that are closely related to the aerodynamic response, thereby improving the prediction accuracy of the pressure coefficient.

Overall, the training hyperparameter sensitivity analysis demonstrates that the PNR model has good optimization stability, while the neighborhood-size analysis further shows that the geometric feature aggregation scale is an important factor affecting model performance. Although the parameter analysis in this work remains preliminary, these results suggest that future work could further optimize sampling strategies, learning-rate settings, and neighborhood-scale selection in a more systematic manner. Exploring adaptive or multiscale aggregation mechanisms may also provide a promising direction for further improving the prediction accuracy and generalization capability of models for complex aircraft aerodynamic characteristics.



**Table 3 Effect of learning rate and weight decay on the pressure coefficient MSE over 100 unseen test samples.**

| Setting | Learning rate | Weight decay | Limited-Data Scenario | Moderate-Data Scenario | Full-Data Scenario |
|---|---|---|---|---|---|
| Baseline | 1e-3 | 1e-4 | 8.66e-4 | 5.49e-4 | 5.72e-4 |
| Smaller learning rate | 5e-4 | 1e-4 | 7.64e-4 (+11.8%) | 5.26e-4 (+4.2%) | 5.08e-4 (+11.2%) |
| Larger learning rate | 2e-3 | 1e-4 | 8.29e-4 (+4.3%) | 4.87e-4 (+11.3%) | 4.60e-4 (+19.6%) |
| Smaller weight decay | 1e-3 | 1e-5 | 8.61e-4 (+0.6%) | 4.86e-4 (+11.5%) | 4.49e-4 (+21.5%) |

**Table 4 Effect of neighborhood size k on the pressure coefficient MSE over 100 unseen test samples.**

| Size | Limited-Data Scenario | Moderate-Data Scenario | Full-Data Scenario |
|---|---|---|---|
| k=4 (Baseline) | 8.66e-4 | 5.49e-4 | 5.72e-4 |
| k=16 | 6.64e-4 (+23.3%) | 4.88e-4 (+11.1%) | 4.72e-4 (+17.5%) |

## 5 Conclusions and Future Work

Focusing on the critical challenge of

generalization for 3D aircraft surface pressure fields under multi-condition scenarios, this paper presents a systematic feature engineering study built upon the PointNextReg architecture. The study evaluates two fundamental design choices: the fusion of geometric and condition features, where we compare static (Concatenation) and dynamic approaches (inspired by DeepONet and Transformer), and the backbone of geometric feature aggregation. Finally, two practical design guidelines are summarized.

(1) **A condition-fusion strategy based on progressive static fusion and DeepONet-style dynamic modulation should be adopted, while simple late-stage concatenation should be avoided, and attention-based reweighting mechanisms should be used with caution.** This study finds that the geometry-condition fusion strategy has a significant impact on model prediction performance. Simply concatenating the condition feature at the final regression stage tends to weaken generalization. In contrast, early fusion, multilevel fusion, and deep global fusion allow condition information to participate more fully in geometric feature extraction and aggregation, leading to performance closer to the early-fusion baseline and providing a more robust modeling foundation. The dynamic fusion experiments further show that the effectiveness of dynamic mechanisms depends strongly on the data scale. Attention-based dynamic reweighting strategies do not provide stable improvements and show stronger data dependence and larger error fluctuations. In contrast, DeepONet-style modulation shows greater potential when sufficient training samples are available. This suggests that directly modulating geometric features with condition information is feasible, but the expressive capacity of the modulation mechanism must be balanced with the preservation of geometric information.

(2) **A hybrid backbone with dynamic aggregation in dense stages and static aggregation in sparse stages should be adopted to balance prediction accuracy and computational cost.** This study finds that the effectiveness of dynamic geometric aggregation modules is highly dependent on their deployment location. Directly replacing all static InvResMLP modules with Point Transformer modules does not necessarily improve performance and may even degrade prediction accuracy. A more effective strategy is to introduce more expressive dynamic aggregation modules in the high-resolution dense point-cloud stages, where rich local neighborhood information can be used to capture critical geometric details. In contrast, in the downsampled sparse point-cloud stages, more robust static aggregation modules should be retained to extract global features stably through their structural inductive biases. This hybrid design improves expressive capacity while reducing the risk that dynamic modules learn spurious correlations from sparse point clouds.

Overall, this study systematically extends and analyzes the PointNextReg baseline for three-dimensional aircraft surface pressure-field prediction. By focusing on condition-geometry feature fusion and geometric feature aggregation, it clarifies how different module designs affect generalization, stability, and computational cost and summarizes practical design principles for subsequent point-cloud surrogate models. For practical applications, the balance between accuracy and efficiency should be considered according to the design stage. Lightweight fusion and aggregation strategies are more suitable for preliminary design, where rapid iteration is needed, whereas the transformer-based upsampling design may be preferable for detailed design, where prediction accuracy is prioritized.

However, the present work remains a data-driven surrogate modeling study based on precomputed CFD samples and a specific family of three-dimensional aircraft configurations. Therefore, the proposed design principles should be regarded as references for similar point-cloud pressure-field modeling tasks, rather than universal rules for all aircraft families and flow regimes. More systematic evaluation across different aircraft families, flow regimes, and distribution shifts will be an important direction for future work. In addition, the models depend on dataset quality and coverage and do not explicitly impose variational physical constraints, directly solve the governing equations, or quantify physical input uncertainty. Integrating point-cloud surrogate models with the deep energy method (Samaniego et al., 2024) or variational physics-informed neural operators (Eshaghi et al., 2025) is another important direction for improving

physical consistency, uncertainty handling, and inverse-design capability.

## Acknowledgments

This work is supported by the Fund of National Key Laboratory of Advanced Propulsion Technology (No. 25-GC-ZZKY-006) and the National Natural Science Foundation of China (No. 12472338).

## Author contributions

Hao Zhang and Yang Shen, designed the research, processed the corresponding data and wrote the first draft of the manuscript. Hao Wang, Wei Huang and Yao-bin Niu, helped to organize the manuscript. Wei Huang, Shuang-xi Liu and Chao-yang Liu revised and edited the final version.

## Conflict of interest

Hao Zhang, Yang Shen, Hao Wang, Wei Huang, Yao-bin Niu, Shuang-xi Liu and Chao-yang Liu declare that they have no conflict of interest.

## Declaration on the use of generative AI tools

As generative AI tools have become increasingly popular, the authors declare have used these tools during the preparation of the manuscript.

## Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.



## References

Martins JR, Lambe AB, 2013. Multidisciplinary design optimization: a survey of architectures. *AIAA Journal*, 51(9), 2049-2075. https://doi.org/10.2514/1.J051895.

Liu J, Chen R, Lou J, et al., 2023. Deep-learning-based aerodynamic shape optimization of rotor airfoils to suppress dynamic stall. *Aerospace Science and Technology*, 133, 108089. https://doi.org/10.1016/j.ast.2022.108089.

Liu F, Han ZH, Zhang Y, et al., 2019. Surrogate-based aerodynamic shape optimization of hypersonic flows considering transonic performance. *Aerospace Science and Technology*, 93, 105345. https://doi.org/10.1016/j.ast.2019.105345.

Mukherjee NK, Rapol S, Unnithan S, et al., 2020. Optimization of a 3D combustion bowl geometry using response surface modeling. *SAE International Journal of Engines*, 14(03-14-02-0011), 173-198. https://doi.org/10.4271/03-14-02-0011.

Kilic D, Nikbay M, 2025. Implementation and assessment of machine learning methods for multidisciplinary aerospace problems. AIAA SCITECH 2025 Forum, p.0492. https://doi.org/10.2514/6.2025-0492.

Sun G, Wang S, 2019. A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233(16), 5863-5872. https://doi.org/10.1177/0954410019864485.

Zhang Y, Han ZH, Shi L, et al., 2016. Multi-round surrogate-based optimization for benchmark aerodynamic design problems. 54th AIAA Aerospace Sciences Meeting, p.1545. https://doi.org/10.2514/6.2016-1545.

Li J, Du X, Martins JRRA, 2022. Machine learning in aerodynamic shape optimization. *Progress in Aerospace Sciences*, 134, 100849. https://doi.org/10.1016/j.paerosci.2022.100849.

Guo Y, Wang H, Hu Q, et al., 2020. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12), 4338-4364. https://doi.org/10.1109/TPAMI.2020.3005434.

Chen F, Akasaka K, 2021. 3D Flow Field Estimation around a Vehicle Using Convolutional Neural Networks. 32nd British Machine Vision Conference, p.396. https://doi.org/10.5244/c.35.272.

Bhatnagar S, Afshar Y, Pan S, et al., 2019. Prediction of aerodynamic flow fields using convolutional neural networks. *Computational Mechanics*, 64(2), 525-545. https://doi.org/10.1007/s00466-019-01740-0.

Duru C, Alemdar H, Baran ÖU, 2021. CNNFOIL: Convolutional encoder decoder modeling for pressure fields around airfoils. *Neural Computing and Applications*, 33(12), 6835-6849. https://doi.org/10.1007/s00521-020-05461-x.

Chen LW, Cakal BA, Hu X, et al., 2021. Numerical investigation of minimum drag profiles in laminar flow using deep learning surrogates. *Journal of Fluid Mechanics*, 919, A34. https://doi.org/10.1017/jfm.2021.398.

Hines D, Bekemeyer P, 2023. Graph neural networks for the prediction of aircraft surface pressure distributions. *Aerospace Science and Technology*, 137, 108268. https://doi.org/10.1016/j.ast.2023.108268.

Massegur D, Da Ronch A, 2024. Graph convolutional multi-mesh autoencoder for steady transonic aircraft aerodynamics. *Machine Learning: Science and Technology*, 5(2), 025006. https://doi.org/10.1088/2632-2153/ad36ad.

Shao X, Liu Z, Zhang S, et al., 2023. PIGNN-CFD: A physics-informed graph neural network for rapid predicting urban wind field defined on unstructured mesh. *Building and Environment*, 232, 110056. https://doi.org/10.1016/j.buildenv.2023.110056.

Kashefi A, Rempe D, Guibas LJ, 2021. A point-cloud deep learning framework for prediction of fluid flow fields on irregular geometries. *Physics of Fluids*, 33(2). https://doi.org/10.1063/5.0033376.

Xiong F, Zhang L, 2023. A point cloud deep neural network metamodel method for aerodynamic prediction. *Chinese*

*Journal of Aeronautics*, 36(4), 92-103. https://doi.org/10.1016/j.cja.2022.11.025.

Shen Y, Huang W, Wang Z, et al., 2023. A deep learning framework for aerodynamic pressure prediction on general three-dimensional configurations. *Physics of Fluids*, 35(10). https://doi.org/10.1063/5.0172437.

Zhang H, Shen Y, Huang W, et al., 2025. Deep transfer learning for three-dimensional aerodynamic pressure prediction under data scarcity. *Theoretical and Applied Mechanics Letters*, 15(2), 100571. https://doi.org/10.1016/j.taml.2025.100626.

Shukla K, Oommen V, Peyvan A, et al., 2024. Deep neural operators as accurate surrogates for shape optimization. *Engineering Applications of Artificial Intelligence*, 129, 107615. https://doi.org/10.1016/j.engappai.2023.107615.

Shih B, Peyvan A, Zhang Z, et al., 2025. Transformers as neural operators for solutions of differential equations with finite regularity. *Computer Methods in Applied Mechanics and Engineering*, 434, 117560. https://doi.org/10.1016/j.cma.2024.117560.

Qi CR, Su H, Mo K, et al., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, p.652-660. https://doi.org/10.1109/cvpr.2017.16.

Qian G, Li Y, Peng H, et al., 2022. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in neural information processing systems*, 35, 23192-23204. https://doi.org/10.52202/068431-1685.

Shi S, Guo C, Jiang L, et al., 2020. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, p. 10529-10538. https://doi.org/10.1109/cvpr42600.2020.01054.

Zhao H, Jiang L, Jia J, et al., 2021. Point transformer. Proceedings of the IEEE/CVF international conference on computer vision, p.16259-16268. https://doi.org/10.1109/iccv48922.2021.01595.

Yuan Z, Song X, Bai L, et al., 2021. Temporal-channel transformer for 3D LiDAR-based video object detection for autonomous driving. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4), 2068-2078. https://doi.org/10.1109/tcsvt.2021.3082763.

Ma L, Kong L, Peng X, et al., 2024. PSTNet: Transformer for aggregating neighborhood features in 3D point cloud semantic segmentation of eggplant plants. *Scientia Horticulturae*, 331, 113158. https://doi.org/10.1016/j.scienta.2024.113158.

Shen Y, Zhang H, Huang W, et al., 2024. Geometric-perspective transfer learning for fast aerodynamic prediction in few-shot tasks. *Physical Review Fluids*, 9(10), 104101. https://doi.org/10.1103/physrevfluids.9.104101.

Perez E, Strub F, De Vries H, et al., 2018. Film: Visual reasoning with a general conditioning layer. Proceedings of the AAAI conference on artificial intelligence, p. 32(1). https://doi.org/10.1609/aaai.v32i1.11671.

Park T, Liu MY, Wang TC, et al., 2019. Semantic image synthesis with spatially-adaptive normalization. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, p. 2337-2346. https://doi.org/10.1145/3306305.3332370.

Lu L, Jin P, Pang G, et al., 2021. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature machine intelligence*, 3(3), 218-229. https://doi.org/10.1038/s42256-021-00302-5.

Vaswani A, Shazeer N, Parmar N, et al., 2017. Attention is all you need. *Advances in neural information processing systems*, 30. https://doi.org/10.65215/nxvz2v36.

Erickson LL, 1990. Panel methods: An introduction. *NASA Technical Paper*, (2995), 68.

Ding F, Liu J, Shen C, et al., 2015. Novel inlet–airframe integration methodology for hypersonic waverider vehicles. *Acta Astronautica*, 111, 178-197. https://doi.org/10.1016/j.actaastro.2015.02.016.

Samaniego E, Anitescu C, Goswami S, et al., 2020. An energy approach to the solution of partial differential equations in computational mechanics via machine learning: Concepts, implementation and applications. *Computer Methods in Applied Mechanics and Engineering*, 362, 112790. https://doi.org/10.1016/j.cma.2019.112790.

Eshaghi MS, Anitescu C, Thombre M, et al., 2025. Variational physics-informed neural operator (VINO) for solving partial differential equations. Computer Methods in Applied Mechanics and Engineering, 437, 117785. https://doi.org/10.1016/j.cma.2025.117785.

## Electronic supplementary materials

Figs. S1–S3

# 中文概要

**题　目：** 面向多工况场景的泛化飞行器压力场建模特征工程研究

**作　者：** 张好，沈洋，王昊，黄伟，钮耀斌，刘双喜，刘朝阳

**机　构：** 国防科技大学，先进推进技术国家级重点实验室，中国长沙，410073

**目　的：** 针对多工况条件下三维飞行器表面压力场预测的

泛化关键难题，系统研究点云深度代理模型中条件特征与几何特征融合、几何局部信息聚合的设计策略，提升模型的预测精度、稳定性和计算效率。

**创新点：** 1. 提出了渐进式静态融合与 DeepONet 风格动态调制的工况信息引入机制，揭示了早期融合、多级融合与后期简单拼接对泛化性能的差异化影响，明确了注意力重加权机制在数据不足时的不稳定性；2. 构建了混合几何聚合骨干网络，在高分辨率密集点云阶段采用动态 Transformer 聚合以捕捉关键几何细节，在降采样的稀疏阶段保留静态聚合以鲁棒提取全局特征，实现了表达力与计算成本的有效平衡；3. 基于系统消融实验，凝练出面向点云气动代理模型的两条实用设计原则，为同类建模任务提供可迁移的参考。

**方　法：** 1. 基于三维飞行器构型，构建马赫数 3.0-10.0、攻角 -10°-10° 的任务数据集，采用 PointNextReg 作为基线架构；2. 设计并对比条件-几何特征融合方案：静态拼接（Concatenation）、DeepONet 风格动态调制，以及基于注意力的 Transformer 重加权，分析其在不同数据规模变化下的表现；3. 对几何聚合骨干网络进行消融实验，比较全静态 InvResMLP 模块、全动态 Point Transformer 模块及混合设计在不同点云分辨率阶段的有效性，评估其对预测精度与计算代价的影响。

**结　论：** 1. 工况特征应采用贯穿式静态融合、样本充分时动态调制，简单的高层拼接会弱化泛化能力，注意力重加权机制在数据不充分时易产生较大误差波动；2. 几何聚合宜采用混合骨干设计：在密集点云阶段使用动态聚合以利用丰富邻域信息，在稀疏点云阶段保留静态聚合以稳定提取全局特征，从而兼顾精度与效率；3. 所得设计原则为同类点云压力场代理模型提供了指导，但尚需跨飞机家族、跨流态和分布偏移的系统性验证；未来可将点云代理模型与深度能量法或变分物理信息神经算子融合，以增强物理一致性、不确定性量化与反向设计能力。

**关键词：** 点云代理模型；条件-几何特征融合；动态聚合；飞行器表面压力场预测