



OPEN Two stage multiobjective topology optimization method via SwinUnet with enhanced generalization

Cheng Xiang^{1,2}, Airong Chen², Hua Li³, Dalei Wang², Baixue Ge² & Haocheng Chang²✉

Topology optimization is a critical tool for modern structural design, yet existing methods often prioritize single objectives (e.g., compliance minimization) and suffer from prohibitive computational costs, especially in multi-objective scenarios. To address these limitations, this paper introduces a novel two-stage multi-objective topology optimization (MOTO) method that uniquely integrates data-driven learning with physics-informed refinement, and both stages are implemented within nearly identical network frameworks, ensuring simplicity and consistency in execution. Firstly, a MOTO mathematical model based on the constraint programming method that considers competing objectives of compliance, stress distribution, and material usage was constructed. Secondly, a novel neural network that incorporates shifted windows attention mechanism and lightweight modules was developed to enhance feature extraction while maintaining computational efficiency. Finally, the proposed model was trained in two stages: In Stage-1, utilizing adaptive input tensors, the network predicts near-optimal geometries across variable design domains (including rectangular and L-shaped configurations) and diverse boundary conditions in real time, requiring only 1,650 samples per condition. In Stage-2, the near-optimal structures from Stage-1 were physically optimized to achieve optimal performance. Experimental results demonstrate that the method's capability to generate high-accuracy, computationally efficient solutions with robust generalization capabilities. It effectively tackles challenges associated with multi-scale design domains and non-convex geometries, various and even untrained boundary conditions while significantly reducing data dependency, a critical advancement for data-driven topology optimization. The novel approach offers new insights for multi-objective structural design and promotes advancements in structural design practices.

Keywords Topology optimization, Deep learning, Multi-object optimization, Self-attention mechanism, Physics-informed neural networks, Generalization ability

Structural optimization seeks to pursue structures with higher efficiency (such as higher performance, lower costs, lightweight, etc.)¹. Within the field of structural optimization, topology optimization (TO) is dedicated to identifying the optimal distribution of material within a given design domain (DD) under different constraints. Over the past few decades, TO has experienced rapid development, leading to the emergence of numerous numerical methods, such as SIMP^{2,3}, evolutionary approaches⁴, level-set method^{5,6}, and moving morphable components method (MMC)⁷. Despite their widespread adoption, traditional methods for solving TO are hindered by their computational expense, necessitating a large number of design variables and iterations during the optimization process. Each iteration requires finite element analysis (FEA) and sensitivity analysis to inform updates to the design variables⁸. This computational inefficiency significantly hampers the practical application of TO, highlighting the urgent need for new methods to enhance its efficiency and practicality.

With the growing popularity of deep learning (DL), significant advancements have been made across various fields. In recent years, several DL-based TO methods have emerged to mitigate computational burdens. Early approaches in this area primarily focused on developing end-to-end TO frameworks that aimed to directly map design constraints to optimal topologies, emphasizing real-time prediction of optimized structures. For examples, Sosnovik and Oseledets⁹ pioneered the integration of TO and DL by framing TO as an image segmentation task. Using the convolutional encoder-decoder architecture, the near-optimal structures can be obtained without iterative processes. With a carefully designed initial physical field input tensor, Wang et al.^{10,11} developed a deep CNN with perceptible generalization ability for TO, which can rapidly predict the 2D or 3D optimized structures

¹College of Civil Engineering, Fuzhou University, Fuzhou 350116, China. ²Department of Bridge Engineering, Tongji University, Shanghai 200092, China. ³Fujian Transportation Planning and Design Institute Co. LTD., Fuzhou 350004, China. ✉email: changhc16@tongji.edu.cn

even with previously unseen boundary conditions (BCs). Rochefort-Beaudoin et al.¹² proposed a two-stage method in which the predicted results of a trained CNN were used as the initial arrangement of components in MMC, which was further refined by executing a limited number of MMC iterations, significantly reducing computational time while maintaining optimal performance. By utilizing optimization history information and combining independent sub-models, Yan et al.¹³ introduced a step-to-step training method which improves the prediction accuracy of a DL model for real-time TO.

The research previously discussed has predominantly focused on compliance as the primary optimization objective. Beyond compliance, studies have also explored other objectives, contributing to a broader scope of TO applications. For examples, Li et al.¹⁴ developed a non-iterative topology optimizer for conductive heat transfer structures, capable of predicting near-optimal structure both in both low and high resolutions. Oh et al.¹⁵ proposed an AI-based deep generative design framework that generates diverse design options, balancing aesthetics with optimized engineering performance. Xiang et al.¹⁶ presented an iteration-free TO method that utilized CNN to minimize stress distribution in 2D and 3D structures. Deng et al.¹⁷ introduced a self-directed online learning optimization method for complex TO problems, such as fluid-structure optimization, heat transfer enhancement, significantly reducing computational time. Recent advances further extend TO to reliability-based design under uncertainty. For instance, Habashneh et al.¹⁸ addressed imperfect structures with uncertain load positions through reliability-based optimization, while Li et al.¹⁹ proposed a proportional-integral-derivative-driven sequential decoupling framework for efficient concurrent TO, achieving enhanced computational robustness.

The aforementioned studies highlight the transformative potential of integrating DL with TO, and has made significant progress in different optimization fields. However, practical structural design often involves balancing multiple conflicting objectives. For instance, ensuring structural stiffness while achieving uniform stress distribution is essential to avoid stress concentrations—a typical multi-objective optimization (MOTO) scenario. Existing research, which largely focuses on single-objective optimization, fails to address the diverse and often competing demands of real-world structural applications. To date, there is a noticeable lack of research on the integration of MOTO with DL, underscoring the need for further exploration in this area. Besides, while the end-to-end approaches have confirmed its remarkable efficiency advantages, it is an image-based method that primarily emphasizes achieving high pixel similarity. However, TO involves strict mechanical constraints. Consequently, this approach may give rise to issues of structural disconnection, a problem that is commonly observed in related research^{20,21}. Additionally, as a typical data-driven approach, end-to-end methods often require a large amount of training data to train an ideal model^{10,13,14,20}. This substantial data requirement can, to some extent, offset the efficiency advantages of such approach. Furthermore, once the design conditions changed, variations in DD sizes or BCs, the model needs to be retrained, and the lack of generalization ability also limits the practical application of this kind of method^{12,16}.

To address the above-mentioned problems, different approaches have been come up with. On the one hand, researchers are attempting to incorporate mechanical information into the network to guide the optimization process. For example, Zhang et al.²² gained the optimized structures by adopting the physics-informed neural networks (PINN)²³ that incorporating physical information of compliance into the loss function and replacing the element density update in TO with the neural networks (NN) parameter update. Jeong et al.²⁴ introduced a novel TO framework based on PINN, which utilizes an energy-based PINN with the multilayer perceptron as the basic framework to replace FEA and achieves optimized topologies without labeled data. These methods explicitly incorporate physical information, thereby avoiding structural discontinuity. Nevertheless, the PINN-based approaches often require hundreds of FEA and frequent network parameter updates, making them even more time-consuming than conventional methods and failing to deal with the efficiency issue²⁵. On the other hand, some recently proposed advanced network architectures have also been introduced into TO task to improve the prediction accuracy. For instance, the attention mechanism, as the core of Transformer and the popular large language model like ChatGPT²⁶, is integrated into the TO task to overcome the limitations of the end-to-end approaches²⁷. Seo and Min²⁸ introduced the graph neural networks for accelerating TO in irregular DDs, achieving near-optimal topologies by encoding complex non-Euclidean data into Euclidean form. Mazé and Ahmed²⁹ proposed a conditional diffusion model-based architecture that introduces a surrogate model-guided strategy that actively favors structures with low compliance and good manufacturability, which significantly outperforms the state-of-the-art conditional GANs.

To summarize, the integration of DL into TO was initially motivated by the need to address the high computational costs associated with iterative procedures. End-to-end methods have emerged as particularly efficient, offering a significant advantage that has driven their widespread adoption. The development of novel NN architectures, combined with innovative training strategies, such as designing specialized input tensors and tailored loss functions, has demonstrated the substantial potential of the end-to-end approach for further advancements. However, these methods face notable limitations in integrating physical information, resulting in potential risks in terms of structural mechanical performance. Additionally, their generalization capabilities remain inadequate and require substantial improvement. In contrast, the PINN-based approaches strictly adhere to principles of mechanical performance, eliminating the need for constructing training datasets. Nevertheless, these methods are fundamentally similar to the Method of Moving Asymptotes (MMA)³⁰ and Optimality Criteria method³¹, differing primarily in their use of NNs weight updates to modify design parameters. With these approaches, each iteration step requires both FEA and the updating of NNs parameters, resulting in significantly higher computational costs compared to traditional TO methods.

Considering the advantages and limitations of both approaches, leveraging their strengths while mitigating their weaknesses could significantly enhance the practical application of DL in MOTO. To this end, this paper proposes a two-stage MOTO method that utilizes almost identical network frameworks in both stages. In the first stage (Stage-1), an end-to-end model is trained using a small dataset, allowing it to predict near-optimal

structures for different DD shapes and sizes and BCs in real time. In the second stage (Stage-2), following the idea of PINN-based approaches, the loss function of the same network is replaced with the optimization objective function, enabling further optimization of the structures using physical information. Since the near-optimal structures predicted in Stage-1 are already very close to the optimal solution, Stage-2 typically requires only a limited number of iterations to achieve the final optimized designs. The proposed approach can effectively realize a balance between the efficiency, accuracy, as well as generalization ability of MOTO.

The remainder of this paper is organized as follows: In Section 2, a typical MOTO problem considering optimization objectives of stiffness, strength, and material usage that have great impact on structural design are presented. In Section 3, a novel high-performance network model that integrating shifted windows attention mechanism and lightweight modules is constructed. In Section 4, the proposed two-step MOTO method, which employs almost the same model architecture in both stages, is introduced. Section 5 evaluates the performance of the method in both stages, as well as its generalization capabilities. Finally, Section 6 concludes the paper.

Typical MOTO problem

Currently, common multi-objective optimization methods include the weighted sum method, the compromise programming method, and the constraint-based method³². The weighted sum method³³ assigns weight coefficients to each objective in proportion to their relative significance within the problem. The objective function is then formulated as the sum of these weighted objectives. The compromise programming method³⁴, on the other hand, is primarily employed to tackle multi-objective optimization problems where the sub-objectives exhibit distinct characteristics. This method normalizes the multiple sub-objective functions to ensure their values fall within the range of 0 to 1, and then the problem is transformed into a single-objective optimization problem by assigning weights to each sub-objective. However, TO is typically approached using sensitivity-driven methods. When it comes to MOTO problems, the use of both the weighted sum method and the compromise programming method often fails to converge due to the significant disparities in both the physical meanings and numerical values among various objectives. Consequently, significant oscillations arise in both the objective function values and the sensitivity values throughout the optimization process. Therefore, given the substantial variations in physical meanings and numerical values among them, merging multiple optimization objectives into a single objective through weighting or employing normalization techniques like compromise coefficients may not be appropriate for the MOTO problem discussed in this paper.

The constraint-based method²² operates on the principle of designating a primary optimization objective among several sub-objective functions, while pre-specifying desired values for the remaining objectives. These additional objectives are then incorporated as constraint conditions within the optimization problem. By solving the constrained optimization problem, the optimized value of the primary objective can be obtained. This approach calculates the sensitivities of the objective function and constraint functions independently, thereby mitigating the oscillation issues in sensitivities and objective functions that can occur in the weighted sum and compromise programming methods due to substantial disparities in physical meanings or numerical values. Therefore, this research employs the constraint-based method to formulate the MOTO problem.

Global P-norm stress measurement

Strength, as assessed by stress level, is a pivotal factor in structural design. To ensure reliability and safety, design defects arising from insufficient strength must be avoided during the design process. Nevertheless, due to the three significant challenges³⁵, namely the singularity phenomenon, the local nature of the constraint, and the highly non-linear stress behavior, the stress optimization problem is significantly more intricate than the compliance optimization problem. Fortunately, diligent efforts by scholars addressing these issues have yielded substantial progress.

The singularity phenomenon predominantly arises in density-based TO methods, where elements with low densities can exhibit high-stress values, thereby complicating the optimization algorithms to remove such elements. Various mitigation strategies have been devised to address this issue, like the stress penalization method³⁶ that penalizes such elements, and the BESO-based method³⁷, which prevents the appearance of intermediate-density elements, thereby eliminating the singularity phenomenon. The second challenge stems from the fact that stress is calculated locally at each element, necessitating control over the stress level of each individual element during optimization, leading to an unacceptable number of constraints. To simplify the stress constraints, Le et al.³⁵ proposed a global stress constraint that aggregates all local stress constraints into a single measure. This method has since been extensively utilized in addressing stress-based TO problems³⁸. The non-linear stress behavior primarily arises from the high sensitivity of stress to topological variations. Currently, the filtering method, which is commonly used in TO, has also been proven effective in enhancing the stability of stress behavior³⁹.

This paper will focus on implementing the SIMP method for MOTO. The Young's modulus E_i is determined by the i th element's density x_i :

$$E_i = E_{min} + x_i^q (E_0 - E_{min}) \quad (1)$$

Where E_0 is the Young's modulus of the material, and E_{min} is a very small value designed to prevent singularity in the stiffness matrix. In this research, the material properties were simplified by setting $E_0 = 1.0$, $E_{min} = 0.001$, and the Poisson's ratio $\nu = 0.3$. These simplifications influence only the relative values of the results and do not alter the final topological configurations. The parameter q , set to 3 in this study, serves as the penalization factor in the SIMP method.

To prevent the singularity phenomenon, the penalization method³⁶ is adopted as follows.

$$\hat{\sigma}_i = (x_i)^{0.5} \sigma_i \quad (2)$$

Where $\hat{\sigma}_i$ is the penalized stress, and σ_i is the effective stress vector. According to Equation (2), $\hat{\sigma}_i$ is equal to σ_i for solid elements and $\hat{\sigma}_i \rightarrow 0$ when $x_i \rightarrow 0$.

To alleviate the local nature of stress, the widely recognized p-norm global stress σ_{PN} is adopted to approximate the maximum stress σ_{max} as follows.

$$\sigma_{PN} = \left(\sum_{i=1}^N \sigma_{vm,i}^p \right)^{\frac{1}{p}} \quad (3)$$

Where N is the total number of elements in the DD, $\sigma_{vm,i}$ is the von Mises stress of the i th element, p is the stress norm parameter. In theory, σ_{PN} is equal to σ_{max} when $p \rightarrow \infty$. However, using an excessively large value of p during the optimization process can lead to severe oscillations, which may result in the convergence to a local optimal solution with a high σ_{max} . In light of related literature^{35,37,39}, our research employs a value of $p = 6$ for the subsequent dataset generation.

Mathematical model of MOTO problem

Consistent with the mainstream research, when formulating the mathematical model for the MOTO problem, we designate compliance as the primary optimization objective, while treating stress and material consumption as constraints, as shown in Equation (4) - (8). This mathematical model aims to minimize structural compliance while ensuring a rational stress distribution and material usage. Consequently, the optimized structure not only exhibits substantial stiffness, but also satisfies the necessary strength requirements and considerations for economic performance.

$$\min : c(\mathbf{x}) = \mathbf{U}^T \mathbf{K} \mathbf{U} = \sum_{i=1}^N E_i(x_i) \mathbf{u}_i^T \mathbf{k}_0 \mathbf{u}_i \quad (4)$$

$$s.t. : \mathbf{K} \mathbf{U} = \mathbf{F}, \quad (5)$$

$$\frac{V(\mathbf{x})}{V_0} = f, \quad (6)$$

$$\sigma_{vm}^{max} \leq \sigma_{vm}^*, \quad (7)$$

$$0 \leq x_i \leq 1, \quad i = 1, 2, \dots, N \quad (8)$$

Where c is the compliance, $\mathbf{K}$ is the global stiffness matrix, $\mathbf{U}$ is the global displacement vector, $\mathbf{F}$ is the global force vector. $\mathbf{k}_0$ is the element stiffness matrix for an element with fully distributed solid material. $\mathbf{u}_i$ is the element displacement vector. N is the number of elements in the discrete DD. $V(\mathbf{x})$ and V_0 are the material volume and the DD volume, respectively. V_f is the pre-defined volume fraction. σ_{vm}^{max} is the maximum σ_{vm} value, and σ_{vm}^* is the target value for limiting the maximum σ_{vm} .

During the optimization process, to avoid the local characteristics of stress, the stress constraint equation was modified by replacing the maximum stress σ_{vm}^{max} with σ_{PN} , as depicted in Equation (3). Given that the MOTO problem is inherently nonlinear, it is solved using the MMA algorithm³⁰, which primarily leverages the sensitivity of both objective and constraint functions with respect to the design variables.

Definition of stress fraction

The degree of stress constraint (denoted as the value of σ_{vm}^*), can significantly influence the final forms of the structure. In the proposed MOTO problem, σ_{vm}^* is introduced to optimize the stress distribution during the compliance optimization process. Thus, it is essential to first determine the stress level (σ_0) under conditions where compliance is minimized with V_f only. The upper and lower limits of σ_{vm}^* are determined iteratively. When the optimized structure under σ_{vm}^* is matched the optimized structure with the minimum compliance under only V_f (for example, the compliance is almost the same and the pixel difference is within 5.0%), it is considered that the σ_{vm}^* at this time has no effect on structural optimization, and this value is denoted as the upper limit of the stress constraint σ_{vm}^u . On the other hand, when the σ_{vm}^* and volume constraint cannot be satisfied simultaneously, it is considered that the σ_{vm}^* at this point represents the minimum stress constraint σ_{vm}^l . Actually, these values can be easily achieved within a limited number of iterations using methods like Bisection method⁴⁰.

Similar to the clear definition of V_f , which represents the ratio of preserved material volume to the DD volume (with $V_f = 0$ indicating no material present, and $V_f = 1$ signifying the DD fully filled with material), we aim to introduce the concept of stress fraction S_f , where stress constraint can be expressed in fractional form. Consequently, it is essential to attribute analogous physical interpretations to the stress constraint: when $S_f = 0$, the stress constraint does not affect the results, and when $S_f = 1$, the stress exerts maximum influence on the results.

Introducing stricter constraints would render the optimization problem unsolvable. The stress constraint values between σ_{vm}^l and σ_{vm}^u can be obtained through weight interpolation. Typically, the value of σ_{vm}^l/σ_0 ranges from 0.8 to 0.95, while the range of σ_{vm}^u/σ_0 is relatively wide, even up to 3.0. The stronger the constraint,

the more pronounced its influence on the topology forms. To fully explore this influence, particular emphasis should be placed on studying the results within the realm of strong stress constraints. Linear interpolation scheme often falls short in addressing high-stress constraint areas. Therefore, an exponential interpolation scheme is adopted, as shown in Equation (9).

$$\sigma_{vm}^* = \frac{a}{b + S_f^k} \sigma_0 = r \sigma_0 \quad (9)$$

Where parameters a and b represent the interpolation function parameters, and k represents the exponent, along with the definition of σ_{vm}^u and σ_{vm}^l , the following relationships can be obtained:

$$\sigma_{vm}^u = \frac{a}{b} \sigma_0 \quad (S_f = 0); \quad \sigma_{vm}^l = \frac{a}{b+1} \sigma_0 \quad (S_f = 1) \quad (10)$$

By combining Equation (9) and (10), the stress constraint coefficient r under S_f can be derived as follows:

$$r = \frac{\sigma_{vm}^u \sigma_{vm}^l}{\sigma_0 (\sigma_{vm}^l + S_f^k (\sigma_{vm}^u - \sigma_{vm}^l))} \quad (11)$$

The variations of coefficient r across different exponents k , as well as the linear interpolation scheme, are shown in Figure 1, where σ_{vm}^u is set to 2, σ_{vm}^l to 0.85, and σ_0 to 1 for illustration. It can be observed that the r values of the linear interpolation approach uniformly decrease with increasing S_f . This decrease exhibits the same magnitude of change in both high-stress constraint and low-stress constraint regions, thereby failing to differentiate the impact of high-stress constraints on the results of TO. By employing exponential interpolation schemes with varying exponents k , it is evident that they can dynamically modulate the extent of changes in response to the degree of influence of the constraint. In regions with low-stress constraints, where stress constraints exert a small impact on TO results, a more pronounced decrease in the r value is applied. Conversely, in high-stress constraint areas where stress constraints profoundly influence optimization results, the decrease in the r value is gradual, effectively accounting for the significant impact of high-stress constraints. Within this interpolation scheme, and taking into account the various trends of the r value across different regions corresponding to various k values, this study ultimately sets k to 0.75.

The novel SwinUnet framework

In current end-to-end methods, a fundamental issue leading to structural disconnections arises from the lack of correlation between adjacent elements in disconnected regions. To address this challenge, the self-attention mechanism, being the core unit of the transformer framework, has inspired research in computer vision and large language models since its introduction⁴¹. This mechanism enables the capture of correlations among different input elements, thereby fostering stronger connections within the model and expanding its receptive field. Consequently, it holds great promise for alleviating structural disconnections in end-to-end TO methods. Building on this insight, our study endeavors to integrate the self-attention mechanism into the UNet framework⁴², which is renowned in DL-based TO for its remarkable feature extraction capability.

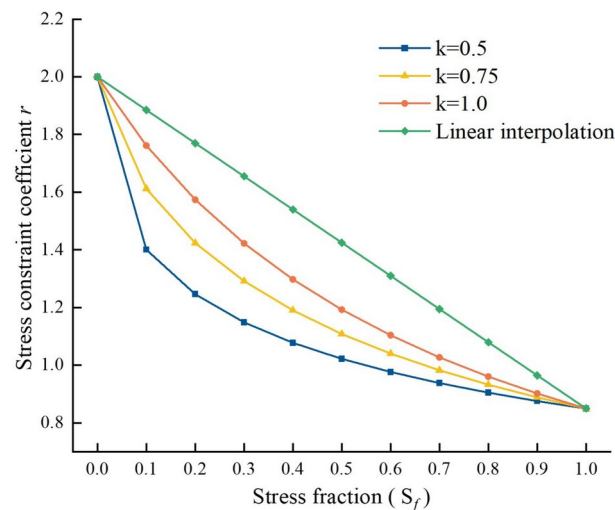


Fig. 1. The variation of stress constraint coefficient r across different exponent k .

The principle of self-attention mechanism

As depicted in Figure 2(a), the self-attention mechanism is implemented through the scaled dot-product operation. Initially, the input sequence, denoted as $\mathbf{X}$, is mapped into three vectors: the query vector $\mathbf{Q}$, key vector $\mathbf{K}$, and value vector $\mathbf{V}$. This mapping is calculated with the following equation:

$$\begin{aligned}\mathbf{Q} &= \mathbf{X}\mathbf{W}^{(Q)} \\ \mathbf{K} &= \mathbf{X}\mathbf{W}^{(K)} \\ \mathbf{V} &= \mathbf{X}\mathbf{W}^{(V)}\end{aligned}\quad (12)$$

Where $\mathbf{X} \in \mathbb{R}^{T \times d_x}$, $\mathbf{W}^{(Q)} \in \mathbb{R}^{d_x \times d_k}$, $\mathbf{W}^{(K)} \in \mathbb{R}^{d_x \times d_k}$, and $\mathbf{W}^{(V)} \in \mathbb{R}^{d_x \times d_v}$ are linear matrices. Here, T and d_x represent the length and dimensions of $\mathbf{X}$, respectively. d_k , d_k , and d_v represent the dimensions of $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$, respectively. Furthermore, $\mathbf{Q}$ and $\mathbf{K}$ are aggregated using the dot-product operation, as described in Equation (13). The resultant values are then normalized by the scale factor $\sqrt{d_k}$ and subjected to the Softmax operation. Finally, the outcome is multiplied by $\mathbf{V}$ to yield the final output vector $\mathbf{Z}$.

$$\begin{aligned}\mathbf{Z} &= \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \\ &= \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}\end{aligned}\quad (13)$$

Nevertheless, the modeling capability of single-head attention is relatively coarse due to the limitations of the feature subspace. To overcome this limitation, Vaswani et al.⁴³ proposed the multi-head self-attention (MHSA), enabling the model to simultaneously focus on information from different representation subspaces at diverse positions. Subsequently, the resulting vectors are then combined and mapped to the final output, thereby enriching the diversity of the feature subspace without incurring additional computational cost, as illustrated in Figure 2(b).

$$\begin{aligned}\mathbf{Z}_i &= \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \\ \text{MHSA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= \text{Concat}(\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_n)\mathbf{W}^O\end{aligned}\quad (14)$$

Where n is the number of MHSA heads, $\mathbf{W}^O \in \mathbb{R}^{n d_v \times d_x}$. All $\mathbf{W}$ are linear weight matrices that are learned from the data, and the output with the same dimension of input is obtained by combining the self-attention of different heads.

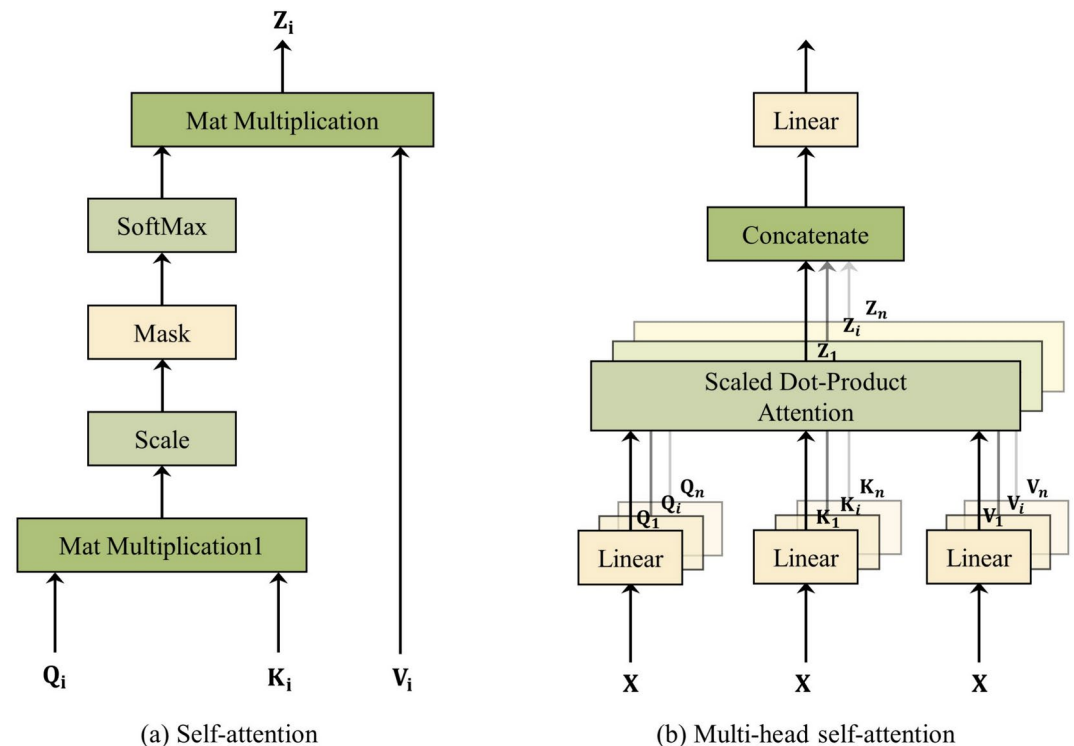


Fig. 2. The architecture of attention mechanism. (a) Scaled dot-product attention. (b) Multihead attention.

Shifted windows based self-attention mechanism

The standard Transformer architecture, along with its adaptation for image processing tasks, employs global self-attention, which computes relationships between a token and all others. However, this global computation results in a quadratic complexity relative to the number of tokens, making it unsuitable for complex tasks necessitating a vast token set for dense prediction or high-resolution image representation. For efficient modeling, this study implements self-attention computation within shifted local windows. These windows are systematically arranged and shifted to evenly partition the image in a non-overlapping manner.

In the shifted windows based self-attention block (SWA, see Figure 3), the shallow feature information from the down sampling is firstly channeled into the Windows Multi-head Self-Attention (W-MHSA) module. This module facilitates segmentation of the feature map into multiple discrete windows to compute self-attention independently within each window. Compared to the global self-attention computation in Transformer, this approach can reduce the sequence length in attention calculations, thereby significantly decreasing computational complexity. With the shifted window partitioning approach, consecutive SWA blocks can be computed as follows:

$$\begin{aligned}\hat{\mathbf{z}}^l &= W\text{-MHSA}(\text{LN}(\mathbf{z}^{l-1})) + \mathbf{z}^{l-1} \\ \mathbf{z}^l &= \text{MLP}(\text{LN}(\hat{\mathbf{z}}^l)) + \hat{\mathbf{z}}^l \\ \hat{\mathbf{z}}^{l+1} &= SW\text{-MHSA}(\text{LN}(\mathbf{z}^l)) + \mathbf{z}^l \\ \mathbf{z}^{l+1} &= \text{MLP}(\text{LN}(\hat{\mathbf{z}}^{l+1})) + \hat{\mathbf{z}}^{l+1}\end{aligned}\quad (15)$$

Where $\hat{\mathbf{z}}^l$ and $\mathbf{z}^l$ denote the output features of the (S)W-MHSA module and the MLP module for block l , respectively.

The windows shifting results in the creation of smaller sub-windows, as illustrated by regions A, B, C, D, and E in Figure 3. A common approach to calculate MHSA within sub-windows is to pad zeros around the sub-windows. However, this leads to an increase in the number of windows used for attention calculations (from 4 to 9), thereby increasing computational cost. Referring to the research of Liu et al.⁴⁴, a cyclic shift and masking mechanism is employed to maintain consistent computational costs between W-MHSA and SW-MHSA. Furthermore, a masking mechanism is employed to restrict self-attention computations within the shifting window to only adjacent sub-windows, where large negative values are applied to non-adjacent sub-windows, causing them to converge to zero after the softmax activation, while the results of adjacent regions remain unaffected.

SwinUnet for MOTO

In this research, based on the shifted windows attention mechanism, a novel extension of Unet named Shifted Windows Attention Unet (SwinUnet) is proposed, as depicted in Figure 4. SwinUnet maintains the core architecture of Unet that comprises three parts: encoding, decoding, and skip connections.

Given that the input of the network is the node information and the output is the element solution, the input tensor needs to be modified before being sent to the encoding blocks. Therefore, a convolution layer with a 2×2 kernel is first applied to decrease its spatial dimensions. Subsequently, the tensor is sent to the encoding part for feature extraction. The feature maps of different dimensions are extracted, and the output tensor of this part is compressed 8 times compared with the beginning. After that, the feature maps of the encoding part are then fed into the decoder for reconstruction. The channels from the preceding convolution and those corresponding to the feature extraction phase after SWA blocks in the skip connection are concatenated, thereby increasing the quantity of information on the decoding part. Finally, the last layer of the network utilizes the sigmoid activation function to derive the probability of each element being classified as a solid element.

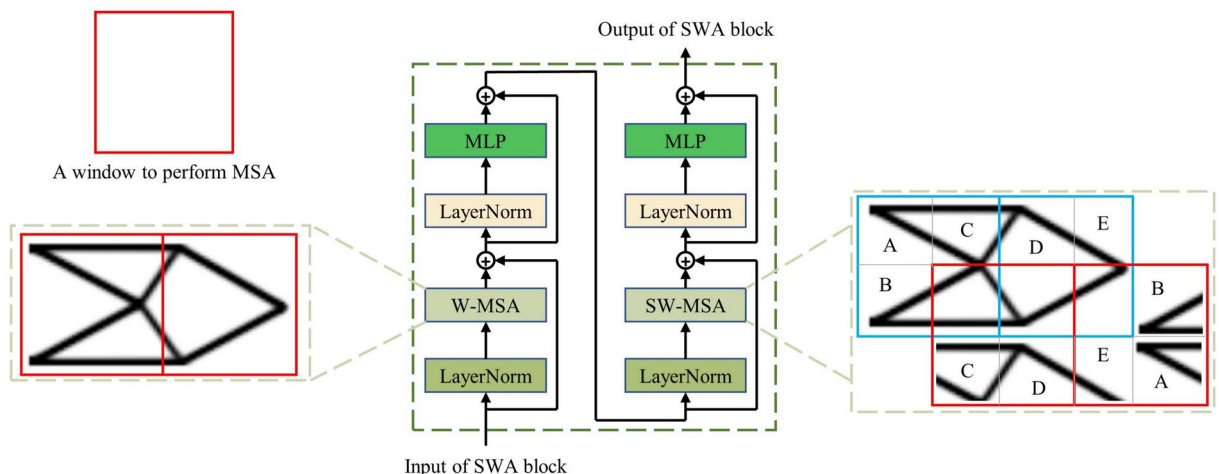


Fig. 3. The architecture of SWA.

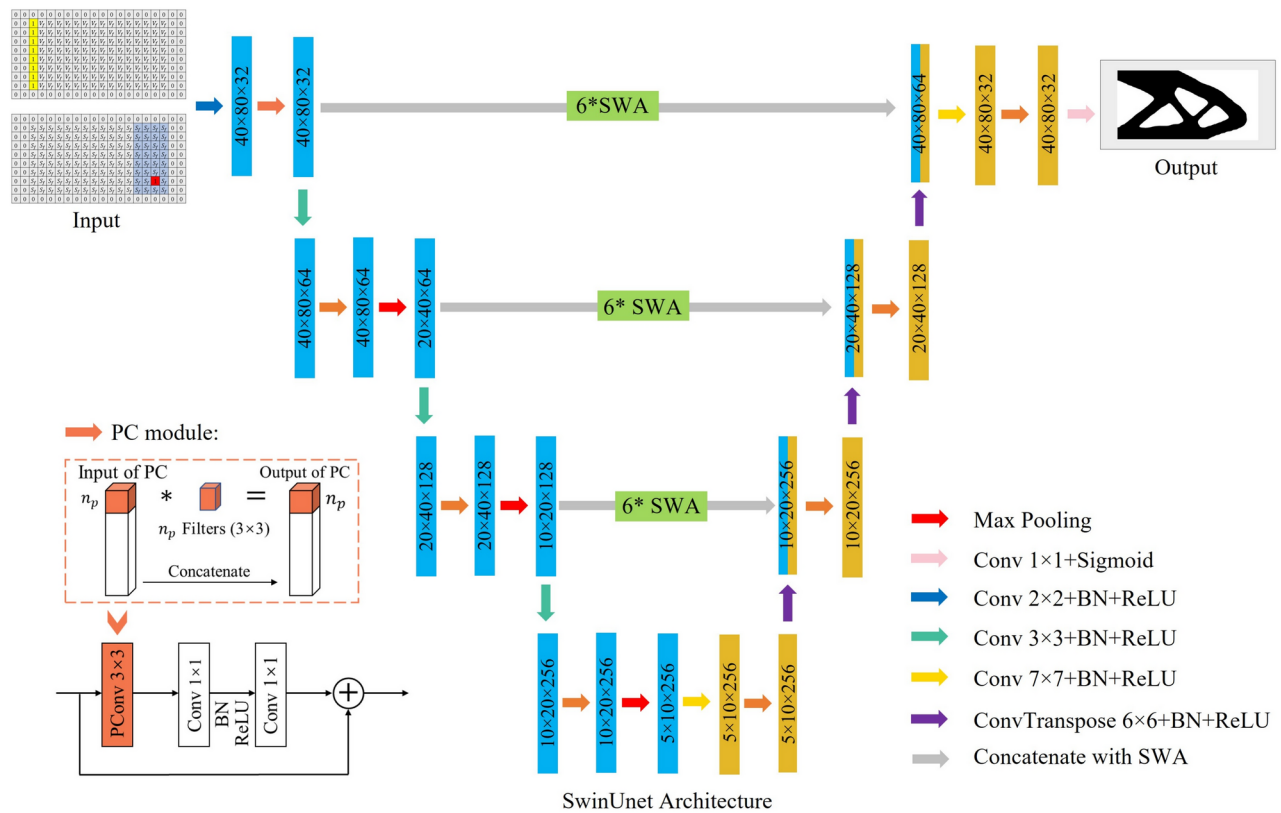


Fig. 4. The architecture of SwinUnet.

Compared to typical Unet, the key improvement of SwinUnet is the incorporation of SWA within the skip connections. This innovation enables the model to more effectively capture correlations between neighboring elements and focus attention on regions with material distribution, as demonstrated in Section 3.2. In addition to SWA, the Partial Convolution (PC) module⁴⁵ is also employed within both the encoding and decoding parts of SwinUnet to substitute the standard convolution operations typically used in conventional CNN architectures. It not only reduces the complexity of the network, but also improves the efficiency of both model training and prediction. PC is an extension of the Depthwise Convolution introduced by Howard et al.⁴⁶ It operates by performing convolution on a subset of channels (n_p) within the input tensor, as illustrated in Figure 4. By integrating Pointwise Convolution (PWConv) and residual learning, the PC module facilitates the comprehensive utilization of channel information. This approach results in a dramatic reduction of trainable parameters to 1/4 of those present in standard convolution operation. Consequently, the training parameters of the network are reduced by a notable 39.4%, from 13,671,777 in the original Unet to 8,285,089.

The proposed two-step MOTO method

The flowchart of the proposed two-stage SwinUnet framework for MOTO is presented in Figure 5, comprising three key steps. First, a small dataset is generated that incorporates variable DDs (rectangular and L-bracket domains with scalable dimensions) and three kinds of BCs. Subsequently, using the small dataset and an input tensor designed with an expansion operation to facilitate the variable DD shapes, the end-to-end 1st-SwinUnet model is trained to enable real-time predictions of near-optimal structures in Stage-1. Finally, the outputs from Stage-1 are further optimized through a physics-informed approach. In Stage-2, with a limited number of PINN-based optimization iterations, the second stage SwinUnet (2nd-SwinUnet) produces the final optimized structures. Detailed descriptions of these steps are provided in the following sections.

Dataset generation and input tensor design

Traditional end-to-end methods, once the network encounters a learning bottleneck, even a slight improvement in prediction accuracy necessitates exponential growth in training data volume. This inherent data dependency explains why conventional methods require large datasets to attain high precision. In contrast, the proposed two-stage framework introduces a physics-informed optimization phase (Stage-2) that compensates for residual errors from the initial data-driven predictions (Stage-1). By decoupling accuracy requirements across stages, Stage-1 prioritizes generating structurally feasible approximations rather than mechanically perfect solutions, thereby substantially alleviating training data demands.

The established mathematical model of MOTO is used for data generation. To ensure numerical stability, the density filtering method with a filtering radius of 1.5 is applied³¹. Benchmark DDs are selected for validation,

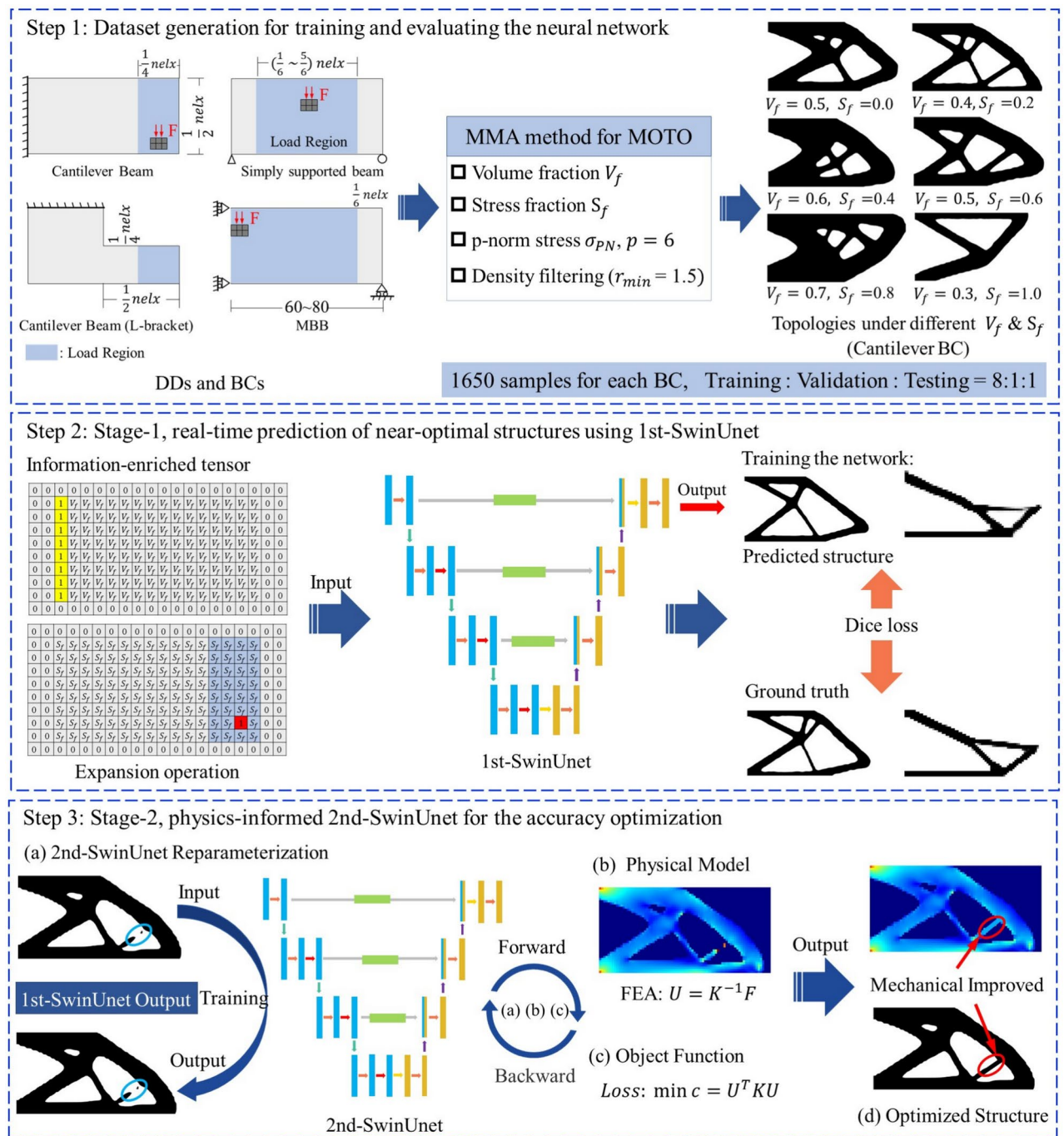


Fig. 5. The flowchart of the proposed method.

including the rectangular beam (2:1 aspect ratio) and the L-bracket with the top-right quadrant removed, both widely recognized in TO studies¹⁶. Three common BCs were considered: cantilever beams, simply supported beams, and MBB-beams⁴⁷. For the L-bracket DD, the cantilever BC is applied to constrain its top nodes. The length of the DD (referred to as $nelx$) was ranged from 60 to 80. To mitigate stress concentration, the load is evenly distributed on two adjacent nodes (i.e., 1.0 N per node), and the 3×2 elements near the loading points are defined as solid elements, and their density remains unchanged throughout the optimization process⁴⁸. Detailed configurations of DDs, BCs, and loading conditions are illustrated in the top-left panel of Figure 5. The V_f ranged from 0.3 to 0.6, and the S_f ranged from 0.0 to 1.0. Parameters of $nelx$, $nely$, V_f , S_f , as well as the loading nodes are all determined using Latin Hypercube Uniform Sampling method⁴⁹. This diverse set of initial BCs, along with the sampling method that covers various design cases, ensures the comprehensiveness and representativeness of the generated samples. Therefore, the model is able to learn a broader range of knowledge, thereby significantly improving its generalization capability and enhancing its performance across different

tasks. As V_f and S_f change, the resulting topologies exhibit distinct material distributions, as shown in the top-right of Figure 5.

For each BC, 1,650 samples were generated, and were divided into training, validation, and test sets with a ratio of 8:1:1. The maximum number of iterations for the MMA algorithm was set as 120, with an average generation time of 9.1 s per sample. On the computer used in this study, four MATLAB scripts for data generation can be run concurrently, allowing the construction of samples for each BC to be completed in approximately one hour, demonstrating high efficiency.

In theory, for TO, there is a complex mapping between the final topology forms and the initial conditions of optimization objectives and constraints. This is also the theoretical basis for using NNs to solve TO problems, as they excel in establishing such mapping relationships. Given the significance of initial conditions in determining optimal topology forms of structures, the way in which these conditions are organized is closely associated with the performance of the network.

Currently, there are two primary methods for organizing initial conditions: the direct input method, where the load action position and the constraint position are directly input into the networks^{20,50}, and the indirect input method, which involves indirectly representing the aforementioned information through the initial physical fields, such as displacement fields, strain fields, and stress fields⁵¹. In this research, a novel information-rich direct input tensor, composed of 2 channels, is proposed. The first channel encodes information about BCs and V_f , while the second channel encompasses load conditions and S_f . Each channel has a size of $(nelx + 1) \times Round((nelx + 1)/2)$, as the input tensor is derived from node information, where the number of nodes exceeds that of elements by one in each dimension. Compared to traditional direct input approaches, this information-rich configuration avoids the issue of limited information within each channel, and the reduction of channels not only enhances training efficiency but also improves prediction accuracy by minimizing irrelevant information.

Typically, NNs require input and output dimensions to remain fixed. However, to improve the generalization capability of the model in this study, the DD shapes are allowed to vary greatly, resulting in input and output with different dimensions. To accommodate this variability, tensor expansion operations are utilized, as depicted in Figure 6. The enriched input tensor is centered within a zero tensor of dimensions $81 \times 41 \times 2$, as shown in Figure 6(b). Similarly, for the output, by applying the same expansion operation to the ground-truth structure, an $80 \times 80 \times 1$ target output tensor is created, as illustrated in Figure 6(a).

Stage-1: Real-time near-optimal structure prediction with 1st-SwinUnet

Loss function

Beyond architectural innovations, the selection of loss functions plays a critical role in improving prediction accuracy and resolving structural discontinuities in TO. While the widely adopted Binary Cross-Entropy Loss ($\mathcal{L}_{BCE}$) enforces pixel-wise consistency between predictions and ground truths, it exhibits inherent limitations in suppressing intermediate densities and ensuring topological connectivity¹⁰. In contrast, the Dice Loss ($\mathcal{L}_{Dice}$) quantifies structural similarity by directly optimizing the overlap between predicted and target geometries at the element level. By minimizing the Dice coefficient, $\mathcal{L}_{Dice}$ prioritizes global shape coherence over local pixel-wise errors, making it particularly effective for tasks requiring precise geometric alignment, such as TO⁵².

Given that the goal of Stage-1 model training aligns with the properties of $\mathcal{L}_{Dice}$, this loss function is selected for use in this study. The mathematical formulation of $\mathcal{L}_{Dice}$ is presented below:

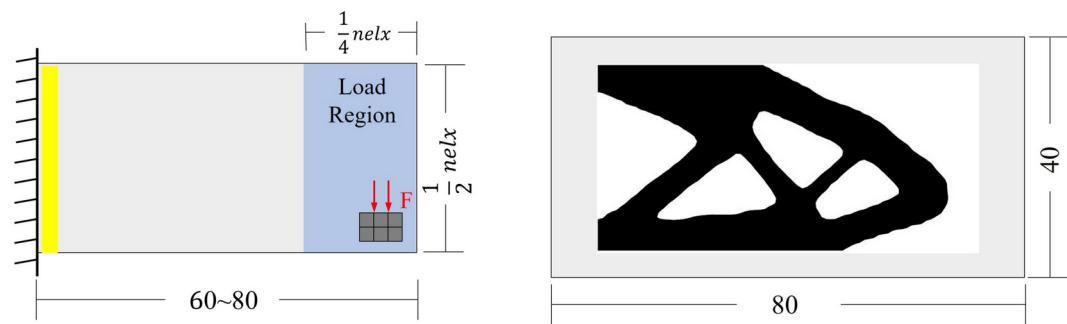
$$\mathcal{L}_{Dice} = 1 - \frac{2|y_{true} \cap y_{pred}|}{|y_{true}| + |y_{pred}|} \quad (16)$$

Where y_{true} is the ground truth, y_{pred} is the prediction result of the network, $|y|$ is the sum of the elements in y , and $\cap$ represents the intersection operation.

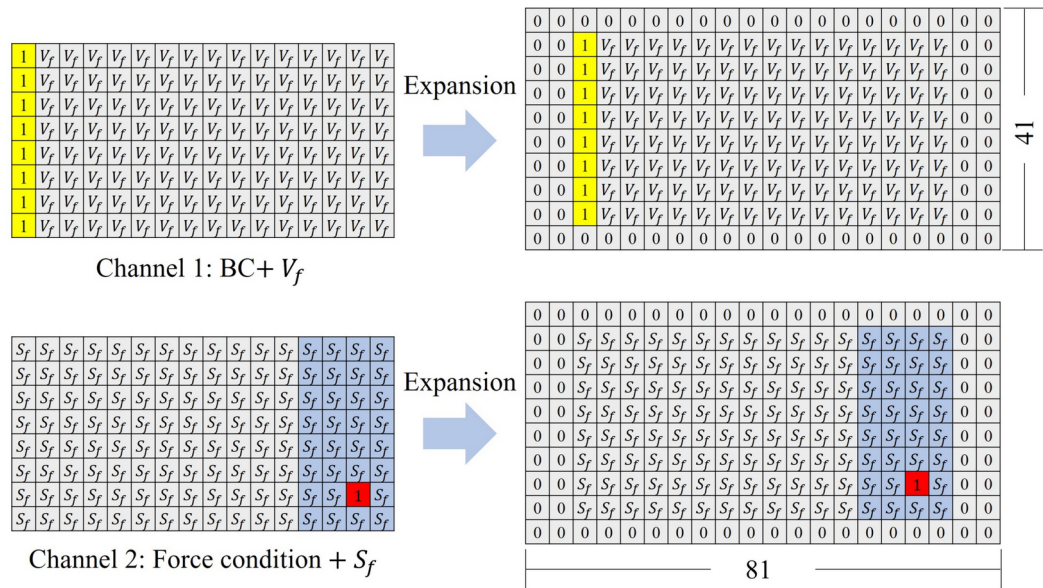
Training the 1st-SwinUnet

All experiments were performed on an Intel® Core™ i7-13700F CPU and NVIDIA GeForce RTX 4070 Ti GPU using the PyTorch framework. The Adam algorithm⁵³ was used during the training process to optimize the network parameters. The initial learning rate was 0.001, with a decay of 0.1 when the validation loss failed to decrease for 10 consecutive epochs. A batch size of 256 was used for training. The validation loss was monitored throughout the training process, and the training was early stopped if the validation loss does not decrease for 30 consecutive epochs, and the model with the lowest validation loss was then selected as the final model for further analysis. Hyperparameter analysis revealed that the model attained optimal performance when all SWA modules in the skip connections were configured with a number of 6 (see Figure 4).

Here, the 1st-SwinUnet trained with $\mathcal{L}_{Dice}$ is given as an example to show the training process. The losses during the training are depicted in Figure 7. Initially, the loss decreases rapidly, with the training loss stabilizing after approximately 70 epochs, while the validation loss continues to fluctuate. At epoch 113, the validation loss reaches its minimum value of 0.0481, with the corresponding training loss being 0.0526, which is very close to the validation loss. This indicates that the model exhibits excellent performance on both the training and validation datasets. By epoch 143, the validation loss no longer decreases, meeting the early stopping criterion. Thus, the model from epoch 113 is selected as the final model for subsequent study. Leveraging high-performance SWA modules and lightweight PC modules, the model training process achieves remarkable efficiency. The entire training process requires only 3,671 s, completing in around an hour.



(a) Initial conditions and the ground truth expansion



(b) Information-enriched direct input tensor and its expansion

Fig. 6. Information-enriched direct input tensor and tensor expansion operation.

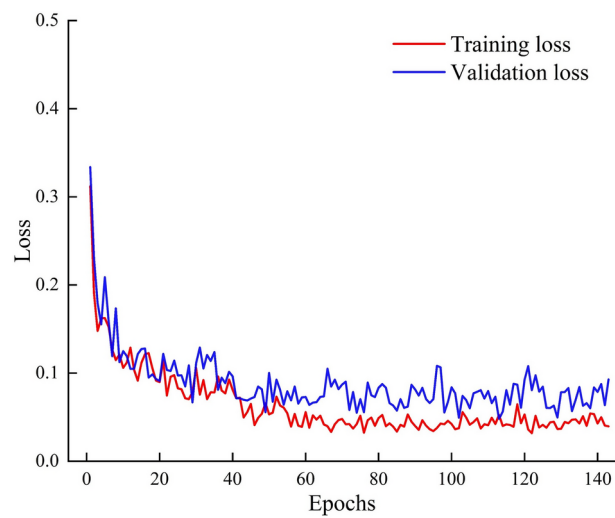


Fig. 7. Training loss and validation loss curves.

Stage-2: Physics-informed SwinUnet for the accuracy optimization

Principle of PINN-based TO

As discussed in the introduction, end-to-end approaches, inspired by advancements in DL for image recognition, often simplify structural design into an image-processing problem. However, this simplification fails to account for the fact that TO is fundamentally a design method driven by mechanical performance. Consequently, such approaches often struggle to ensure both the accuracy and mechanical reliability of the predicted optimization results. To overcome the limitations of using DL as a surrogate for MOTO, Stage-2 developed a PINN-based method for further refinement. This approach leverages the same SwinUnet to implicitly represent the structural topology, reparameterizing the design variables of MOTO into NN parameters. By iteratively updating these parameters under the guidance of the objective function, optimized structures with excellent mechanical performance can be obtained. Notably, in contrast to full-PINN TO methods, which necessitate hundreds of iterations to converge to the optimized structures^{22,24,25}, the second stage of the proposed approach significantly reduces computational effort. This efficiency is achieved because the near-optimal structures predicted in Stage-1 are already very close to the optimal solution, thereby requiring only a limited number of iterations to reach the final optimized designs.

In this research, the primary objective of MOTO is to minimize compliance, which aligns conceptually with the goal of training a NN to minimize the loss function. Leveraging this similarity, the PINN approach directly incorporates compliance as the loss function for the 2nd-SwinUnet, as shown in Equation (17). By reparameterizing structural information into NN parameters, the structure is iteratively updated and optimized during Stage-2 training, ultimately achieving a design with minimal compliance. Throughout the training process, the gradients of the compliance loss with respect to the network parameters are computed using automatic differentiation in PyTorch, ensuring efficient and accurate optimization.

$$L_{Stage2} = c(x) = \mathbf{U}^T \mathbf{K} \mathbf{U} = \sum_{i=1}^N E_i(x_i) \mathbf{u}_i^T \mathbf{k}_0 \mathbf{u}_i \quad (17)$$

Where the meaning of the symbols here are the same as in Equation (4).

Training the 2nd-SwinUnet

The flowchart for Stage-2 is shown in Figure 8. The 2nd-SwinUnet takes the near-optimal output of Stage-1 as its input. Since this input is element-based rather than the node-based input used in Stage-1, the size of the preprocessing convolution kernel is adjusted from 2×2 to 1×1 . Moreover, the network parameters in Stage-1 function solely serve as weights, whereas in Stage-2, these parameters represent the reparameterized structure, giving them with entirely different physical meanings. Due to this distinction, the network weights of 1st-SwinUnet cannot be directly applied to Stage-2, and the parameters of 2nd-SwinUnet must be reinitialized to ensure that the weights of 2nd-SwinUnet are capable of reparameterizing the near-optimal structure produced in Stage-1. To facilitate this process, prior to Stage-2 training, the output from Stage-1 is used as both the input and the target output, and it is effortless for the network to learn the mapping between identical inputs and outputs.

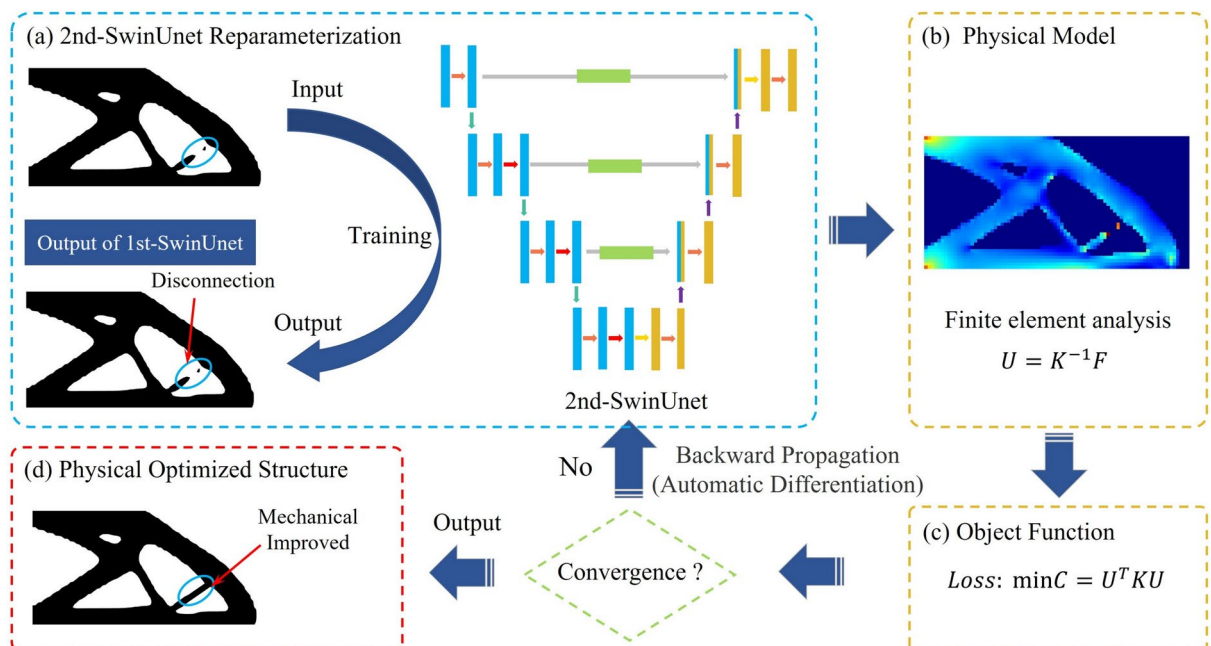


Fig. 8. Flowchart of the 2nd-SwinUnet training process.

Subsequently, the initial parameters of Stage-2 effectively become a reparameterization of the Stage-1 output, providing the foundation for efficient optimization in the subsequent iterations.

The training of the 2nd-SwinUnet exhibits a rapid convergence trend, usually completing within 20 epochs. Figure 9 illustrates the compliance (representing the model's loss) variation for a test sample during Stage-2 training. It can be found that the output of 1st-SwinUnet (labeled as the 1st structure in the figure) is already very close to both the final optimized structure and the ground truth. As training progresses, the compliance of the cantilever beam decreases significantly within the first five epochs, reaching a minimum value of 173.2. To prevent the 2nd-SwinUnet from overfitting and to improve training efficiency, the early stopping strategy in Stage-1 is also applied here. Specifically, if the training loss does not decrease for 5 consecutive epochs in Stage-2, it is deemed that convergence has been reached, and the one with the minimal compliance is the final optimal structure. Based on this early stopping criterion, the training terminates at the 10th epoch, with the structure from the 5th epoch selected as the final result. Compared to the 1st structure, the mechanical performance of the structure after Stage-2 optimization is significantly improved. For instance, as shown in Figure 9, the connection between the upper part of the web rod and the main structure is strengthened.

Performance of the proposed method

The trained SwinUnet models from both stages were employed to generate optimized structures using samples from the test dataset. Subsequently, the performance of the proposed method was evaluated by comparing the prediction results with those obtained from the conventional MMA-based method. According to the considered optimization objectives, three indicators, i.e., the compliance error (E_c), σ_{PN} error ($E_{\sigma_{PN}}$), and V_f error (E_v) were used during the evaluation. Additionally, three other global indicators were also introduced. The first one is the outperforming sample ratio (OSR), used to quantify the percentage of cases where the DL model surpasses the MMA-based method. We considered the DL outcomes superior if they exhibited $|E_v| \leq 1\%$ with $E_c, E_{\sigma_{PN}} \leq -1\%$. These designs achieve superior compliance/stress performance with minimal volume constraint violations. The second indicator is the bad sample ratio (BSR), which identifies the proportion of samples displaying significant errors. Samples with E_c or $E_{\sigma_{PN}} > 100\%$ are categorized as bad cases due to their substantial deviation from the optimization objectives. Finally, samples with E_c or $E_{\sigma_{PN}} > 20\%$ are classified as suboptimal. The suboptimal sample ratio (SSR) quantifies the proportion of samples that closely approximate the ground-truth structures but require further optimization. Together, the OSR, BSR, and SSR provide a comprehensive evaluation of the proposed method's performance across various scenarios.

Based on the evaluation indicators outlined above, the assessment of the proposed two-stage method was conducted. Specifically, this includes evaluating the performance of the novel network and the selected Dice loss in Stage-1 by comparing them with the typical Unet¹⁰. Additionally, an evaluation of the overall performance of the two-stage method was carried out, followed by an in-depth discussion of the model's generalization capacity.

Evaluation of the 1st-SwinUnet performance

The performance of the proposed novel network architecture

During the design of the network for MOTO, significant effort was invested in enhancing its performance. Key improvements include the introduction of SWA modules to enhance model accuracy and the implementation of the PC module to lighten the network. With the well-trained model, near-optimal structures can be obtained in

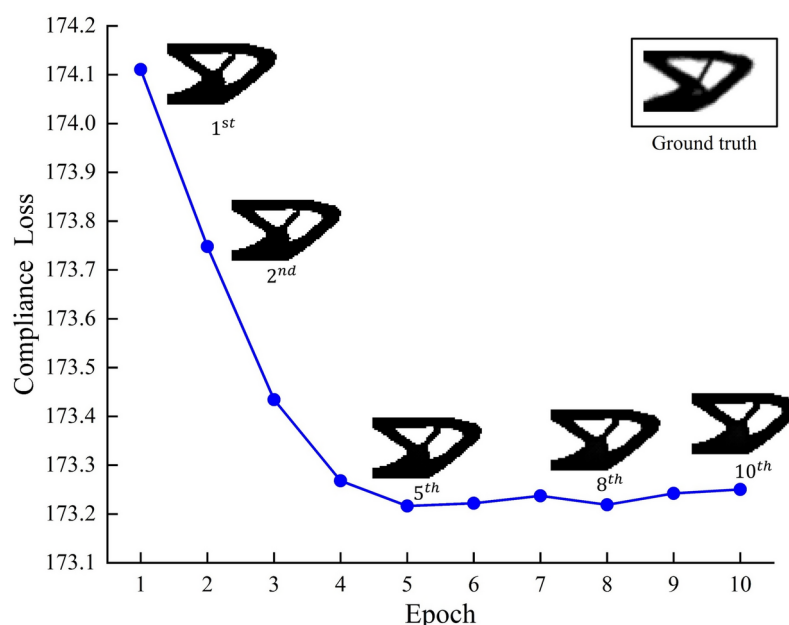


Fig. 9. Example of the 2nd-SwinUnet training process.

just 0.02 s. Additionally, the proposed method requires generating only 1,650 samples for each BC, substantially reducing offline computational costs compared to similar studies^{10,13,20}, which rely on thousands or even tens of thousands of samples.

To evaluate the performance of the novel SwinUnet, the prediction results of the 1st-SwinUnet and the typical Unet model¹⁰ are compared with those of the MMA-based method. As depicted in Table 1, negative values of E_c and $E_{\sigma_{PN}}$ are observed, indicating that even with only 1,650 samples under each BC, the proposed innovative network can predict outcomes that closely align with the ground truth. Compared to the typical Unet, SwinUnet outperforms it significantly across all indicators. Notably, it demonstrates substantial reductions in E_c , $E_{\sigma_{PN}}$, and E_v . Regarding the global indicators, the BSR value of SwinUnet is only 41.5% that of the typical Unet, and its OSR value improves by 44.2%.

Examples of the prediction results are shown in Figure 10. Compared to the ground truth obtained using the MMA-based method, the predictions from both DL models exhibit clearer boundaries, significantly reducing the presence of intermediate density elements. However, many of the Unet predictions display structural disconnections, as evident in the predicted structures (a), (b), and (e). In contrast, SwinUnet effectively eliminates most of these disconnections, demonstrating considerable improvements in model performance. Furthermore, SwinUnet achieves lower compliance and σ_{PN} values in most cases compared to both traditional methods and the Unet. These results align with the findings in Table 1, highlighting SwinUnet’s superior ability to meet the optimization objectives of compliance and stress minimization. While Unet produces lower compliance values, it exhibits higher deviations in stress distribution.

Overall, these results demonstrate the superior accuracy of the proposed SwinUnet network over typical Unet, emphasizing its effectiveness in achieving the objectives of Stage-1.

Effectiveness of the Dice loss function

Besides the architectural improvements, the model adopts the $\mathcal{L}_{Dice}$ to improve the similarity between predicted and ground-truth structures, replacing the common $\mathcal{L}_{BCE}$. To show the advantages of $\mathcal{L}_{Dice}$, it is compared with $\mathcal{L}_{BCE}$ as shown in Equation (18). Additionally, a hybrid loss function ($\mathcal{L}_{B\&D}$), defined in Equation (19), which aims to leverage the strengths of both losses while mitigating their individual limitations⁵⁴, is also tested.

$$\mathcal{L}_{BCE} = -\frac{1}{N} \sum_{i=1}^N [y_{true} \log(y_{pred}) + (1 - y_{true}) \log(1 - y_{pred})] \tag{18}$$

Where N is the number of pixels in the output, and the other symbols are defined in the same manner as in Equation (16).

$$\mathcal{L}_{B\&D} = \alpha \mathcal{L}_{Dice} + (1 - \alpha) \mathcal{L}_{BCE} \tag{19}$$

Where α is the weight coefficient of the hybrid loss function, with $\alpha = 0$ representing only $\mathcal{L}_{BCE}$ and $\alpha = 1$ representing only $\mathcal{L}_{Dice}$. Referring to relevant research⁵⁴, the weight coefficient α in the hybrid loss function was 0.9.

Three SwinUnet models with identical configurations were trained using loss functions of $\mathcal{L}_{Dice}$, $\mathcal{L}_{BCE}$, and $\mathcal{L}_{B\&D}$, respectively. The performance of these models was then evaluated to identify the optimal one for handling similarity-based end-to-end models with limited sample sizes.

The results, summarized in Table 2, clearly demonstrate that the model trained with $\mathcal{L}_{Dice}$ significantly outperforms the other two. It delivers the lowest E_c and $E_{\sigma_{PN}}$ values, along with the smallest volume constraint violations, closely aligning with the optimization objectives. The model trained with $\mathcal{L}_{B\&D}$ ranked second, while the $\mathcal{L}_{BCE}$ -based model performs the worst. Regarding the global indicators, $\mathcal{L}_{Dice}$ achieves a remarkable 25.6% reduction in BSR compared to $\mathcal{L}_{B\&D}$ and 33.3% reduction compared to $\mathcal{L}_{BCE}$. Additionally, it achieves a slightly higher OSR value than the others, further underscoring its superiority. These findings highlight the substantial performance improvements brought by $\mathcal{L}_{Dice}$ over the alternatives, and emphasize the critical importance of selecting an appropriate loss function.

Moreover, when compared with the Unet performance in Table 1, it is evident that even with different loss functions, the SwinUnet framework consistently outperforms Unet. This further underscores the exceptional capabilities of the innovative network framework proposed in this study.

Evaluation of the two-stage framework

Despite considerable efforts to enhance the performance of the 1st-SwinUnet model, it still remains an end-to-end framework. As shown in Table 1 and Figure 10, the prediction results still exhibit issues such as structural disconnections and unclear material distributions. For instance, the prediction outcome of the 1st-SwinUnet, illustrated in Figure 10(b), demonstrates notable improvements in pixel-wise similarity compared to the Unet.

Model	E_c	$E_{\sigma_{PN}}$	E_v	BSR	SSR	OSR
Typical Unet	1.03%	1.85%	0.13%	11.67%	21.97%	40.76%
1st-SwinUnet	−6.15%	−1.86%	0.01%	4.85%	8.18%	58.79%

Table 1. Performance comparison between 1st-SwinUnet and typical Unet.

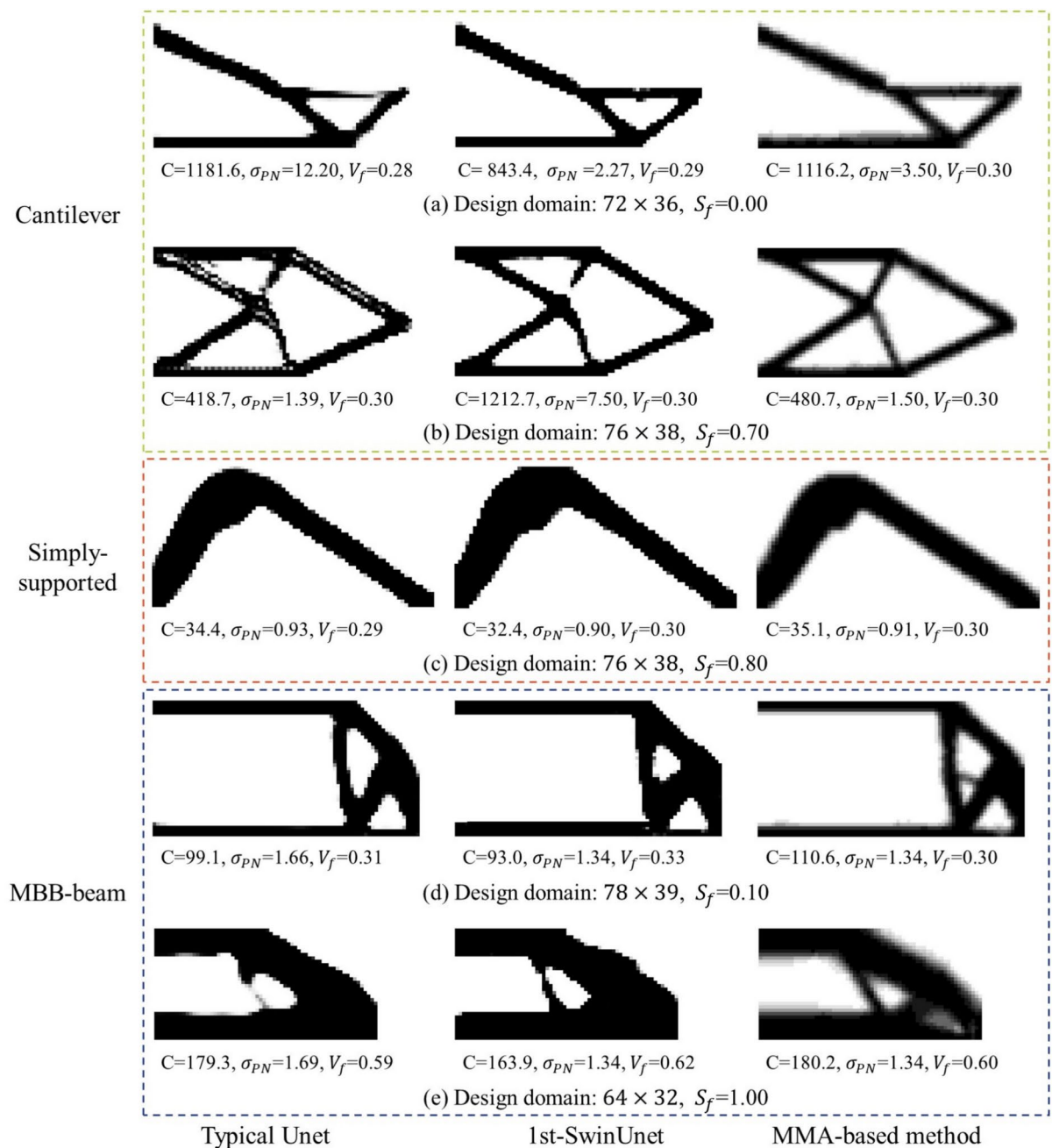


Fig. 10. Comparison between the traditional Unet and the 1st SwinUnet.

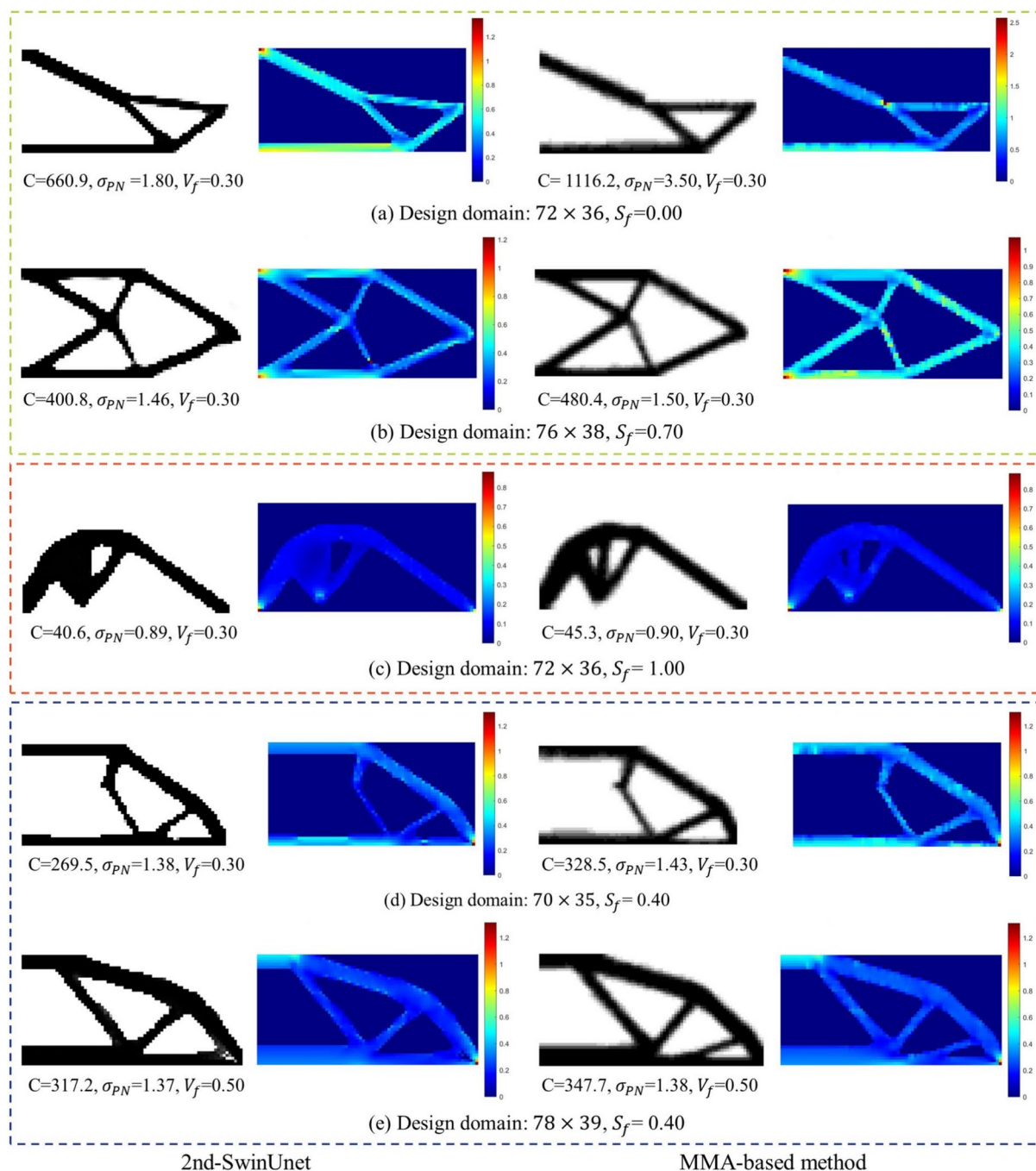
Loss function	E_c	$E_{\sigma_{PN}}$	E_v	BSR	SSR	OSR
$\mathcal{L}_{Dice}$	-6.15%	-1.86%	0.01%	4.85%	8.18%	58.79%
$\mathcal{L}_{BCE}$	-2.52%	0.63%	-0.55%	7.27%	16.06%	53.33%
$\mathcal{L}_{B\&D}$	-5.22%	-1.25%	-0.33%	6.52%	10.30%	54.54%

Table 2. Performance comparison among different loss functions.

However, structural disconnections persist in the predicted structure, underscoring the need for further optimization.

In Stage-2, following the flowchart in Figure 8, the predicted near-optimal structures were optimized using the mechanically driven approach implemented in the 2nd-SwinUnet. The corresponding results are presented in Table 3, and examples are given in Figure 11. It is observed that the Stage-2 optimization effectively addresses

	E_c	$E_{\sigma_{PN}}$	E_v	BSR	SSR	OSR	Iterations
1st-SwinUnet	-6.15%	-1.86%	0.01%	4.85%	8.18%	58.79%	/
2nd-SwinUnet	-5.13%	-1.77%	-0.23%	0	0	83.48%	20

Table 3. Performance evaluation of the 2nd-SwinUnet.**Fig. 11.** The results of the two-stage model.

the limitations of the initial predictions. Specifically, all near-optimal samples are successfully optimized, with structural disconnections resolved in the final designs. For example, the disconnection evident in Figure 10(a) and (b) is rectified in Figure 11(a) and (b). Additionally, the structural boundaries become more distinct, further improving the overall design quality. Among the testing results, an impressive 83.48% OSR is achieved. Furthermore, it is noteworthy that the indicators of E_c , $E_{\sigma_{PN}}$, E_v of the final structures are all negative,

indicating that the structures after the physical optimization of Stage-2 exhibit lower compliance and stress values compared to the ground truth, while strictly satisfying the volume constraints.

In terms of efficiency, the computational efficiency of the proposed two-stage framework is compared with the conventional gradient-based methods (e.g., MMA) and existing PINN approaches. In Stage-1, once trained, the neural network generates near-optimal designs in 0.02 s per case, which is hundreds of times faster than the MMA-based method (9.1 s/case). This acceleration is achieved without iterative FEA, as the network directly maps design parameters to optimized structures. In Stage-2, while PINN-based refinement inherently involves FEA computations, the high-quality initial predictions from Stage-1 drastically reduce the required iterations. The proposed method converges in 20 epochs on average, compared to 120 epochs for MMA (a sixfold reduction). This efficiency gain offsets the computational overhead of PINN, addressing a well-documented limitation of pure physics-driven methods^{22,24,25}, which typically demand hundreds of epochs due to their poor initialization.

Generalization ability of the proposed method

A critical limitation of many DL-based TO studies lies in their restricted applicability of the trained models to specific DDs and BCs. Given the substantial time investment required for dataset generation and model training, it is impractical to train a new model for every variation in DDs or BCs. This study therefore systematically investigates the generalization capabilities of the proposed method.

Compared with conventional approaches, the proposed framework exhibits superior adaptability to geometric variations and BC diversity. While traditional methods typically operate within fixed rectangular domains, our framework successfully processes both scalable rectangular configurations (with n_{elx} spanning 60–80) and geometrically complex non-convex L-bracket domains without requiring retraining. The model's robustness is further validated through comprehensive testing across three fundamental BC categories: cantilever beams, simply supported beams, and MBB-beams, achieving consistent performance with only 1,650 training samples per BC. Results can be found in Figure 11. The two-stage methodology consistently generates optimized topologies despite significant geometric and BC variations. This generalization capability originates from the physics-informed training mechanism in Stage-2, which incorporates fundamental physical principles (e.g., equilibrium constraints, stress regularization) to ensure feasible solution. This physics-based refinement complements the data-driven patterns learned in Stage-1, enabling effective generalization with limited training data.

To rigorously evaluate generalization boundaries, we extend validation to three challenging scenarios beyond the original training specifications. (1) A scaled-down rectangular domain (50×25) under standard BCs confirming size invariance (Figure 12 I(a)); (2) A new hybrid BC combining cantilever and simply supported constraints to evaluate performance on unseen BC (Figure 12 I(b)); (3) Concurrent modifications of both domain dimensions (50×25) and BC configurations (Figure 12 I(c)). Comparative results in Figure 12 II–III demonstrate that the framework produces solutions comparable to MMA-based optimization benchmarks. As defined by Woldseth et al.²¹, our framework demonstrates task-shift generalization, meaning it performs reliably on tasks that differ moderately from the training data.

However, performance degradation occurs when handling BC/DD combinations beyond the training distribution, as exemplified in Figure 12 III(b), where both topological and performance discrepancies emerge compared with conventional methods. This limitation stems from the hybrid data-physics architecture's dependence on first-stage network initialization. When novel parameters exceed the training distribution, suboptimal initial solutions constrain the subsequent physics-based optimization search space. Future developments will implement transfer learning strategies⁵⁵, where limited additional training for new design conditions could refine the first-stage network, thereby extending the method's applicability while preserving computational efficiency.

Conclusion

This paper proposes a novel two-stage MOTO method based on DL, addressing critical challenges in computational efficiency, generalization, and mechanical feasibility. A MOTO mathematical model that incorporates important design objectives of stiffness, strength, and material usage was established, and a unified SwinUnet framework that executes both stages, employing SWA and lightweight modules, was constructed to support this approach. Stage-1 employs data-driven learning to generate near-optimal designs in 0.02 s, using only 1,650 samples per condition. Stage-2 refines these designs via the PINN framework, achieving convergence in 20 epochs while yielding structures with 5.13% lower compliance and 1.86% reduced peak stress compared to ground truths, and 83.48% of them perform better than the ground truth obtained by the MMA-based method, all under strict volume constraints. The proposed method demonstrates substantial improvements in generalization capability across DD sizes and BCs. These improvements also highlight the practicality and competitiveness of the proposed approach. The proposed method effectively addresses the challenges associated with the reliance on large datasets and the lack of mechanical considerations in end-to-end approaches. Besides, it achieves a significant reduction in computational cost compared to PINN-based methods. Furthermore, the method also demonstrates substantial improvements in generalization capability across multi-scale DDs and non-convex geometries, as well as various, even untrained BC. These improvements also highlight the practicality and competitiveness of the proposed approach.

Although substantial progress has been made, the proposed method still has some limitations that warrant further exploration. On the one hand, compared to existing one-stage TO studies using a single network, the proposed two-stage method entails an increase in complexity during execution. However, by employing nearly identical networks across both stages, the complexity associated with transitioning between different models has been effectively minimized^{12,25,56}. On the other hand, given the complexity of the MOTO problem, this study primarily focuses on relatively simple 2D structures, whereas real-world applications often involve intricate 3D

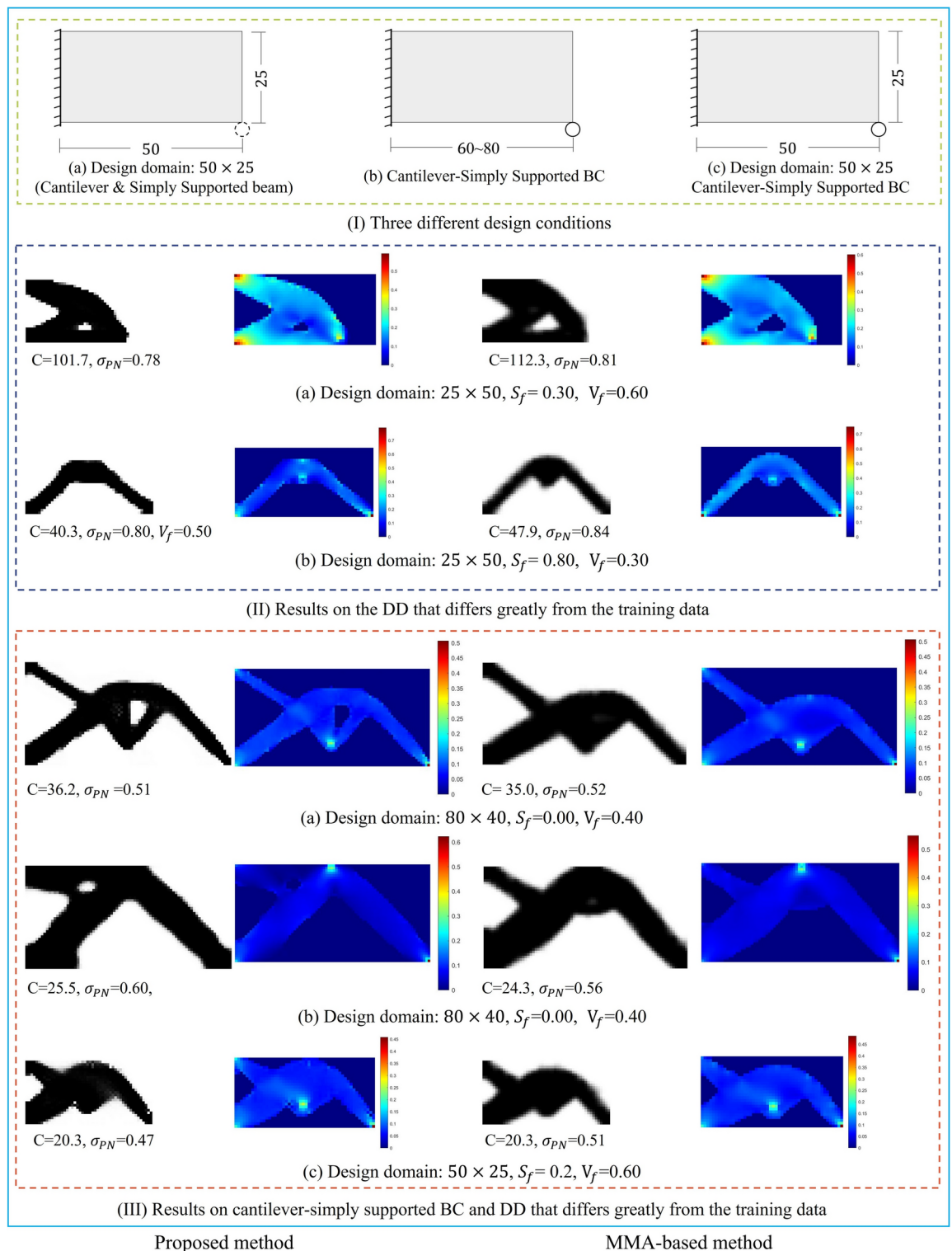


Fig. 12. Validation the generalization ability of the proposed framework.

designs. Extending the current method to 3D poses several challenges. In a data-driven framework, preparing datasets for 3D cases requires detailed characterization of highly complex and diverse structures, significantly increasing the number of design variables. This necessitates larger, more diverse datasets to ensure robust model training, which inherently amplifies computational expenses. Besides, PINN-based optimization in Stage-2 involves a limited number of FEA analyses. For 3D scenarios, this could severely impact efficiency. Additionally, both the mathematical model and the NN architecture must be reformulated to accommodate the demands of 3D optimization while maintaining high efficiency. Despite the aforementioned challenges, extending the

method to 3D scenarios offers promising opportunities, such as enabling more realistic and practical structural designs and facilitating manufacturing through 3D printing.

Addressing these challenges and extending the proposed method to complex 3D problems represents a crucial direction for our future work. Subsequent efforts will focus on developing a robust 3D MOTO mathematical model, efficiently generating high-resolution 3D datasets using techniques such as GPU acceleration and parallel computing, and designing advanced NN architectures tailored to 3D scenarios. Furthermore, enhancing the model's generalization capability to adapt to a broader range of 3D DDs and BCs will also be a priority.

Data availability

Relevant codes and dataset used in this paper have been uploaded to GitHub, which can be found at <https://github.com/FZU-chengxiang/SwinUnet-for-MOTO>.

Received: 5 September 2024; Accepted: 3 March 2025

Published online: 18 March 2025

References

- Wang, Z.Jr. J. H. S. A., Ashok, A., Wang, Z. & Castro, S. Lightweight design of variable-angle filament-wound cylinders combining kriging-based metamodels with particle swarm optimization. *Structural and Multidisciplinary Optimization* **65**, 140 (2022).
- Bendsoe, M. P. Optimal shape design as a material distribution problem. *Structural Optimization* **1**, 193–202 (1989).
- Zhou, M. & Rozvany, G. The coc algorithm, part ii: Topological, geometrical and generalized shape optimization. *Computer Methods in Applied Mechanics and Engineering* **89**, 309–336. [https://doi.org/10.1016/0045-7825\(91\)90046-9](https://doi.org/10.1016/0045-7825(91)90046-9) (1991).
- Xie, Y. M. & Steven, G. P. A simple evolutionary procedure for structural optimization. *Computers & Structures* **49**, 885–896 (1993).
- Wang, M. Y., Wang, X. & Guo, D. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering* **192**, 227–246 (2003).
- Du, Z., Zhou, X.-Y., Picelli, R. & Kim, H. A. Connecting microstructures for multiscale topology optimization with connectivity index constraints. *Journal of Mechanical Design* **140**, 111417 (2018).
- Guo, X., Zhang, W. & Zhong, W. Doing topology optimization explicitly and geometrically—a new moving morphable components based framework. *Journal of Applied Mechanics* **81**, 81009 (2014).
- Sanfui, S. & Sharma, D. Gpu-based mesh reduction strategy utilizing active nodes for structural topology optimization. *Structures* **55**, 570–586. <https://doi.org/10.1016/j.istruc.2023.05.079> (2023).
- Sosnovik, I. & Oseledets, I. Neural networks for topology optimization. *Russian Journal of Numerical Analysis and Mathematical Modelling* **34**, 215–223. <https://doi.org/10.1515/rnam-2019-0018> (2019).
- Wang, D. et al. A deep convolutional neural network for topology optimization with perceptible generalization ability. *Engineering Optimization* **54**, 973–988 (2022).
- Xiang, C. et al. Accelerated topology optimization design of 3d structures based on deep learning. *Structural and Multidisciplinary Optimization* **65**, 1–18 (2022).
- Rocheffort-Beaudoin, T., Vadean, A., Gamache, J.-F. & Achiche, S. Supervised deep learning for the moving morphable components topology optimization framework. *Engineering Applications of Artificial Intelligence* **123**, <https://doi.org/10.1016/j.engappai.2023.106436> (2023).
- Yan, J., Geng, D., Xu, Q. & Li, H. Real-time topology optimization based on convolutional neural network by using retrain skill. *Engineering with Computers* <https://doi.org/10.1007/s00366-023-01846-3> (2023).
- Li, B., Huang, C., Li, X., Zheng, S. & Hong, J. Non-iterative structural topology optimization using deep learning. *Computer Aided Design* **115**, 172–180. <https://doi.org/10.1016/j.cad.2019.05.038> (2019).
- Oh, S., Jung, Y., Kim, S., Lee, I. & Kang, N. Deep Generative Design: Integration of Topology Optimization and Generative Models. *Journal of Mechanical Design* **141**, 111405. <https://doi.org/10.1115/1.4044229> (2019).
- Xiang, C., Chen, A. & Wang, D. Real-time stress-based topology optimization via deep learning. *Thin-Walled Structures* **181**, <https://doi.org/10.1016/j.tws.2022.110055> (2022).
- Deng, C., Wang, Y., Qin, C., Fu, Y. & Lu, W. Self-directed online machine learning for topology optimization. *Nature Communications* **13**, <https://doi.org/10.1038/s41467-021-27713-7> (2022).
- Habashneh, M., Cucuzza, R., Aela, P. & Rad, M. M. Reliability-based topology optimization of imperfect structures considering uncertainty of load position. *Structures* **69**, <https://doi.org/10.1016/j.istruc.2024.107533> (2024).
- Li, Z., Wang, L. & Gu, K. Efficient reliability-based concurrent topology optimization method under pid-driven sequential decoupling framework. *Thin-Walled Structures* **203**, <https://doi.org/10.1016/j.tws.2024.112117> (2024).
- Yu, Y., Hur, T., Jung, J. & Jang, I. G. Deep learning for determining a near-optimal topological design without any iteration. *Structural and Multidisciplinary Optimization* **59**, 787–799 (2018).
- Woldseth, R. V., Aage, N., Baerentzen, J. A. & Sigmund, O. On the use of artificial neural networks in topology optimisation. *Structural and Multidisciplinary Optimization* **65**, <https://doi.org/10.1007/s00158-022-03347-1> (2022).
- Zhang, Z. et al. Tonr: An exploration for a novel way combining neural network with topology optimization. *Computer Methods in Applied Mechanics and Engineering* **386**, <https://doi.org/10.1016/j.cma.2021.114083> (2021).
- Raissi, M., Yazdani, A. & Karniadakis, G. E. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science* **367**, 1026+. <https://doi.org/10.1126/science.aaw4741> (2020).
- Jeong, H. et al. A physics-informed neural network-based topology optimization (pinnto) framework for structural optimization. *Engineering Structures* **278**, <https://doi.org/10.1016/j.engstruct.2022.115484> (2023).
- Jeong, H. et al. A complete physics-informed neural network-based framework for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering* **417**, <https://doi.org/10.1016/j.cma.2023.116401> (2023).
- Arasteh, S. T. et al. Large language models streamline automated machine learning for clinical studies. *Nature Communications* **15**, <https://doi.org/10.1038/s41467-024-45879-8> (2024).
- Zheng, S., Fan, H., Zhang, Z., Tian, Z. & Jia, K. Accurate and real-time structural topology prediction driven by deep learning under moving morphable component-based framework. *Applied Mathematical Modelling* **97**, <https://doi.org/10.1016/j.apm.2021.04.009> (2021).
- Seo, M. & Min, S. Graph neural networks and implicit neural representation for near-optimal topology prediction over irregular design domains. *Engineering Applications of Artificial Intelligence* **123**, <https://doi.org/10.1016/j.engappai.2023.106284> (2023).
- Mazé, F. & Ahmed, F. Diffusion models beat gans on topology optimization. *Proceedings of the AAAI Conference on Artificial Intelligence* **37**, 9108–9116. <https://doi.org/10.1609/aaai.v37i8.26093> (2023).
- Svanberg, K. The method of moving asymptotes - a new method for structural optimization. *International Journal for Numerical Methods in Engineering* **24**, 359–373. <https://doi.org/10.1002/nme.1620240207> (1987).
- Andreassen, E., Clausen, A., Schevenels, M., Lazarov, B. S. & Sigmund, O. Efficient topology optimization in matlab using 88 lines of code. *Structural and Multidisciplinary Optimization* **43**, 1–16. <https://doi.org/10.1007/s00158-010-0594-7> (2011).

32. Fan, W., Xu, Z., Wu, B., He, Y. & Zhang, Z. Structural multi-objective topology optimization and application based on the criteria importance through intercriteria correlation method. *Engineering Optimization* **54**, 830–846. <https://doi.org/10.1080/0305215X.2021.1901087> (2022).
33. Mitchell, S. L. & Ortiz, M. Computational multiobjective topology optimization of silicon anode structures for lithium-ion batteries. *Journal of Power Sources* **326**, 242–251. <https://doi.org/10.1016/j.jpowsour.2016.06.136> (2016).
34. Xin, Q. C., Yang, L. & Huang, Y. N. Digital design and manufacturing of spherical joint base on multi-objective topology optimization and 3d printing. *Structures* **49**, 479–491. <https://doi.org/10.1016/j.istruc.2023.01.101> (2023).
35. Le, C., Norato, J., Bruns, T., Ha, C. & Tortorelli, D. Stress-based topology optimization for continua. *Structural and Multidisciplinary Optimization* **41**, 605–620. <https://doi.org/10.1007/s00158-009-0440-y> (2010).
36. Bruggi, M. & Duysinx, P. Topology optimization for minimum weight with compliance and stress constraints. *Structural and Multidisciplinary Optimization* **46**, <https://doi.org/10.1007/s00158-012-0759-7> (2012).
37. Xia, L., Zhang, L., Xia, Q. & Shi, T. Stress-based topology optimization using bi-directional evolutionary structural optimization method. *Computer Methods in Applied Mechanics and Engineering* **333**, <https://doi.org/10.1016/j.cma.2018.01.035> (2018).
38. Deng, H., Vulimiri, P. & To, A. An efficient 146-line 3d sensitivity analysis code of stress-based topology optimization written in matlab. *Optimization and Engineering* <https://doi.org/10.1007/s11081-021-09675-3> (2021).
39. Nabaki, K., Shen, J. & Huang, X. Stress minimization of structures based on bidirectional evolutionary procedure. *Journal of Structural Engineering* **145**, [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0002264](https://doi.org/10.1061/(ASCE)ST.1943-541X.0002264) (2019).
40. Islam, M. M., Islam, M. S. & Saiduzzaman, M. An advanced approach of regula-falsi method exploiting the bisection method. *Journal of Interdisciplinary Mathematics* **25**, 1999–2006. <https://doi.org/10.1080/09720502.2022.2133227> (2022).
41. Min, B. *et al.* Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys* **56**, <https://doi.org/10.1145/3605943> (2024).
42. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. *18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 234–241, https://doi.org/10.1007/978-3-319-24574-4_28 (2015).
43. Vaswani, A. *et al.* Attention is all you need. *Advances in neural information processing systems* **30** (2017).
44. Liu, Z. *et al.* Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021).
45. Chen, J. *et al.* Run, don't walk: Chasing higher flops for faster neural networks. arXiv:2303.03667 (2023).
46. Howard, A. G. *et al.* Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017).
47. Sigmund, O. On benchmarking and good scientific practise in topology optimization. *Structural and Multidisciplinary Optimization* **65**, <https://doi.org/10.1007/s00158-022-03427-2> (2022).
48. Holmberg, E., Torstenfelt, B. & Klarbring, A. Stress constrained topology optimization. *Structural and Multidisciplinary Optimization* **48**, 33–47. <https://doi.org/10.1007/s00158-012-0880-7> (2013).
49. Steinberg, D. M. & Lin, D. K. J. A construction method for orthogonal latin hypercube designs. *Biometrika* **93**, 279–288. <https://doi.org/10.1093/biomet/93.2.279> (2006).
50. Zheng, S., He, Z. & Liu, H. Generating three-dimensional structural topologies via a u-net convolutional neural network. *Thin-Walled Structures* **159**, 107263 (2021).
51. Nie, Z., Lin, T., Jiang, H. & Kara, L. B. Topologygan: Topology optimization using generative adversarial networks based on physical fields over the initial domain. *Journal of Mechanical Design* **143**, <https://doi.org/10.1115/1.4049533> (2021).
52. Islam, M. M. & Liu, L. Deep learning accelerated topology optimization with inherent control of image quality. *Structural and Multidisciplinary Optimization* **65**, <https://doi.org/10.1007/s00158-022-03433-4> (2022).
53. Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. *ArXiv Preprint ArXiv:1412.6980* (2014).
54. Wang, D., Ning, Y., Xiang, C. & Chen, A. A two-stage network framework for topology optimization incorporating deep learning and physical information. *Engineering Applications of Artificial Intelligence* **133**, <https://doi.org/10.1016/j.engappai.2024.108185> (2024).
55. Xu, L., Zhang, W., Yao, W., Youn, S.-K. & Guo, X. Machine learning accelerated mmc-based topology optimization for sound quality enhancement of serialized acoustic structures. *Structural and Multidisciplinary Optimization* **67**, <https://doi.org/10.1007/s00158-024-03800-3> (2024).
56. Ates, G. C. & Gorguluarslan, R. M. Two-stage convolutional encoder-decoder network to improve the performance and reliability of deep learning models for topology optimization. *Structural and Multidisciplinary Optimization* **63**, 1927–1950 (2021).

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 52078365), Shanghai Pujiang Program [grant number 22PJD079], Shanghai Pujiang Program [grant number 21PJD077] and China Postdoctoral Science Foundation [grant number 2022M722426].

Author contributions

C.X. conducted the methodology and wrote the main manuscript text. A.C. conceptualized the study and acquired funding. H.L. provided data resources and contributed to network development. D.W. assisted with network development and data curation. B.G. was responsible for visualization and data curation. H.C. supervised the study, reviewed and edited the manuscript, and acquired funding. All authors reviewed the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025