ASME Journal of Mechanical Design
Online journal at:
https://asmedigitalcollection.asme.org/mechanicaldesign

Neural Network Surrogate Modeling for Stochastic Finite Element Method Using Three-Dimensional Graph Representations: A Comparative Study

Jessica Ezemba[1]
Department of Mechanical Engineering,
Carnegie Mellon University,
5000 Forbes Avenue,
Pittsburgh, PA 15213
e-mail: jezemba@andrew.cmu.edu

Christopher McComb
Department of Mechanical Engineering,
Carnegie Mellon University,
5000 Forbes Avenue,
Pittsburgh, PA 15213
e-mail: ccm@andrew.cmu.edu

Conrad Tucker
Department of Mechanical Engineering,
Carnegie Mellon University,
5000 Forbes Avenue,
Pittsburgh, PA 15213
e-mail: conradt@andrew.cmu.edu

Modern engineering design increasingly relies on probabilistic simulation to account for uncertainties in geometry and loading conditions. The stochastic finite element method (SFEM) has become standard as a way to address this need, using thousands of deterministic FEM evaluations to estimate the uncertainty. However, this creates a prohibitively high computational cost that can inhibit efficient design exploration. Neural network (NN) surrogate models offer a promising alternative, shifting computation to a one-time upfront training cost while enabling near-instantaneous subsequent evaluations for iterative design tasks. However, effective NN surrogates for SFEM must learn to directly predict distributions that traditionally emerge from iterative sampling and aggregation of system responses across varying parameter spaces. Although previous research has explored various NN architectures for physical simulations, their effectiveness specifically for SFEM problems that combine geometric complexity with stochastic loading conditions, particularly in predicting converged distribution of physical fields that require understanding relationships between global and local features, remains inadequately addressed. This work addresses this gap by systematically evaluating 11 NN architectures organized into three distinct learning mechanism categories: attention-based, message passing, and hierarchical approaches. Our evaluation using 3D geometries with stochastic point elastic loading reveals that while these surrogate models achieve inference speeds orders of magnitude faster than traditional SFEM (milliseconds versus hours for traditional SFEM), their accuracy remains below the level required for full replacement of SFEM in iterative design applications. Our findings identify specific architectural trade-offs, highlighting avenues for hybrid approaches that may better balance computational efficiency with predictive accuracy. [DOI: 10.1115/1.4069278]

Keywords: stochastic finite element method, graph neural network, surrogate model, engineering design

1 Introduction

One approach used to account for uncertainties in modern engineering design is the stochastic finite element method (SFEM), which extends traditional FEM to consider stochastic variations in boundary conditions, materials, or geometries [1,2].

FEM simulation models can be computationally expensive [3]. SFEM requires thousands of deterministic FEM evaluations, creating prohibitively high computational costs that can inhibit efficient design exploration. Consequently, relying on simulation models alone can be time-consuming, especially when there is a requirement to explore a vast number of environment loads or geometries as is the case with SFEM [3]. These repeated simulations can be replaced by data-driven surrogate models that eliminate the need to evaluate the true data-generating process repeatedly [4]. Neural network (NN) surrogate models offer a promising alternative, shifting computation to a one-time upfront training cost while enabling near-instantaneous subsequent evaluations for iterative design

[1]Corresponding author.

Contributed by the Design Automation Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received March 4, 2025; final manuscript received July 24, 2025; published online September 23, 2025. Assoc. Editor: Ikjin Lee.

tasks. The development of NN surrogate models can be framed as a supervised learning problem, with the objective of identifying and representing the governing relationship between system inputs and outputs.

However, the development of NN surrogate models for SFEM requires optimizing for fundamentally different objectives compared to traditional NN surrogate models [4]. While traditional surrogate models learn direct input–output mappings for single simulations, SFEM surrogates must learn the mapping between input parameters and the converged statistical distributions that emerge from multiple FEM evaluations with varying probabilistic loads and boundary conditions. The key challenge lies in training models to directly predict stress distributions that traditionally only emerge after aggregating a large number of simulations—requiring surrogates to internalize patterns from repeated sampling of geometries and loading conditions. Despite this complexity, the challenge remains tractable for two key reasons. First, the underlying physics governing both deterministic FEM and SFEM are consistent; it is only the parameter sampling that introduces additional complexity, not the fundamental physical relationships. Second, while SFEM involves a random sampling process, it ultimately converges to consistent statistical values determined by the geometry and the probabilistic loading environment. This convergence means that for any given geometry and probabilistic loading description, a deterministic mapping exists between the input parameters and the converged statistical quantities, making it suitable for surrogate modeling. Consequently, with appropriate architectural choices, NN surrogate models could potentially transform the iterative process of SFEM into a single predictive step.

Recent advances in the field of geometric deep learning (GDL) [5] have produced two promising directions for NN surrogate modeling of SFEM. First, neural operators represent a novel class of methods designed to learn mappings between infinite-dimensional function spaces, making it possible to directly learn partial differential equation (PDE) solution operators instead of just point-wise relationships [6]. These approaches aim to capture the underlying mathematical structure of PDEs, making them particularly well-suited for problems where solutions need to be learned across entire spatial domains. Second, graph neural networks (GNNs) offer a different advantage through their ability to directly operate on mesh structures, providing natural handling of irregular geometries and spatial relationships [7]. These approaches have led to various learning mechanisms for handling complex geometric data, including message passing architectures that iteratively update features through local neighborhood interactions, attention-based methods that dynamically weight relationships based on their relevance, and hierarchical approaches that process information at multiple scales [8,9]. Each of these mechanisms offers potential benefits for SFEM surrogate modeling—message passing for capturing local physical relationships, attention for handling long-range dependencies, and hierarchical processing for managing multiple scales of geometric features.

However, both approaches face distinct challenges in capturing the full complexity of SFEM. Neural operators often struggle with irregular 3D geometries and complex boundary conditions [10], and GNNs face difficulties with long-range interactions, as solutions at any point depend on points far from the local neighborhood [11]. This is particularly crucial for predicting high spatial frequency features in stress fields, where local geometric features must be understood in the context of global structural behavior. The diverse challenges faced by these approaches highlight a critical need: understanding how different learning mechanisms perform when simultaneously confronted with irregular 3D geometries, stochastic boundary conditions, and high spatial frequency features in SFEM applications.

The current work responds to this need by performing a comprehensive evaluation of 11 different neural network architectures, which span three fundamentally different learning mechanisms. In doing so, this work makes several key contributions. First, we contribute a benchmark dataset to assess neural network performance in the prediction of structural stress fields under combined geometric and stochastic loading variations available.[2] Second, we provide a systematic evaluation of how different learning mechanisms (attention-based, message-passing-based, and hierarchical processing) handle both geometric variations and stochastic loading conditions simultaneously. Third, our comparative analysis reveals performance trade-offs between computational efficiency and predictive accuracy across varying 3D geometries, identifying which architectural choices are most effective for practical applications. These contributions provide a foundation for developing more efficient and robust NN surrogate models for SFEM.

2 Related Works

Table 1 shows a comparative analysis of existing methods for mesh-based physical simulation surrogate modeling that aim to replace computationally expensive physics-based simulations with neural network approximations. Specifically, this analysis reveals several key challenges in using these NN surrogate models for SFEM applications: handling random, unstructured 3D meshes, supporting stochastic boundary conditions and loading, managing nonparametric geometries, and capturing high spatial frequency features. While various approaches have addressed some aspects of these challenges, no method addresses these requirements simultaneously. This combination of features is particularly crucial for SFEM applications where both geometric complexity and loading uncertainty must be considered simultaneously.

2.1 Finite Element Method.

Traditional FEM relies on deterministic models, often employing idealized geometries and average material properties [16]. This approach, however, fails to capture the inherent uncertainties present in real-world systems [2]. These uncertainties can stem from various sources, including:

- Material variability: Fluctuations in material properties (e.g., Young's modulus and Poisson's ratio) due to manufacturing processes or inherent material randomness.
- Geometric uncertainties: Variations in dimensions arising from manufacturing tolerances or imperfections.
- Loading uncertainties: Fluctuations in applied loads, such as point loads, wind, or seismic loads.

Traditionally, safety factors have been employed as a deterministic approach to mitigate these effects of uncertainty. While factors of safety are valuable tools for assessing product reliability, they do not directly quantify or predict the influence and sources of randomness in a system [16]. Moreover, safety factors offer only a subjective and indirect method of addressing uncertainty, providing a bulk overestimation that fails to account for geometric variability [17].

2.1.1 *Stochastic Finite Element Method.*

To address these limitations, SFEM has emerged as an extension of traditional FEM. SFEM consists of running multiple deterministic FEM models and aggregating the results. By incorporating random variables into the model parameters, SFEM allows for a probabilistic assessment of system behavior. This extension enables the analysis of systems with inherent uncertainty, making SFEM a critical tool for studying complex phenomena where deterministic methods fall short [16].

Multiple variants of SFEM have been developed, with three of these being the most used and accepted: Monte Carlo simulation (MCS), the perturbation method, and the spectral stochastic finite element method [18]. Each method uses a different approach to represent, solve, and study the randomness of a system. MCS is the most common method used to represent the randomness of a system and has been used in many engineering design systems to

[2]https://github.com/cmudrc/SFEM

Table 1 Feature comparison of recent neural network architectures for mesh-based analysis

Features	M-GNN [11]	Geom-DeepONet [12]	MagNET [13]	TAGUNet [14]	GINO [15]	AMG [10]
Random unstructured 3D mesh	×		×		×	×
Dynamic loading	×	×	×	×		
Stochastic BC/loading		×	×	×	×	×
Nonparametric geometry		×				
High spatial frequency features				×		

study uncertainty such as multiaxial fatigue on wheel/rail contact [19], axial loading of composite pipes [20], corrosion degradation prediction [21], deep excavation [22], life prediction for wafer packaging [23], manufacturing of wind turbines [24], and uncertainty analysis of rotor systems [25]. Within SFEM specifically, MCS is commonly used as a validation baseline to compare novel SFEM approaches such as stochastic stability of columns with random elastic modulus fields [26], free vibration of beams with three-dimensional material property randomness [27], nonintrusive stochastic analysis of plate structures [28], and geometric nonlinear behavior in composite shells [29].

Despite advancements in SFEM methods, challenges remain. The computational cost of SFEM remains prohibitive for complex systems with many random variables. Additionally, the dependency of accuracy on the number of realizations or expansion terms presents a trade-off between efficiency and precision. These limitations have motivated two parallel research directions. Ongoing research [30] focuses on developing more efficient algorithms to reduce the computational cost of SFEM, particularly for complex systems with many random variables. Alternatively, NN surrogate modeling approaches approximate the SFEM process through machine learning models trained on representative datasets of stochastic inputs and their corresponding responses. This latter approach could potentially transform the iterative SFEM process into a single predictive step, further reducing computational costs.

2.1.2 Neural Network Surrogate Models. FEM has been used in engineering design to run simulations on design configurations to test if they meet system requirements. However, FEM simulations are known to be computationally expensive, often requiring iterative development to meet convergence criteria set by the simulation engineer [31]. To address this, surrogate models (or metamodels) have been introduced as an inexpensive-to-evaluate way to emulate expensive-to-evaluate ground truth models [3,32].

Traditional surrogate models use many techniques such as polynomials, NNs, Gaussian processes, Kriging, radial basis functions, support vector machines, dynamic mode analysis, polynomial chaos expansions, genetic programming, multivariate adaptive regression splines, and Bayesian networks to learn the underlying function mapping [33]. Surrogate models that use NNs learn an implicit function from a dataset of input–output pairs generated using FEM. This results in an implicit "black-box" function, denoted as $\mathcal{M}$. This function maps the input parameters $\mathbf{X}$ of the system to the outputs $\mathbf{Y}$, represented by the equation:

$$\mathbf{Y} = \mathcal{M}(\mathbf{X}) \tag{1}$$

where $\mathcal{M} : \mathbb{R}^{n\times d_1} \mapsto \mathbb{R}^{n\times d_2}$ represents the mapping from SFEM input parameters to converged statistical outputs. In the context of SFEM, $\mathbf{X}$ typically includes geometric parameters, material properties, and probabilistic loading descriptions, while $\mathbf{Y}$ represents the converged statistical quantities such as stress distributions. NN surrogate models are particularly suitable for SFEM applications because the underlying mapping from input parameters to converged statistical quantities in SFEM satisfies the continuity and boundedness assumptions required for universal function approximation [34]. This approximation allows properly designed NNs to approximate the complex physical relationships that emerge from multiple simulations, provided sufficient training data capture the variability in both geometric and loading parameters. Consequently, NN surrogate models have been used in recent engineering design research and have shown high predictive accuracy [35–38].

2.1.3 Neural Network Surrogate Models for SFEM. NN surrogate models for SFEM have been utilized to address uncertainties across several applications [39–41], but these studies primarily focused on capturing the stochastic nature of material properties while maintaining fixed geometric configurations and loading conditions.

The current work explores the impact of varying both static geometric configurations and loading conditions on structural response. Unlike prior research that typically examined single geometries or loading scenarios, such as 2D tensile test specimens [39] or propagating 3D geometry to 2D fields [42], this study compares different model architectures across multiple static 3D geometric configurations and loading conditions to evaluate their predictive capabilities.

2.2 Geometric Deep Learning. GDL [5] has become an important tool for addressing challenges associated with irregular geometries [43,44]. GDL is a field of study that aims to generalize deep neural networks to be used on input data with non-Euclidean domains such as meshes, manifolds, and graphs. Unlike direct machine learning methods that require 3D geometry to be reduced to a lower representation (e.g., an image), GDL learns directly over 3D surfaces, preserving more complex information of the 3D geometry. GNNs are a central algorithm for GDL and have emerged as versatile tools for modeling complex systems, finding applications in diverse fields such as urban wind field prediction [45], soft-tissue mechanics [46], fracture and stress evolution modeling [47], and physical dynamics simulations [48]. These applications underscore the ability of GNNs to effectively process unstructured data representations, making them valuable assets across various domains.

GNNs primarily function by aggregating information from neighboring nodes to improve the representation of each node. This approach is often grounded in the homophily assumption [49], which suggests that nodes connected by edges tend to have similar features. GNNs aim to leverage this assumption to enhance intraclass similarities while distinguishing between different classes. To address challenges like varying neighborhood sizes and the need for local invariance during convolution, GNNs employ diverse mechanisms for aggregating information from neighboring nodes. For example:

- GraphSAGE [50]: Employs a fixed-size neighbor sampling strategy with multiple aggregation schemes and treats all neighboring nodes as equally important.
- Graph attention network (GAT) [51]: Introduces a self-attention mechanism to assign adaptive importance weights to neighbors, enabling more nuanced representation learning.
- Multiresolution graphs [15] have also been employed in GNNs to facilitate faster information propagation.

The use of GNN-based frameworks is growing in engineering design, with applications ranging from predicting physical fields from loading and boundary conditions [52–55] to establishing correlations between material properties and physical responses [56]. These approaches naturally extend to mesh-based problems,

specifically those using finite element meshes, as is the case with SFEM. Here, GNN architectures can leverage the inherent graph structure of the mesh to capture physical relationships between elements. For time-independent solid mechanics problems, where stress and deformation are influenced by the entire domain, mesh-based GNNs process information through sequential message passing steps between neighboring nodes. However, these models face challenges with generalizability under varying conditions [52], and their local message passing operations create additional considerations when propagating information across distant nodes [11]. To address these limitations, various architectural innovations have been studied.

2.2.1 MeshGraphNets. While MeshGraphNets [57] have demonstrated strong generalization and scalability for dynamic physical systems, their application to static solid mechanics problems remains relatively unexplored [11]. Although recent work [11,48] has addressed some limitations of MeshGraphNets across different geometries, these studies have primarily focused on 2D shapes. This work extends the investigation to 3D irregular geometries, which present significantly greater challenges in generalizing across geometry [58].

2.2.2 U-Nets. While traditional U-Net approaches are limited to grid data, their graph-based version, known as GraphUNet [7], can provide the desired generality. Building off graph U-Nets, graph coarsening approaches have also been proposed to address these limitations [14,59]. MAgNET [13] is one such approach that has built upon graph U-Nets using graph pooling/unpooling layers to capture local regularities in the input data while maintaining the structural information for the 3D graph mesh structure. However, this research did not evaluate across multiple 3D geometries and boundary conditions. TAGUNet [14] extends the MagNET study by evaluating across multiple 3D geometries but does not address stochastic boundary conditions and loading environments. Additionally, it was only assessed for continuous displacement across 3D geometry, which may not be applicable for all time-independent solid mechanics problems that also aim to evaluate discontinuous stress fields with high spatial frequency (i.e., there are neighboring nodes of high-stress regions and many nodes with low-stress regions).

2.2.3 Kolmogorov–Arnold Networks. The majority of existing GNN architectures employ multilayer perceptrons (MLPs) as their fundamental building blocks, capitalizing on their well-established ability to approximate any continuous function [34]. This reliance on MLPs has been a cornerstone of GNN development. Recently, a novel approach has emerged, leveraging the Kolmogorov–Arnold representation theorem [60]. This approach, embodied in Kolmogorov–Arnold networks (KANs), replaces the traditional MLP framework with learnable activations and summations. GraphKAN represents an adaptation of this concept to the GNN domain, incorporating spline-based functions to model relationships between nodes [61]. This innovative approach presents a promising avenue for advancing GNN architectures, although further research is necessary to understand its potential and address the unique challenges of specific application domains. Our work contributes to this understanding by evaluating GraphKAN's [61] performance in time-independent solid mechanics problems, particularly assessing whether its spline-based approach offers advantages over traditional MLP-based architectures for capturing complex stress distributions in SFEM applications.

2.2.4 Neural Operator Learning. Neural operators represent a class of machine learning methods designed to learn mappings between infinite-dimensional function spaces, in contrast to traditional neural networks that operate on finite-dimensional vectors [6]. The key innovation lies in their ability to learn operators–mathematical objects that map one function space to another—rather than just point-wise relationships [62]. These methods enable AI-driven numerical simulations to train fast NN surrogate models for approximating solutions of parametric partial differential equations [63], making them particularly suited for PDE solutions where the relationship between input conditions and solutions needs to be learned across entire spatial domains.

However, applying these methods to real-world problems poses significant challenges. A prominent operator learning architecture is the Fourier neural operator (FNO) first proposed by Li et al. [64], which encodes input functions via multiple Fourier layers. Lu et al. [6] also devised a deep operator neural network architecture called DeepONet for effective operator learning. Due to its dual network nature (trunk and branch network), DeepONet can map an infinite space of parameters to the entire solution field on the computational domain. However, DeepONet and FNO, while transformative, are inherently limited to uniform grids due to their reliance on the fast Fourier transform and dual network architecture. Recent attempts to address these limitations have faced their own challenges: Geom-DeepONet [12] requires parametrization of 3D geometry which limits the flexibility of representing arbitrary geometries [65,66], while flexible graph-focused adaptations like GINO [15] and Geo-FNO [67] can struggle with performance degradation in complex geometries due to their Fourier bases and periodic boundary assumptions [10].

For complex geometries, transformer-based models such as Transolver [68] and GNOT [69] previously set the standard for state-of-the-art performance in handling complex data structures. However, these models still encounter significant challenges in capturing high spatial frequency details [10], especially in time-independent solid mechanics problems. AMG [10] demonstrated better performance by integrating specialized graph techniques that enhance feature processing capabilities across various scales, utilizing the graph neural operators, GNO [70], ability to effectively manage local high-frequency features combined with the global graph and physics graph's capability for capturing global low-frequency features. However, the authors did not test the generalizability across varying boundary conditions which was evaluated in previous works [68].

This work evaluates neural operator architectures' ability to handle both irregular 3D geometries and stochastic boundary conditions in an SFEM with high spatial frequency details. While recent works have made progress in handling either geometric complexity through graph-based adaptations or high spatial frequency features through specialized transformer mechanisms, this work extends these research studies by evaluating these methods' capabilities when faced with both challenges simultaneously. Additionally, the lack of evaluation across varying boundary conditions, which play a pivotal role in defining system behavior [71], in recent approaches like AMG leaves a critical gap in understanding these models' practical applicability for SFEM problems.

3 Methods

This work consists of three main components: geometry generation, SFEM simulation, and NN surrogate model comparison. First, we generate synthetic 3D geometries and mesh them for FEM. We then simulate stochastic point elastic loading environments by applying varying loads and boundary conditions to these meshes. Each 3D geometry undergoes SFEM simulation, where multiple stochastic loading scenarios are applied according to the Monte Carlo sampling procedure. The results are processed into node and edge features suitable for graph-based learning. Finally, we evaluate 11 neural network architectures across three categories—attention-based, message passing-based, and hierarchical processing—on their ability to directly predict converged stress distributions based on geometry.

3.1 Dataset Generation. In this work, we model a stochastic point elastic loading environment with varying load magnitudes and boundary conditions where there is an external force from an

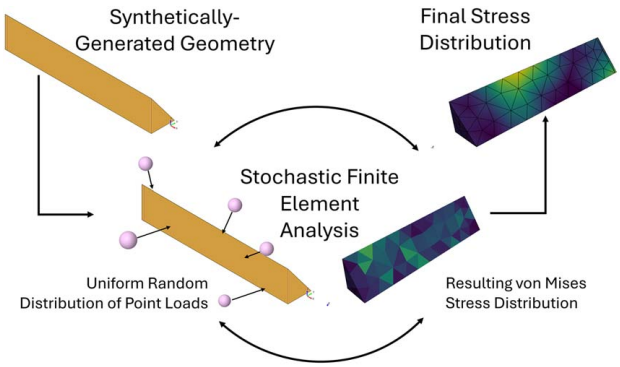

Fig. 1 Data generation process of stochastic point elastic environment loading across ≈ 16,000 3D meshes

object, similar to the analysis conducted during the design of wind turbines [24,72] or safety-critical electronic enclosures [73–75] (Fig. 1).

3.1.1 Geometry Generation and Meshing. To capture the impact of geometric variance, we created a wide set of synthetic 3D geometries. This was accomplished with Brepgen [76], an autoregressive generative model for 3D CAD models in a structured latent tree format that directly supports boundary representation (B-rep) geometry with free-form surfaces, such as those found in the Fusion 360 dataset [77]. These B-Rep 3D geometries were subsequently meshed with a maximum element size (characteristic length) of 0.1 m. A first-order (linear) mesh was created using the Frontal-Delaunay algorithm for 2D elements [78] and the HXT algorithm for 3D elements [79]. The mesh was optimized using Laplacian smoothing for surface and edge refinement, advancing front strategies for local element quality improvement, and metric-based element relaxation techniques to enhance uniformity and minimize distortion [80]. This maximum size constraint and optimization ensure that even for larger geometric features, the element size remains appropriate, while the meshing algorithm automatically generates smaller elements for regions with finer geometric details. The resulting meshes consisted of tetrahedral elements, which are suitable for FEM.

3.1.2 Point Elastic Loading Environment. Loading conditions were simulated by generating uniform randomized loads over the free nodes in three classes: small, medium, and large, as shown in Fig. 2. This approach was motivated by previous research demonstrating that variance in load classes and geometry influences random failures in product life cycles [81]. Each load class was defined by a set of parameters describing the normal force magnitude, the number of load classes, and the number of nodes, as detailed in Table 2. These parameters generated specific load configurations using a uniform distribution Monte Carlo sampling.

In static point load analysis, external loads are characterized by three key components: (1) force magnitude and direction, (2) load location, and (3) contact area [73,74]. For this study, we focused on the normal force component representing the contact area through loaded nodes. The force magnitudes across load classes (20 N, 200 N, and 2000 N) were selected to create a logarithmic progression that explores different loading magnitudes while staying within the material's elastic limit. Specifically, these values were chosen to ensure that even the most extreme combination—maximum force with minimum contact area—would not exceed the yield stress of the material. The contact area is controlled through the minimum and maximum number of nodes. These bounds (3–10 nodes for small, 6–20 for medium, and 12–40 for large) define the range for the uniform random distribution, from which the number of loaded nodes is sampled for each simulation. The number of load cases (5, 10, and 20) acts as a pseudo-position parameter by determining how many different locations on the 3D geometry will experience loading. The largest planar surface area of each geometry was fixed to prevent free-body motion, applying Dirichlet boundary conditions to these nodes. The remaining surface nodes, referred to as "free nodes" throughout this article, are available for load application. For each load case, a "cluster" refers to a subset of these free nodes randomly selected based on a uniform probability distribution. For each load case, a cluster of nodes (whose size is determined by the random sampling between the minimum and maximum bounds) is randomly selected from the available free nodes, distributing loads across different regions of the geometry. The increasing number of load cases with class size increases the coverage of possible loading scenarios for larger, more complex cases. All these parameters can be adjusted based on specific design requirements or environmental conditions. The current values create a simulated environment that explores a wide range of point elastic loading conditions within the linear elastic regime.

3.1.3 Stochastic Finite Element Method. We used FEniCS [82], a finite element analysis engine, to solve for von Mises stress under linear elastic deformation for all simulations. Specifically, we used DOLFINx (version 0.9.0), the next-generation version of the FEniCS library. The variational problem was formulated using continuous Lagrange elements of order 1 for all function spaces to ensure dimensional consistency between node inputs and stress outputs required for graph-based representation. The main equations used for these calculations can be found in other works [83]. This static analysis assumed a single isotropic material (i.e., polycarbonate; Young's modulus: 2.303 GPa, Poisson's ratio: 0.4002). The largest planar surface area of each geometry was fixed to prevent free-body motion. Forces were applied to surface nodes not already fixed (see Fig. 2) using a uniform random sampling process described in Sec. 3.1.2. We applied a Dirichlet boundary condition at the fixed face, and we applied a Neumann boundary condition at loaded nodes.

Algorithm 1 summarizes the simulation process, where force magnitudes were assigned based on load class, and load cases (force locations) and number of nodes selected were uniformly randomized within the bounds for the load class. Each load case corresponds to a cluster of surface nodes, representing a potential impact region. To generate each cluster, a set of free nodes is randomly drawn within the load class-specific bounds, and a consecutive sequence of free surface nodes is selected starting from a random

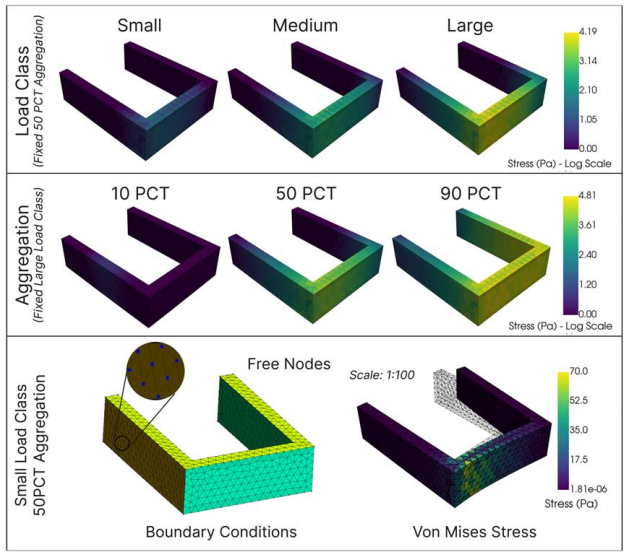

Fig. 2 Visualization of key dataset features: (*a*) sample stress predictions across small, medium, and large load classes, (*b*) the effect of aggregation on final converged distribution, and (*c*) example of the result for a small load class and 50 Percentile (PCT) aggregation

Table 2 Environment loading parameters

Environmental parameters	Small	Medium	Large
Normal force (N)	20.0	200.0	2000.0
# Load cases	5	10	20
Minimum number of nodes	3	6	12
Maximum number of nodes	10	20	40

index. Following N simulations, the von Mises stress results were aggregated by taking the median (50th percentile) of the distribution. This aggregation value can be changed to study different converged distributions as shown in Fig. 2. Specifically, we performed 50 simulations in this work for each design and for each load case. A convergence study showed that the median von Mises stress on average converges to within 10% of the asymptotic value.

Algorithm 1 Dataset generation algorithm using Monte Carlo sampling

$k \leftarrow 1$
while $k \leq N$ **do**
  Select load class $L \in \{small, medium, large\}$
  Set force magnitude $F_i \leftarrow F_L$
  Randomly select number of nodes $A_i \sim Random(A_{L,min}, A_{L,max})$
  Randomly select P_L load cases from the available set
  Compute the displacement u_i
  Calculate von Mises stress σ_i
  Store computed $\sigma_{L,i}$
  $k \leftarrow k + 1$
end while
Compute the final percentile from N simulations
$\sigma_{L,i,\text{perc}} = \text{Percentile}\left(\{\sigma_{L,i}\}_{i=1}^{N}\right)$

3.1.4 Node and Edge Features. In order to efficiently complete the proposed study, we precomputed a set of node and edge features for use across models. Specifically, we precomputed five core node features (position coordinates, signed distance function (SDF), boundary condition (BC) flags, and loading conditions) represented by 12 numerical variables, and two sets of optional edge features based on previous synthetic dataset generated features for stress-based analysis [84,85]. All models use the node features. The GNO and EncoderProcessorDecoder also make use of additional edge features. Node positional features are derived from a signed distance function, computed as the distance between the 3D mesh and an enclosing cube. Positional coordinates are normalized between $[-3, 3]$, and target stress values undergo log normalization followed by unit Gaussian normalization. The output node feature is a scalar value representing von Mises stress as summarized in Table 3.

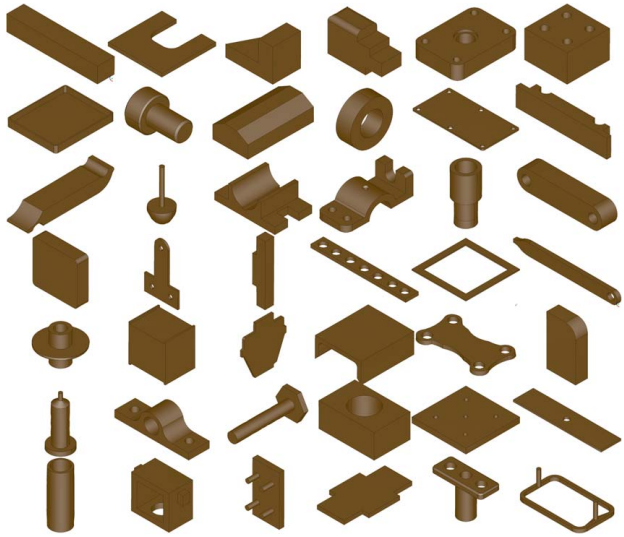

Fig. 3 Sample of 3D geometries generated using Brepgen [76]. These examples demonstrate the variety of shapes possible in the synthetic dataset drawn from a subset of ≈ 16, 000 geometries.

Each design has approximately 2000 nodes. Figure 3 shows a sample of 3D geometries generated using Brepgen, illustrating some of the geometric variations in our dataset. Ultimately, our synthetic dataset consisted of roughly 48,000 samples (approximately 16,000 3D meshes for three load cases). We apply an 85:15 train-evaluation split to the dataset for all cases in this article.

3.2 Network Architectures. In order to explore the space of model architectures, we select models with three primary learning mechanisms: attention-based, message passing-based, and hierarchical processing. Attention-based mechanisms, such as GAT [51], Transolver [68], and AMG [10], focus on dynamically highlighting critical graph components. Attention-based mechanisms enhance model flexibility by learning dynamic importance weights for nodes in each neighborhood. This approach enables a more nuanced understanding of structural relationships, particularly beneficial for SFEM where the influence of random variables varies across the topology. The attention mechanism's ability to selectively aggregate physical information from adjacent nodes has been shown to outperform standard averaging approaches, especially in capturing local stress concentrations and load transfer patterns across mesh elements [86].

Message passing-based mechanisms, including EncoderProcessorDecoder [87], GraphSAGE [50], GNO [70], and GraphKAN [61], facilitate the exchange of information between nodes and edges. Message passing-based approaches leverage the spatial relationships inherent in finite element meshes, working directly with the principle

Table 3 Summary of node and edge features for the graph representation

Feature type	Category	Dimension	Description	Used by model
Node	Position	3	3D coordinates of each vertex	All models
Node	SDF	1	Signed distance field value	All models
Node	Normal vectors	3	Surface normal vectors	All models
Node	Boundary conditions	2	Binary fixed and free node information	All models
Node	Load class	3	One-hot encoded load class (small/medium/large)	All models
Edge	Relative position	3	Difference in position between connected nodes	GNO
Edge	SDF features	2	SDF values for connected nodes	GNO
Edge	Normal features	6	Normal vectors for connected nodes	GNO
Edge	Cartesian features	3	Normalized relative position between connected nodes	EncoderDecoder
Edge	Point pair features	4	Rotation-invariant features	EncoderDecoder

Table 4 Performance comparison across different learning mechanisms for point elastic loading environments

Model	Learning mechanism	RMSE	MAE	90 PCT R^2	50 PCT R^2	10 PCT R^2
GAT	Attention	0.691	0.496	0.846	0.571	0.059
Transolver[a]	Attention	0.688	0.484	0.867	0.588	0.034
AMG[a]	Attention	**0.684**	**0.480**	**0.870**	**0.596**	0.043
EncoderProcessorDecoder	Message passing	0.733	0.536	0.789	0.490	0.006
GraphSAGE	Message passing	0.691	0.493	0.851	0.585	0.048
GNO	Message passing	0.703	0.502	0.833	0.558	0.052
GraphKAN	Message passing	0.687	0.498	0.850	0.578	0.071
GraphUNet	Hierarchical	0.906	0.679	0.669	0.317	−0.605
TAGUNet	Hierarchical	0.927	0.693	0.705	0.329	−0.709
Geo-FNO	Hierarchical	0.704	0.510	0.806	0.549	**0.115**
GINO[a]	Hierarchical	0.967	0.766	0.219	0.061	−0.088

The best result is in bold and the second best is italicized.
[a]Models that required a reduced learning rate of 1.0×10^{-4} for stable training, compared to the standard 1.0×10^{-3} used for other models.

that connected nodes' share-related physical properties. Through iterative information exchange between neighboring nodes, these models can effectively learn and propagate both local deformation patterns and, after sufficient message passing steps, global structural patterns. This class of models uses specialized kernel techniques to capture complex stress distributions across different spatial scales, which makes them particularly suited for problems where stress fields exhibit strong spatial correlations.

Hierarchical processing models, such as GraphUNet [7], TAGUNet [14], Geo-FNO [67], and GINO [15], leverage multilevel graph representations to capture local and global structures. Hierarchical models operate at multiple resolution levels, enabling them to simultaneously capture both fine-grained stress concentrations and broad structural responses. This multiscale approach is particularly valuable for SFEM surrogate modeling, where stress patterns often exhibit both localized peaks and global distribution patterns. By processing the mesh at different levels of granularity, these models could efficiently learn the relationship between loading conditions and resulting stress distributions while preserving critical information at each scale.

To ensure compatibility with our research objectives, some models required minor architectural modifications. Geo-FNO and GINO were adapted to automatically determine the dimensions of the grid based on the input of the geometry. Geo-FNO and GINO, without modifications, require fixed input dimensions, but it was modified to automatically determine grid dimensions as $\lceil\sqrt{[3]N}\rceil^3$ based on input coordinates N (i.e., each 3D mesh had input coordinates of size $NX3$), with padding for noncubic-perfect counts. These modifications enable these models to process different input geometry sizes. The GraphKAN architecture was modified by removing the logistic output layer while maintaining its core network structure of coordinate-based Kolmogorov–Arnold transformations, enabling direct regression predictions rather than classification probabilities. This adaptation preserves the fundamental interpretability advantages of the Kolmogorov–Arnold representation, as the modification only affects the final output transformation rather than the underlying spline-based functions that model relationships between nodes.

To ensure consistency in performance comparisons, all models were trained for 50 epochs using the Adam optimizer on a Tesla V100-DGXS-32GB GPU. We used a base learning rate of 1×10^{-3}, except for Transolver, AMG, and GINO, which required 1×10^{-4} for stable convergence [88]. Models were trained for 50 epochs without learning rate scheduling (fixed learning rate) to maintain uniformity across the training setup. The Huber loss function ($\delta = 0.5$) was chosen as the optimization objective to handle right-skewed stress distributions. Final model weights were selected based on minimum validation loss.

Model performance was evaluated using three standard regression metrics: coefficient of determination (R^2), root-mean-squared error (RMSE), and mean absolute error (MAE). To characterize performance across different geometric configurations, we analyze the distribution of R^2 values using 90th, 50th, and 10th percentiles, providing insight into best-case, typical, and worst-case performance, respectively. RMSE and MAE values were calculated across all test cases to provide absolute measures of prediction error. Model predictions were also qualitatively compared against ground truth stress distributions from the synthetic dataset for prediction accuracy and precision.

To assess differences between architectural approaches, we employed a three-stage statistical analysis. First, one-way ANOVA was conducted to test for overall differences between learning mechanisms. This was followed by Tukey honestly significant difference (HSD) post hoc tests to identify specific pairwise differences between approaches. Finally, effect sizes were calculated using eta-squared (η^2) to quantify the proportion of variance explained by learning mechanism categories and Cohen's d to measure the practical significance of pairwise differences between mechanisms.

4 Results

Table 4 presents a comprehensive comparison of 11 model architectures across three learning mechanisms for NN surrogate SFEM modeling.

4.1 Model Architecture Performance. The moderate R^2 values across all models (median $R^2 < 0.6$) reflect the fundamental complexity of this task. These results highlight several inherent challenges: first, these models attempt to predict in a single step what traditionally emerges from the statistical aggregation of several simulations, requiring them to internalize patterns from repeated sampling. Second, the models must generalize across varying boundary conditions and multiple synthetic geometries, adding significant complexity compared to traditional single-geometry SFEM applications. Third, the physical nature of point loading problems demands simultaneous handling of both local and global dependencies—models must capture localized high-stress regions while relating them to overall geometric structure and load distributions. This is particularly challenging for stress predictions, which, unlike displacement fields in linear elastic analysis, can be discontinuous between neighboring nodes. Despite these challenges, certain architectural approaches show promise in capturing these complex relationships, with some models achieving the 90th percentile R^2 values above 0.85.

4.1.1 Attention-Based Models. The attention-based mechanisms demonstrated overall better performance. AMG achieved the best results with a 90th percentile R^2 of 0.870, a median R^2 of 0.596, and the lowest RMSE of 0.684. Transolver performed closely with a 90th percentile R^2 of 0.867 and median R^2 of

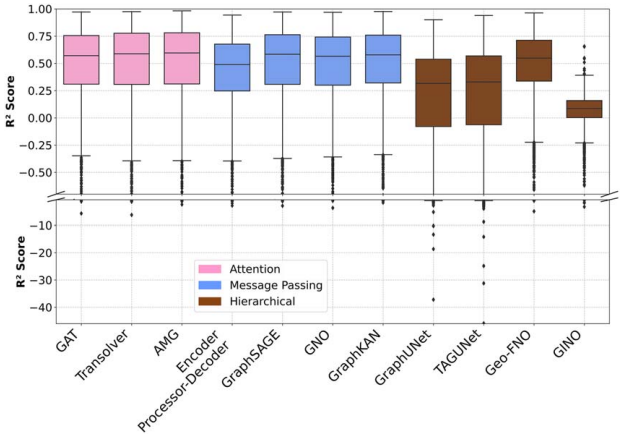

Fig. 4 Distribution of R^2 values across learning mechanisms (attention-based, message passing, and hierarchical), highlighting performance consistency and variability between different architectural approaches

0.588. Both models achieved these results with a reduced learning rate (1×10^{-4}), compared to the standard 1×10^{-3} used for other architectures because of training instabilities, but the baseline GAT model showed competitive performance with a 90th percentile R^2 of 0.846 and median R^2 of 0.571 without requiring learning rate adjustments.

4.1.2 Message Passing Approaches. Message passing mechanisms showed consistent performance across metrics. GraphKAN achieved particularly strong results with a 90th percentile R^2 of 0.850 and the second-best 10th percentile R^2 of 0.071, indicating robust performance even in challenging cases. GrapSAGE and GNO demonstrated similar stability with the 90th percentile R^2 values of 0.851 and 0.833, respectively, while the EncoderProcessorDecoder showed slightly lower performance with a 90th percentile R^2 of 0.460.

4.1.3 Hierarchical Methods. Hierarchical approaches generally underperformed compared to attention and message passing mechanisms. Performance varied significantly, with Geo-FNO achieving the best results in this category (90th percentile $R^2 = 0.806$) and notably the best 10th percentile R^2 (0.115) across all models. However, GINO exhibited limited predictive accuracy, with a median R^2 of 0.061. GraphUNet and TAGUNet showed a high degree of variability in their predictive performance with 10th percentile R^2 values below −0.6.

4.2 Performance Distribution and Statistical Analysis. Figure 4 shows the distribution of R^2 values across all models, categorized by their learning mechanism. Attention-based models exhibit the most similar distributions, indicating more consistent performance. However, hierarchical models demonstrate wider variability in their R^2 values. Message passing approaches show intermediate levels of performance consistency.

In Table 5, statistical analysis across learning mechanisms shows significant differences in model performance distributions (ANOVA: $F = 2684.05$, $p < 0.001$). Tukey HSD post hoc tests confirm that all pairwise comparisons between learning mechanisms are statistically significant. However, the practical significance of these differences, measured through Cohen's d effect sizes, varies. The comparison between attention-based and hierarchical methods shows a medium effect size ($d = 0.5546$), as does the comparison between hierarchical and message passing approaches ($d = -0.5173$). In contrast, attention-based and message passing methods show only a very small effect size ($d = 0.0699$), suggesting that despite statistical significance, their practical differences are minimal. The overall $\eta^2 = 0.0754$ indicates a medium effect size for the learning mechanism classification as a whole.

Table 5 Statistical analysis of learning mechanism performance

	ANOVA			Tukey HSD		Cohen's d	
Comparison	F-Statistic	p-Value	Sig.	Mean Diff.	Sig.	d	Effect size
Among groups	2684.05	<0.001	Yes	–	–	–	–
Attention versus hierarchical	–	–	–	−0.2966	Yes	0.5546	Medium
Attention versus message passing	–	–	–	−0.0224	Yes	0.0699	Very small
Hierarchical versus message passing	–	–	–	0.2742	Yes	−0.5173	Medium

Table 6 Kolmogorov–Smirnov test results across learning mechanisms

		Kolmogorov–Smirnov test		
Category	Model pair	KS-statistic	p-Value	Significant
Attention	AMG versus GAT	0.0460	3.63×10^{-7}	Yes
	AMG versus Transolver	0.0131	0.5548	No
	GAT versus Transolver	0.0402	1.48×10^{-5}	Yes
Message passing	GNO versus GraphKAN	0.0347	2.97×10^{-4}	Yes
	GNO versus GraphSAGE	0.0383	4.45×10^{-5}	Yes
	GNO versus EncoderProcessorDecoder	0.1079	1.58×10^{-37}	Yes
	GraphKAN versus GraphSAGE	0.0154	3.48×10^{-1}	No
	GraphKAN versus EncoderProcessorDecoder	0.1245	8.52×10^{-50}	Yes
	GraphSAGE versus EncoderProcessorDecoder	0.1364	1.05×10^{-59}	Yes
Hierarchical	GINO versus GraphUNet	0.5040	0.00×10^{-0}	Yes
	GINO versus TAGUNet	0.5117	0.00×10^{-0}	Yes
	GINO versus Geo-FNO	0.7711	0.00×10^{-0}	Yes
	GraphUNet versus TAGUNet	0.0463	3.02×10^{-7}	Yes
	GraphUNet versus Geo-FNO	0.2785	3.75×10^{-250}	Yes
	TAGUNet versus Geo-FNO	0.2636	7.72×10^{-224}	Yes

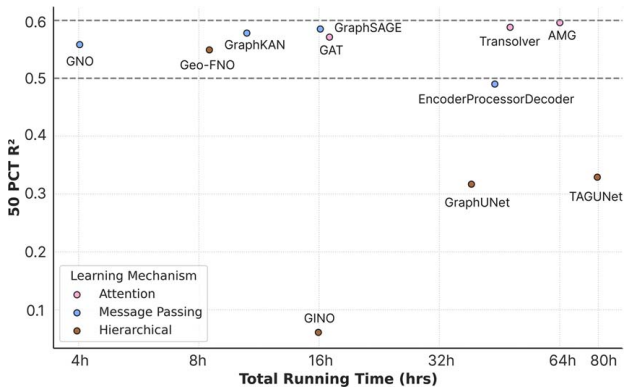

Fig. 5 Time-accuracy trade-off comparing model architectures. Each point represents a model's 50 PCT R^2 value versus total training time.

Kolmogorov–Smirnov (KS) tests were also done to compare the R^2 prediction distributions between individual model pairs. The tests reveal distinct patterns in architectural behavior (Table 6). Within attention-based approaches, AMG and Transolver show statistically similar distributions ($p = 0.5548$), while both differ significantly from GAT ($p < 0.05$). Among message passing methods, only GraphKAN and GraphSAGE demonstrate statistical similarity ($p = 0.348$), with all other pairs showing significant differences ($p < 0.05$). Hierarchical methods show significant differences in distributions both within their category and compared to other approaches (all $p < 0.05$), with particularly large KS statistics (ranging from 0.0463 to 0.7711).

4.3 Computational Efficiency Analysis. Figure 5 illustrates the relationship between computational efficiency (total training time) and predictive performance across different architectures. Message passing methods generally achieve competitive performance with the lowest computational overhead. While attention mechanisms generally achieved the highest predictive accuracy, they also required more training time than message passing methods. Hierarchical models, despite showing lower overall performance, require significantly more training time.

While total training time provides insights into model development costs, inference time is typically more important for practical applications as it determines the model's usability in real-world engineering workflows where rapid analysis of new designs is crucial. Analysis of inference time performance reveals distinct patterns across learning mechanisms (Fig. 6).

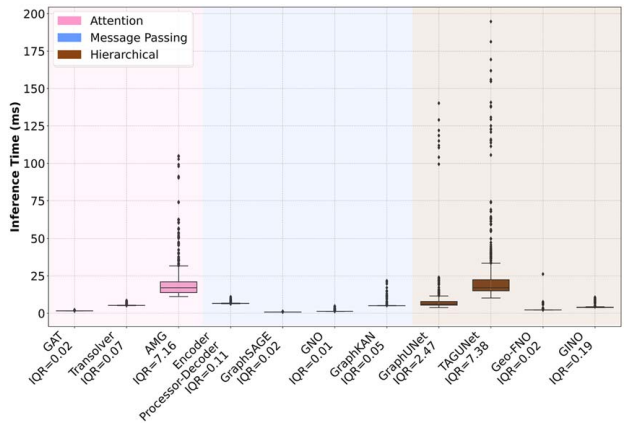

Fig. 6 Distribution of inference time across learning mechanisms (attention-based, message passing, and hierarchical). The interquartile range for each model is given by IQR.

Table 7 Effect of the number of nodes on the inference time across model architectures

Model	Correlation (ρ)	Min time (ms)	Max time (ms)
GAT	0.246	1.52	2.32
Transolver	0.058	5.10	8.57
AMG	0.991	11.19	104.93
EPD[a]	0.289	6.29	10.99
GraphSAGE	0.526	0.81	1.24
GNO	0.709	1.25	4.77
GraphKAN	0.717	4.97	21.70
GraphUNet	0.516	3.77	140.18
TAGUNet	0.730	10.23	194.72
Geo-FNO	0.092	2.17	26.18
GINO	0.106	3.80	10.69

[a]EPD, EncoderProcessorDecoder.

Among attention-based approaches, AMG shows the highest variability (interquartile range (IQR) = 7.16 ms), while GAT and Transolver demonstrate more consistent performance (IQR = 0.02 ms and 0.07 ms, respectively). Message passing approaches maintain consistent inference times, with EncoderProcessorDecoder showing the highest variability (IQR = 0.11 ms) within this category, while GNO maintains more consistent timing (IQR = 0.01 ms). Hierarchical approaches exhibit the most significant variations, with GraphUNet and TAGUNet showing high interquartile ranges (IQR = 2.47 ms and 7.38 ms, respectively).

Table 7 illustrates how inference time scales with input geometry size. AMG shows a strong linear correlation with node count ($\rho = 0.991$), while message passing methods demonstrate moderate correlations (GraphKAN: $\rho = 0.717$, GNO: $\rho = 0.709$). Hierarchical methods show varying correlations (ρ ranging from 0.516 to 0.730).

4.4 Qualitative Model Performance Analysis. Figure 7 shows a qualitative comparison of stress predictions for representative 3D geometries, focusing on the best-performing models from each learning mechanism: AMG, GraphKAN, and Geo-FNO. Visual inspection reveals distinct patterns in model behavior across different geometric complexities.

In some cases, all three models demonstrate high precision and accuracy in their predictions, suggesting that certain 3D geometries are consistently well captured by all learning mechanisms. However, significant discrepancies in predictions are also observed in other scenarios. For example, AMG generally maintains better accuracy in regions of high stress. Geo-FNO, while potentially exhibiting lower absolute accuracy, demonstrates more consistent precision across the entire geometry. GraphKAN, despite achieving the second-best worst-case (10 PCT) performance metrics, sometimes struggles with the precise location of stress.

5 Discussion

This study highlights practical considerations and persistent challenges in developing NN surrogate models for SFEM. While certain architectures demonstrate potential for approximating the iterative SFEM process, distinct trade-offs exist between accuracy, computational efficiency, and stability across learning mechanisms. Our findings suggest that the success of these models depends on their ability to balance local geometric features with global structural understanding, particularly in the context of complex geometries and loading conditions.

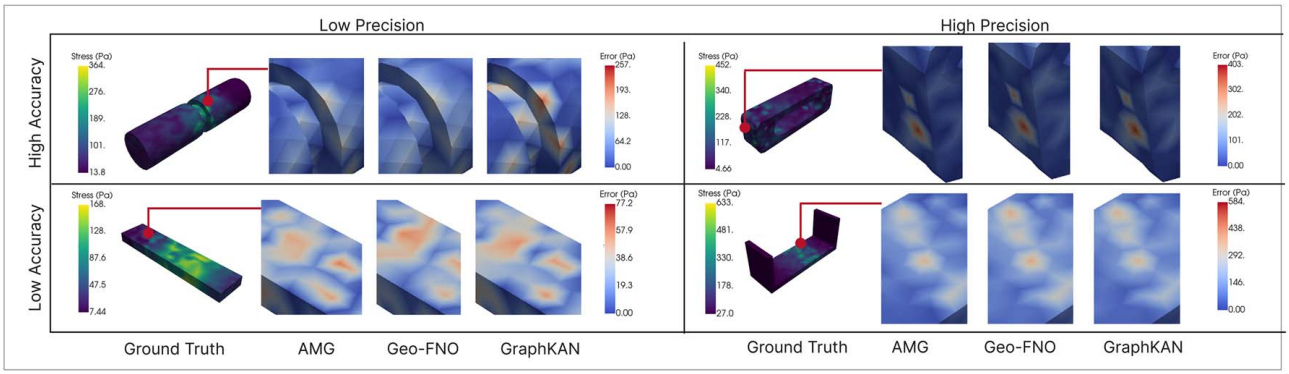

Fig. 7 Comparison of error between von Mises stress predictions and synthetic dataset across the best-performing models (AMG, GraphKAN, and Geo-FNO) for representative 3D geometries, showing variations in prediction accuracy and precision

5.1 Performance Analysis and Limitations. Attention-based models demonstrated better performance through their ability to dynamically weight node relationships based on their relevance to stress prediction. This capability is particularly valuable for capturing stress concentrations, where the influence of neighboring nodes varies significantly. The performance of AMG and Transolver suggests that selective focus on relevant node relationships is crucial for accurate stress prediction. However, these benefits come with trade-offs. During training, the $O(N \cdot \text{Degree})$ [15] computational complexity of the attention mechanisms may present scalability challenges for larger meshes. This quadratic scaling means that doubling the mesh size quadruples the computational requirements, potentially limiting their application to larger-scale engineering problems. Additionally, the requirement for reduced learning rates indicates potential training stability issues that must be carefully managed during implementation.

Message passing approaches offer an alternative with $O(N \cdot \text{Degree})$ training complexity [15] which makes them more scalable than attention mechanisms. Models like GraphKAN demonstrate this advantage through robust performance across different scenarios (90th percentile R^2 of 0.850), due to their ability to effectively propagate information through multiple neighborhood iterations. However, the lower performance of the EncoderProcessorDecoder (90th percentile R^2 of 0.789) highlights a limitation—without specialized kernels for message aggregation, these models require careful tuning of message passing steps to achieve optimal performance. The performance of GraphKAN and GraphSAGE suggests that specialized message aggregation strategies are useful for capturing both local and global features for message passing approaches.

The limited performance of hierarchical approaches (particularly GINO with 50PCT R^2 of 0.061 and GraphUNet with poor stability) can be attributed to several fundamental challenges. While these methods theoretically offer efficient scaling ($O(N \cdot \log \cdot N)$ for GINO) during training [15], their coarsening operations and reduced representation space often fail to capture high-stress regions. The process of transforming irregular grids to uniform latent structures, while computationally efficient, appears to lose essential local geometric information needed for accurate stress prediction. This trade-off between computational efficiency and geometric fidelity limits its applications for stress fields, which can exhibit sharp discontinuities and require local geometry capture. The better performance of Geo-FNO (90PCT $R^2 = 0.806$) compared to other hierarchical methods suggests that maintaining some form of geometric awareness during the coarsening process is crucial to match the performance of attention or message passing approaches.

The performance gap between attention-based/message passing approaches and hierarchical methods warrants careful consideration. While GINO and Geo-FNO have shown promise in adapting irregular grids to uniform latent structures in fluid simulation fields, their performance in our experiments (50PCT of $R^2 = 0.061$ and $R^2 = 0.549$, respectively) suggests limitations in handling the complex physics of point elastic loading environments. This aligns with known challenges of Fourier-based techniques [10], particularly their dependence on periodic boundary conditions, which can lead to performance degradation in complex geometries.

While our attention-based models achieved the best performance metrics (with AMG reaching a 90th percentile R^2 of 0.870), the practical utility of these results requires careful consideration. Although there is no universally accepted threshold for surrogate model accuracy in engineering design applications, the substantial gap between best-case and median performance across all models raises concerns about consistency. This variability, combined with poor worst-case performance (10 PCT R^2 values often near-zero or negative), suggests that current architectures would be insufficient for safety-critical applications where reliability across diverse cases is essential. Currently, these surrogate models might serve as complementary tools for preliminary design exploration, potentially accelerating certain aspects of the design process, while still requiring traditional SFEM for final validation.

5.2 Technical Implications for SFEM Neural Network Surrogate Modeling. Our results highlight several key technical considerations for developing effective SFEM NN surrogate models. First, the ability to capture discontinuous stress fields appears more important than computational efficiency, as evidenced by the success of computationally intensive attention mechanisms. Second, the relationship between local geometric features and global stress distributions requires careful architectural design—simple pooling or coarsening operations often prove insufficient. Third, the stability of predictions across different geometric configurations suggests that model architecture influences generalization capability. The varying performance across different geometric cases (as shown in Fig. 7) also reveals important patterns in how different architectures handle the competing demands of local accuracy and global consistency. Attention-based models capture local stress concentrations but at a higher computational cost, while message passing approaches offer a more balanced solution for practical applications.

The statistical analysis also provides important insights for practical model selection in SFEM applications. While all learning mechanisms show statistically significant differences, the effect sizes suggest that the choice between attention-based and message passing approaches may have minimal impact on practical outcomes. The medium effect sizes observed when comparing hierarchical methods to both attention-based and message passing approaches ($d = 0.5546$ and $d = -0.5173$, respectively) indicate meaningful practical differences in their performance distributions. These findings suggest that while hierarchical methods might offer

theoretical advantages in multiscale analysis, their current implementations result in materially different prediction patterns compared to other approaches. For engineering applications, this suggests that the choice between attention-based and message passing architectures can be primarily driven by computational considerations rather than prediction quality, while the adoption of hierarchical methods should be carefully evaluated based on specific application requirements.

The statistical similarity between AMG and Transolver, despite their architectural differences, suggests that certain attention mechanisms may converge on similar prediction strategies. Similarly, the comparable distributions of GraphKAN and GraphSAGE indicate that different message passing implementations can achieve functionally equivalent results. The significant differences observed in hierarchical methods' distributions, combined with their lower performance metrics, suggest fundamental limitations in how current hierarchical architectures process SFEM-relevant information. These statistical patterns reinforce our performance analysis while revealing subtler aspects of architectural influence. The fact that most model pairs show significant differences in their prediction distributions, even within the same learning mechanism category, emphasizes how architectural choices fundamentally shape how models learn and generalize across different geometric configurations.

5.3 Computational Efficiency and Scaling Considerations. The computational demands of traditional SFEM versus NN surrogate models represent a critical consideration for practical engineering applications. For our dataset generation, each geometry required an average of 50 Monte Carlo simulations to achieve convergence. While industrial engineering geometries typically contain 100, 000+ nodes, our synthetic dataset used meshes with an average of 2000 nodes per geometry to make the extensive computational experiments feasible. With these reduced-size geometries, each complete SFEM simulation took approximately 5 min on a Tesla V100-DGXS-32GB GPU, resulting in approximately 200 h of total computation time for dataset generation across all geometries.

Although generating the dataset and training NN surrogate models demanded substantial computational resources, these costs are incurred only once. The real challenge lies in the efficiency of inference in determining whether these models can provide fast predictions at scale. Inference speed and scalability determine the viability of a model as a replacement for traditional SFEM simulations. Inference time analysis reveals important practical considerations for model selection that extend beyond prediction accuracy alone. The strong linear correlation between AMG's inference time and node count ($R^2 = 0.991$) suggests potential scalability challenges for larger engineering models despite its predictive performance. This scaling concern becomes particularly significant when considering industrial applications, where typical engineering geometries contain 100, 000+ nodes. Our additional testing on such full-scale geometries showed that traditional SFEM requires approximately 500 h per geometry to achieve convergence, a dramatic increase from 5 min per geometry for our 2000-node test cases. Message passing approaches offer a more favorable balance, maintaining consistent inference times while achieving comparable accuracy. Their moderate scaling relationships ($R^2 \approx 0.71$) indicate better suitability for these larger-scale applications.

The variable performance of hierarchical methods, both in prediction accuracy and computational efficiency, suggests that their current implementations may need refinement before practical adoption. While their linear scaling is encouraging, the high variability in inference times (IQR up to 7.38 ms) could pose challenges in time-sensitive applications. These computational trade-offs become particularly relevant when considering the minimal practical differences in prediction accuracy between attention-based and message passing approaches (Cohen's $d = 0.0699$). For many applications, the more consistent and scalable performance of message passing architectures may outweigh the marginal accuracy advantages of attention-based approaches.

5.4 Future Research Directions. Future work may explore nonlinear elastic analysis, dynamic modeling, and more variable hyperelastic materials beyond our current scope. Additionally, investigating the effects of mesh quality on prediction and conducting comprehensive convergence studies could provide valuable insights for improving model robustness.

Future work should also explore hybrid architectures that could combine the strengths of different learning mechanisms—for example, using message passing for efficient local feature extraction while selectively applying attention mechanisms for critical high-stress regions. Additionally, investigating adaptive mesh refinement strategies that could dynamically adjust the model's focus based on stress gradients could help address the current limitations in capturing both local and global features. For the SFEM random node selection, future work for applications where geometric control is important, e.g., ensuring clusters represent compact, physically contiguous zones, a more advanced clustering algorithm (e.g., region growing or graph-based connectivity) could be incorporated to enforce path-connectedness or shape constraints. Developing more physically realistic stochastic loading patterns that better reflect specific engineering domains also represents an important direction for future work. Our use of Lagrangian functional spaces for von Mises stress represents a limitation and an avenue for future work, which can investigate alternative mesh representations that could accommodate discontinuous Galerkin formulations while maintaining compatibility with graph-based learning architectures. Moving from node-based to element-based graph representations could potentially allow the use of discontinuous Galerkin elements for stress fields, which better represent physical stress discontinuities at element boundaries. This approach might better capture the high spatial frequency features in stress distributions while potentially improving the surrogate models' predictive capabilities, although it would require significant modifications to the graph construction and feature engineering processes. Future work could also explore the relationship between SFEM convergence criteria and surrogate model performance, potentially identifying optimal trade-offs between computational cost and predictive accuracy for specific application domains.

Our use of fixed learning rates without scheduling represents a limitation of the current study. Future work can explore more sophisticated optimization strategies, including learning rate scheduling and adaptive optimization methods, which could further improve model performance. The observed training instabilities in certain architectures suggest that optimization strategies may need to be tailored to specific learning mechanisms for optimal performance. Another particularly important avenue for future work is the systematic characterization of failure cases observed in the current study (near-zero or negative R^2 values at the 10th percentile). This could involve developing geometric complexity metrics (such as surface area to volume ratios, feature count, or curvature distribution) and correlating these with model performance. This analysis would help identify whether failures are systematically related to specific geometric features or are distributed randomly across the dataset. Identifying whether failures cluster around specific geometric characteristics could provide valuable insights for architectural improvements. Developing more sophisticated heuristics for boundary condition assignment that better reflect physical reality while maintaining cross-geometry comparability represents another important direction for future research. This could include geometry-aware boundary condition assignment that considers functional intent rather than applying a single rule across all cases such as fixing the largest surface area presented. While our approach incorporates geometric variability through stochastically generated geometries (with Brepgen [76]), it differs from approaches that parametrically vary specific dimensions within specific ranges. Our method cannot directly accept geometric

uncertainty parameters as inputs, representing a limitation for applications requiring explicit parametric analysis. Future work could explore hybrid approaches that combine stochastic geometry generation with parametric geometric uncertainty handling.

More fundamental challenges remain in the field of NN surrogate modeling for SFEM that also need to be addressed. Current approaches struggle with high-dimensional objects containing multiple modalities, particularly when dealing with graphs with many nodes [4]. The inherent difficulty in modeling stochastic systems where inputs follow probabilistic data generation processes, as in SFEM, presents another significant challenge [4].

Effectively addressing these challenges requires a deeper understanding of uncertainty, which plays a crucial role in determining the reliability of surrogate models. Uncertainties can be broadly categorized into aleatoric uncertainty, arising from inherent process randomness (e.g., variability in experimental data), and epistemic uncertainty, which stems from insufficient training data or model imperfections [89,90]. Our focus is on aleatoric uncertainty, as epistemic uncertainty has been well-addressed in other communities [90–94]. Aleatoric uncertainty manifests through variations in controllable and uncontrollable parameters like geometry and loading conditions, making probabilistic modeling approaches essential. However, no current method provides highly accurate aleatoric uncertainty estimates, underscoring the need for improved benchmarks, methods, and training resources [93,95]. Despite advances in neural networks and neural operators for handling 3D geometries, developing models capable of handling broader spectrums of geometries, loads, and boundary conditions remains an active area of research that requires more diverse datasets and generalizable NN surrogate models [52,96].

These findings ultimately underscore the continuing challenge of NN surrogate models for SFEM in complex engineering systems. While NN approaches offer compelling computational advantages, they may not be inherently well-suited to the specific challenges of SFEM prediction. Future work can also explore alternatives to direct SFEM surrogate modeling such as using neural networks as surrogates for single static load cases within the traditional Monte Carlo simulation process. This approach could accelerate each iteration of SFEM by replacing computationally expensive FEM evaluations with rapid surrogate predictions. The observed limitations across multiple architectural paradigms indicate that we should also explore alternative surrogate modeling techniques discussed in our related work, such as Kriging, polynomial chaos expansions, or hybrid physics-informed methods that more explicitly incorporate the underlying mechanical principles. While these results demonstrate incremental progress, they also highlight the substantial work still needed to develop more robust and generalizable methods for engineering design applications [97,98].

6 Conclusion

This work investigated the capabilities of various neural architectures for SFEM in point elastic loading environments, with a particular focus on capturing aleatoric uncertainty using NN surrogate SFEM models. Through an evaluation of 11 model architectures, we identified both promising advances and significant challenges in the field. The performance patterns across the different architectures highlight the challenges in capturing both local and global dependencies in SFEM NN surrogate modeling. The performance of attention-based models suggests the importance of dynamically weighting node relationships, particularly for capturing the discontinuous nature of stress fields. The stability of message passing approaches indicates their effectiveness in learning local patterns while maintaining computational efficiency. The varying performance of hierarchical methods, particularly their struggles with worst-case (10 PCT) scenarios, suggests that current multiscale approaches may not yet be optimal for handling the complex interplay between geometry, boundary conditions, and stress distributions in SFEM applications.

Consequently, these findings demonstrate that current approaches to NN surrogate models capturing SFEM processes, while showing progress, still fall short of the reliability requirements for practical engineering applications. The performance contrast between best-case (90 PCT) ($R^2 = 0.870$) and worst-case (10 PCT) ($R^2 < -0.709$) scenarios across models emphasizes the need for more stable and generalizable approaches.

This study has several promising directions for advancing SFEM NN surrogate modeling. Future architectural developments should focus on reducing the performance variability across different geometric configurations while maintaining the computational advantages demonstrated by message passing approaches. Technical improvements should address the challenges of simultaneously handling irregular geometries and stochastic boundary conditions, particularly for complex 3D structures. Extending this framework to nonlinear elastic analysis and diverse material properties will be crucial for broader engineering applications, alongside investigations into mesh quality sensitivity for enhanced model robustness.

By exposing both the current capabilities and limitations of various approaches, this work aims to advance further research into more robust methods for NN surrogate models for SFEM in engineering design. The challenges identified here underscore the substantial work still needed to develop NN surrogate models that can reliably support real-world engineering applications.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The data and information that support the findings of this article are freely available.[3]

Appendix: ANOVA and Tukey Honestly Significant Difference Assumptions

Robustness of ANOVA to Assumption Violations. ANOVA is generally robust to moderate violations of the normality assumption, particularly when sample sizes are large [99]. In our study, with sample sizes exceeding 7000 per group, the central limit theorem ensures that the sampling distribution of the means approximates normality, even if the underlying data are not perfectly normal. Additionally, research has shown that ANOVA tolerates nonnormal data especially when group sizes are equal or large [100]. In cases where the assumption of homogeneity of variances might be violated, ANOVA remains robust when group sizes are equal or nearly equal [101].

Nonparametric Comparisons. To further validate our findings, we included nonparametric tests, such as the Kolmogorov–Smirnov test, which do not rely on normality assumptions [102]. These tests provided additional support for the robustness of our conclusions. Beyond statistical significance, we reported effect sizes (e.g., η^2 and Cohen's d) throughout our results. These measures are less sensitive to assumption violations and provide a clearer picture of the practical significance of our findings.

References

[1] Salomon, S., and Salomon, S., 2019, *Active Robust Optimization*, Springer, Cham, Switzerland.

[2] Nerenst, T. B., Ebro, M., Nielsen, M., Eifler, T., and Nielsen, K. L., 2021, "Exploring Barriers for the Use of FEA-Based Variation Simulation in Industrial Development Practice," Des. Sci., **7**, p. e21.

[3] Kudela, J., and Matousek, R., 2022, "Recent Advances and Applications of Surrogate Models for Finite Element Method Computations: A Review," Soft Comput., **26**(24), pp. 13709–13733.

[3]See Note 2.

[4] Yang, Y., 2022, "Deep Learning and Uncertainty Quantification: Methodologies and Applications," Ph.D. thesis, University of Pennsylvania.
[5] Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P., 2017, "Geometric Deep Learning: Going Beyond Euclidean Data," IEEE Signal Process. Mag., **34**(4), pp. 18–42.
[6] Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E., 2021, "Learning Nonlinear Operators Via DeepONet Based on the Universal Approximation Theorem of Operators," Nat. Mach. Intell., **3**(3), pp. 218–229.
[7] Gao, H., and Ji, S., 2019, "Graph U-Nets," International Conference of Machine Learning, Long Beach, CA, June 9–15, PMLR, pp. 2083–2092.
[8] Asif, N. A., Sarker, Y., Chakrabortty, R. K., Ryan, M. J., Ahamed, M. H., Saha, D. K., Badal, F. R., et al., 2021, "Graph Neural Network: A Comprehensive Review on Non-Euclidean Space," IEEE Access, **9**, pp. 60588–60606.
[9] Han, J., Cen, J., Wu, L., Li, Z., Kong, X., Jiao, R., and Yu, Z., 2025, "A Survey of Geometric Graph Neural Networks: Data Structures, Models and Applications," Front. Comput. Sci., **19**, p. 1911375.
[10] Li, Z., Song, H., Xiao, D., Lai, Z., and Wang, W., 2025, "Harnessing Scale and Physics: A Multi-Graph Neural Operator Framework for PDEs on Arbitrary Geometries," Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Toronto, Canada, Aug. 3–7.
[11] Gladstone, R. J., Rahmani, H., Suryakumar, V., Meidani, H., D'Elia, M., and Zareei, A., 2024, "Mesh-Based GNN Surrogates for Time-Independent PDEs," Sci. Rep., **14**(1), p. 3394.
[12] He, J., Koric, S., Abueidda, D., Najafi, A., and Jasiuk, I., 2024, "Geom-DeepONet: A Point-Cloud-Based Deep Operator Network for Field Predictions on 3D Parameterized Geometries," Comput. Methods Appl. Mech. Eng., **429**, p. 117130.
[13] Deshpande, S., Bordas, S. P., and Lengiewicz, J., 2024, "Magnet: A Graph U-Net Architecture for Mesh-Based Simulations," Eng. Appl. Artif. Intell., **133**, p. 108055.
[14] Ferguson, K., Chen, Y.-H., Chen, Y., Gillman, A., Hardin, J., and Burak Kara, L., 2025, "Topology-Agnostic Graph U-Nets for Scalar Field Prediction on Unstructured Meshes," ASME J. Mech. Des., **147**(4), p. 041701.
[15] Li, Z., Kovachki, N., Choy, C., Li, B., Kossaifi, J., Otta, S., and Nabian, M. A., 2024, "Geometry-Informed Neural Operator for Large-Scale 3D PDEs," Adv. Neural Inf. Process. Syst., **36**, pp. 35836–35854.
[16] Arregui-Mena, J. D., Margetts, L., and Mummery, P. M., 2016, "Practical Application of the Stochastic Finite Element Method," Arch. Comput. Methods Eng., **23**, pp. 171–190.
[17] Acar, E., Bayrak, G., Jung, Y., Lee, I., Ramu, P., and Ravichandran, S. S., 2021, "Modeling, Analysis, and Optimization Under Uncertainties: A Review," Struct. Multidiscipl. Optim., **64**(5), pp. 2909–2945.
[18] Ghanem, R. G., and Spanos, P. D., 2003, *Stochastic Finite Elements: A Spectral Approach*, Courier Corporation, North Chelmsford, MA.
[19] Liu, Y., 2006, *Stochastic Modeling of Multiaxial Fatigue and Fracture*, Vanderbilt University, Nashville, TN.
[20] Wang, Y.-Y., Lou, M., Wang, Y., Wu, W.-G., and Yang, F., 2022, "Stochastic Failure Analysis of Reinforced Thermoplastic Pipes Under Axial Loading and Internal Pressure," China Ocean Eng., **36**(4), pp. 614–628.
[21] Li, Z., Zhou, J., Nassif, H., Coit, D., and Bae, J., 2023, "Fusing Physics-Inferred Information From Stochastic Model With Machine Learning Approaches for Degradation Prediction," Reliab. Eng. Syst. Saf., **232**, p. 109078.
[22] Momeni, E., Poormoosavian, M., Tehrani, H. S., and Fakher, A., 2021, "Reliability Analysis and Risk Assessment of Deep Excavations Using Random-Set Finite Element Method and Event Tree Technique," Transp. Geotech., **29**, p. 100560.
[23] Hsiao, H., and Chiang, K., 2020, "AI-Assisted Reliability Life Prediction Model for Wafer-Level Packaging Using the Random Forest Method," J. Mech., **37**, pp. 28–36.
[24] Haghshenas, A., Hasan, A., Osen, O., and Mikalsen, E. T., 2023, "Predictive Digital Twin for Offshore Wind Farms," Energy Inform., **6**(1), p. 1.
[25] Fu, C., Sinou, J.-J., Zhu, W., Lu, K., and Yang, Y., 2023, "A State-of-the-Art Review on Uncertainty Analysis of Rotor Systems," Mech. Syst. Signal Process., **183**, p. 109619.
[26] Diem, N., and Hien, T., 2025, "Stochastic Finite Element for Stability of Columns Considering the Two-Dimensional Random Field of Elastic Modulus," Results Eng., **25**, p. 104496.
[27] Nguyen, D. D., and Ta, H. D., 2025, "Free Vibration of Beams Using Stochastic Finite Element Method Considering Three-Dimensional Randomness of Material Properties," Forces Mech., **19**, p. 100312.
[28] Huo, H., Xu, W., Wang, W., Chen, G., and Yang, D., 2022, "New Non-intrusive Stochastic Finite Element Method for Plate Structures," Comput. Struct., **268**, p. 106812.
[29] Panther, L., Wagner, W., and Freitag, S., 2025, "A Consistently Linearized Spectral Stochastic Finite Element Formulation for Geometric Nonlinear Composite Shells," Comput. Mech., **75**(5), pp. 1665–1683.
[30] Zakian, P., Khaji, N., and Kaveh, A., 2017, "Graph Theoretical Methods for Efficient Stochastic Finite Element Analysis of Structures," Comput. Struct., **178**, pp. 29–46.
[31] Nath, D., Ankit, Neog, D. R., and Gautam, S. S., 2024, "Application of Machine Learning and Deep Learning in Finite Element Analysis: A Comprehensive Review," Arch. Comput. Methods Eng., **31**(5), pp. 1–40.
[32] Samaddar, A., Ravi, S. K., Ramachandra, N., Luan, L., Madireddy, S., Bhaduri, A., Pandita, P., Sun, C., and Wang, L., 2025, "Data-Efficient Dimensionality Reduction and Surrogate Modeling of High-Dimensional Stress Fields," ASME J. Mech. Des., **147**(3), p. 031701.
[33] Samadian, D., Muhit, I. B., and Dawood, N., 2024, "Application of Data-Driven Surrogate Models in Structural Engineering: A Literature Review," Arch. Comput. Methods Eng., **32**(2), pp. 735–784.
[34] Hornik, K., Stinchcombe, M., and White, H., 1989, "Multilayer Feedforward Networks Are Universal Approximators," Neural Netw., **2**(5), pp. 359–366.
[35] Zhang, Y., Gong, Z., Zhou, W., Zhao, X., Zheng, X., and Yao, W., 2023, "Multi-fidelity Surrogate Modeling for Temperature Field Prediction Using Deep Convolution Neural Network," Eng. Appl. Artif. Intell., **123**, p. 106354.
[36] Lieu, Q. X., Nguyen, K. T., Dang, K. D., Lee, S., Kang, J., and Lee, J., 2022, "An Adaptive Surrogate Model to Structural Reliability Analysis Using Deep Neural Network," Expert Syst. Appl., **189**, p. 116104.
[37] Taghizadeh, M., Nabian, M. A., and Alemazkoor, N., 2024, "Multifidelity Graph Neural Networks for Efficient and Accurate Mesh-Based Partial Differential Equations Surrogate Modeling," Comput. Aided Civ. Infrastruct. Eng, **40**(7), pp. 841–858.
[38] Whalen, E., and Mueller, C., 2022, "Toward Reusable Surrogate Models: Graph-Based Transfer Learning on Trusses," ASME J. Mech. Des., **144**(2), p. 021704.
[39] Nastos, C., Komninos, P., and Zarouchas, D., 2023, "Non-destructive Strength Prediction of Composite Laminates Utilizing Deep Learning and the Stochastic Finite Element Methods," Compos. Struct., **311**, p. 116815.
[40] Martínez, E. R., Chakraborty, S., and Tesfamariam, S., 2021, "Machine Learning Assisted Stochastic-XFEM for Stochastic Crack Propagation and Reliability Analysis," Theor. Appl. Fract. Mech., **112**, p. 102882.
[41] Salazar, F., and Hariri-Ardebili, M. A., 2022, "Coupling Machine Learning and Stochastic Finite Element to Evaluate Heterogeneous Concrete Infrastructure," Eng. Struct., **260**, p. 114190.
[42] Liang, L., Liu, M., Martin, C., and Sun, W., 2018, "A Deep Learning Approach to Estimate Stress Distribution: A Fast and Accurate Surrogate of Finite-Element Analysis," J. R. Soc. Interface, **15**(138), p. 20170844.
[43] Heidari, N., and Iosifidis, A., 2025, "Geometric Deep Learning for Computer-Aided Design: A Survey," IEEE Access, **13**, pp. 119305–119334.
[44] Sirico Jr, A., and Herber, D. R., 2024, "On the Use of Geometric Deep Learning for the Iterative Classification and Down-Selection of Analog Electric Circuits," ASME J. Mech. Des., **146**(5), p. 051703.
[45] Shao, X., Liu, Z., Zhang, S., Zhao, Z., and Hu, C., 2023, "PIGNN-CFD: A Physics-Informed Graph Neural Network for Rapid Predicting Urban Wind Field Defined on Unstructured Mesh," Build. Environ., **232**, p. 110056.
[46] Dalton, D., Husmeier, D., and Gao, H., 2023, "Physics-Informed Graph Neural Network Emulation of Soft-Tissue Mechanics," Comput. Methods Appl. Mech. Eng., **417**, p. 116351.
[47] Perera, R., Guzzetti, D., and Agrawal, V., 2022, "Graph Neural Networks for Simulating Crack Coalescence and Propagation in Brittle Materials," Comput. Methods Appl. Mech. Eng., **395**, p. 115021.
[48] Fortunato, M., Pfaff, T., Wirnsberger, P., Pritzel, A., and Battaglia, P., 2022, "Multiscale Meshgraphnets," preprint arXiv:2210.00612.
[49] Grover, A., and Leskovec, J., 2016, "node2vec: Scalable Feature Learning for Networks," Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, Aug. 13–17, pp. 855–864.
[50] Hamilton, W., Ying, Z., and Leskovec, J., 2017, "Inductive Representation Learning on Large Graphs," Adv. Neural Inf. Process. Syst., **30**.
[51] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y., 2017, "Graph Attention Networks," stat, **1050**, p. 10-48550.
[52] Gulakala, R., Markert, B., and Stoffel, M., 2023, "Graph Neural Network Enhanced Finite Element Modelling," PAMM, **22**(1), p. e202200306.
[53] Heyrani Nobari, A., Rey, J., Kodali, S., Jones, M., and Ahmed, F., 2024, "MeshPointNet: 3D Surface Classification Using Graph Neural Networks and Conformal Predictions on Mesh-Based Representations," ASME J. Mech. Des., **146**(5), p. 051712.
[54] Wang, J., Westermann, R., and Wu, J., 2023, "A Streamline-Guided Dehomogenization Approach for Structural Design," ASME J. Mech. Des., **145**(2), p. 021702.
[55] Hou, W., Li, Y., and Wang, C., 2024, "FrameGraph: A Scalable Performance Evaluation Method for Frame Structure Designs Using Graph Neural Network," ASME J. Mech. Des., **146**(12), p. 121703.
[56] Jin, H., Zhang, E., and Espinosa, H. D., 2023, "Recent Advances and Applications of Machine Learning in Experimental Solid Mechanics: A Review," Appl. Mech. Rev., **75**(6), p. 061001.
[57] Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., and Battaglia, P. W., 2020, "Learning Mesh-Based Simulation With Graph Networks," International Conference on Learning Representations, Addis Ababa, Ethiopia, Apr. 26–30.
[58] Fan, L., Zhao, Q., Shi, J., Lin, F., and Rao, W., 2024, "Learn to Simulate Finite Element Analysis Via Mesh-Based Graph Networks," 2024 27th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Tianjin, China, May 8–10, IEEE, pp. 1073–1078.
[59] Cai, C., Wang, D., and Wang, Y., 2021, "Graph Coarsening With Neural Networks," preprint arXiv:2102.01350.
[60] Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., Hou, T. Y., and Tegmark, M., 2025, "KAN: Kolmogorov–Arnold Networks," The Thirteenth International Conference on Learning Representations, Singapore, Apr. 24–28.
[61] Zhang, F., and Zhang, X., 2024, "GraphKAN: Enhancing Feature Extraction With Graph Kolmogorov Arnold Networks," preprint arXiv:2406.13597.
[62] Kaewnuratchadasorn, C., Wang, J., and Kim, C.-W., 2024, "Physics-Informed Neural Operator Solver and Super-Resolution for Solid Mechanics," Comput. Aided Civ. Infrastructure Eng., **39**(22), pp. 3435–3451.

[63] Azizzadenesheli, K., Kovachki, N., Li, Z., Liu-Schiaffini, M., Kossaifi, J., and Anandkumar, A., 2024, "Neural Operators for Accelerating Scientific Simulations and Design," Nat. Rev. Phys., **6**(5), pp. 320–328.
[64] Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A., 2020, "Fourier Neural Operator for Parametric Partial Differential Equations, preprint arXiv:2010.08895.
[65] Koolman, E., Clay, J. Z., Li, X., Jiang, R., Goldstein, M. H., Xie, C., Demirel, H. O., and Sha, Z., 2024, "A Multi-case Study of Traditional, Parametric, and Generative Design Thinking of Engineering Students," International Conference on-Design Computing and Cognition, Montreal, Canada, July 8–10, Springer, pp. 89–104.
[66] Aranburu, A., Justel, D., Contero, M., and Camba, J. D., 2022, "Geometric Variability in Parametric 3D Models: Implications for Engineering Design," Procedia CIRP, **109**, pp. 383–388.
[67] Li, Z., Huang, D. Z., Liu, B., and Anandkumar, A., 2023, "Fourier Neural Operator With Learned Deformations for PDEs on General Geometries," J. Mach. Learn. Res., **24**(388), pp. 1–26.
[68] Wu, H., Luo, H., Wang, H., Wang, J., and Long, M., 2024, "Transolver: A Fast Transformer Solver for PDEs on General Geometries," Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria, July 21–27.
[69] Hao, Z., Wang, Z., Su, H., Ying, C., Dong, Y., Liu, S., Cheng, Z., Song, J., and Zhu, J., 2023, "GNOT: A General Neural Operator Transformer for Operator Learning," International Conference on Machine Learning, Honolulu, HI, July 23–29, PMLR, pp. 12556–12569.
[70] Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A., 2020, "Neural Operator: Graph Kernel Network for Partial Differential Equations," preprint arXiv:2003.03485.
[71] Cohen, S. N., Jiang, D., and Sirignano, J., 2023, "Neural Q-Learning for Solving PDEs," J. Mach. Learn. Res., **24**(1), pp. 1–49.
[72] Djojodihardjo, H., 2023, "Introduction to Aircraft Loads," *Introduction to Aeroelasticity: With Case-Studies*, Springer, Singapoe, pp. 591–619.
[73] Yau, Y. H., Hua, S. N., and Kok, C. K., 2018, "Structural Failure Analysis of Polycarbonate Enclosures of Electronic Devices Subjected to Multiple Ball Impacts," Polym. Test., **65**, pp. 374–386.
[74] Shaikh, S. A., Cherukuri, H., Balusu, K., Devanathan, R., and Soulami, A., 2024, "Probabilistic Surrogate Model for Accelerating the Design of Electric Vehicle Battery Enclosures for Crash Performance," preprint arXiv:2408.03450.
[75] Gharaibeh, M. A., and Pitarresi, J. M., 2022, "A Methodology to Calculate the Equivalent Static Loading for Simulating Electronic Assemblies Under Impact," Microelectron. Reliab., **139**, p. 114842.
[76] Xu, X., Lambourne, J., Jayaraman, P., Wang, Z., Willis, K., and Furukawa, Y., 2024, "Brepgen: A B-Rep Generative Diffusion Model With Structured Latent Geometry," ACM Trans. Graph., **43**(4), pp. 1–14.
[77] Willis, K. D., Pu, Y., Luo, J., Chu, H., Du, T., Lambourne, J. G., Solar-Lezama, A., and Matusik, W., 2021, "Fusion 360 Gallery: A Dataset and Environment for Programmatic CAD Construction From Human Design Sequences," ACM Trans. Graph., **40**(4), pp. 1–24.
[78] Rebay, S., 1993, "Efficient Unstructured Mesh Generation by Means of Delaunay Triangulation and Bowyer-Watson Algorithm," J. Comput. Phys., **106**(1), pp. 125–138.
[79] Marot, C., Pellerin, J., and Remacle, J.-F., 2019, "One Machine, One Minute, Three Billion Tetrahedra," Int. J. Numer. Methods Eng., **117**(9), pp. 967–990.
[80] Schöberl, J., 1997, "NETGEN an Advancing Front 2D/3D-Mesh Generator Based on Abstract Rules," Comput. Vis. Sci., **1**(1), pp. 41–52.
[81] Brix Nerenst, T., Ebro, M., Nielsen, M., Eifler, T., and Nielsen, K. L., 2019, "Barriers for Virtual Assessment of Structural Robustness," ASME International Mechanical Engineering Congress and Exposition, Salt Lake City, UT, Nov. 11–14.
[82] Logg, A., Mardal, K.-A., and Wells, G., 2012, *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*, Springer Science & Business Media, Vol. 84.
[83] Sudret, B., and Der Kiureghian, A., 2000, *Stochastic Finite Element Methods and Reliability: A State-of-the-Art Report*, Department of Civil and Environmental Engineering, University of California, Berkeley, CA.
[84] Hong, S., Kwon, Y., Shin, D., Park, J., and Kang, N., 2025, "DeepJEB: 3D Deep Learning-Based Synthetic Jet Engine Bracket Dataset," ASME J. Mech. Des., **147**(4), p. 041703.
[85] Ribeiro, B. A., Ribeiro, J. A., Ahmed, F., Penedones, H., Belinha, J., Sarmento, L., Bessa, M. A., and Tavares, S., 2023, "SimuStruct: Simulated Structural Plate With Holes Dataset With Machine Learning Applications," Workshop on ''Machine Learning for Materials'' ICLR, Kigali, Rwanda, May 1–5.
[86] Zhao, Y., Li, H., Zhou, H., Attar, H. R., Pfaff, T., and Li, N., 2024, "A Review of Graph Neural Network Applications in Mechanics-Related Domains," Artif. Intell. Rev., **57**(11), p. 315.
[87] Maurizi, M., Gao, C., and Berto, F., 2022, "Predicting Stress, Strain and Deformation Fields in Materials and Structures With Graph Neural Networks," Sci. Rep., **12**(1), p. 21834.
[88] Yang, H., Li, Z., Wang, X., and Wang, J., 2024, "An Implicit Factorized Transformer With Applications to Fast Prediction of Three-Dimensional Turbulence," Theor. Appl. Mech. Lett., **14**(6), p. 100527.
[89] Mangado, N., Piella, G., Noailly, J., Pons-Prats, J., and Ballester, M. Á. G., 2016, "Analysis of Uncertainty and Variability in Finite Element Computational Models for Biomedical Engineering: Characterization and Propagation," Front. Bioeng. Biotechnol., **4**, p. 85.
[90] Li, L., Chang, J., Vakanski, A., Wang, Y., Yao, T., and Xian, M., 2024, "Uncertainty Quantification in Multivariable Regression for Material Property Prediction With Bayesian Neural Networks," Sci. Rep., **14**(1), p. 10543.
[91] Yang, Y., Kissas, G., and Perdikaris, P., 2022, "Scalable Uncertainty Quantification for Deep Operator Networks Using Randomized Priors," Comput. Methods Appl. Mech. Eng., **399**, p. 115399.
[92] Hüllermeier, E., and Waegeman, W., 2021, "Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods," Mach. Learn., **110**(3), pp. 457–506.
[93] Mucsányi, B., Kirchhof, M., and Oh, S. J., 2024, "Benchmarking Uncertainty Disentanglement: Specialized Uncertainties for Specialized Tasks," Adv. Neural Inform. Process. Syst., **37**, pp. 50972–51038.
[94] Shaikh, S. A., 2024, "Machine Learning-Based Approaches for Forward and Inverse Problems in Engineering Design," Ph.D. thesis, The University of North Carolina at Charlotte.
[95] Franco, N. R., Manzoni, A., and Zunino, P., 2023, "Mesh-Informed Neural Networks for Operator Learning in Finite Element Spaces," J. Sci. Comput., **97**(2), p. 35.
[96] Block, P., Boller, G., DeWolf, C., Pauli, J., and Kaufmann, W., 2024, "Fast Prediction of Stress Distribution: A GNN-based surrogate model for unstructured mesh FEA," Proceedings of the IASS 2024 Symposium: Redefining the Art of Structural Design, Zurich, Switzerland, Aug. 26–30.
[97] Chen, Y., Shi, J., He, R., Lu, C., Shi, P., Feng, J., and Sareh, P., 2023, "A Unified Inverse Design and Optimization Workflow for the Miura-oRing Metastructure," ASME J. Mech. Des., **145**(9), p. 091704.
[98] Sareh, P., Chermprayong, P., Emmanuelli, M., Nadeem, H., and Kovac, M., 2018, "Rotorigami: A Rotary Origami Protective System for Robotic Rotorcraft," Sci. Robot., **3**(22), p. eaah5228.
[99] Sawyer, S. F., 2009, "Analysis of Variance: The Fundamental Concepts," J. Man. Manip. Ther., **17**(2), pp. 27E–38E.
[100] Sainani, K. L., 2012, "Dealing With Non-normal Data," Pm&r, **4**(12), pp. 1001–1005.
[101] Kim, Y. J., and Cribbie, R. A., 2018, "ANOVA and the Variance Homogeneity Assumption: Exploring a Better Gatekeeper," Br. J. Math. Stat. Psychol., **71**(1), pp. 1–12.
[102] Wilcox, R. R., 1997, "Some Practical Reasons for Reconsidering the Kolmogorov–Smirnov Test," Br. J. Math. Stat. Psychol., **50**(1), pp. 9–20.