Supplementary Information: Surface-Based Stress Tomography of Architected Metamaterials via Physics-Constrained Generative Learning

Donggeun Park[a+], Minwoo Park[a+], Junhee Cho[a+], Seunghwa Ryu[a,*]

Affiliations

[a] Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea

[+] These authors contributed equally: Donggeun Park, Minwoo Park, Junhee Cho[a]

[*]Corresponding author e-mail: ryush@kaist.ac.kr

Supplementary Note 1. GRF-based spinodoid generation and dataset construction

Spinodoid geometries in this study are generated using a Gaussian random field (GRF) representation as a computationally efficient surrogate to spinodal decomposition concepts (**Fig. S1**). The GRF scalar field $\phi(\mathbf{x})$ is evaluated on a regular three-dimensional grid according to

$$\phi(\mathbf{x}) = \sqrt{\frac{2}{N_w}} \sum_{i=1}^{N_w} cos\,(k\,\mathbf{n}_i \cdot \mathbf{x} + \gamma_i),$$

where N_w denotes the number of wave components, $k = 10\pi/L$ is the wave number with domain size $L = 12$, $\mathbf{n}_i \in \mathbb{R}^3$ are unit wave-direction vectors, and $\gamma_i \in [0,2\pi]$ are phase shifts. Binary spinodoid topology is obtained by level-set thresholding,

$$X(\mathbf{x}) = \mathbb{H}(\phi(\mathbf{x}) - u_c),$$

where $\mathbb{H}$ is the Heaviside function and u_c is a cutoff selected by a density-matching procedure.

We adopt a global–local parameterization of morphology. Global anisotropy is controlled through orientation constraints specified by $\theta \in \{15°, 30°, 45°, 60°, 90°\}$. Local morphology is governed by the high-dimensional sets $\{\mathbf{n}_i\}_{i=1}^{N_w}$ and $\{\gamma_i\}_{i=1}^{N_w}$ (and implicitly by k), which modulate fine-scale patterns under a fixed global setting. Wave directions are sampled from a standard normal distribution and normalized, and candidates are accepted if they satisfy an angular rule with respect to coordinate axes:

$$\min\,(\angle(\mathbf{n}_i, \pm\mathbf{e}_j)) < \theta_j \text{ for at least one axis } j \in \{1,2,3\},$$

where $\mathbf{e}_j$ are canonical basis vectors. This provides a simple, controllable mechanism to bias wave-direction sampling and generate lamellar/columnar/cubic-to-hybrid-like anisotropic responses through θ.

To satisfy a target relative density ρ, the cutoff $u_c$is computed by binary search. Starting from $u_{\min} = \min(\phi)$and $u_{\max} = \max(\phi)$, the midpoint $u_c = (u_{\min} + u_{\max})/2$is iteratively updated, and the induced density is evaluated as

$$\rho(u_c) = \langle \mathbb{H}(\phi(\mathbf{x}) - u_c) \rangle.$$

Bounds are updated until $| \rho(u_c) - \rho | <$ tol. In the implementation, tol $= 10^{-3}$and the maximum number of iterations is 1000. If the density target is not met within tolerance, the median of $\phi$is used as a fallback threshold to ensure robust binarization.

To ensure physically meaningful bicontinuous architectures, we enforce single-component continuity of the solid phase using 3D 6-connectivity labeling. Structures with more than one disconnected solid component are rejected (**Fig. S1**). For each θ-group, 800 realizations are generated. FE outputs are used to compute an effective stiffness metric (estimated from the corresponding von Mises stress response, consistent with the manuscript). Samples are ranked within each θ-group, and the top 20% highest-stiffness configurations are reserved for testing to focus evaluation on the upper stiffness regime, while the remainder is used for training and validation.

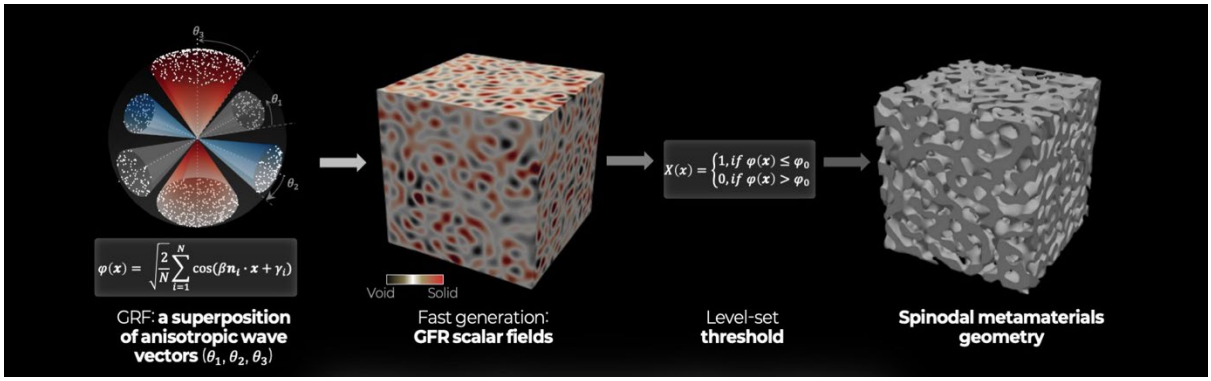


Fig. S1. GRF-based spinodoid generation pipeline (Cahn–Hilliard concept → GRF scalar field → level-set thresholding → spinodoid geometry) and continuity filtering.

Supplementary Note 2 PDG3Net architecture details and training modules

PDG^{3}Net takes six-face surface images as inputs and reconstructs (i) six face-wise surface stress fields and (ii) a volumetric stress tensor field. Each face image is processed by a shared surface-generator backbone (weight sharing across faces), producing a face-wise latent representation. These six latent tensors are aggregated into a 3D latent volume, which is subsequently decoded by a 3D volume generator to produce volumetric stress components. This design explicitly implements a staged mapping from surface geometry to intermediate surface stress cues and finally to volumetric stress fields, reducing the burden of directly regressing full 3D tensors from surface images alone.

For each face $m \in \{1, \dots, 6\}$, the input is a binary image $I^{(m)} \in \mathbb{R}^{H \times W}$(here $H = W = 64$). The surface generator uses a multi-kernel encoder–decoder with kernels (3,6,9) to capture multi-scale geometric cues (local curvature/connectivity, mid-scale motifs, and global orientation signatures). Outputs are (a) a face-wise surface stress map $\hat{s}^{(m)} \in \mathbb{R}^{H \times W \times C_s}$and (b) a face-wise latent feature tensor $\mathbf{z}^{(m)} \in \mathbb{R}^{h \times w \times C_z}$, where h, w denote the latent spatial resolution.

Face-wise latent tensors $\{\mathbf{z}^{(m)}\}_{m=1}^{6}$are fused by concatenation and a learnable fusion block to form a unified latent tensor $\mathbf{z}_{2D} \in \mathbb{R}^{h \times w \times C_f}$. This fused representation is reshaped (and/or projected via an MLP) into a 3D latent volume $\mathbf{Z}_{3D} \in \mathbb{R}^{d \times h \times w \times C_{3D}}$, which is decoded by a 3D convolutional decoder with upsampling to produce the volumetric stress tensor $\hat{\sigma} \in \mathbb{R}^{D \times H \times W \times 6}$. The output channels correspond to $(\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{xz}, \sigma_{yz}$.

A conditional discriminator (cGAN) is trained to distinguish FE-derived volumetric stress tensors from generated tensors conditioned on the same surface input, improving distributional realism beyond voxel-wise matching. In addition, a pretrained representation model is used for topology-aware feature-space regularization. Specifically, an encoder $E(\cdot)$is pretrained on unlabeled spinodoid geometries to learn structural signatures shared across samples. During PDG^{3}Net training, feature embeddings $E(\hat{\sigma})$ and $E(\sigma^{FE})$ are compared to compute $\mathcal{L}_{feat}$, and selected decoder components are frozen to stabilize feature extraction (**Fig. S2**).

To obtain geometry-aware structural representations without additional labeling cost, we generated 10,000 unlabeled spinodoid geometries within the same GRF parameterization and pretrained the encoder $E(\cdot)$using a reconstruction-based objective (autoencoder) and/or contrastive feature learning. Because this pretraining uses only geometry (no FE stress labels), it provides a cost-effective mechanism to encode bicontinuous structural continuity for downstream stress reconstruction. Details of the hyperparameters are provided below (and summarized in **Table. S1**).

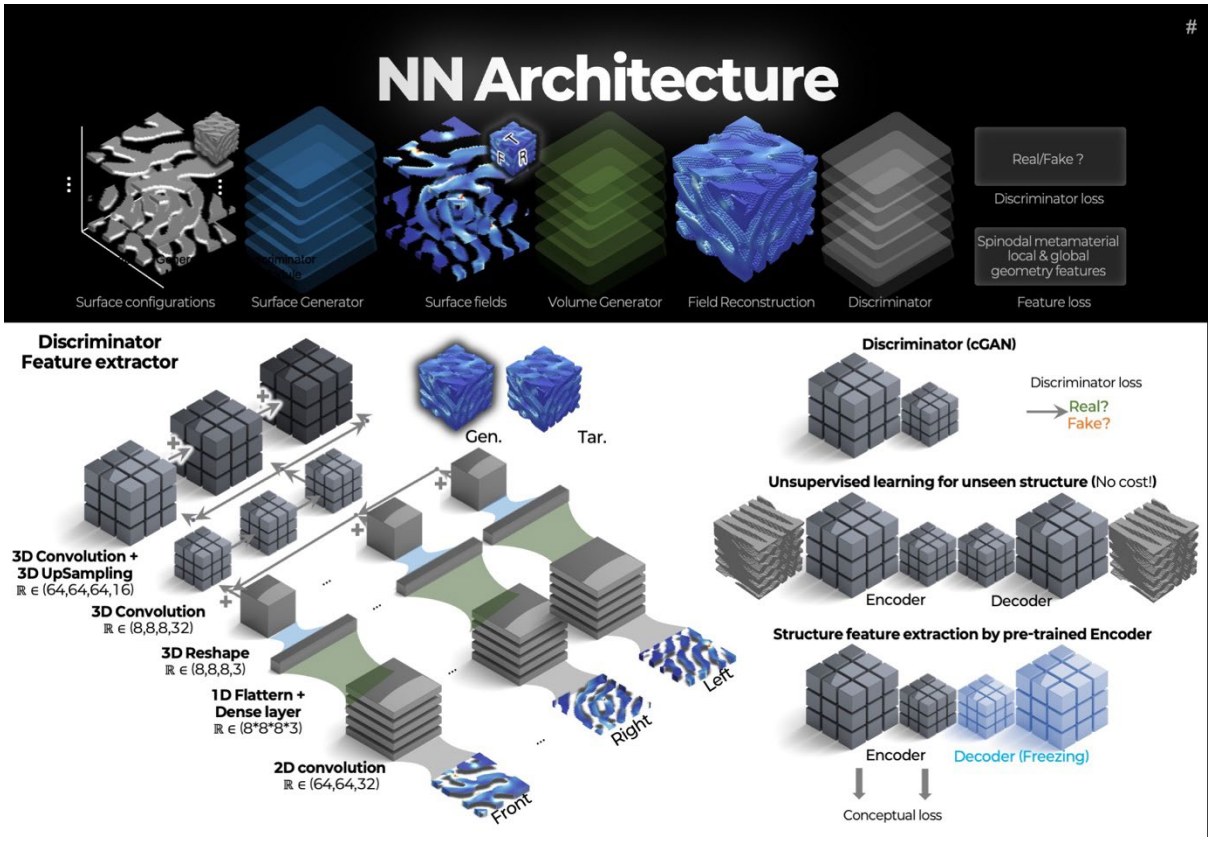


Fig. S2. Multi-kernel surface generator: small/mid/large receptive fields and mapping from surface configurations to surface stress targets. Volume generator: fusion of six surface stress fields and 3D convolutional decoding to volumetric stress tensors. cGAN discriminator and pretrained encoder-based feature-space regularization, including decoder freezing strategy.

Table S1. Detailed PDG3Net architecture configuration (kernel sizes, feature maps, tensor shapes).

Layer	Module	Input	Output	Notes
0	Six-face input	6 × (64×64×1)	6 × (64×64×1)	Six-face binary surface images
1	Surface encoder (multi-kernel fusion)	(64×64×1)	(64×64×32)	3×3, 6×6, 9×9 branches fused to 32ch
2	Down-1	(64×64×32)	(32×32×64)	Stride-2 conv
3	Down-2	(32×32×64)	(16×16×128)	Stride-2 conv
4	Down-3 (bottleneck entry)	(16×16×128)	(8×8×256)	Stride-2 conv
5	Bottleneck blocks	(8×8×256)	(8×8×256)	Residual/conv blocks

6	Up-1	(8×8×256)	(16×16×128)	Upsample ×2 + conv
7	Up-2	(16×16×128)	(32×32×64)	Upsample ×2 + conv
8	Up-3	(32×32×64)	(64×64×32)	Upsample ×2 + conv
9	Surface stress head	(64×64×32)	(64×64×1)	Face-wise surface **von Mises** stress map
10	Face latent head	(8×8×256)	(8×8×128)	Per-face latent used for volumetric lifting
11	Multi-view fusion	6 × (8×8×128)	(8×8×512)	1×1 conv 512 bottleneck
12	2D→3D lift (projection)	(8×8×512)	(8×8×8×128)	Project to 3D latent volume (depth=8)
13	3D decoder up-1	(8×8×8×128)	(16×16×16×128)	3D upsample ×2
14	3D decoder up-2	(16×16×16×128)	(32×32×32×64)	3D upsample ×2
15	3D decoder up-3	(32×32×32×64)	(64×64×64×32)	3D upsample ×2
16	Volumetric stress head	(64×64×64×32)	(64×64×64×6)	Six stress components
17	Discriminator (cGAN)	cond: surface + $(64^3\times6)$	real/fake	Conditional 3D discriminator (patch-based)
18	Feature encoder (pretrained)	$(64^3\times6)$	embedding	pretrained on 10k unlabeled geometries

Table S2. Detailed PDG3Net hyperparameters.

Category	Item	Setting
Dataset	Global orientation groups	($\theta \in \{15\circ, 30\circ, 45\circ, 60\circ, 90\circ\}$)
Dataset	Samples per θ-group	800
Dataset	Test split	Top 20% highest-stiffness samples by K_{eff}) within each θ-group
Dataset	Train/validation split	Remaining 80% randomly shuffled and split into 70% training and 10% validation
Training	Random seed	77
Training	Optimizer	Adam
Training	Adam parameters	β_1=0.9, β_2=0.999, $\epsilon = 1\times10^{-7}$
Training	Initial learning rate	1×10^{-4}
Training	Learning-rate schedule	Learning-rate decay from 1×10^{-4} during training
Training	Batch size	8

Training	Epochs	1000
Training	Early stopping	Used
Training	Weight initialization	He initialization
Training	Batch normalization	Used
Loss weights	cGAN loss weight	0.7
Loss weights	Physics-constraint loss weight	0.3
Loss weights	Shape/topology-constraint loss weight	1
Hardware	GPU	3 × NVIDIA GeForce RTX 4090-class GPUs, 24 GB memory per GPU
Hardware	Multi-GPU setting	TensorFlow/Keras data-parallel training
Runtime	Approximate training time	~40 h for 1000 epochs
Runtime	Approximate inference time	~0.5 s per sample

Supplementary Note 3. Loss function details and training strategy (including ablation logic)

Training is performed using a unified multi-objective loss:

$$\mathcal{L}_{total} = \lambda_{adv}(\mathcal{L}_{gen} + \mathcal{L}_{dis}) + \lambda_{feat}\mathcal{L}_{feat} + \lambda_{phys}\mathcal{L}_{phys}.$$

Here $\mathcal{L}_{gen}$ denotes voxel-wise reconstruction loss (MSE) between predicted and FE-derived volumetric stress tensors. The adversarial term $\mathcal{L}_{\mathrm{dis}}$ corresponds to the conditional GAN objective, encouraging generated stress fields to match the target distribution conditioned on the same surface input. The topology-aware feature-space loss $\mathcal{L}_{feat}$ penalizes deviations in the pretrained encoder's latent representations between generated and target fields, thereby preserving bicontinuous structural signatures. The physics residual term $\mathcal{L}_{phys}$ enforces static equilibrium through the residual $\nabla \cdot \sigma + \mathbf{f} = 0$ (with $\mathbf{f} = 0$ in the present setting).

The divergence $\nabla \cdot \sigma$is computed on a structured grid using central finite differences. For example,

$$\partial_x \sigma_{xx}(i,j,k) \approx \frac{\sigma_{xx}(i+1,j,k) - \sigma_{xx}(i-1,j,k)}{2\Delta x},$$

and similarly for other components. The residual vector is evaluated as

$$\mathbf{r} = \begin{bmatrix} \partial_x \sigma_{xx} + \partial_y \sigma_{xy} + \partial_z \sigma_{xz} \\ \partial_x \sigma_{yx} + \partial_y \sigma_{yy} + \partial_z \sigma_{yz} \\ \partial_x \sigma_{zx} + \partial_y \sigma_{zy} + \partial_z \sigma_{zz} \end{bmatrix},$$

and the physics loss is minimized as

$$\mathcal{L}_{phys} = \langle \| \mathbf{r} \|^2 \rangle$$

over the volumetric domain.

To interpret the role of each constraint, we consider staged objectives consistent with **Fig. S3**. Starting from reconstruction-only training ($\mathcal{L}_G$), we add distribution matching via the discriminator ($\mathcal{L}_G + \mathcal{L}_D$), then incorporate feature-space regularization ($\mathcal{L}_G + \mathcal{L}_D + \mathcal{L}_C$), and finally include the equilibrium constraint $\mathcal{L}_{phys}$. **Fig S3** summarizes qualitative improvements in continuity and stress localization as objectives are added, providing ablation logic for the architecture and training strategy.

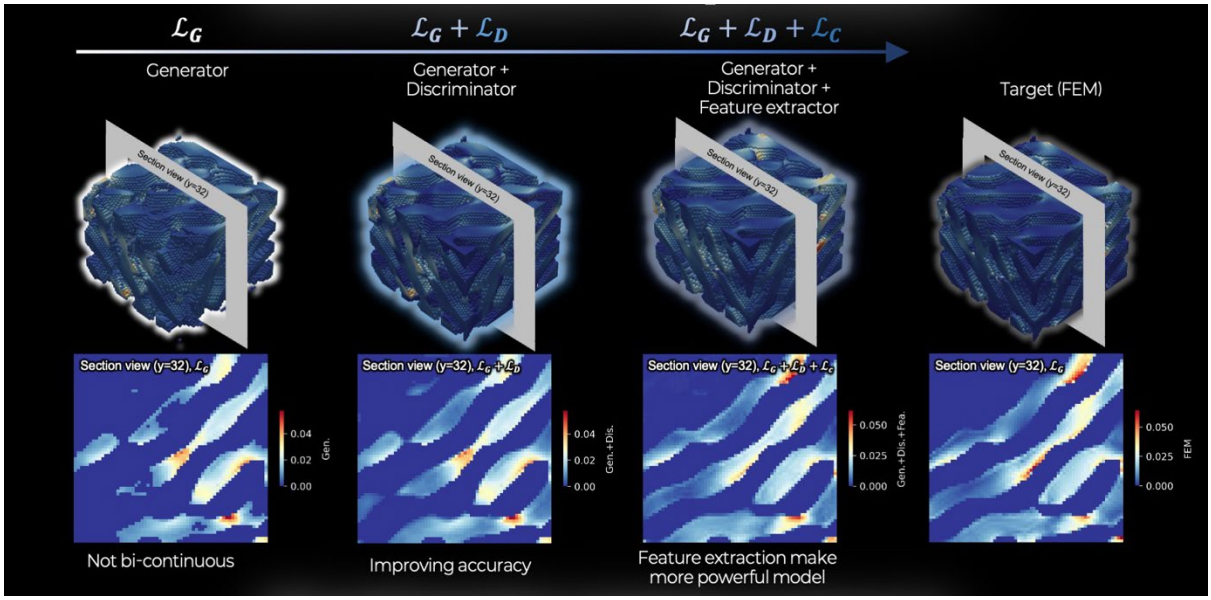


Fig. S3. Stage-wise objective ablation used for architecture optimization ($\mathcal{L}_G$, $\mathcal{L}_G + \mathcal{L}_D$, $\mathcal{L}_G + \mathcal{L}_D + \mathcal{L}_C$, + physics).

Supplementary Note 4. Genetic algorithm (GA) optimization: full workflow, encoding, and hyperparameters

This note corresponds to the "Optimization Framework with PDG3Net" slides and the implementation code used for stress-field-driven design exploration. Each GA individual is encoded by a global anisotropy constraint (θ setting) and local GRF descriptors, namely the wave directions and phases $\{\mathbf{n}_i\}_{i=1}^{N_w}$ and $\{\gamma_i\}_{i=1}^{N_w}$ with $N_w = 1000$. For each decoded geometry, the pretrained surrogate predicts volumetric stress fields, and fitness is computed from a direction-wise stress-field homogenization objective consistent with the paper:

$$J = \frac{1}{M}\sum_{k=1}^{M} \frac{1}{V}\int_{\Omega} \parallel \sigma^{(k)}(\mathbf{x}) \parallel \, dV,$$

where M is the number of loading directions, V is the volume, and Ω denotes the volumetric domain. The GA minimizes J to identify parameter sets producing more uniformly distributed stress fields across directions. If a specific run uses a single representative direction, this is stated explicitly in the manuscript, and multi-direction evaluation is reported in the main experiments.

The GA uses selection of the top 50% parents (with tournament selection as an optional variant), multi-point crossover with $K = 3$ crossover points applied to segments of the wave arrays, and mutation with probability $m = 0.1$. In mutation, one wave direction is replaced by a newly sampled unit vector satisfying the θ constraint, and the corresponding phase is resampled uniformly from $[0,2\pi]$. The population size is $P = 30$, and the generational loop is repeated until convergence or a maximum generation count as used in the manuscript. **Fig. S4**

summarizes the full optimization algorithmic flow and illustrates the genetic operators (selection, crossover, mutation) in detail.

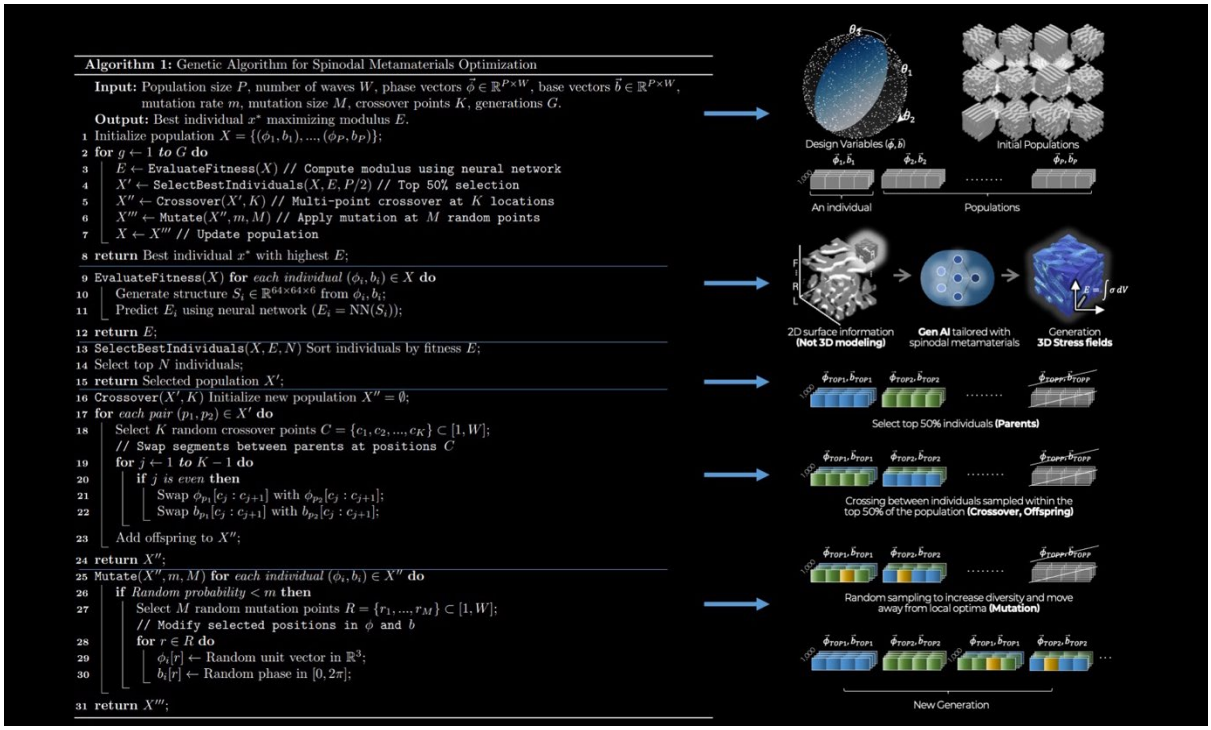


Fig. S4. Genetic algorithm optimization workflow: fitness evaluation → parent selection → multi-point crossover → mutation → next generation.

Supplementary Note 5. FEM validation: elastic surface computation (directional modulus distribution)

This note corresponds to the “FEM Validation (3D Elastic Surface)” slide set. The validation pipeline converts voxelized geometries to ABAQUS input files, performs a set of elastic simulations under varied boundary conditions/loading directions, extracts the stiffness tensor and/or directional modulus distribution from stress–strain responses, and visualizes the resulting elastic surface in Python. **Fig .S5** summarizes the full pipeline from geometry generation to stiffness extraction and visualization.

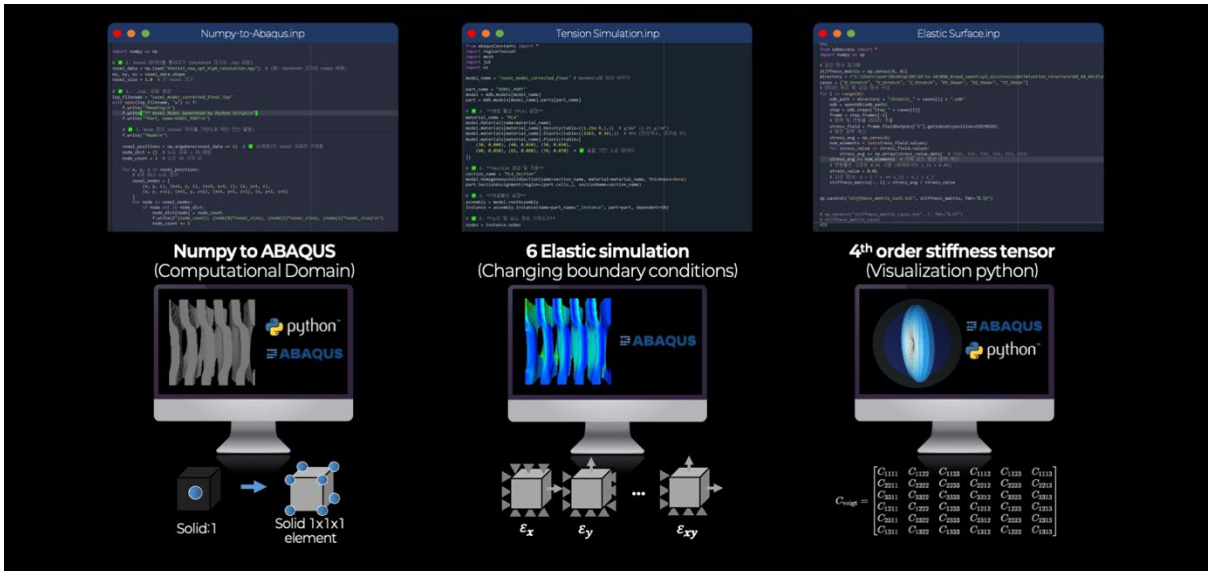


Fig. S5. FEM validation pipeline for 3D elastic surface computation (voxel→Abaqus→elastic simulations→stiffness extraction→visualization

Supplementary Note 6. Additive manufacturing (FDM) and mechanical testing: detailed procedures and raw data

To experimentally validate the designed and optimized spinodoid architectures, test specimens were fabricated using a material extrusion–based additive manufacturing process, specifically fused deposition modeling (FDM). FDM was selected because it can reliably produce complex three-dimensional architected geometries with relatively low setup overhead, and because it allows consistent reproduction of internal porous networks without extensive post-processing when printing parameters are carefully controlled.

All specimens were printed using commercially available PLA filament (Prusament, Czech Republic). Printing was performed using PrusaSlicer (v2.9.1), and all printer configurations were held constant across specimens to ensure comparability. Because the geometries include complex internal networks, the tool-path strategy was adjusted such that material deposition follows external boundaries to promote smooth and continuous external surfaces and to reduce defects associated with material accumulation in locally confined regions. To ensure internal uniformity, a rectangular infill pattern was used with an infill ratio of 99%. To reduce potential anisotropy induced by layerwise deposition, raster orientation was alternated by 90° between successive layers. Specimens were fabricated under controlled ambient conditions (25 °C, 40% relative humidity).

Each specimen was designed as a cube with an edge length of 50 mm and a target relative density of 40%. The build direction corresponds to the global z-axis in the mechanical testing coordinate system. To confirm fabrication consistency relative to design specifications, relative density was evaluated for each specimen. The bulk density of the printed structure was computed by dividing the oven-dried specimen mass by the geometric volume obtained from

the supplied STL file. The effective bulk density of printed PLA structures was measured as 1.15 g/cm^3 using this procedure, and the results were consistent across specimens, supporting reproducibility of the fabrication process.

In this study, Young's modulus was evaluated through uniaxial compression testing conducted at room temperature. All experiments were performed using a universal testing machine (Instron 5583, Instron, USA) equipped with a 100 kN load cell. Compression was applied under displacement-controlled conditions at a nominal strain rate of 1% min^{-1}, corresponding to quasi-static loading. Data were recorded at a frequency of 50 Hz.

To account for potential manufacturing imperfections on the contacting surfaces, each specimen was subjected to three loading–unloading cycles within the elastic regime up to a compressive strain of 0.1% prior to the main measurements. Following this pre-conditioning procedure, the specimens were monotonically compressed until densification occurred, and the tests were terminated when the displacement reached 70% of the initial specimen height. In addition, the printed PLA material properties were specified based on supplier-reported values and supplementary compression testing. According to the Prusament PLA technical datasheet, the tensile modulus of printed specimens is (2.3 ± 0.1 GPa) for the horizontal print direction and (2.4 ± 0.1 GPa) for the vertical xz print direction. In the present study, compression testing followed ASTM D695 and resulted in a Young's modulus of (1630 ± 21 MPa).

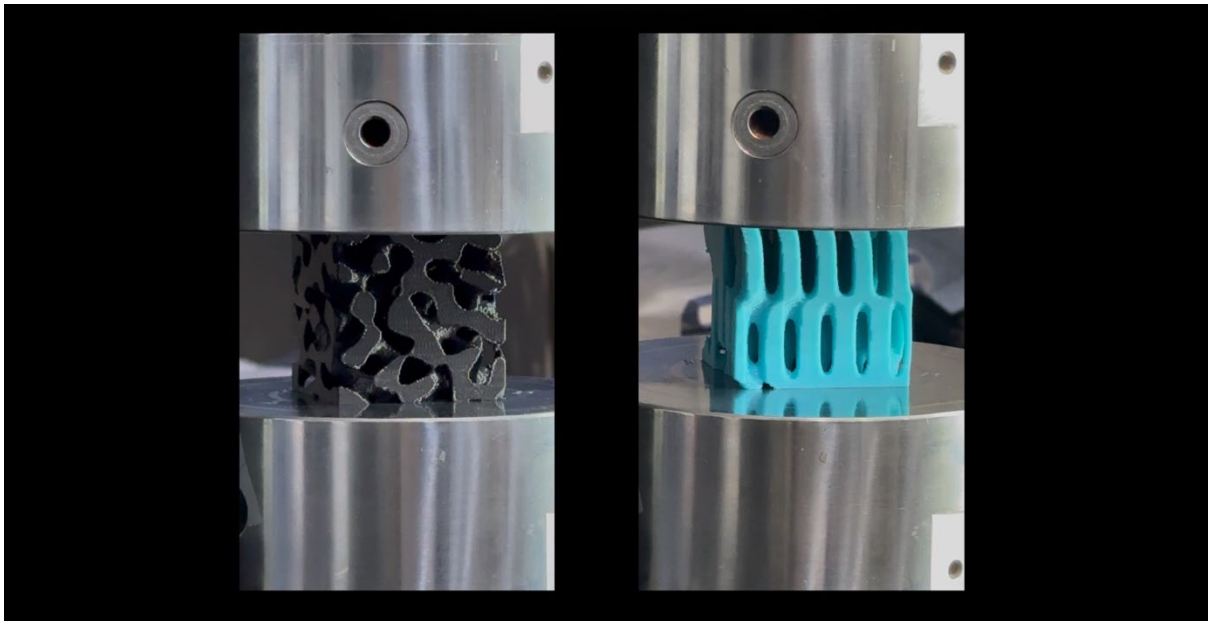


Fig S6. The fabricated sample and mechanical compression test

Supplementary Note S7. Training and quantitative verification of the intermediate surface-stress representation

The intermediate surface-stress representation was introduced to provide mechanically informative multi-view anchors for volumetric stress reconstruction. For each spinodoid specimen and prescribed loading direction, the FE-derived volumetric stress tensor field was first converted into a von Mises stress field. The corresponding six external face maps were then extracted from the volumetric von Mises stress field and used as face-wise surface-stress targets.

The surface generator was trained to map each binary surface image to its corresponding FE-derived surface von Mises stress map. Thus, the intermediate learning task was formulated as a supervised image-to-surface-stress mapping problem for each of the six external faces. The predicted six-face surface-stress maps were subsequently fused by the volume generator and lifted into the volumetric domain for internal stress reconstruction.

The data split followed the same upper-stiffness extrapolation protocol used in the main reconstruction task. Within each global orientation group of GRF, $\theta \in \{15^\circ, 30^\circ, 45^\circ, 60^\circ, 90^\circ\}$, samples were ranked by the effective stiffness metric (K_{eff}). The top 20% highest-stiffness samples in each θ-group were reserved as the extrapolation test set. The remaining 80% were randomly shuffled using a fixed random seed of 77 and split into 70% for training and 10% for validation.

To quantify the prediction accuracy of the surface-stress intermediate representation, the predicted surface-stress maps were compared with the FE-derived surface-stress targets on

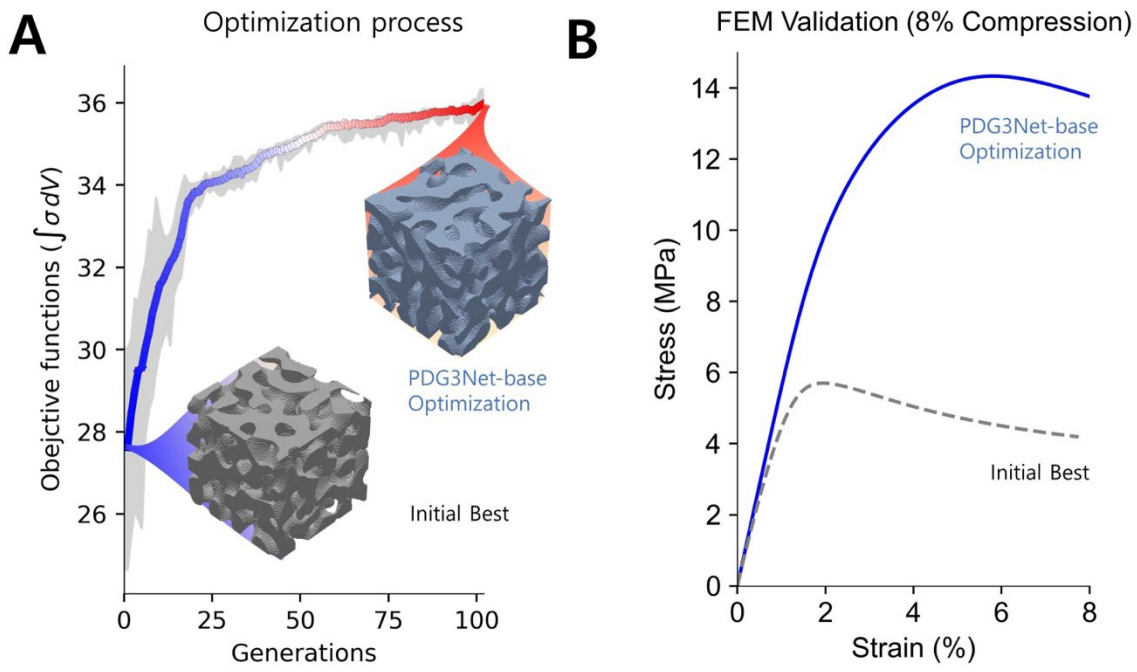
the held-out top-20% test set. Evaluation was performed separately for each external face using R^2. As summarized in **Supplementary Table S3**, the R^2 values range from 0.993 to 0.997 across the six faces, indicating that the surface generator accurately captures the face-wise surface-stress distributions. These results support the use of the predicted surface-stress maps as structured mechanical anchors for the subsequent surface-to-volume stress reconstruction.

Table S3. Quantitative verification of the intermediate six-face surface-stress prediction. (R^2) was computed between the predicted and FE-derived surface von Mises stress maps for each external face on the held-out upper-stiffness test set.

Metric	(+x)	(-x)	(+y)	(-y)	(+z)	(-z)	Mean
R^2	0.997	0.996	0.995	0.996	0.994	0.993	0.995

Supplementary Note S8. Target-density inverse optimization at relative density 0.5

To clarify how the proposed framework can be applied to inverse design with prescribed geometric constraints, we performed an additional density-constrained optimization case with a target relative density of 0.5. Candidate geometries were generated in the same global–local GRF parameter space, while the target density was enforced during the thresholding step. The optimization objective was evaluated using the PDG^{3}Net-predicted volumetric stress field. As shown in **Fig. S7A**, the objective function improved progressively over the course of the genetic algorithm. The final optimized design was then re-evaluated using FEM under compression. As shown in **Fig. S7B**, the optimized structure exhibited a substantially improved compressive response compared with the initial best design. This example demonstrates that the proposed PDG3Net-assisted framework can be extended to inverse optimization problems with prescribed relative-density constraints.



Supplementary Fig. S7. Target-density inverse optimization under a prescribed relative density of 0.5. (A) Optimization history of the PDG3Net-assisted genetic algorithm under the density-constrained design setting. The objective function increases progressively across generations, and representative structures corresponding to the initial best design and the final optimized design are shown. (B) FEM validation under 8% compression for the initial best and PDG3Net-based optimized structures. The optimized structure exhibits a markedly improved compressive response, confirming the effectiveness of the density-constrained inverse optimization framework.