FOURIER-LATENT DIFFUSION FOR CONSTRAINED GENERATION OF TRIPLY PERIODIC MINIMAL SURFACES

Anonymous authors
Paper under double-blind review

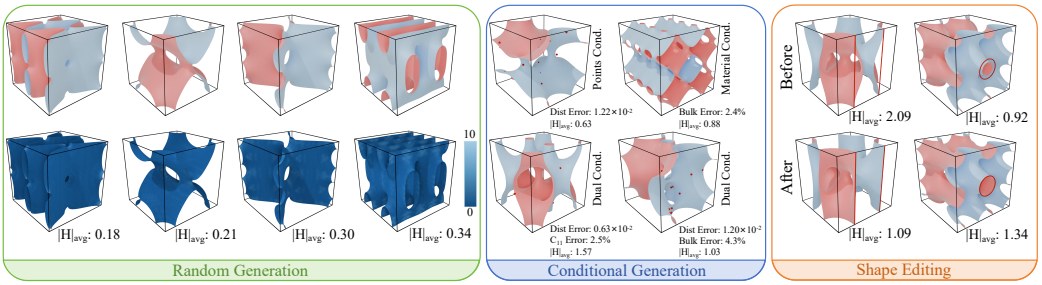


Figure 1: **Our framework enables diverse and near-minimal TPMS generation given user requirements. Left:** unconditional TPMS samples. **Middle:** generation conditioned on sparse points, selected elastic properties, or both. **Right:** local editing. Across all these tasks, the generated TPMS structures maintain low mean curvature over the entire surface.

ABSTRACT

We present a generative framework for the constrained design of D_{2h}-symmetric triply periodic minimal surfaces (TPMS) with low residual mean curvature. Existing TPMS design pipelines either explore a limited set of canonical analytical families or rely on computationally expensive numerical procedures, making it difficult to efficiently generate diverse candidates under geometric and/or mechanical requirements. Our approach learns a distribution of solver-generated TPMS in a compact Fourier space where periodicity and D_{2h} symmetry are guaranteed by construction. This representation eliminates the need for a learned geometric decoder and enables the training of an effective diffusion model that generates low-mean-curvature candidates conditioned on sparse geometric constraints, selected homogenized elastic properties, or their combination. A subsequent coefficient-space refinement further reduces the residual mean curvature. Experiments show that our approach outperforms trigonometric, SDF-based, and standard Fourier-space baselines and enables controllable, high-quality generation under geometric and low-dimensional mechanical constraints. Overall, the proposed framework provides a compact and efficient design space for generating near-minimal periodic structures under user-specified requirements.

1 INTRODUCTION

Triply periodic minimal surfaces (TPMS) are periodic surfaces with zero mean curvature that repeat along three independent spatial directions. The near-zero mean curvature of TPMS is not only a defining geometric property, but also has important implications for their physical performance. In particular, TPMS can attain the optimal bulk modulus when the thickness approaches zero, establishing a direct connection between zero mean curvature and mechanical efficiency (Torquato et al., 2003; Zhang & Liu, 2025). Together with their smooth and bicontinuous geometry, high surface-area-to-volume ratio, and interconnected transport pathways, these properties have motivated broad applications of TPMS in architected materials, heat and mass transfer, and related engineering systems (Feng et al., 2022). For such applications, the design problem extends beyond reproducing a

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

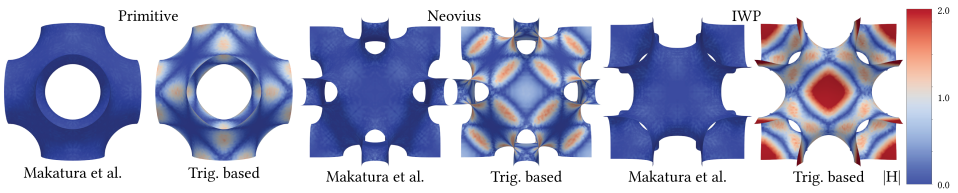


Figure 2: **TPMS represented by low-order trigonometric fields exhibit large absolute mean curvature.** We compare three classical TPMS: Primitive, Neovius, and IWP constructed using the accurate method from Makatura et al. (2023) and low-order trigonometric fields, respectively. The latter exhibits substantially higher absolute mean curvature.

small set of familiar TPMS families: designers often seek geometrically diverse candidates that satisfy prescribed spatial and functional requirements. For example, a structure may be required to pass through selected locations while simultaneously achieving a target bulk modulus. Addressing such problems requires a design space that can explore diverse TPMS morphologies while preserving low mean curvature and supporting quantitative geometric and mechanical constraints.

Existing approaches provide complementary but incomplete solutions. A widely used strategy represents canonical TPMS families, such as the Schwarz Primitive, Diamond, and Gyroid surfaces, using a small number of trigonometric terms (Al-Ketan & Abu Al-Rub, 2021; Jones et al., 2021; Hsieh & Valdevit, 2020). These representations are simple, efficient, and convenient for modeling and fabrication, but their restricted parameterizations cover only a limited portion of the broader TPMS design space and generally introduce non-negligible residual mean curvature. Because mean curvature is directly related to both minimal-surface fidelity and the mechanical optimality, such approximation errors are not merely geometric artifacts and may compromise the desirable physical properties of TPMS. Procedural TPMS construction provides more accurate surfaces and access to a substantially broader family by numerically solving minimal-surface problems associated with different boundary configurations (Makatura et al., 2023). However, repeatedly specifying boundary configurations and solving the corresponding numerical problems is computationally expensive and does not directly provide a mechanism for efficiently sampling new candidates under geometric and mechanical requirements.

We address this gap within a D_{2h}-symmetric design space by learning the distribution of high-quality, solver-generated TPMS. Our key idea is to represent these surfaces in a symmetry-reduced Fourier space in which periodicity and coordinate-reflection symmetry are enforced exactly by construction, rather than learned approximately from data. By explicitly incorporating D_{2h} symmetry into the basis, the representation removes symmetry-related Fourier redundancy and provides a compact coefficient tensor for generative modeling. At the same time, these coefficients define a continuous implicit field whose spatial derivatives can be evaluated analytically, enabling direct surface extraction and curvature-based refinement without requiring an additional learned geometric decoder. Then, we train a conditional diffusion model in this compact space to capture the geometric distribution of solver-generated TPMS. The model supports unconditional sampling as well as conditional generation from sparse geometric constraints and selected homogenized elastic properties. Periodicity and D_{2h} symmetry are guaranteed by the representation, whereas near-minimal curvature is learned from the solver-generated data distribution. A subsequent coefficient-space optimization further reduces residual mean curvature while remaining within the same periodic and symmetry-constrained representation. Although the resulting surfaces are not exact zero-mean-curvature minimal surfaces, they provide near-minimal TPMS candidates with lower residual mean curvature than the evaluated trigonometric representation and the SDF- and standard-Fourier-space generative baselines.

Our contributions are threefold. **First**, we construct a dataset of 18,704 aligned TPMS candidates by combining procedural minimal-surface construction with boundary-loop enumeration, filtering, and canonicalization. **Second**, we develop a conditional generative framework in a compact D_{2h}-invariant Fourier space that guarantees periodicity and symmetry by construction, captures low-curvature TPMS morphologies through learning, and supports coefficient-space curvature refinement. **Third**, we evaluate unconditional generation and conditional generation under sparse point constraints, low-dimensional elastic-property targets, and their combination, together with comparisons that examine the effects of the representation and curvature refinement.

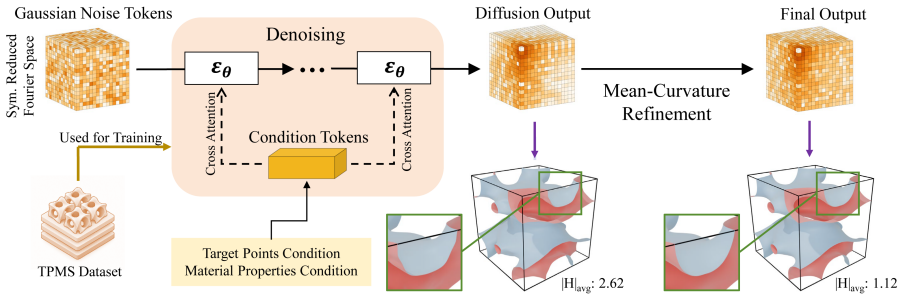


Figure 3: **Overview of our generation pipeline.** Starting from Gaussian noise in our D_{2h}-reduced space, the shared denoising network ε_θ iteratively removes noise while optionally incorporating point and/or material-property conditions as tokens through cross-attention. The denoised coefficients correspond to a surface with low mean curvature, and our postprocessing step further reduces the residual curvature, bringing the surface closer to satisfying the minimality requirement of TPMS.

Collectively, the proposed framework broadens the accessible design space of D_{2h}-symmetric TPMS while maintaining low residual mean curvature and enabling inverse design under both geometric and mechanical requirements. To the best of our knowledge, it is the first learned generative framework for TPMS design that jointly supports geometric and material-property conditioning while maintaining low residual mean curvature.

2 Related Work

Periodic and minimal-surface construction. Classical minimal-surface constructions and geometric solvers provide a foundation for TPMS modeling. The Enneper–Weierstrass representation describes exact minimal surfaces when the associated period conditions are satisfied, but obtaining suitable data and enforcing global compatibility make broad enumeration difficult (Meeks III & Pérez, 2011). Discrete methods recover particular families through specialized constructions (Reitebuch et al., 2019), while Plateau-type solvers compute minimal surfaces from prescribed boundary (Pinkall & Polthier, 1993; Wang & Chern, 2021). Our dataset builds on the procedural boundary-loop construction of Makatura et al. (2023), which computes a minimal patch on a fundamental domain and extends it by symmetry. Low-order trigonometric fields are the most used TPMS design tools (Al-Ketan & Abu Al-Rub, 2021; Jones et al., 2021; Maskery et al., 2022; Hsieh & Valdevit, 2020). Weighted combinations further support hybridization and property variation (Feng et al., 2021; Khaleghi et al., 2021), although these fields generally define approximations rather than exact minimal surfaces. Figure 2 illustrates residual-curvature differences between solver-derived examples and low-order trigonometric approximations. Other approaches construct shell or surface architectures from labyrinths and boundary parameterizations (Akbari et al., 2020; 2022; Xu et al., 2023). Our method also uses a finite Fourier representation, but learns its coefficient distribution from solver-generated data instead of restricting design to a small set of canonical formulas.

Generative modeling. Three-dimensional generative models operate on representations ranging from point clouds and meshes to volumetric and continuous implicit fields (Yang et al., 2019; Liu et al., 2023; Zheng et al., 2022). For high-resolution geometry, representation size affects both learning cost and reconstruction accuracy. Latent diffusion addresses this issue through learned compression (Cheng et al., 2023; Chou et al., 2023), while transform-domain methods exploit wavelet, spectral, or cosine structure (Hu et al., 2024; Fan et al., 2025; Ning et al., 2024). We follow the latter principle with a basis tailored to periodic, reflection-symmetric surfaces. Its coefficient relations are imposed explicitly, so the denoiser need not learn redundant encodings of the same symmetry orbit. The backbone uses established diffusion and transformer components (Ho et al., 2020; Bao et al., 2023). For generative inverse design, MIND (Xue et al., 2025) generates tileable microstructures using a hybrid neural representation, and guided diffusion for voxel-based metamaterials (Yang et al., 2026) targets specified elastic responses. These methods address closely related inverse-design problems. Our emphasis is the combination of sparse geometric conditions, mechanical targets, and thin shells with low residual curvature. This specialization motivates the representation but does not make prior inverse-design methods inapplicable.

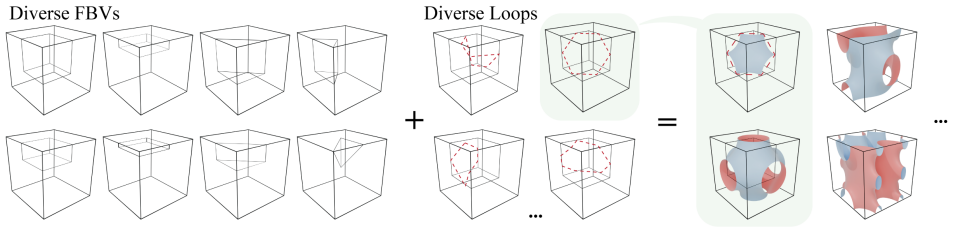


Figure 4: **TPMS dataset synthesis.** By exhaustively combining diverse admissible boundary loops with multiple mirrorable FBV types, we generate a large set of geometrically diverse TPMS.

Learning Solution Fields of PDEs. Because minimal surfaces satisfy a zero-mean-curvature partial differential equation (PDE), our problem is related to learned solution fields, spanning PINNs (Raissi et al., 2019), geometry-informed methods (Berzins et al., 2024), operator-learning approaches (Li et al., 2020; 2023; Wu et al., 2024), and physics-constrained generative modeling (Chung et al., 2022; Baldan et al., 2026). However, standard operator-learning methods typically assume known inputs such as boundary conditions, forcing terms, or geometric parameters, whereas TPMS boundary data are not specified a priori.

3 METHOD

Our pipeline combines solver-based data construction, symmetry-reduced Fourier encoding, conditional diffusion, and curvature refinement (Figure 3). Let $\Omega = [0, L]^3$ denote a cubic periodic cell with side length L. We use $L = 1$. The model operates on field coefficients rather than extracted meshes; the geometry is recovered as the zero set of the decoded field.

3.1 SOLVER-DERIVED DATASET

We use the procedural construction of Makatura et al. (2023) to compute free-boundary minimal-surface patches from admissible loops defined on mirrorable fundamental bounding volumes (FBVs). An FBV is a subdomain whose reflected copies tile the periodic unit cell. A boundary loop specifies a minimal-surface problem within the FBV, and reflecting the resulting patch across the FBV faces produces a TPMS with D_{2h} symmetry. We enumerate loops containing at most ten vertices across eight FBV types (Figure 4), retain successfully solved instances, and augment the resulting meshes using distinct canonical rotations of the unit cell. This procedure explores diverse boundary configurations within the selected construction, rather than varying only a small number of parameters within a fixed surface family. Each solved TPMS is converted into a signed distance field (SDF), using a consistent sign convention that requires the eight corners of the unit cell to have positive values. We then symmetrize the fields, remove the constant Fourier component, and use the dominant Fourier coefficients to align representations that differ only by half-period shifts or global sign flips. Duplicate geometries and candidates with an average absolute mean curvature above 7.5 are discarded. The resulting 18,704 aligned candidates are split into training and validation sets in a 9:1 ratio after preprocessing. Appendix A provides further details on the encoding and canonicalization procedures. Canonicalization is important because the same periodic geometry can admit multiple equivalent field representations: translating the unit-cell origin by half a period changes the signs of selected Fourier coefficients, whereas reversing the SDF sign leaves the zero level set unchanged. Selecting a consistent representative therefore reduces this redundancy and aligns the structures before learning.

After canonicalization and projecting the shape into our representation, the average absolute mean curvature decreases from 2.2 for the raw meshes to 1.4, while the geometric reconstruction error remains small. Across the dataset, the mean absolute SDF difference is approximately 5×10^{-4}, and the 99th-percentile absolute SDF difference remains below 6.5×10^{-3} for every structure (Figure 7). These measurements show that the processed dataset preserves the original geometry while exhibiting low residual mean curvature. This provides the diffusion model with a consistent distribution of low-curvature, reflection-symmetric TPMS geometries from which to learn.

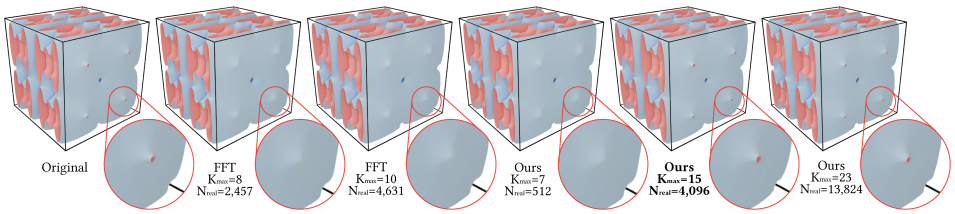


Figure 5: **Our latent representation is more compact than a standard Fourier representation.** Our D_{2h}-symmetric Fourier basis accurately captures fine TPMS features, including small holes, with 4,096 coefficients. In contrast, a standard Fourier representation with a similar coefficient budget fails to recover these features. Further increasing the number of coefficients in our basis yields negligible improvement.

3.2 Symmetry-Reduced Fourier Representation

Directly learning a generative model over raw 3D shapes is challenging because high-resolution field- or point-based representations are high-dimensional and computationally expensive. Implicit neural representations such as DeepSDF (Park et al., 2019) can represent high-quality continuous geometry, but accurate reconstruction typically requires dense spatial queries during training and surface extraction. Likewise, image-based architectures such as convolutional U-Nets (Ronneberger et al., 2015) do not naturally enforce three-dimensional periodicity. In contrast, periodic scalar fields admit Fourier expansions, making their coefficients a natural representation for our data. The additional reflection symmetry enables a substantial reduction in dimensionality. After alignment, the fields are invariant under $p_x \mapsto L - p_x$, $p_y \mapsto L - p_y$, and $p_z \mapsto L - p_z$. These three commuting reflections generate the eight-element group D_{2h}. Invariance ties Fourier coefficients under independent sign changes of the coordinate frequencies, eliminating sine factors from the corresponding tensor-product basis. We therefore write

$$f(\mathbf{p};\boldsymbol{a}) = \sum_{h,k,\ell=0}^{K_{\max}} a_{hk\ell} \cos\left(\frac{2\pi h p_x}{L}\right) \cos\left(\frac{2\pi k p_y}{L}\right) \cos\left(\frac{2\pi \ell p_z}{L}\right), \tag{1}$$
$$\mathcal{S}(\boldsymbol{a}) = \{\mathbf{p} \in \Omega : f(\mathbf{p};\boldsymbol{a}) = 0\},$$

where $K_{\max}$ is the largest retained frequency index in each coordinate and $\boldsymbol{a}$ is the real coefficient tensor. Every coefficient tensor defines a periodic, D_{2h}-invariant field, regardless of whether it is obtained from the dataset, diffusion sampling, or refinement.

A general real Fourier expansion containing harmonics up to absolute frequency $K_{\max}$ requires $(2K_{\max}+1)^3$ real degrees of freedom, whereas our invariant basis uses only $(K_{\max}+1)^3$ coefficients. For $K_{\max} = 15$, the counts are 29,791 and 4,096, respectively, before fixing the constant coefficient. In this work, we store the coefficients in a 16^3 tensor and set $a_{000} = 0$ during preprocessing. The coefficients are obtained by combining symmetry-related standard Fourier coefficients. Each nonzero cosine factor corresponds to a pair of positive- and negative-frequency complex Fourier modes. Thus, if q denotes the number of nonzero indices among h, k, ℓ, the corresponding cosine-product coefficient is obtained from the normalized DFT coefficient as $a_{hk\ell} = 2^q \operatorname{Re}(F_{hk\ell})$, excluding the usual Nyquist-frequency special cases. The detailed derivation is in Appendix B. Figure 5 illustrates how the selected cutoff captures openings in a representative reconstruction.

This compactness results from restricting the function space rather than eliminating Fourier truncation error. The decoded field is an implicit level-set function and need not remain a metric SDF. Its zero set is smooth wherever $\nabla f \neq \mathbf{0}$, but neither regularity nor zero mean curvature is guaranteed for arbitrary coefficients. The 16^3 tensor represents spectral coefficients, so the value and the curvature of the field can therefore be computed analytically.

3.3 Conditional Diffusion

Normalization and tokenization. Coefficient magnitudes vary substantially across frequency modes. We apply the componentwise signed-logarithmic transform $\tilde{a}_i = \operatorname{sign}(a_i)\log(1 + |a_i|/\tau)$, where $\tau > 0$ controls the scale, followed by affine rescaling to obtain normalized coefficients

$\hat{\boldsymbol{a}} \in [-1,1]^{(K_{\max}+1)^3}$. The normalized 16^3 tensor is partitioned into nonoverlapping $2 \times 2 \times 2$ blocks, yielding 512 tokens with eight coefficients each. Let $\boldsymbol{y}_0 = \mathcal{T}(\boldsymbol{a})$ denote this complete encoding operation. Its inverse undoes patchification and affine rescaling before applying the inverse signed-logarithmic transform, ensuring that geometry is always reconstructed from coefficients in the original field scale.

Denoiser and conditions. Inspired by Bao et al. (2023), our UViT-style denoiser linearly embeds the coefficient tokens into 256 dimensions and processes them using four encoder blocks, one bottleneck block, and four decoder blocks with skip fusion and four attention heads. A final linear head predicts noise in the original token space. The diffusion timestep is provided as a time token. For conditional generation, sparse point coordinates, selected homogenized elastic properties, or both are embedded as condition tokens and incorporated through cross-attention. The condition is supplied at every reverse step, while cross-attention is omitted for unconditional generation. This formulation allows the denoiser to incorporate design requirements while generating the full coefficient tensor, rather than modifying only a prescribed subset of geometric parameters.

Training. We use a variance-preserving forward process (Song et al., 2020b) with the standard noise-prediction objective,

$$\begin{aligned} \boldsymbol{y}_t &= \alpha(t)\boldsymbol{y}_0 + \sigma(t)\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad t \sim \mathcal{U}(0,1), \\ \mathcal{L}_{\text{diff}} &= \mathbb{E}_{\boldsymbol{y}_0, \boldsymbol{\epsilon}, t, \mathbf{c}}\big[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{y}_t, t, \mathbf{c})\|_2^2\big]. \end{aligned} \tag{2}$$

Here, $\alpha(t) = \exp\Big(-\frac{1}{2}\int_0^t \beta(s)\,ds\Big)$ and $\sigma(t) = \sqrt{1-\alpha(t)^2}$ determine the signal and noise amplitudes, $\beta(\cdot)$ is the noise schedule, and $\mathbf{c}$ denotes the optional condition. The condition may encode sparse geometric constraints, target homogenized material properties, or both. At each training step, a clean example is perturbed at a randomly sampled time, and the denoiser is trained to predict the injected Gaussian noise. The final objective includes neither an Eikonal loss nor an explicit curvature penalty, as our ablation study shows that these terms are either unnecessary or detrimental to training (Appendix F.4). Instead, the low-curvature character of the geometry is learned directly from the data distribution.

Sampling, inversion, and editing. Generation starts from Gaussian tokens and uses deterministic DDIM updates (Song et al., 2020a) to recover a clean coefficient representation. Each update combines the current clean-sample estimate with the network-predicted noise; no additional randomness is introduced once the initial state is fixed. Decoding with $\mathcal{T}^{-1}$ yields the coefficients used in Eq. (1). Starting from an existing TPMS, DDIM inversion maps its Fourier coefficients along a deterministic trajectory from the clean latent state to a noisier latent state. Reversing the same trajectory approximately reconstructs the input shape, allowing the noisy latent to serve as a starting point for controlled editing. At each denoising step, we evaluate a differentiable constraint loss on the predicted clean surface and take a gradient step with respect to the latent to move the surface toward prescribed points or boundary curves before continuing the DDIM update. For example, to force the generated surface to pass through a set of user-specified points $\mathcal{P}_{\text{con}}$, we use $\mathcal{L}_{\text{geom}} = \frac{1}{|\mathcal{P}_{\text{con}}|}\sum_{\mathbf{p}\in\mathcal{P}_{\text{con}}} |f(\mathbf{p}; \hat{\boldsymbol{a}}_0)|^2$, where $\hat{\boldsymbol{a}}_0$ is obtained by inverse-normalizing $\hat{\boldsymbol{y}}_0$, the predicted clean latent at the current timestep.

3.4 Post Curvature Refinement

Although our diffusion model already produces structures with sufficiently low mean curvature, we further reduce residual curvature by directly optimizing the decoded coefficients. At iteration k, we sample a point set $\mathcal{P}^{(k)}$ from the current surface and perform a gradient-based update using

$$\mathcal{L}_{\text{ref}}^{(k)}(\boldsymbol{a}) = \mathbb{E}_{\mathbf{p}\sim\mathcal{P}^{(k)}}[H_\delta(\mathbf{p}; \boldsymbol{a})^2] + \lambda_{\boldsymbol{a}}\mathbb{E}_{\mathbf{p}\sim\mathcal{P}^{(k)}}|f(\mathbf{p}; \boldsymbol{a})|, \tag{3}$$

where the regularized mean curvature is evaluated as

$$H_\delta(\mathbf{p}; \boldsymbol{a}) = \frac{1}{2}\nabla \cdot \left(\frac{\nabla f(\mathbf{p}; \boldsymbol{a})}{\sqrt{\max\left(\|\nabla f(\mathbf{p}; \boldsymbol{a})\|_2^2, \delta\right)}}\right), \qquad \delta > 0. \tag{4}$$

Table 1: **Generation-pipeline comparison.** Curvature ($|H|_{\text{avg}}$) and maximum genus ("Max. genus") are reported. Inversion CD is a separate reconstruction metric, scaled by 10^{-3}. "Periodic" indicates whether periodicity is enforced by construction. "Raw" refers to the structure before the curvature refinement, while "Aft" denotes the structure after the curvature refinement.

Method	Training (h) ↓	Inversion CD ↓	Raw $\lvert H\rvert_{\text{avg}}$ ↓	Aft $\lvert H\rvert_{\text{avg}}$ ↓	Max. genus ↑	Periodic
Trigonometric	**0**	—	1.23	1.07	8	✓
SDF-Diffusion	71	10.13	1.15	0.90	12	✗
SDFusion	103	4.06	1.19	1.09	15	✗
Standard Fourier Space	35	3.94	2.28	1.40	**17**	✓
Ours	30	**3.71**	**1.00**	**0.67**	**17**	✓

δ is a very small value to prevent division by zero. The first term penalizes residual curvature, while the second discourages deviation from the reference surface. The point set is resampled at each iteration rather than fixed on the initial level set, so each update is evaluated relative to the preceding geometry.

The Fourier basis provides analytic first- and second-order spatial derivatives, avoiding finite-difference approximations. The normalized-gradient curvature formula applies to any regular implicit field and does not require a metric SDF. We use $8{,}192$ sampled vertices, 80 optimization steps, and $\lambda_{\boldsymbol{a}} = 0.1$. Because refinement remains within the same coefficient space, periodicity and reflection symmetry remain exact throughout optimization, while the compact representation enables efficient refinement.

4 Results

Our 7.8M-parameter model is implemented in PyTorch 2.2.1 and trained for one million steps using AdamW (Loshchilov & Hutter, 2017; Kingma & Ba, 2014), a batch size of 128, and a learning rate of 10^{-4}. Training takes approximately 30 hours on two NVIDIA H200 GPUs. On one H200, sampling takes 2.1 s and refinement takes 4.4 s, for a total of 6.5 s per candidate.

We evaluate generation quality, geometric and mechanical conditioning, and the effects of the representation and refinement. The implicit field of the structure is on a 150^3 grid. We extract the zero set using Marching Cubes and compute curvature analytically at the extracted vertices. The statistic $|H|_{\text{avg}}$ is the average of per-sample absolute mean curvature. Chamfer distance (CD) uses unsquared Euclidean distances, while point-conditioning error is the mean prescribed-point-to-surface distance. Detailed definitions are in Appendix G. The common cell size makes the reported distances directly comparable across tasks. Curvature is expressed in inverse-length units, so its interpretation also depends on this normalization. We use the mean to summarize overall residual curvature.

4.1 Generation Quality and Baseline Comparisons

Evaluation protocol. For each method, we generate 300 candidates and retain the 150 with the lowest $|H|_{\text{avg}}$ for generation comparison. Table 1 therefore evaluates a *design-screening* setting. Our selected set has a mean curvature of 0.67, whereas the average over all 300 candidates is 1.16. We distinguish these populations because screening changes the evaluated distribution. Inversion CD is measured separately by reconstructing existing examples from the dataset.

Baseline results. We compare our method with the widely used trigonometric baseline combining six standard implicit terms (Yang & Buehler, 2022), SDF-Diffusion (Shim et al., 2023), which learns directly from volumetric SDFs, SDFusion (Cheng et al., 2023), which uses a VQ-VAE representation before diffusion, and a standard Fourier-space representation. All generative baselines are trained on our dataset. The results are reported in Table 1. For each method, we report $|H|_{\text{avg}}$ both before and after curvature refinement. For our method and the trigonometric baseline, we apply our Fourier-space refinement, whereas for the rest methods, we use the widely adopted mean-curvature flow (MCF) (Desbrun et al., 1999). Whether with or without curvature refinement, our pipeline achieves the lowest mean curvature and the largest maximum genus, while consistently producing single-component structures. The training-free trigonometric baseline exhibits approximately $1.6\times$ higher residual mean curvature and lower structural diversity, as reflected by its maximum genus.

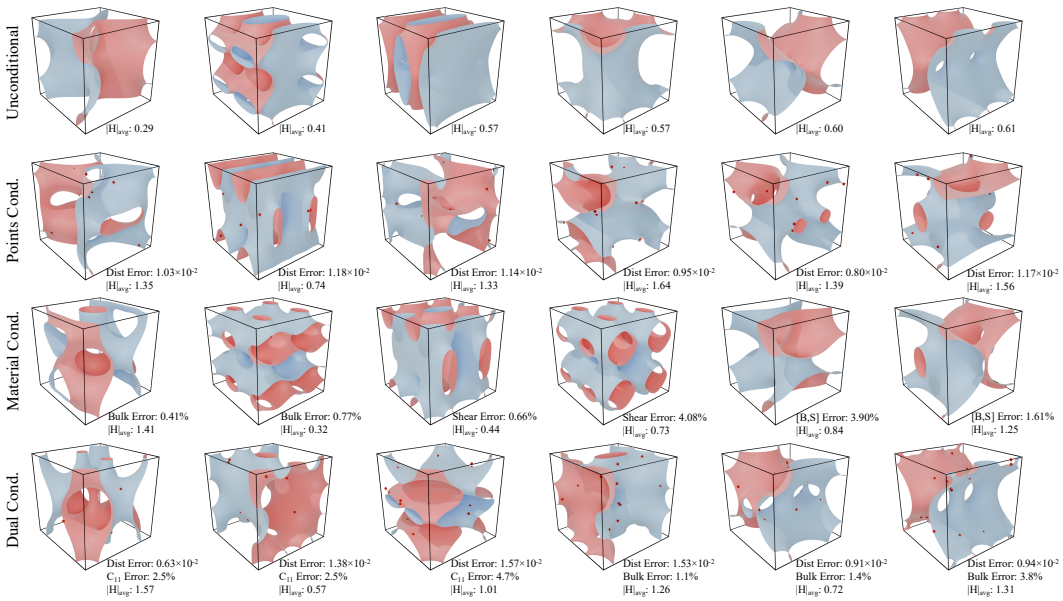


Figure 6: **Representative generation results under different conditioning settings.** From top to bottom: unconditional generation, point conditioning, material-property conditioning, and joint point–property conditioning. For material conditioning, the columns correspond to bulk, shear, and joint bulk–shear conditions; for joint conditioning, they correspond to point constraints combined with either C_{11} or bulk modulus.

Histogram statistics are shown in Figure 14. The other generative baselines require longer training times than our method while yielding higher mean curvature and lower structural diversity. The two SDF baselines also exhibit periodic-boundary inconsistencies in our implementations. These results show that our pipeline performs better on the reported metrics.

Geometric variation. Samples generated by our method also exhibit geometric novelty, with an average CD of 9.1×10^{-3} to the nearest training sample, approximately $2.5\times$ the average inversion CD. A selected set of visual results is provided in the Appendix, in which Figure 12 reports the distribution, Figure 10 compares candidates with their nearest dataset neighbors, and Figure 11 shows additional samples. These examples demonstrate geometric variation beyond direct reproduction of the illustrated neighbors.

4.2 Geometric and Mechanical Conditioning

Evaluation protocol. We evaluate sparse-point conditions, material-property conditions, and their combinations (Table 2). Each point condition consists of 16 3D points specified by their coordinates within the cell. For mechanical evaluation, surfaces are uniformly thickened to 0.02 and assigned an isotropic constituent with Young's modulus $E_s = 1$ and Poisson's ratio $\nu_s = 0.3$. The same periodic homogenization solver provides training labels and evaluates generated structures (Zhang et al., 2023). Bulk and shear values are Voigt–Reuss–Hill scalar averages of the orthotropic stiffness tensor (Appendix I). For a nonzero scalar target m^*, the relative error is $100|\hat{m} - m^*|/|m^*|$. The selected properties capture complementary responses: bulk and shear summarize volumetric and shear behavior, while C_{11} specifies an individual stiffness component. They provide low-dimensional targets for evaluating controllability without requiring every independent tensor entry to be prescribed.

Single-modality conditions. Across 30 point-conditioned samples, the mean point-to-surface distance is 0.017, or 1.7% of the cell period, with mean curvature 1.28. For bulk, shear, and joint bulk–shear conditioning, we evaluate 20 samples per setting. Bulk-only conditioning yields a 2.8% mean relative error, compared with 9.2% for shear; both retain low residual curvature in the re-

Table 2: **Conditional-design results.** Distances are reported for a unit-period cell; property errors are percentages. "—" denotes an inapplicable or unreported quantity, rather than zero error.

Condition	Samples	Avg. Dist	Prop. err (%)	$\lvert H\rvert_{\mathrm{avg}}$
Points	30	0.017	—	1.28
Bulk modulus	20	—	2.8	0.91
Shear modulus	20	—	9.2	1.21
Bulk + shear	20	—	6.2, 5.7	1.09
Points + C_{11}	20	0.019	9.3	1.30
Points + bulk	20	0.021	6.9	1.19

ported evaluation. Simultaneously conditioning on bulk and shear lead to errors of 6.2% and 5.7%, respectively. Figure 6 shows representative constraints and generated geometries.

Joint geometric and mechanical conditions. Point constraints control local surface placement, whereas homogenized properties depend on the global geometry. Combining points with either C_{11} or bulk modulus tests whether the model can satisfy local spatial requirements and an effective mechanical response simultaneously. Our method is capable of doing so. The respective property errors are 9.3% and 6.9%, with point distances of 0.019 and 0.021 across 20 samples per setting, as reported in Table 2. These results support simultaneous, approximate control within the selected design family. Figure 6 shows representative constraints and generated geometries.

4.3 Inversion and Editing

To assess reconstruction, we invert and decode $2{,}000$ training shapes and separately evaluate validation examples. The mean CDs are 3.71×10^{-3} and 3.82×10^{-3}, with corresponding mean curvature values of 1.43 and 1.47. Similar errors across the two splits support accurate reconstruction of held-out inputs, complementing rather than replacing the generation evaluation. Appendix D provides additional details and the deterministic update equations, while Figure 9 provides visual comparisons. Our model also supports TPMS editing: users specify target curves on selected cell faces to guide modifications of an existing candidate. As shown in Figure 1, the edited shapes match the prescribed boundary curves while retaining low mean curvature and remaining close to the original geometry.

4.4 Ablation Study

Although the raw diffusion outputs already exhibit low residual mean curvature, curvature refinement further reduces the average $\lvert H\rvert_{\mathrm{avg}}$ from 1.00 to 0.67. Meanwhile, the genus remains unchanged, and the average CD to the original generated shape is only 1.60×10^{-3}, confirming that the optimization acts primarily as a local refinement, while the generative model plays the central role in producing low-mean-curvature structures. Additionally, we find that neither increasing $K_{\max}$ nor using a standard Fourier representation improves generation quality. Moreover, performing inverse homogenization directly in our compact latent space fails to reliably identify structures that satisfy user requirements while maintaining low residual mean curvature. These results are reported in Appendix F.

5 Conclusion

We presented a generative-model-based framework for controllable generation of TPMS with low residual mean curvature. Through qualitative and quantitative experiments, we demonstrated that the proposed model supports deterministic shape inversion, unconditional generation, and conditional generation under sparse geometric and material-property constraints. The generated TPMS exhibit low mean curvature and compare favorably with existing baselines. These results show that our framework provides a practical approach for TPMS generation and inverse design. Several limitations remain. First, although the generated surfaces have low residual mean curvature, they are not guaranteed to be exactly minimal. Second, the current latent representation is restricted to D_{2h}-symmetric TPMS, which excludes broader classes of TPMS. Failure cases are discussed in Appendix J.

AI USE STATEMENT

In this work, we used OpenAI's ChatGPT (GPT-5.6, GPT-6) and Codex to assist in polishing the language and presentation of the manuscript, and improving the layout and presentation of figures. These tools were not involved in developing the research idea, formulating the proposed methodology, designing or selecting models, generating experimental data, conducting experiments, or interpreting the results. All AI-assisted text, including literature descriptions and citations, was manually reviewed and verified against the original sources by the authors. We take full responsibility for the final content of this work, including all text, claims, results, and artifacts produced with the aid of generative AI.

REFERENCES

Mostafa Akbari, Armin Mirabolghasemi, Hamid Akbarzadeh, and Masoud Akbarzadeh. Geometry-based structural form-finding to design architected cellular solids. In *Proceedings of the ACM Symposium on Computational Fabrication*, pp. 1–11, 2020.

Mostafa Akbari, Armin Mirabolghasemi, Mohammad Bolhassani, Abdolhamid Akbarzadeh, and Masoud Akbarzadeh. Strut-based cellular to shellular funicular materials. *Advanced Functional Materials*, 32(14):2109725, 2022.

Oraib Al-Ketan and Rashid K Abu Al-Rub. Mslattice: A free software for generating uniform and graded lattices based on triply periodic minimal surfaces. *Material Design & Processing Communications*, 3(6):e205, 2021.

Giacomo Baldan, Qiang Liu, Alberto Guardone, and Nils Thuerey. Physics vs distributions: Pareto optimal flow matching with physics constraints. In *International Conference on Learning Representations*, volume 2026, pp. 101215–101243, 2026.

Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22669–22679, 2023.

Arturs Berzins, Andreas Radler, Eric Volkmann, Sebastian Sanokowski, Sepp Hochreiter, and Johannes Brandstetter. Geometry-informed neural networks. *arXiv preprint arXiv:2402.14009*, 2024.

Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4456–4465, 2023.

Gene Chou, Yuval Bahat, and Felix Heide. Diffusion-sdf: Conditional generative modeling of signed distance functions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2262–2272, 2023.

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pp. 317–324, 1999.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Jiajie Fan, Amal Trigui, Andrea Bonfanti, Felix Dietrich, Thomas Bäck, and Hao Wang. A mesh is worth 512 numbers: Spectral-domain diffusion modeling for high-dimension shape generation. *arXiv preprint arXiv:2503.06485*, 2025.

Jiawei Feng, Bo Liu, Zhiwei Lin, and Jianzhong Fu. Isotropic porous structure design methods based on triply periodic minimal surfaces. *Materials & Design*, 210:110050, 2021.

Jiawei Feng, Jianzhong Fu, Xinhua Yao, and Yong He. Triply periodic minimal surface (tpms) porous structures: from multi-scale design, precise additive manufacturing to multidisciplinary applications. *International Journal of Extreme Manufacturing*, 4(2):022001, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Meng-Ting Hsieh and Lorenzo Valdevit. Minisurf–a minimal surface generator for finite element modeling and additive manufacturing. *Software Impacts*, 6:100026, 2020.

Jingyu Hu, Ka-Hei Hui, Zhengzhe Liu, Ruihui Li, and Chi-Wing Fu. Neural wavelet-domain diffusion for 3d shape generation, inversion, and manipulation. *ACM transactions on graphics*, 43(2): 1–18, 2024.

Alistair Jones, Martin Leary, Stuart Bateman, and Mark Easton. Tpms designer: A tool for generating and analyzing triply periodic minimal surfaces. *Software Impacts*, 10:100167, 2021.

Saeed Khaleghi, Fayyaz N Dehnavi, Mostafa Baghani, Masoud Safdari, Kui Wang, and Majid Baniassadi. On the directional elastic modulus of the tpms structures and a novel hybridization method to control anisotropy. *Materials & Design*, 210:110074, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Zongyi Li, Nikola Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, et al. Geometry-informed neural operator for large-scale 3d pdes. *Advances in Neural Information Processing Systems*, 36:35836–35854, 2023.

Zhen Liu, Yao Feng, Michael J Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3d mesh modeling. *arXiv preprint arXiv:2303.08133*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Liane Makatura, Bohan Wang, Yi-Lu Chen, Bolei Deng, Chris Wojtan, Bernd Bickel, and Wojciech Matusik. Procedural metamaterials: A unified procedural graph for metamaterial design. *ACM Transactions on Graphics*, 42(5):1–19, 2023.

Ian Maskery, Luke A Parry, Daniel Padrão, Richard JM Hague, and Ian A Ashcroft. Flatt pack: A research-focussed lattice design program. *Additive Manufacturing*, 49:102510, 2022.

William Meeks III and Joaquín Pérez. The classical theory of minimal surfaces. *Bulletin of the American Mathematical Society*, 48(3):325–407, 2011.

Mang Ning, Mingxiao Li, Jianlin Su, Haozhe Jia, Lanmiao Liu, Martin Beneš, Wenshuo Chen, Albert Ali Salah, and Itir Onal Ertugrul. Dctdiff: Intriguing properties of image generative modeling in the dct space. *arXiv preprint arXiv:2412.15032*, 2024.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 165–174, 2019.

Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental mathematics*, 2(1):15–36, 1993.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Ulrich Reitebuch, Martin Skrodzki, and Konrad Polthier. Discrete gyroid surface. In *Proceedings of bridges 2019: Mathematics, art, music, architecture, education, culture*, pp. 461–464, 2019.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Jaehyeok Shim, Changwoo Kang, and Kyungdon Joo. Diffusion-based signed distance fields for 3d shape generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20887–20897, 2023.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

S Torquato, S Hyun, and Aleksandar Donev. Optimal design of manufacturable three-dimensional composites with multifunctional characteristics. *Journal of Applied Physics*, 94(9):5748–5755, 2003.

Stephanie Wang and Albert Chern. Computing minimal surfaces with differential forms. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. *arXiv preprint arXiv:2402.02366*, 2024.

Yonglai Xu, Hao Pan, Ruonan Wang, Qiang Du, and Lin Lu. New families of triply periodic minimal surface-like shell lattices. *Additive Manufacturing*, 77:103779, 2023.

Tianyang Xue, Longdu Liu, Lin Lu, Paul Henderson, Pengbin Tang, Haochen Li, Jikai Liu, Haisen Zhao, Hao Peng, and Bernd Bickel. Mind: Microstructure inverse design with generative hybrid neural representation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–12, 2025.

Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4541–4550, 2019.

Yanyan Yang, Lili Wang, Xiaoya Zhai, Kai Chen, Wenming Wu, Yunkai Zhao, Falai Chen, Ligang Liu, and Xiao-Ming Fu. Guided diffusion for fast inverse design of voxel-based mechanical metamaterials. *Smart Materials in Manufacturing*, 4:100129, 2026.

Zhenze Yang and Markus J Buehler. High-throughput generation of 3d graphene metamaterials and property quantification using machine learning. *Small Methods*, 6(9):2200537, 2022.

Di Zhang and Ligang Liu. Asymptotic analysis and design of linear elastic shell lattice metamaterials. *ACM Transactions on Graphics (TOG)*, 44(4):1–18, 2025.

Di Zhang, Xiaoya Zhai, Ligang Liu, and Xiao-Ming Fu. An optimized, easy-to-use, open-source gpu solver for large-scale inverse homogenization problems. *Structural and Multidisciplinary Optimization*, 66(9):207, 2023.

Xinyang Zheng, Yang Liu, Pengshuai Wang, and Xin Tong. Sdf-stylegan: implicit sdf-based stylegan for 3d shape generation. In *Computer Graphics Forum*, volume 41, pp. 52–63. Wiley Online Library, 2022.

A DATASET DETAILS

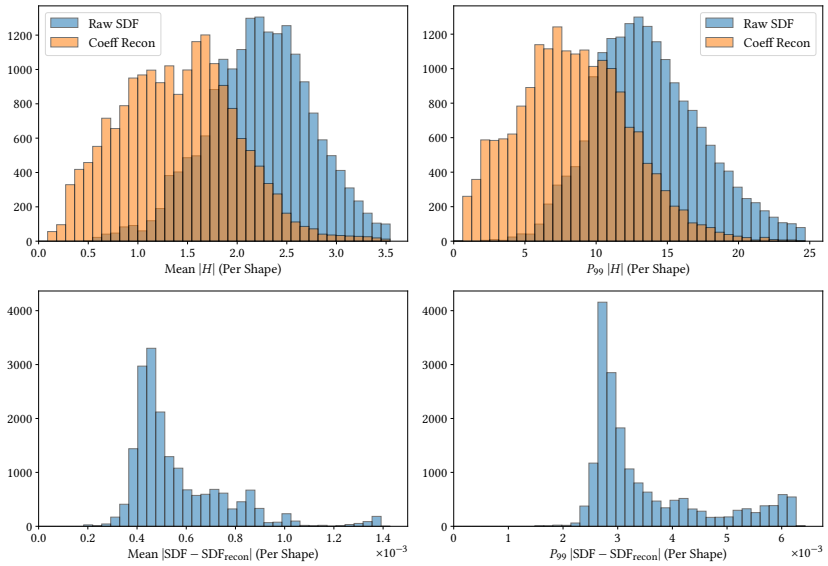


Figure 7: **Numerical results of data processing.** Top row: after our symmetry processing and frequency-domain truncation, the shapes exhibit a significant reduction in mean curvature, better satisfying the basic requirements of TPMS. Bottom row: meanwhile, we maintain a tiny difference between the original and processed shapes, with the average SDF difference remaining on the order of 10^{-4}.

Notation. Every shape lives in the periodic unit cell $\Omega = [0,1)^3$. A field sampled on an n^3 grid is written $f_{\boldsymbol{p}}$ with $\boldsymbol{p} = (p_1, p_2, p_3) \in \{0, \dots, n-1\}^3$, indices taken modulo n. Its discrete Fourier transform is $\hat{f}_{\boldsymbol{q}} = \sum_{\boldsymbol{p}} f_{\boldsymbol{p}}\, e^{-2\pi \mathrm{i}\, \boldsymbol{q}\cdot\boldsymbol{p}/n}$. For $\boldsymbol{b} \in \{0,1\}^3$, $\tau_{\boldsymbol{b}}$ is the circular shift by half a period along every axis a with $b_a = 1$, $(\tau_{\boldsymbol{b}} f)_{\boldsymbol{p}} = f_{\boldsymbol{p} + \frac{n}{2}\boldsymbol{b}}$. $\bar{f}$ is the mean of f over the grid, and $[\,\cdot\,]$ is 1 if the condition holds and 0 otherwise.

Step 1. Mesh generation and curvature filtering. TPMS surface meshes are generated with the procedural graph of Makatura et al. (2023). We use the conditions that check if a loop is valid from Makatura et al. (2023) and combine the loop variations with eight different boundary types. The mean curvature H is computed on each mesh, and meshes whose average mean curvature exceeds 7.5 are discarded.

Step 2. Rotation augmentation. Each mesh is rotated by the rotations of the cube that map the unit cell onto itself (rotations by multiples of 90° about the coordinate axes). Only the distinct rotated copies are kept, and each is added to the dataset as a separate shape.

Step 3. Signed distance field.

(a) Sample the cell on a $257 \times 257 \times 257$ grid of points $\boldsymbol{x}_{\boldsymbol{p}} = \boldsymbol{p}/256$, $\boldsymbol{p} \in \{0, \dots, 256\}^3$; both boundary faces of the cell are included.

(b) Compute the unsigned distance $d_{\boldsymbol{p}}$ from $\boldsymbol{x}_{\boldsymbol{p}}$ to the mesh.

(c) Assign signs by flood fill: the 8 corners of the cell are assumed to be positive, and the sign is propagated through the grid, giving the signed distance field $\phi_{\boldsymbol{p}} = \pm d_{\boldsymbol{p}}$.

(d) Keep the field only if the maximum distance error reported by the distance computation is below half a voxel, $0.5/256$.

(e) Because ϕ is periodic, the samples with $p_a = 256$ repeat those with $p_a = 0$. They are dropped, leaving the periodic 256^3 grid $\phi_{\boldsymbol{p}} = \phi(\boldsymbol{p}/256)$, $\boldsymbol{p} \in \{0, \dots, 255\}^3$, which all later steps use.

Step 4. DC removal, canonical alignment, and deduplication. The field is mirror-symmetric about the origin in x, y and z, and by periodicity also about every point of $\{0, \frac{1}{2}\}^3$. The only remaining freedom is therefore the global sign and which of these 8 symmetry centers sits at the origin, which gives 16 equivalent fields $s\,\tau_{\boldsymbol{b}}\phi$ with $s \in \{+1, -1\}$ and $\boldsymbol{b} \in \{0, 1\}^3$. Their Fourier coefficients are $s\,(-1)^{\boldsymbol{q}\cdot\boldsymbol{b}}\,\hat{\phi}_{\boldsymbol{q}}$: a half-period shift along x multiplies mode (h, k, l) by $(-1)^h$, and a sign flip multiplies every mode by -1.

(a) Remove the DC component: $\phi \leftarrow \phi - \bar{\phi}$. Note that since the original shape is already close to TPMS, $\bar{\phi}$ is close to zero. Thus, this adjustment does not change the represented surface much.

(b) Take every 32nd voxel, $\psi_{\boldsymbol{p}} = \phi_{32\boldsymbol{p}}$ for $\boldsymbol{p} \in \{0, \ldots, 7\}^3$, and compute its 8^3 DFT $\hat{\psi}$.

(c) Dominant mode: $\boldsymbol{q}^\star = \arg\max_{\boldsymbol{q}\neq\boldsymbol{0}} |\Re\hat{\psi}_{\boldsymbol{q}}|$.

(d) Keep the 8 of the 16 versions in which this coefficient is positive,

$$\mathcal{V} = \big\{(s, \boldsymbol{b}) :\ s\,(-1)^{\boldsymbol{q}^\star\cdot\boldsymbol{b}}\,\Re\hat{\psi}_{\boldsymbol{q}^\star} > 0\big\}.$$

(e) Tie-break: choose $(s^\star, \boldsymbol{b}^\star) \in \mathcal{V}$ whose subsample $s\,\tau_{\boldsymbol{b}}\psi$, flattened in row-major order, is lexicographically smallest; entries that differ by at most 10^{-12} count as equal. The first entry is the value at the origin, so in practice the version with the smallest ϕ at the origin wins.

(f) Apply the winning shift and sign at full resolution, $\phi^\star = s^\star\,\tau_{\boldsymbol{b}^\star}\phi$ on the 256^3 grid, and store it in single precision.

(g) Downsample $\phi^\star$ to 16^3, 32^3, and 64^3. We first compare each pair of SDF samples at the coarsest resolution. If they are identical, we repeat the comparison at progressively finer resolutions up to 64^3. This procedure removes all duplicate samples.

Step 5. Curvature filtering of the aligned field.

(a) On the 256^3 grid, with periodic central differences of spacing $1/256$, compute H as described in Appendix H.

(b) Extract the zero level set of $\phi^\star$ with marching cubes and sample 4,096 of its vertices uniformly at random. Move each sample onto the surface with one Newton step, $\boldsymbol{x} \leftarrow \boldsymbol{x} - \phi^\star(\boldsymbol{x})\,\nabla\phi^\star(\boldsymbol{x})/\|\nabla\phi^\star(\boldsymbol{x})\|^2$, and evaluate H there by trilinear interpolation.

(c) Keep the shape if, over the samples, $\mathrm{mean}\,|H| < 7.5$ and $\max|H| < 100$.

After this step, 18,704 shapes remain.

Output. The final dataset has 18,704 shapes. Figure 7 shows the statistics.

B BASIS COMPUTATION DETAILS

Let $r_x(\mathbf{p}) = (L - p_x, p_y, p_z)$, with r_y and r_z defined analogously. These commuting reflections generate an eight-element group,

$$G = \{r_x^\alpha r_y^\beta r_z^\gamma : \alpha, \beta, \gamma \in \{0, 1\}\} \cong \mathbb{Z}_2^3 \cong D_{2h}. \tag{5}$$

The requirement is invariance of the signed field, $f(g\mathbf{p}) = f(\mathbf{p})$, rather than only invariance of its unsigned zero set. A general real periodic field has expansion

$$f(\mathbf{p}) = \sum_{\mathbf{n}\in\mathbb{Z}^3} c_{\mathbf{n}} \exp\left(\frac{2\pi i}{L}\mathbf{n}\cdot\mathbf{p}\right), \qquad c_{-\mathbf{n}} = \overline{c_{\mathbf{n}}}. \tag{6}$$

Coordinate-reflection invariance ties coefficients under independent sign reversals of the three frequency indices. Grouping these modes yields the cosine-product expansion in Eq. (1). This derivation explains the reduction: the model need not learn repeated coefficients corresponding to the same symmetry orbit. It does not add information beyond the corresponding symmetry-constrained Fourier subspace.

For normalized DFT coefficients $F_{hk\ell}$ of the symmetrized field, define $q(h, k, \ell) = \mathbf{1}_{h>0} + \mathbf{1}_{k>0} + \mathbf{1}_{\ell>0}$. Away from Nyquist-index special cases, the conversion to cosine-product coefficients is

$$a_{hk\ell} = 2^{q(h,k,\ell)}\,\mathrm{Re}\,F_{hk\ell}. \tag{7}$$

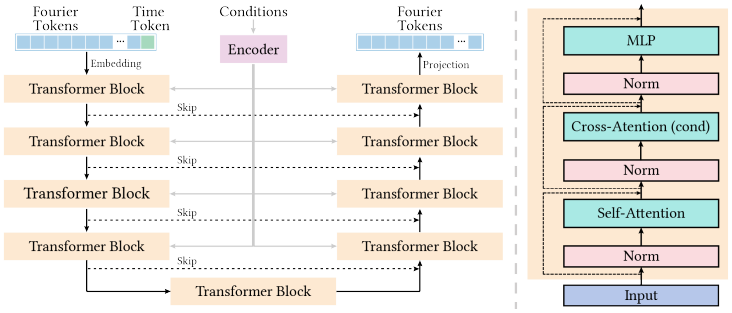


Figure 8: **Our neural network architecture.** We use a transformer-based diffusion model with skip connections. Geometric or material-property conditions are encoded as tokens and injected into each transformer block through cross-attention.

Each nonzero cosine factor contributes two exponential modes with weights $1/2$. The constant coefficient is set to zero as described above.

The count $(2K_{\max}+1)^3$ for the general real Fourier field already accounts for conjugate symmetry; it counts independent real scalars, not independently stored complex numbers. The invariant basis has $(K_{\max}+1)^3$ real coefficients, a reduction approaching a factor of eight as $K_{\max}$ increases. At $K_{\max} = 15$, the ratio is approximately 7.27. The $4{,}096$ entries in our 16^3 tensor include the constant component; fixing that component leaves $4{,}095$ adjustable entries if the same restriction is maintained during generation.

C Implementation and Diffusion Details

C.1 Normalization and Its Inverse

For a positive scale parameter τ, define the componentwise signed-logarithmic transform

$$\phi_\tau(a) = \operatorname{sign}(a)\log\left(1+\frac{|a|}{\tau}\right), \qquad \phi_\tau^{-1}(u) = \tau\operatorname{sign}(u)\left(e^{|u|}-1\right). \tag{8}$$

In this paper, we set $\tau = 10^{-5}$. Let $\mathcal{A}$ denote the affine rescaling used to map the transformed training coefficients to $[-1,1]$, and let $\mathcal{P}$ denote patchification. The encoding and decoding operations are

$$\boldsymbol{y} = \mathcal{T}(\boldsymbol{a}) = \mathcal{P}\mathcal{A}\phi_\tau(\boldsymbol{a}), \qquad \boldsymbol{a} = \mathcal{T}^{-1}(\boldsymbol{y}) = \phi_\tau^{-1}\left(\mathcal{A}^{-1}\mathcal{P}^{-1}\boldsymbol{y}\right). \tag{9}$$

Thus decoding first undoes patchification and affine rescaling, then applies the inverse logarithmic transform. Neither transform imposes the Eikonal equation on the generated field.

C.2 Neural Network Architecture

The network follows a U-shaped transformer design consisting of encoder blocks, a bottleneck block, and decoder blocks with skip fusion. A final linear prediction head maps the decoded features back to the original token dimension, as shown in Figure 8. In our experiments, we use an embedding dimension of $D = 256$, four encoder blocks, one bottleneck block, four decoder blocks, and four attention heads.

C.3 Forward Schedule

The variance-preserving forward process is

$$d\boldsymbol{y} = -\tfrac{1}{2}\beta(t)\boldsymbol{y}\,dt + \sqrt{\beta(t)}\,d\mathbf{w}, \qquad t\in[0,1], \tag{10}$$

where $\mathbf{w}$ is a Wiener process. Its perturbation coefficients in Eq. (2) are

$$\alpha(t) = \exp\left(-\tfrac{1}{2}\int_0^t \beta(s)\,ds\right), \qquad \sigma(t) = \sqrt{1-\alpha(t)^2}. \tag{11}$$

The reported linear schedule uses $\beta_{\min} = 0.1$ and $\beta_{\max} = 10.0$. The denoiser predicts the Gaussian perturbation, using the squared-error objective in Eq. (2).

D INVERSION DETAILS

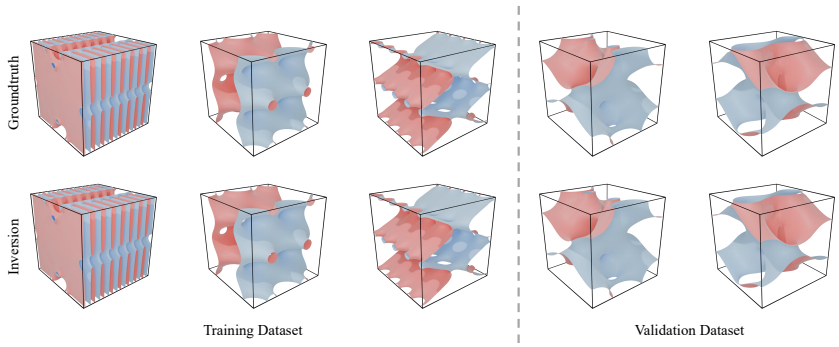


Figure 9: **Visualization of deterministic shape inversion.** Our model achieves high-fidelity reconstruction for both seen and unseen shapes, showing that the learned diffusion trajectory is stable and generalizable.

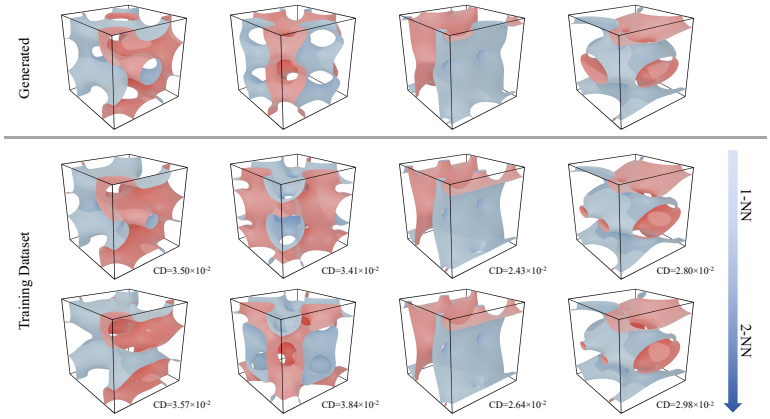


Figure 10: **Shape novelty analysis.** First row: randomly generated shapes; second and third rows: the shapes from the training set with the smallest and second-smallest CD, respectively. Our generated results exhibit clear topological differences from the training data.

D.1 DETERMINISTIC UPDATES

At time t_i, the denoiser predicts a noise tensor and a clean-token estimate,

$$\hat{\boldsymbol{\epsilon}}_i = \boldsymbol{\epsilon}_\theta(\boldsymbol{y}_{t_i}, t_i, \mathbf{c}), \qquad \hat{\boldsymbol{y}}_0 = \frac{\boldsymbol{y}_{t_i} - \sigma(t_i)\hat{\boldsymbol{\epsilon}}_i}{\alpha(t_i)}. \tag{12}$$

The deterministic DDIM-style update is

$$\boldsymbol{y}_{t_j} = \alpha(t_j)\hat{\boldsymbol{y}}_0 + \sigma(t_j)\hat{\boldsymbol{\epsilon}}_i. \tag{13}$$

Here $\hat{\boldsymbol{\epsilon}}_i$ is a network prediction, not an independent random draw. For a fixed initial state, condition, and timestep sequence, the update is deterministic. Generation starts from a Gaussian state and uses decreasing noise levels, $t_j < t_i$. Inversion starts from encoded geometry and uses increasing levels, $t_j > t_i$; traversing the reverse sequence then approximately reconstructs the input.

D.2 GUIDED EDITING

For local editing, the predicted clean coefficients are $\boldsymbol{a}_{\text{pred}} = \mathcal{T}^{-1}(\hat{\boldsymbol{y}}_0)$. Following gradient-guided inference (Dhariwal & Nichol, 2021; Chung et al., 2022), we evaluate a differentiable loss on this

prediction. For prescribed points $\mathcal{P}_{\text{con}}$, an implicit-field penalty is

$$\mathcal{L}_{\text{geom}}(\hat{\boldsymbol{y}}_0; \mathcal{P}_{\text{con}}) = \frac{1}{|\mathcal{P}_{\text{con}}|} \sum_{\mathbf{p} \in \mathcal{P}_{\text{con}}} |f(\mathbf{p}; \mathcal{T}^{-1}(\hat{\boldsymbol{y}}_0))|^2. \tag{14}$$

The guided estimate is

$$\tilde{\boldsymbol{y}}_0 = \hat{\boldsymbol{y}}_0 - \eta_t \nabla_{\hat{\boldsymbol{y}}_0} \mathcal{L}_{\text{guide}}(\hat{\boldsymbol{y}}_0; \mathbf{c}_{\text{edit}}), \tag{15}$$

where η_t controls guidance strength and $\mathcal{L}_{\text{guide}}$ may be the geometric penalty above. We substitute $\tilde{\boldsymbol{y}}_0$ for $\hat{\boldsymbol{y}}_0$ in Eq. (13). This operation encourages the zero set to pass through the target points, but $|f|$ is not a Euclidean distance unless the field has the SDF property. Geometric evaluation therefore uses the actual point-to-surface distance defined in Appendix G. The editing examples in Figure 1 use target curves on selected cell faces and retain the overall morphology in the illustrated cases.

D.3 Reconstruction Evaluation

We invert and reconstruct 2,000 randomly selected training shapes and separately evaluate validation shapes. Table 3 reports the provided statistics. Representative reconstructions appear in Figure 9.

Table 3: **Inversion and reconstruction.** CD values use unsquared distances in a unit-period cell. The curvature columns average the per-surface mean and per-surface 99th percentile, respectively.

Split	Mean CD	99th-percentile CD	Mean $\lvert H\rvert_{\text{avg}}$	Mean $\lvert H\rvert_{\text{p99}}$
Training	3.71×10^{-3}	9.56×10^{-3}	1.43	9.18
Validation	3.82×10^{-3}	1.11×10^{-2}	1.47	9.31

E Additional Results

Figure 11 shows a gallary of randomly generated TPMS structures using our pipeline. Figure 12 shows the statistics of randomly generate TPMS structures. Figure 13 presents another set of conditional generation results.

F Ablation study

F.1 Post Curvature Refinement

We evaluate the effect of the mean-curvature refinement step described in Eq. (3). For each generated shape, we sample 8,192 vertices from the extracted mesh, evaluate the absolute mean curvature $|H|$, and backpropagate the corresponding loss to optimize the Fourier coefficients. We perform 80 optimization steps per shape with $\lambda_{\boldsymbol{a}} = 0.1$. The refinement takes approximately 4 seconds per shape. As shown in Table 1, curvature refinement reduces the average $|H|_{\text{avg}}$ from 1.00 to 0.67. At the same time, the genus remains unchanged, and the average CD to the original generated shape is only 1.60×10^{-3}, confirming that the optimization acts only as a local refinement of the generated surface. Selected visual comparisons are shown in Figure 15.

Compared with MCF in Euclidean space, our corefinement method better aligns with our representation and provides better performance in geometric preservation and efficiency. This is because 1) our space preserves D_{2h} symmetry, 2) the Euclidean space MCF requires linear system solve with DOF equal to the number of vertices, which is huge compared with ours. Our experiment shows that MCF requires approximately $10\times$ more computation time per shape than our refinement. The curvature refinement also preserves the conditioned material properties, with their values typically changing by only about 1%, thus introducing negligible drift from the specified conditions.

F.2 Higher D_{2h} Fourier dimension.

We study the effect of increasing the Fourier latent resolution from $Dim = 16$ to 24. The $Dim = 24$ representation contains approximately $3.4\times$ as many coefficients as the $Dim = 16$ representation.

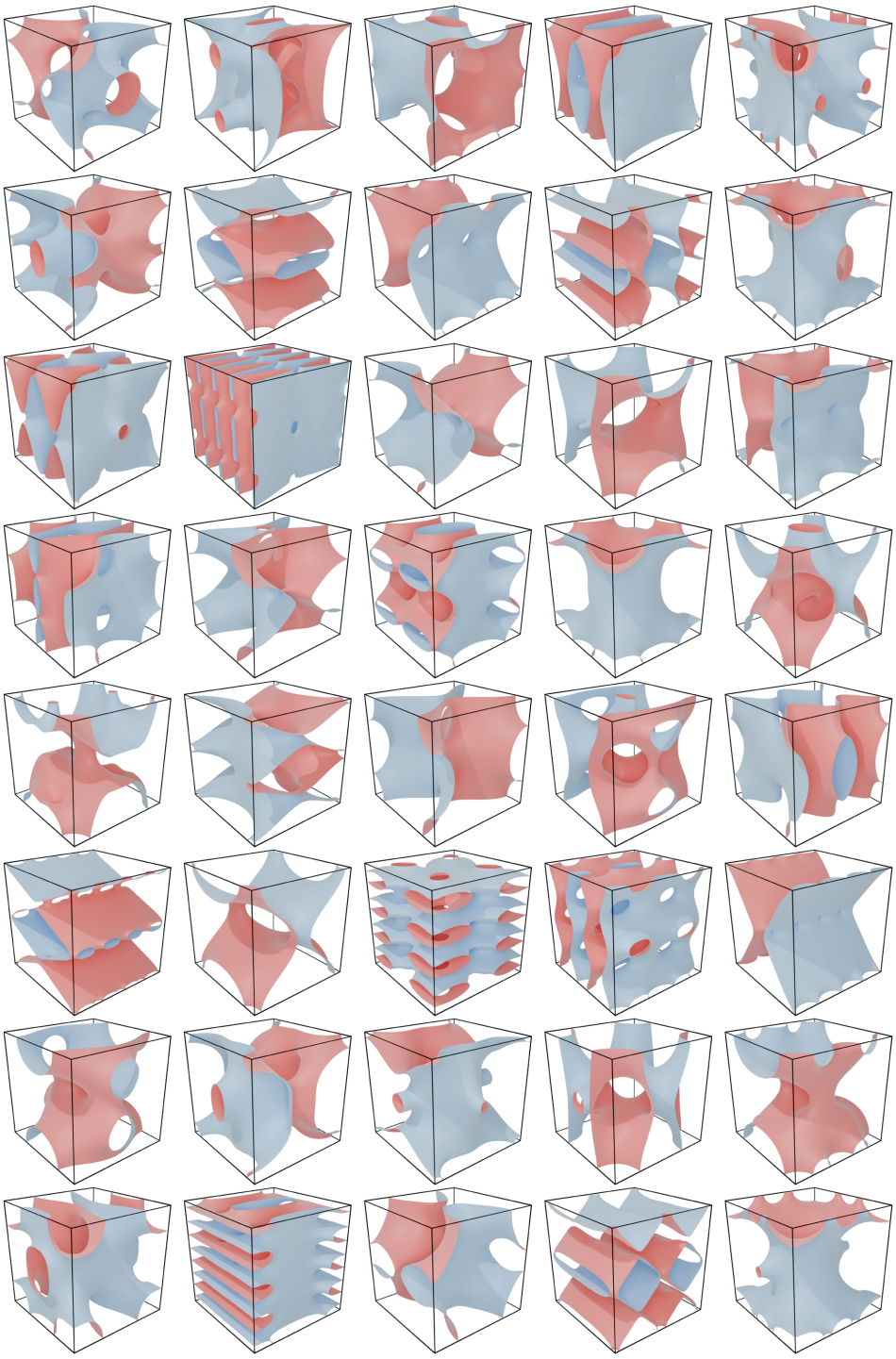


Figure 11: **Randomly generated TPMS.** Our model enables unconditional generation of TPMS from randomly initialized Gaussian noise. The generated shapes demonstrate rich geometric and topological diversity while preserving low mean curvature.

During training, we keep the patch partitioning strategy and all training hyperparameters unchanged. This higher-dimensional representation increases the training time by approximately $4\times$. However, the generation results show that the increased dimensionality does not improve overall performance.

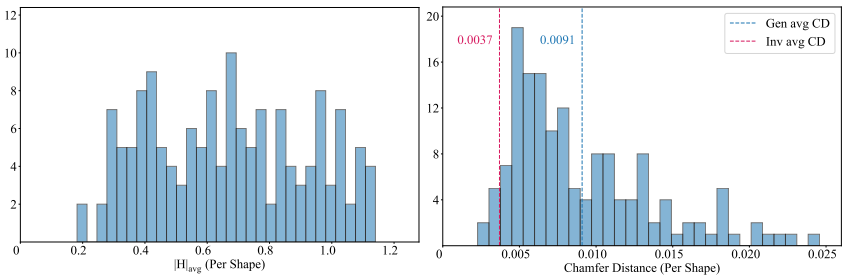


Figure 12: **Quantitative results for the random generation task.** We randomly select 300 generated shapes and analyze the 150 shapes with the lowest $|H|_{\text{avg}}$. Left: a large portion of the generated shapes have $|H|_{\text{avg}}$ <1, satisfying the curvature requirement of TPMS. Right: the minimum CD from the generated shapes to the training set is also much larger than that in the inversion task, demonstrating the diversity and novelty of the generated shapes rather than simple memorization of the training set.

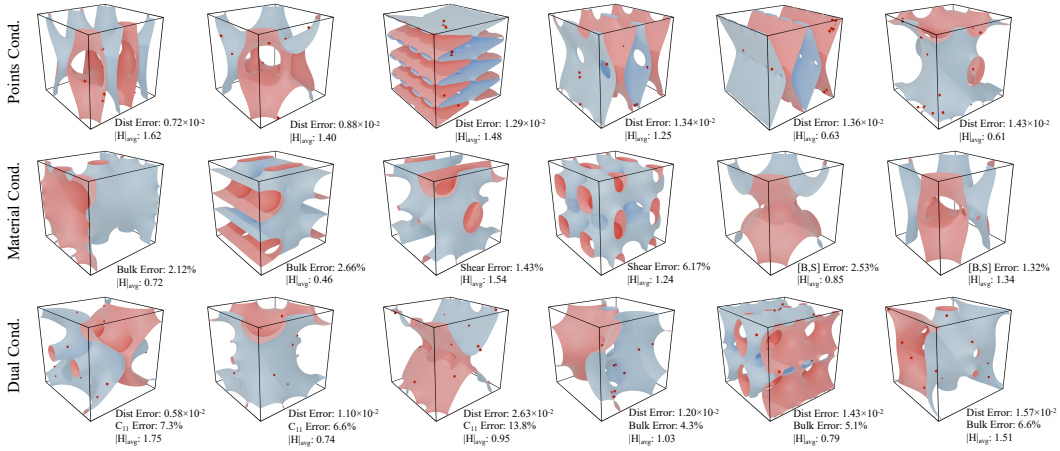


Figure 13: **Additional conditional generation results.** We present more examples under different target conditions. These results further demonstrate that our method consistently satisfies the prescribed conditions while producing diverse TPMS geometries with low mean curvature.

Although $Dim = 24$ slightly increases the minimum CD between randomly generated samples and the training shapes, suggesting marginally greater novelty, it results in more than $2\times$ worse average $|H|_{\text{avg}}$ compared with $Dim = 16$. These results suggest that the compact $Dim = 16$ representation provides better performance.

F.3 Standard Fourier Representation.

We trained a diffusion model on standard Fourier coefficients for comparison with our D_{2h} representation. To obtain a comparable minimum resolvable feature size, we used frequencies ranging from -15 to $+16$. Since the TPMS in our dataset are highly symmetric, the Sine coefficients produced by the standard Fourier transform are close to zero and are therefore set to zero before training. The standard Fourier representation requires 32^3 coefficients per shape, whereas our D_{2h} representation requires only 16^3 coefficients for a comparable feature resolution. All training settings were kept unchanged except for the coefficient block size, which was increased from 2^3 to 4^3 to maintain a comparable training time. As shown in Table 1, the standard Fourier representation achieves a similar maximum generated genus but performs worse on all other metrics, with its mean-curvature error more than twice as large as ours. These results demonstrate that our D_{2h} representation is substantially more compact and better suited to diffusion training.

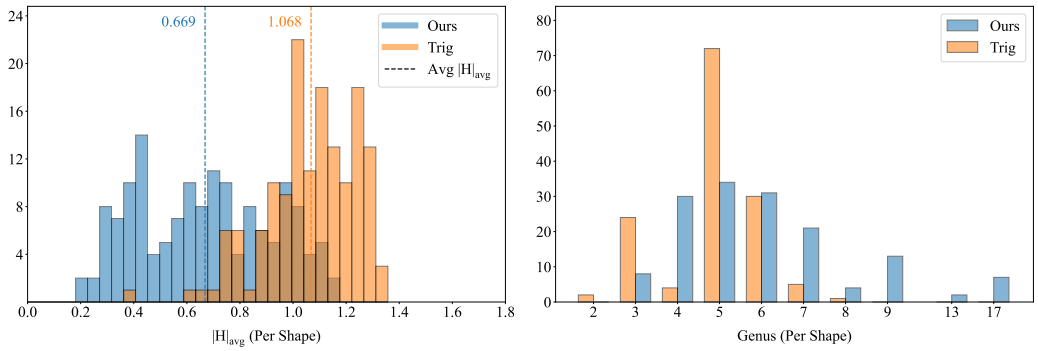


Figure 14: **Core metrics comparison between ours and the Trigonometric baseline.** Our method achieves significantly lower mean curvature while preserving richer high-frequency structures and higher topological diversity. These results demonstrate that our model substantially expands the expressiveness and complexity beyond traditional trigonometric functions.

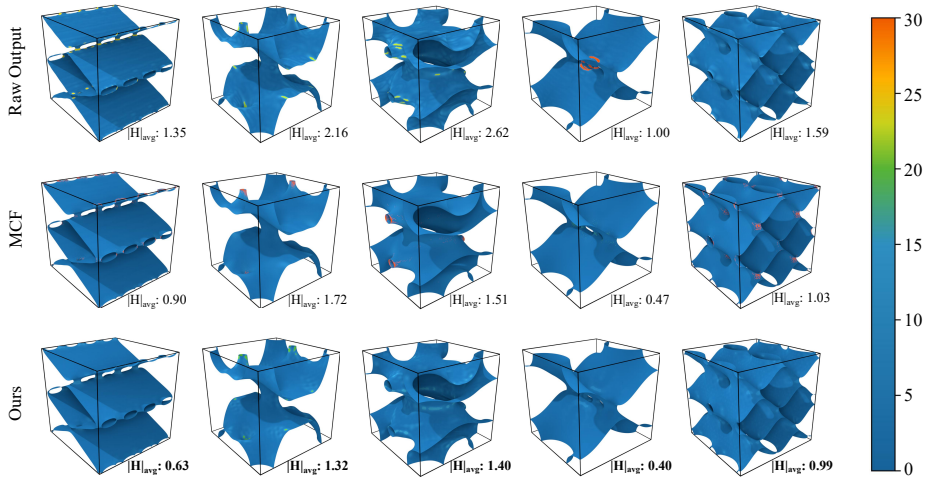


Figure 15: **Curvature colormaps for the curvature refinement ablation.** We compare the diffusion raw output (without curvature refinement), MCF, and our refinement method. Both MCF and our method effectively reduce the overall mean curvature. However, our method is $10\times$ faster than MCF.

F.4 Additional Training Regularizations

We considered adding Eikonal and mean-curvature losses during training to improve performance, but ultimately did not include either in the final training objective. The reasons for excluding these losses are discussed below.

Eikonal Loss. We use f as an implicit level-set function rather than a standard signed distance function (SDF). An Eikonal loss can be written as

$$\mathcal{L}_{\text{eik}} = \mathbb{E}_{\mathbf{p}} \left[\left(\|\nabla f(\mathbf{p})\|_2 - 1 \right)^2 \right]. \tag{16}$$

Although this loss encourages the SDF property, such a property is not required for defining the zero level set. It is also unnecessary for our curvature computation because the computation of the mean curvature explicitly normalizes ∇f and remains valid for any regular implicit field satisfying $\nabla f \neq \mathbf{0}$. We therefore do not use Eikonal loss, which would additionally require spatial sampling and gradient evaluation during diffusion training.

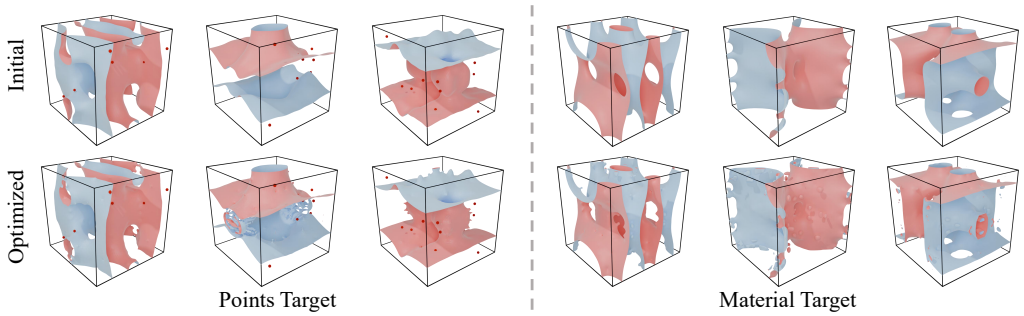


Figure 16: **Direct coefficient-space optimization.** We directly optimize the 16^3 D_{2h} Fourier coefficients to satisfy the conditioning constraints while reducing mean curvature. All resulting shapes exhibit invalid structures, highlighting the difficulty of balancing these two objectives through direct optimization.

Mean-Curvature Loss. We also tested adding a mean-curvature loss to the diffusion training objective:

$$\mathcal{L}_{\text{curv}} = \mathbb{E}_{\mathbf{p}\sim\mathcal{P}}\left[H_f(\mathbf{p};\hat{\mathbf{a}}_0)^2\right], \tag{17}$$

where $\hat{\mathbf{a}}_0$ denotes the predicted clean Fourier coefficients. This loss requires near-surface sampling and second-order spatial derivatives at every training iteration for every shape, substantially increasing the computational cost. From our experiments, it did not further reduce the mean curvature or improve the generated results. Moreover, the raw diffusion outputs already achieve an average $|H|_{\text{avg}}$ of 1.00, indicating that the diffusion model can learn the low-curvature TPMS distribution without an explicit mean-curvature loss. We therefore apply curvature optimization only as a lightweight postprocessing refinement.

F.5 Direct Fourier-Space Optimization

We additionally tested direct optimization in the 16^3 D_{2h} Fourier coefficient space to simultaneously satisfy the specified conditions and reduce mean curvature. For point conditioning, we adopted a decay-based initialization scheme following the empirical distribution of the Fourier coefficients. For bulk-modulus conditioning, we employed the inverse homogenization technique to optimize the coefficients toward a higher bulk modulus, initialized from the D_{2h} Fourier coefficients of a shape with a relatively low bulk modulus. As shown in Figure 16, direct optimization performed poorly in both settings. Both point and material constraints cannot be satisfied, and the residual mean curvature is high. We attribute this to two factors. First, without a learned structural prior, the high-dimensional 16^3 coefficient space is too broad to explore. Second, the conditioning and curvature objectives often conflict, making them difficult to satisfy simultaneously. These results highlight the effectiveness of our pipeline.

G Definition of Evaluation Metrics

We define the three error metrics used in the paper as follows.

Chamfer distance. Given two sets of points $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^{N_P}$ and $\mathcal{Q} = \{\mathbf{q}_j\}_{j=1}^{N_Q}$ sampled from two surfaces, their symmetric Chamfer distance is defined as

$$\text{CD}(\mathcal{P},\mathcal{Q}) = \frac{1}{2}\Big(\frac{1}{N_P}\sum_{i=1}^{N_P}\min_{1\le j\le N_Q}\|\mathbf{p}_i-\mathbf{q}_j\|_2 + \frac{1}{N_Q}\sum_{j=1}^{N_Q}\min_{1\le i\le N_P}\|\mathbf{q}_j-\mathbf{p}_i\|_2\Big). \tag{18}$$

We use the Euclidean distance rather than its squared value.

Point-to-surface distance. For a set of prescribed condition points $\mathcal{P}_{\text{con}}$ and a generated surface $\mathcal{S}$, we define

$$\text{Dist}(\mathcal{P}_{\text{con}},\mathcal{S}) = \frac{1}{|\mathcal{P}_{\text{con}}|}\sum_{\mathbf{p}\in\mathcal{P}_{\text{con}}}\min_{\mathbf{x}\in\mathcal{S}}\|\mathbf{p}-\mathbf{x}\|_2. \tag{19}$$

Unlike CD, this metric is one-directional and directly measures how closely the generated surface passes through the prescribed points.

Material-property relative error. For a target material property m^* and the corresponding property $\hat{m}$ computed from the generated structure, the relative error is

$$\mathrm{Err}_{\mathrm{prop}} = \frac{|\hat{m} - m^*|}{|m^*|} \times 100\%, \qquad \mathrm{Err}_{[\mathrm{B,S}]} = \frac{1}{2}(\mathrm{Err}_{\mathrm{bulk}} + \mathrm{Err}_{\mathrm{shear}}) \tag{20}$$

This definition is used for the bulk modulus, shear modulus, and C_{11}.For joint property conditioning, the relative error for each conditioned property is reported separately in Table 2, while their average is shown in Figures 1, 6, and 13.

H Computation of Surface Mean Curvature H

The diffusion model produces a normalized coefficient tensor $\hat{\mathbf{a}}$, we first denormalize it to obtain $\mathbf{a}$. Given $\mathbf{a}$, the continuous implicit field over one unit cell $\Omega = [0, L]^3$ is reconstructed as

$$f(\mathbf{p}; \mathbf{a}) = \sum_{h=0}^{K_{\max}} \sum_{k=0}^{K_{\max}} \sum_{\ell=0}^{K_{\max}} a_{hk\ell} \cos\left(\frac{2\pi h p_x}{L}\right) \cos\left(\frac{2\pi k p_y}{L}\right) \cos\left(\frac{2\pi \ell p_z}{L}\right). \tag{21}$$

The reconstructed surface is its zero-level set:

$$\mathcal{S}(\mathbf{a}) = \{\mathbf{p} \in \Omega : f(\mathbf{p}; \mathbf{a}) = 0\}. \tag{22}$$

We evaluate f on a regular 150^3 grid and apply Marching Cubes at isovalue zero. All metrics are evaluated over one period with $L = 1$.

The spatial gradient ∇f and Hessian $\nabla^2 f$ are evaluated analytically from the representation. For any point $\mathbf{p}$ on the zero-level surface, its unit normal is

$$\mathbf{n}(\mathbf{p}) = \frac{\nabla f(\mathbf{p}; \mathbf{a})}{\|\nabla f(\mathbf{p}; \mathbf{a})\|_2}. \tag{23}$$

The mean curvature is then given by

$$H(\mathbf{p}) = \frac{1}{2} \nabla \cdot \mathbf{n}(\mathbf{p}) = \frac{1}{2\|\nabla f\|_2^3} \Big[\|\nabla f\|_2^2 \Delta f - \nabla f^{\mathsf{T}} (\nabla^2 f) \nabla f \Big]. \tag{24}$$

where all derivatives are evaluated at $\mathbf{p}$. For numerical stability, we use a small constant $\delta > 0$ and evaluate

$$H_\delta(\mathbf{p}) = \frac{1}{2 \max(\|\nabla f\|_2^2, \delta)^{3/2}} \Big[\|\nabla f\|_2^2 \Delta f - \nabla f^{\mathsf{T}} (\nabla^2 f) \nabla f \Big]. \tag{25}$$

The curvature is evaluated at each vertex of the extracted zero-level surface. Since the formula explicitly normalizes ∇f, it is valid for any twice-differentiable implicit field satisfying $\nabla f \neq \mathbf{0}$ on the surface; therefore, f is not required to be a metric SDF. For a surface containing N_v evaluated vertices, the reported average absolute mean curvature is

$$|H|_{\mathrm{avg}} = \frac{1}{N_v} \sum_{i=1}^{N_v} |H_\delta(\mathbf{v}_i)|. \tag{26}$$

I Computation of Homogenized Elastic Properties

Each implicit TPMS surface is uniformly thickened to a thickness of 0.02 and discretized over one periodic unit cell. The solid phase is modeled as an isotropic linear elastic material with Young's modulus $E_s = 1.0$ and Poisson's ratio $\nu_s = 0.3$. We use the same periodic finite-element homogenization solver for both the training data and the generated samples.

The solver returns the homogenized stiffness tensor $\mathbf{C}^{\mathrm{H}} \in \mathbb{R}^{6\times 6}$ using the Voigt ordering

$$(11, 22, 33, 12, 23, 13) \longleftrightarrow (1, 2, 3, 4, 5, 6). \tag{27}$$

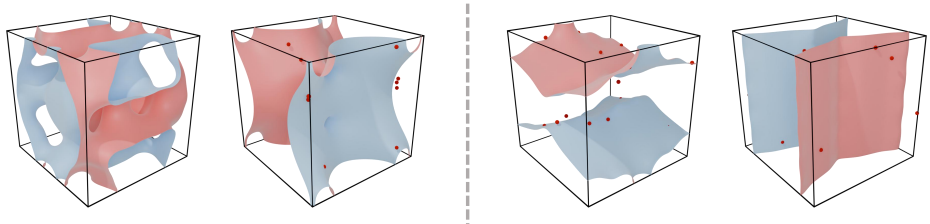


Figure 17: **Failure cases under unseen conditions.** Left: the generated surfaces fail to fully satisfy the unseen target point conditions. Right: conflicts between the local point constraints and the global material-property targets result in disconnected or degenerate surfaces.

Since the generated structures possess D_{2h} symmetry, their homogenized responses are orthotropic with respect to the coordinate axes. Up to numerical error, the stiffness tensor has the form

$$\mathbf{C}^{\mathrm{H}} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix}. \tag{28}$$

To obtain the scalar bulk and shear modulus from $\mathbf{C}^{\mathrm{H}}$, we use the Voigt–Reuss–Hill average. Let

$$\mathbf{S}^{\mathrm{H}} = (\mathbf{C}^{\mathrm{H}})^{-1} \tag{29}$$

denote the corresponding compliance matrix. The Voigt estimates are

$$\begin{aligned} K_{\mathrm{V}} &= \frac{1}{9}\Big[C_{11} + C_{22} + C_{33} + 2(C_{12} + C_{13} + C_{23})\Big], \\ G_{\mathrm{V}} &= \frac{1}{15}\Big[C_{11} + C_{22} + C_{33} - C_{12} - C_{13} - C_{23} + 3(C_{44} + C_{55} + C_{66})\Big]. \end{aligned} \tag{30}$$

The corresponding Reuss estimates are

$$\begin{aligned} K_{\mathrm{R}} &= \Big[S_{11} + S_{22} + S_{33} + 2(S_{12} + S_{13} + S_{23})\Big]^{-1}, \\ G_{\mathrm{R}} &= 15\Big/\Big[4(S_{11} + S_{22} + S_{33} - S_{12} - S_{13} - S_{23}) + 3(S_{44} + S_{55} + S_{66})\Big]. \end{aligned} \tag{31}$$

Finally, the bulk and shear modulus used in our experiments are the Hill averages

$$K = \frac{K_{\mathrm{V}} + K_{\mathrm{R}}}{2}, \qquad G = \frac{G_{\mathrm{V}} + G_{\mathrm{R}}}{2}. \tag{32}$$

We select bulk modulus, shear modulus, and C_{11} as representative low-dimensional conditions because they capture complementary mechanical responses while allowing diverse geometries to share similar target values. Joint [bulk, shear] conditioning further evaluates simultaneous control of volumetric and shear responses. In contrast, the full 9- or 12-dimensional property vector is nearly unique to each shape and sparsely distributed in the high-dimensional space, previously unseen vectors may easily fall outside the training distribution and produce invalid geometries.

J Failure Cases under Joint Conditioning

Our method performs well in most cases, but failures exist as shown in Figure 17. In joint conditioning, the local point constraints may conflict with the material-property targets, causing the generated geometry to become disconnected or degenerate. These examples highlight the difficulty of simultaneously satisfying unseen geometric and material conditions while preserving valid geometry and low mean curvature.