

GAN-DUF: Hierarchical Deep Generative Models for Design Under Free-Form Geometric Uncertainty

Wei (Wayne) Chen, Doksoo Lee, Oluwaseyi Balogun, Wei Chen*

Dept. of Mechanical Engineering

Northwestern University

Evanston, IL 60208

Email: {wei.wayne.chen, dslee, o-balogun, weichen}@northwestern.edu

Deep generative models have demonstrated effectiveness in learning compact and expressive design representations that significantly improve geometric design optimization. However, these models do not consider the uncertainty introduced by manufacturing or fabrication. Past work that quantifies such uncertainty often makes simplifying assumptions on geometric variations, while the “real-world”, “free-form” uncertainty and its impact on design performance are difficult to quantify due to the high dimensionality. To address this issue, we propose a Generative Adversarial Network-based Design under Uncertainty Framework (GAN-DUF), which contains a deep generative model that simultaneously learns a compact representation of nominal (ideal) designs and the conditional distribution of fabricated designs given any nominal design. This opens up new possibilities of 1) building a universal uncertainty quantification model compatible with both shape and topological designs, 2) modeling free-form geometric uncertainties without the need to make any assumptions on the distribution of geometric variability, and 3) allowing fast prediction of uncertainties for new nominal designs. We can combine the proposed deep generative model with robust design optimization or reliability-based design optimization for design under uncertainty. We demonstrated the framework on two real-world engineering design examples and showed its capability of finding the solution that possesses better performance after fabrication.

1 INTRODUCTION

Many engineering design problems boil down to geometric optimization. However, geometric optimization remains a grand challenge because of its extreme dimensional complexity and often hard-to-achieve performance objective. Recent work has shown that deep generative models can learn a compact¹ and expressive design representation that remarkably improves geometric design optimization performances (indicated by both the quality of optimal solutions and the computational cost) [1, 2, 3]. However, past work based on deep generative models only considers the ideal scenario where manufacturing or fabrication imperfections do not occur, which is unrealistic due to the existence of uncertainties in reality, such as limited tool precision or wear. Such imperfections sometimes have a high impact on a design’s performance or properties. Consequently, the originally optimal solution (obtained by only considering the ideal scenario) might not possess high performance or desired properties after fabrication.

Past work has developed task-specific robust optimization techniques to identify geometric design solutions that are insensitive to variations of load, materials, and geometry [4, 5, 6]. However, due to the lack of generalizable uncertainty representation that is compatible with different geometric representations, previous work often makes simplifying assumptions on geometric variations (e.g., the distribution or the upper/lower bound of uncertain parameters), while the “real-world”, “free-form”

* Address all correspondence to this author.

¹Here “compact” means (1) the learned design representation has a much lower dimensionality than the original representation and (2) the learned representation contains mostly valid designs.

geometric uncertainty and its impact on design performance are difficult to quantify due to the high dimensionality². In this paper, we propose a *Generative Adversarial Network-based Design under Uncertainty Framework (GAN-DUF)* to allow uncertainty quantification (UQ) of free-form geometric variability under real-world scenarios. The term “free-form” refers to two aspects: 1) the geometric variability has no shape or topological restrictions and 2) no assumption on the form of uncertainty is needed. Therefore, this framework is generalizable to any shape or topological designs. It improves existing geometric design under uncertainty from four aspects: 1) The generative adversarial network (GAN) uses a compact representation to reparameterize geometric designs, allowing accelerated optimization; 2) The GAN associates real-world, free-form fabrication uncertainty with ideal designs (*i.e.*, *nominal designs*) by learning a conditional distribution of fabricated designs given any nominal design; 3) The optimization process accounts for the distribution of geometric variability underlying any manufacturing processes, and allows UQ for robust design optimization or reliability-based design optimization; 4) The compact representation of nominal designs allows gradient-free global optimization due to the representation’s low-dimensionality.

We list the contributions of this work as follows:

1. We propose a hierarchical deep generative model to simultaneously learn a compact representation of designs and quantify their real-world, free-form geometric uncertainties.
2. We combine the proposed model with a robust design optimization framework and demonstrate its effectiveness on two realistic robust design examples.
3. We build two benchmark datasets, containing nominal and fabricated designs, which will facilitate future studies on data-driven design under manufacturing uncertainty.

The rest of the paper is organized as follows: Section 2 introduces two core concepts used in our current work — the GAN and design under uncertainty, including a brief review of related past work on design under geometric uncertainty. Section 3 describes the method in detail. Section 4 demonstrates the effectiveness of the proposed method by using two case studies. Section 5 concludes the paper and discusses future work.

²Due to the free-form nature of the geometry and its uncertainty, there is no easy way to parameterize them using lower-dimensional parameters. This is the reason for their high dimensionality.

2 BACKGROUND

In this section, we introduce Generative Adversarial Networks and design under uncertainty.

2.1 Generative Adversarial Networks

The generative adversarial network [7] models a game between a *generator* G and a *discriminator* D . The goal of G is to generate samples (designs in our case) that resemble those from data; while D tries to distinguish between real data and generated samples by predicting the probability of a sample being real data. Both models improve during training via the following minimax optimization:

$$\min_G \max_D \mathbb{E}_{\mathbf{x} \sim P_{\text{data}}} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim P_{\mathbf{z}}} [\log(1 - D(G(\mathbf{z})))] \quad (1)$$

where \mathbb{E} denotes expectation, P_{data} is the data distribution, $P_{\mathbf{z}}$ is a predefined noise distribution (*e.g.*, a standard normal distribution), $\mathbf{z} \sim P_{\mathbf{z}}$ is the noise that serves as G ’s input, $G(\mathbf{z})$ denotes the generated sample, and \mathbf{x} denotes the sample from a training dataset. In this paper, \mathbf{x} represents geometric designs (*e.g.*, pixelated, voxelated, point cloud, point sequence, or surface mesh representation, usually with high-dimensionality). A trained generator thus can map from a predefined noise distribution to the distribution of designs. Due to the lower dimensionality of \mathbf{z} compared to the original design representation \mathbf{x} , we can use \mathbf{z} to more efficiently control the geometric variation of high-dimensional designs.

Despite the ability to generate high-dimensional data from low-dimensional noise, standard GANs do not have a way of regularizing the noise; so it usually cannot reflect an intuitive design variation, which is unfavorable in many design applications. To compensate for this weakness, the InfoGAN adds the *latent codes* \mathbf{c} as G ’s another input and maximizes the mutual information between \mathbf{c} and $G(\mathbf{c}, \mathbf{z})$, as a way of regularizing \mathbf{c} and obtaining an interpretable and disentangled latent representation [8]. The generated design is now expressed as $G(\mathbf{c}, \mathbf{z})$. Maximizing the mutual information prevents the information loss of \mathbf{c} in the generation process. However, directly maximizing the mutual information is difficult, the InfoGAN instead maximizes the mutual information lower bound L_I :

$$L_I(G, Q) = \mathbb{E}_{\mathbf{c} \sim P(\mathbf{c}), \mathbf{x} \sim G(\mathbf{c}, \mathbf{z})} [\log Q(\mathbf{c}|\mathbf{x})] + H(\mathbf{c}), \quad (2)$$

where $H(\mathbf{c})$ is the entropy of the latent codes, and Q is the auxiliary distribution for approximating $P(\mathbf{c}|\mathbf{x})$. The

InfoGAN's training objective becomes:

$$\min_{G,Q} \max_D \mathbb{E}_{\mathbf{x} \sim P_{\text{data}}} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{c} \sim P_{\mathbf{c}}, \mathbf{z} \sim P_{\mathbf{z}}} [\log(1 - D(G(\mathbf{c}, \mathbf{z}))) - \lambda L_I(G, Q)], \quad (3)$$

where λ is a weight parameter. In practice, $H(\mathbf{c})$ is usually treated as a constant as $P_{\mathbf{c}}$ is fixed. Please see [8] for more details about InfoGAN.

Past works studied how the maximization of L_I in InfoGAN affects the latent representation of designs [1, 9]. Specifically, Ref. [1] showed that by maximizing the mutual information lower bound between the latent vector and the generated designs, the latent vector \mathbf{c} encoded major geometric variation in the data, while the noise vector \mathbf{z} encoded minor geometric variation. Ref. [9] showed that without maximizing L_I , the latent vector \mathbf{c} either failed to capture major geometric variation or led to inconsistent geometric variation along latent space bases. The ability to capture major geometric variation is necessary, since the generator needs to cover the complete range of both the nominal and the fabricated designs. The property of consistent geometric variation is also essential for designers to explore the latent space in an intuitive way and disentangle the objective function of a design problem.

Since InfoGAN provides an interpretable and disentangled latent representation that is also compact and low-dimensional, searching for design solutions in this latent space is much more efficient than searching in the original high-dimensional design space [9, 1, 2, 3]. Building on the InfoGAN model, this work proposes a new deep generative model that constructs a hierarchical latent representation — the *parent latent representation* encodes nominal design variation and the *child latent representation* encodes fabricated design variation. In this way, we can simultaneously model 1) the compact latent representation of nominal designs and 2) the distribution of fabricated designs conditioned on any nominal design. We will elaborate on our proposed hierarchical generative model in Sec. 3.1.

2.2 Design under Uncertainty

Design under uncertainty aims to identify optimal designs that are robust and/or reliable under the variations associated with various sources of uncertainties (e.g., material, geometry, and operating conditions) [10, 11]. Two common approaches are robust design optimization (RDO) [12] and reliability-based design optimization (RBDO) [13, 14].

We assume that the design is represented by a vector of deterministic design variables \mathbf{b} and a vector of random variables $\boldsymbol{\xi}$. The vector $\boldsymbol{\xi}$ represents the sources of uncertainty (e.g., noise or control factor). The goal of RDO is to minimize the effects of variation without eliminating the sources of uncertainties [12]. Given a performance function $f(\cdot)$, RDO approaches simultaneously maximize the mean performance $\mu(f(\mathbf{b}, \boldsymbol{\xi}))$ and minimize the variance of the cost $\sigma^2(f(\mathbf{b}, \boldsymbol{\xi}))$ over $\boldsymbol{\xi}$. The design goal, in general, involves the following optimization problem:

$$\min_{\mathbf{b}} J_{\text{RDO}}(\mathbf{b}, \boldsymbol{\xi}) = -F(\mu(f(\mathbf{b}, \boldsymbol{\xi})), \sigma(f(\mathbf{b}, \boldsymbol{\xi}))), \quad (4)$$

where F is the multi-objective cost function that is typically formulated as

$$F(\cdot) = \mu(\cdot) - k\sigma(\cdot), \quad (5)$$

where $k > 0$ is the tuning parameter. $\mu(f(\cdot))$ and $\sigma^2(f(\cdot))$ are the statistical moments of $f(\cdot)$ w.r.t. the associated uncertainty $\boldsymbol{\xi}$ and can be expressed as:

$$\begin{aligned} \mu(f(\mathbf{b}, \boldsymbol{\xi})) &= \mathbb{E}_{\boldsymbol{\xi}}[f(\mathbf{b}, \boldsymbol{\xi})] = \int_{\boldsymbol{\xi}} p(\boldsymbol{\xi}) f(\mathbf{b}, \boldsymbol{\xi}) d\boldsymbol{\xi}, \\ \sigma^2(f(\mathbf{b}, \boldsymbol{\xi})) &= \mathbb{E}_{\boldsymbol{\xi}}[(f(\mathbf{b}, \boldsymbol{\xi}) - \mu(f(\mathbf{b}, \boldsymbol{\xi})))^2] \\ &= \int_{\boldsymbol{\xi}} p(\boldsymbol{\xi}) (f(\mathbf{b}, \boldsymbol{\xi}) - \mu(f(\mathbf{b}, \boldsymbol{\xi})))^2 d\boldsymbol{\xi}. \end{aligned} \quad (6)$$

Instead of Eq. (5), we can also use one of $P(f(\mathbf{b}, \boldsymbol{\xi}))$'s quantiles as $F(\cdot)$. The 100% quantile represents the “worst-case” scenario, while other less conservative quantile orders are also employed [15].

On the other hand, RBDO refers to the optimization scheme where reliability analysis is incorporated into deterministic optimization methods [14]. Herein, reliability is defined as the probability that a system is expected to successfully operate under risks of interest, such as deflection, leakage, and local damage. RBDO approaches exploit stochastic methods to address the statistical nature of constraints and design problems. Given m risk factors, a representative formulation of RBDO reads:

$$\begin{aligned} \min_{\mathbf{b}} \quad & J_{\text{RBDO}}(\mathbf{b}, \mu_{\boldsymbol{\xi}}) = -f(\mathbf{b}, \mu_{\boldsymbol{\xi}}) \\ \text{s.t.} \quad & P_j[g_j(\mathbf{b}, \boldsymbol{\xi}) \geq 0] \geq R_j, \quad j = 1, \dots, m, \end{aligned} \quad (7)$$

where $\mu_{\boldsymbol{\xi}}$ is the mean of $\boldsymbol{\xi}$, $g_j(\cdot)$ denotes the j -th limit-state function that indicates the margin of safety with respect to the j -th risk factor, and R_j is the specified reliability level with respect to the j -th factor. Given the

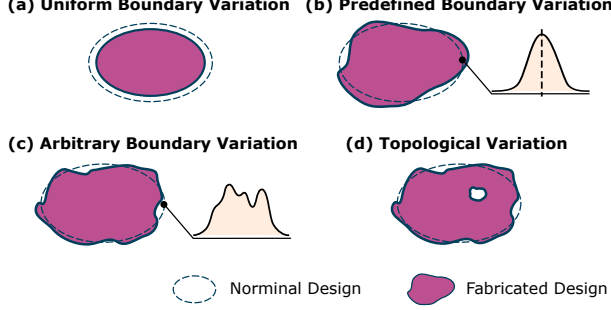


Fig. 1. Types of geometric uncertainty modeling: (a) Uniform boundary variation where the boundary of the geometry is uniformly “eroded” (*e.g.*, over-etched) or “dilated” (*e.g.*, under-etched) [6, 16, 17]; (b) Predefined boundary variation where the distribution or the correlation of boundary points is predefined [5]; (c) Arbitrary boundary variation where no assumption is imposed on the distribution of boundary points; (d) Topological variation where the design’s topological change (*e.g.*, hole nucleation) is also possible. To the best of our knowledge, past work only considers (a) and (b) when modeling geometric uncertainty, while our proposed method can address “free-form” uncertainties that include all four cases.

j -th factor, $g_j(\cdot) < 0$ and $g_j(\cdot) \geq 0$ denote the associated failure region and safe region, respectively.

Both approaches have been developed for design optimization under geometric uncertainty at various levels of geometric complexity (*i.e.*, size, shape, and topology). Unlike parametric uncertainty that only considers the uncertainties of a few independent design parameters representing the design [18, 19], geometric uncertainty can account for the uncertainties at every point on the boundary of a geometry as well as its topological changes. Due to the high-dimensionality of variables to be considered in geometric uncertainty quantification, previous work either assumes uniform boundary variation [6, 16, 17] (Fig. 1a) or imposes predefined distribution/correlation on boundary points [5, 20, 21, 22, 23] (Fig. 1b). Specifically, [5] models geometric uncertainty by perturbing the boundary using a random normal velocity field; [20, 21, 22, 23] model geometric uncertainty by applying a predefined random field as the threshold of a level-set design representation; [24, 25, 26, 27] build shape deviation models and use Bayesian approaches to estimate the posterior of model parameters. While those methods can simplify geometric uncertainty quantification by making assumptions on the form of shape deviations, the modeled uncertainties do not necessarily conform to realistic scenarios, which usually involve much

more complicated geometric variability. For example, in real applications, the boundary variation does not necessarily follow canonical distributions (Fig.s 1c) and manufacturing defects do not only happen on the boundary [28] (Fig. 1d). Therefore, how to model “real-world”, “free-form” geometric uncertainty without making any simplifying assumptions is still an open challenge.

In this work, we overcome this challenge by using a hierarchical deep generative model to learn 1) the underlying distribution of free-form nominal designs and 2) the conditional distribution of fabricated design given any nominal design, under free-form geometric uncertainty (Figures 1c and 1d). We demonstrate the efficacy using two real-world design examples. The ability to model free-form geometry and uncertainties allows us to address topological uncertainties.

3 METHODOLOGY

In this section, we introduce our Generative Adversarial Network-based Design under Uncertainty Framework (GAN-DUF). It consists of two parts: 1) A generative adversarial network to learn a hierarchical latent representation that compactly captures both the variability of nominal designs and the uncertainty of any given nominal design³ (Sec. 3.1); 2) A design under uncertainty module that uses the trained generator to perform robust design optimization or reliability-based design optimization (Sec. 3.2).

3.1 Quantifying Uncertainty using Generative Adversarial Networks

Let \mathcal{I}_{nom} and \mathcal{I}_{fab} denote the datasets of nominal and fabricated designs, respectively:

$$\mathcal{I}_{\text{nom}} = \{\mathbf{x}_{\text{nom}}^{(1)}, \dots, \mathbf{x}_{\text{nom}}^{(N)}\}$$

$$\mathcal{I}_{\text{fab}} = \left\{ \left(\mathbf{x}_{\text{fab}}^{(1,1)}, \dots, \mathbf{x}_{\text{fab}}^{(1,M)} \right), \dots, \left(\mathbf{x}_{\text{fab}}^{(N,1)}, \dots, \mathbf{x}_{\text{fab}}^{(N,M)} \right) \right\},$$

where $\mathbf{x}_{\text{fab}}^{(i,j)}$ is the j -th realization (fabrication) of the i -th nominal design. The **goals** are to 1) learn a lower-dimensional, compact representation \mathbf{c} of nominal designs to allow accelerated design optimization and 2) learn the conditional distribution $P(\mathbf{x}_{\text{fab}}|\mathbf{c})$ to allow the quantification of manufacturing uncertainty at any given nominal design (represented by \mathbf{c}).

To achieve these two goals, we propose a generative adversarial network (Fig. 2a) that enables the hierarchical

³Code and data are available at <https://github.com/wchen459/GAN-DUF>.

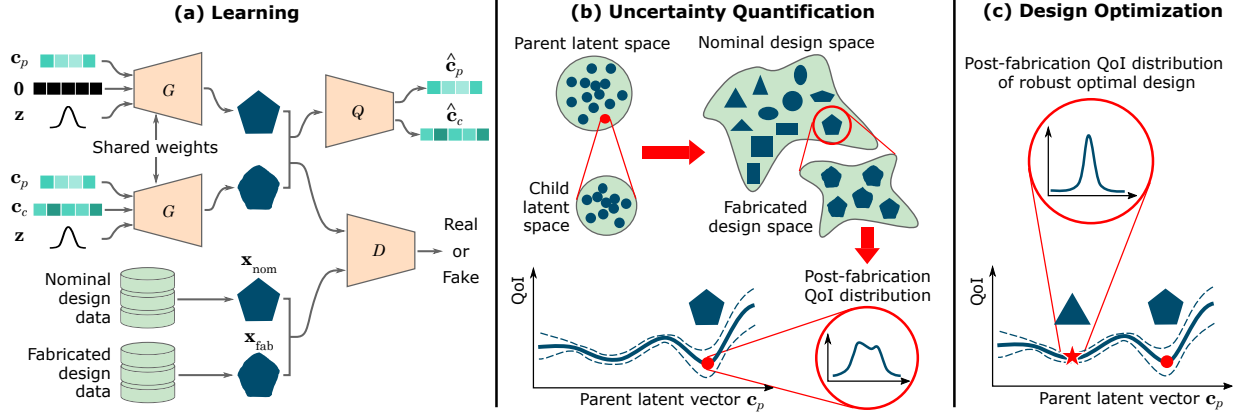


Fig. 2. Illustration of proposed Generative Adversarial Network-based Design under Uncertainty Framework (GAN-DUF): (a) The proposed Hierarchical GAN architecture for simultaneously learning the compact representation of nominal designs and the conditional distributions of fabricated designs; (b) Fabricated designs can be generated by sampling \mathbf{c}_c at any fixed \mathbf{c}_p representing a nominal design. The uncertainty of a nominal design's QoIs can be quantified by evaluating the QoIs of these generated fabricated designs via simulation or experiments; (c) We can optimize \mathbf{c}_p to obtain a nominal design \mathbf{x}_{nom} with desired post-fabrication QoIs.

modeling of nominal designs and fabricated designs. Its generator G generates fabricated designs when feeding in the parent latent vector \mathbf{c}_p , the child latent vector \mathbf{c}_c , and the noise \mathbf{z} ; whereas it generates nominal designs simply by using the same generator G but setting $\mathbf{c}_c = \mathbf{0}$. By doing this, we can control the generated nominal designs through \mathbf{c}_p and the generated fabricated designs through \mathbf{c}_c . Note that the noise \mathbf{z} represents the portion of geometric variation that cannot be captured by either the parent latent vector or the child latent vector. This is useful to be included as one of the inputs to the generator because sometimes the data contain noise (*e.g.*, noisy or blurred images of fabricated samples) that is irrelevant to either the variation of nominal geometries or the fabrication imperfection. When using the trained model to generate nominal or fabricated designs, \mathbf{z} is set to $\mathbf{0}$ to eliminate the effect of noise.

Given the pair of a generated nominal design $G(\mathbf{c}_p, \mathbf{0}, \mathbf{z})$ and a generated fabricated design $G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z})$, the discriminator D predicts whether the pair is generated or drawn from data (*i.e.*, \mathcal{I}_{nom} and \mathcal{I}_{fab}). Similar to InfoGAN, we also predict the conditional distribution $Q(\mathbf{c}_p, \mathbf{c}_c | \mathbf{x}_{\text{nom}}, \mathbf{x}_{\text{fab}})$ to promote disentanglement of latent spaces and ensure the latent spaces capture major geometric variability [1]. The GAN is trained using

the following loss function:

$$\min_{G, Q} \max_D \mathbb{E}_{\mathbf{x}_{\text{nom}}, \mathbf{x}_{\text{fab}}} [\log D(\mathbf{x}_{\text{nom}}, \mathbf{x}_{\text{fab}})] + \mathbb{E}_{\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}} [\log(1 - D(G(\mathbf{c}_p, \mathbf{0}, \mathbf{z}), G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}))) - \lambda \mathbb{E}_{\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}} [\log Q(\mathbf{c}_p, \mathbf{c}_c | G(\mathbf{c}_p, \mathbf{0}, \mathbf{z}), G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}))]]. \quad (8)$$

As a result, G decouples the variability of the nominal and the fabricated designs by using \mathbf{c}_p to represent the nominal design (**Goal 1**) and \mathbf{c}_c to represent the fabricated design of any nominal design. By fixing \mathbf{c}_p and sampling from the prior distribution of \mathbf{c}_c , we can produce the conditional distribution $P(\mathbf{x}_{\text{fab}} | \mathbf{c}_p) = P(G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}) | \mathbf{c}_p)$ (**Goal 2**). As a neural network, the generator has the flexibility of generating fabricated designs completely based on real data, without making any unrealistic assumptions on the types of uncertainties.

Compared to existing uncertainty quantification methods, this GAN-based model opens up possibilities of 1) building a universal UQ model compatible with both shape and topological designs, 2) modeling *free-form* geometric uncertainties without the need to make any assumptions on the distribution of geometric variability, and 3) allowing fast prediction of uncertainties for new nominal designs.

The trained generator allows us to sample fabricated designs given any nominal design, simply by sampling the low-dimensional \mathbf{c}_c with a fixed \mathbf{c}_p representing the nominal design (Fig. 2b). We can then evaluate the quantities of interest (QoIs) of these generated fabricated de-

signs using computational (e.g., physics simulation such as finite element methods, wave analysis, and computational fluid dynamics) or experimental methods. The QoIs may include performance, quality, properties, and/or cost. The resulted QoI distribution (*i.e.*, *post-fabrication QoI distribution*) allows us to quantify the uncertainty of QoIs for the nominal design. Note that the proposed framework is agnostic to both the type of designs (e.g., how designs are represented or what geometric variability is presented) and downstream tasks like design evaluation and design optimization.

3.2 Design Optimization under Uncertainty using Trained Generator

We can integrate the evaluated uncertainty into optimization frameworks such as robust optimization, where we simultaneously optimize the mean QoIs and minimize the influence of uncertainty [12] (Fig. 2c), as well as reliability-based optimization, where we optimize the QoIs subject to constraints such as the probability of failure or reliability index [14]. The solution is expected to maintain high real-world performance/quality, desired properties, or a lower chance of failure even under fabrication imperfection.

Specifically, in robust design optimization, the formulation of Eq. (4) is modified as

$$\min_{\mathbf{c}_p} -F(\mu(f(\mathbf{x})), \sigma(f(\mathbf{x}))), \quad (9)$$

where $\mathbf{x} = G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{0})$. Here \mathbf{c}_p and \mathbf{c}_c corresponds to the vector of deterministic design variables and random variables, respectively. In reliability-based design optimization, the formulation of Eq. (7) is modified as

$$\begin{aligned} \min_{\mathbf{c}_p} & -f(G(\mathbf{c}_p, \mathbf{0}, \mathbf{0})) \\ \text{s.t. } & P_j [g_j(\mathbf{x}) \geq 0] \geq R_j, \quad j = 1, \dots, m, \end{aligned} \quad (10)$$

where $g_j(\mathbf{x}) \geq 0$ denotes that the j -th QoI is within an acceptable range. For example, the QoI can be the performance deviation from a given target performance.

In both formulations, we only need to optimize \mathbf{c}_p , which has a much lower dimensionality than the original design representation. The functionality of \mathbf{c}_c is to introduce stochasticity and it reflects the sources of uncertainty.

4 RESULTS

We use two real-world robust design examples to demonstrate the effectiveness of our proposed framework.

Ideally, to obtain fabricated design data \mathcal{I}_{fab} , we can take the nominal designs from \mathcal{I}_{nom} , fabricate them, and use the actual fabricated designs as data. However, in this study, we simulate the fabrication effects by deforming the geometry of nominal designs based on the following approaches, as a way to save time and cost. Note that how well the simulated manufacturing uncertainty resembles the real-world uncertainty is not central to this proof of concept study. We treat the simulated uncertainty as the real uncertainty only to demonstrate our design under uncertainty framework. In the ideal scenario, we can directly use the real-world fabricated designs to build \mathcal{I}_{fab} and our proposed framework can still model the fabricated design distribution, since the framework is agnostic to the form of uncertainty. Also note that the required amount of data and latent vector dimensions will depend on the complexity level of geometric variation in data. For example, if the fabricated designs have a higher variation, we may need more fabricated design data and a higher-dimensional child latent vector to maintain the same level of accuracy for modeling the uncertainty.

4.1 Case Study: Airfoil Design

An airfoil is the cross-sectional shape of an airplane wing or a propeller/rotor/turbine blade. The shape of the airfoil determines the aerodynamic performance of a wing or a blade. By optimizing the airfoil shapes, we can improve their aerodynamic performance, minimize fuel consumption, and reduce greenhouse gas emissions.

4.1.1 Dataset Construction

We use the UIUC airfoil database⁴ as our nominal design dataset \mathcal{I}_{nom} . The preprocessing of \mathcal{I}_{nom} and the creation of the fabricated design dataset \mathcal{I}_{fab} are described as follows:

Nominal design data. The original UIUC database contains invalid airfoil shapes and the number of surface coordinates representing each airfoil is inconsistent. Therefore, we used the preprocessed data from Chen *et al.* [1] so that outliers are removed and each airfoil is consistently represented by 192 surface points (*i.e.*, $\mathbf{x}_{\text{nom}} \in \mathbb{R}^{192 \times 2}$).

Fabricated design data. For airfoil designs, we simulate the effect of manufacturing uncertainty by randomly perturbing the free-form deformation (FFD) control points of each airfoil design from \mathcal{I}_{nom} [29]. Specifically, the original FFD control points fall on a 3×8 grid

⁴http://m-selig.ae.illinois.edu/ads/coord_database.html

and are computed as follows:

$$\mathbf{P}_{\text{nom}}^{l,m} = \left(x_{\text{nom}}^{\min} + \frac{l}{7}(x_{\text{nom}}^{\max} - x_{\text{nom}}^{\min}), y_{\text{nom}}^{\min} + \frac{m}{2}(y_{\text{nom}}^{\max} - y_{\text{nom}}^{\min}) \right),$$

$$l = 0, \dots, 7 \text{ and } m = 0, \dots, 2, \quad (11)$$

where x_{nom}^{\min} , x_{nom}^{\max} , y_{nom}^{\min} , and y_{nom}^{\max} define the 2D minimum bounding box of the design \mathbf{x}_{nom} . To create fabricated designs, we add Gaussian noise $\epsilon \sim \mathcal{N}(0, 0.02)$ to the y -coordinates of control points except those at the left and the right ends. This results in a set of deformed control points $\{\mathbf{P}_{\text{fab}}^{l,m} | l = 0, \dots, 7; m = 0, \dots, 2\}$. The airfoil shape also deforms with the new control points and is considered as a fabricated design. The surface points of fabricated airfoils are expressed as

$$\mathbf{x}_{\text{fab}}(u, v) = \sum_{l=0}^7 \sum_{m=0}^2 B_l^7(u) B_m^2(v) \mathbf{P}_{\text{fab}}^{l,m}, \quad (12)$$

where $0 \leq u \leq 1$ and $0 \leq v \leq 1$ are parametric coordinates, and the n -degree Bernstein polynomials $B_i^n(u) = \binom{n}{i} u^i (1-u)^{n-i}$. We set the parametric coordinates based on the surface points of the nominal shape:

$$(\mathbf{u}, \mathbf{v}) = \left(\frac{\mathbf{x}_{\text{nom}} - x_{\text{nom}}^{\min}}{x_{\text{nom}}^{\max} - x_{\text{nom}}^{\min}}, \frac{\mathbf{y}_{\text{nom}} - y_{\text{nom}}^{\min}}{y_{\text{nom}}^{\max} - y_{\text{nom}}^{\min}} \right). \quad (13)$$

Perturbing nominal designs via FFD ensures that the deformed (fabricated) shapes are still continuous, which conforms to conventional manufacturing methods for aerodynamic shapes. Figure 3 shows an example of nominal design and its corresponding fabricated designs.

The final dataset contains 1,528 nominal designs and 10 fabricated designs per nominal design. Note that since similar nominal designs also have similar fabricated designs, we may need even fewer fabricated designs as training data. Please see Appendix A for the study on how the sample size of fabricated design data affects uncertainty quantification using the proposed deep generative model.

4.1.2 Generative Model Training and Evaluation

We trained the proposed GAN on \mathcal{I}_{nom} and \mathcal{I}_{fab} . We set the parent latent vector to have a uniform prior distribution $\mathcal{U}(\mathbf{0}, \mathbf{1})$ (so that we can search in a bounded space

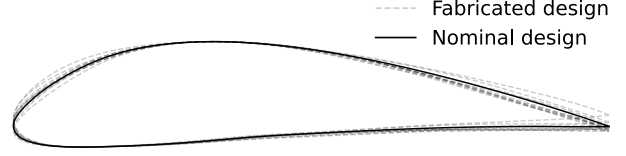


Fig. 3. An example of nominal airfoil design and its corresponding fabricated designs.

during the design optimization stage), whereas both the child latent vector and the noise have normal prior distributions $\mathcal{N}(\mathbf{0}, 0.5\mathbf{I})$. The generator/discriminator architecture and the training configurations were set according to Chen *et al.* [1]. During training, we set both the generator's and the discriminator's learning rate to 0.0001. We trained the model for 20,000 steps with a batch size of 32.

We conducted parametric studies over parent and child latent dimensions to investigate their effects on the generative performances (we fix the dimension of the noise \mathbf{z} to 10)⁵. Particularly, we care about two performances: (1) how well the parent latent representation can cover nominal designs, and (2) how well the performance distributions of fabricated designs are approximated. The experimental settings and results are described as follows.

We evaluated the first performance (*i.e.*, nominal design coverage) via a fitting test, where we found the parent latent vector that minimizes the Euclidean distance between the generated nominal design and a target nominal design sampled from the dataset (*i.e.*, fitting error). We use Sequential Least Squares Programming (SLSQP) [30] as the optimizer and set the number of random restarts to 3 times the parent latent dimension. We repeated this fitting test for 100 randomly sampled target designs under each parent latent dimension setting. A parent latent representation with good coverage of the nominal design data will result in low fitting errors for most target designs. Figure 4a indicates that a parent latent dimension of 7 achieves relatively large design coverage (*i.e.*, low fitting errors).

We evaluated the second performance (*i.e.*, fabricated design performance approximation) by measuring the Wasserstein distance between two conditional distributions — $P(f(\mathbf{x}_{\text{fab}}) | \mathbf{x}_{\text{nom}})$ and $P(f(G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z})) | \mathbf{x}_{\text{nom}})$, where f denotes the objective function. In this example, f is the simulation that computes the lift-to-drag ratio C_L/C_D . For each generated nominal design \mathbf{x}_{nom} , we created 100 “simulated” fabri-

⁵Compared to parent and child latent dimensions, the noise dimension has a relatively small effect on the results as long as it is sufficiently large to capture the noise in training data [1].

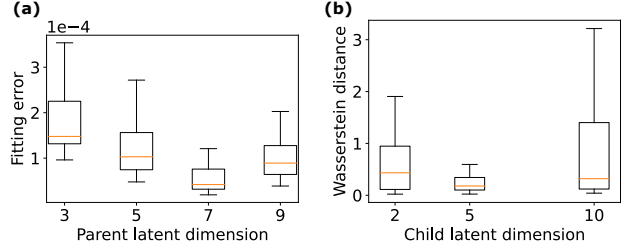


Fig. 4. Parametric study for the airfoil design example: (a) The effect of parent latent dimensions on design space coverage indicated by fitting errors; (b) The effect of child latent dimensions on conditional distribution approximation indicated by Wasserstein distances.

cated designs as \mathbf{x}_{fab} , the same way as we create training data, to be used as the “ground truth” fabricated designs. Note that compared to the number of fabricated designs per nominal design in the training data, we created a much larger number of “ground truth” fabricated designs for evaluation purposes. We also generated the same number of fabricated designs using the trained generator. We compute the aforementioned Wasserstein distance by using these two sets of samples. We repeated this test for 30 randomly generated nominal designs under each child latent dimension setting. Figure 4b shows that when the child latent dimension is 5, we have relatively low Wasserstein distances with the smallest variation (the parent latent dimension was fixed to 7). When the child latent dimension further increases to 10, the uncertainty of the Wasserstein distances increase, possibly due to the higher dimensionality. Note that the training data only contains 10 fabricated designs per nominal design, while in the test phase we use many more samples per nominal design to faithfully approximate the conditional distributions. We do not need that many samples in the training phase because the generative model does not learn independent conditional distributions for each nominal design, but can extract information across all nominal designs.

4.1.3 Design Optimization

The objective of the design optimization is to maximize the lift-to-drag ratio C_L/C_D , which is computed by using the computational fluid dynamics (CFD) solver SU2 [31].

We compared two optimization scenarios:

1. Standard optimization, where we only consider the deterministic performance of the nominal design. The objective is expressed as $\max_{\mathbf{c}_p} f(G(\mathbf{c}_p, \mathbf{0}, \mathbf{0}))$.

2. Robust design optimization, which accounts for the performance variation caused by manufacturing uncertainty. The objective is expressed as $\max_{\mathbf{c}_p} Q_\tau(f(G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{0}))|\mathbf{c}_p)$, where Q_τ denotes the conditional τ -quantile. We set $\tau = 0.05$ in this example.

Under both scenarios, we fixed the parent and the child latent dimensions to 7 and 5, respectively, based on the aforementioned parametric study. In each scenario, we performed Bayesian optimization (BO) to optimize \mathbf{c}_p . We evaluate 21 initial samples of \mathbf{c}_p selected by Latin hypercube sampling (LHS) [32] and 119 sequentially selected samples based on BO’s acquisition function of expected improvement (EI) [33]⁶. In standard optimization, we evaluate the nominal design performance $f(G(\mathbf{c}_p, \mathbf{0}, \mathbf{0}))$ at each sampled point. In robust design optimization, we estimate the quantile of fabricated design performances $f(G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{0}))$ by Monte Carlo (MC) sampling using 100 randomly sampled $\mathbf{c}_c \sim P(\mathbf{c}_c)$ at each \mathbf{c}_p . Figure 5 shows the design solutions and the distributions of ground-truth fabricated design performances⁷ (*i.e.*, post-fabrication performance distributions) of these solutions. We also performed permutation tests [34] to evaluate the difference between the post-fabrication performances of the standard and the robust design solutions (Table 1). The results show that, by accounting for manufacturing uncertainty, the post-fabrication performances of the robust design solution $\mathbf{x}_{\text{robust}}^*$ are significantly improved (indicated by small p -values), compared to the standard design solution $\mathbf{x}_{\text{std}}^*$, even though the nominal performance of $\mathbf{x}_{\text{robust}}^*$ is worse than $\mathbf{x}_{\text{std}}^*$. This result illustrates the possibility that the solution discovered by standard optimization can have high nominal performance but is likely to possess low performance when it is fabricated. The robust design optimization enabled by GAN-DUF can substantially mitigate this risk.

4.2 Case Study: Optical Metasurface Absorber Design

Optical metasurfaces are artificially engineered structures that can support exotic light propagation building on subwavelength inclusions [35, 36]. Among many, metasurface absorbers [37] have been intensely studied for engineering applications such as medical imaging,

⁶The settings of the initial and the total evaluation times in BO are based on the parent latent dimension d_p . Specifically, we performed $3d_p$ initial LHS evaluations and $20d_p$ total evaluations, where $d_p = 7$ as mentioned earlier.

⁷“Ground-truth fabricated design” refers to designs created by the same means by which the designs from \mathcal{I}_{fab} were created.

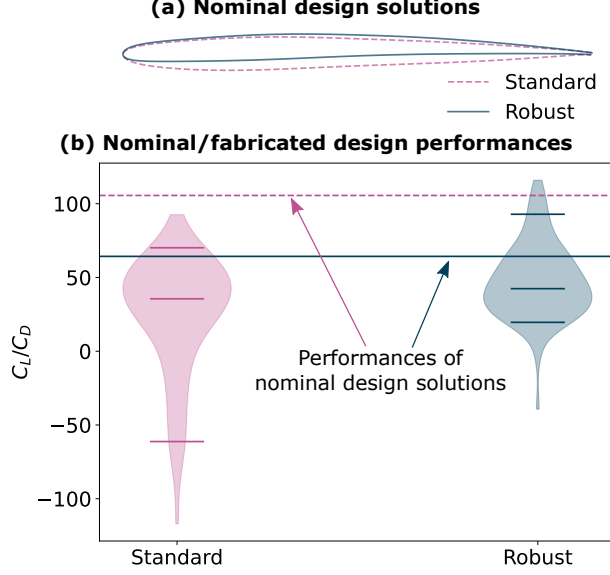


Fig. 5. Solutions for the airfoil design example: (a) Nominal airfoil design solutions obtained by standard optimization and robust design optimization; (b) When considering the manufacturing uncertainty, the robust design solution $\mathbf{x}_{\text{robust}}^*$ shows improved quantile values for the post-fabrication performance distribution compared to the standard design solution $\mathbf{x}_{\text{std}}^*$, even though the nominal performance of $\mathbf{x}_{\text{robust}}^*$ is slightly worse than $\mathbf{x}_{\text{std}}^*$. The short horizontal lines indicate 95% quantiles, medians, and 5% quantiles.

sensing, and wireless communications. In this work, the functionality of interest is large energy absorbance at a range of incident frequencies in the terahertz regime. Fig. 6 shows a schematic of wave analysis. A free-form unit cell made of Au is placed on top of a dielectric substrate whose relative permittivity is set as $2.88 - 0.09i$, where the imaginary term is associated with energy loss. The periodic boundary condition on electromagnetic fields is imposed on the lateral surfaces of the analysis domain. The energy absorbance performance at a single frequency ω is quantified as $A(\omega) = 1 - |S_{11}(\omega)|^2$, where ω is the excitation frequency of the incident wave, and S_{ij} is the component of the S -parameter matrix that characterizes an electrical field intensity from port i to port j in a complex network. This absorbance behavior largely depends on the cross-sectional topology of the unit cell and, by extension, is subject to deviation under fabrication uncertainty.

Table 1. Statistical significance p -values obtained by permutation tests on the post-fabrication performances of the standard and the robust airfoil design solutions. The test statistics are the improvement of the quantiles and the mean. A lower p -value indicates a more significant improvement of the robust design solution over the standard design solution in terms of the post-fabrication performances.

Measures of location	p -value
5% quantile	1e-5
25% quantile	8e-5
Median	0.0397
Mean	1e-5

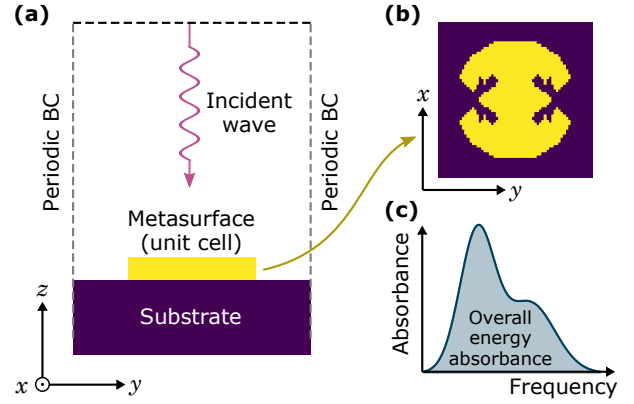


Fig. 6. Conceptual illustration of a metasurface absorber: (a) A schematic of wave analysis. A unit cell is placed on top of the dielectric substrate. A plane wave is normally incident to the metasurface. Periodic boundary conditions are imposed on the lateral surfaces of the analysis domain; (b) An example of a free-form unit cell (nominal design); (c) A hypothetical frequency response of absorbance A at a target frequency band.

4.2.1 Dataset Construction

We created 1,000 nominal designs and 10 fabricated designs per nominal design (Fig. 7a) by using the following method:

Nominal design data. The nominal design dataset builds on three topological motifs (*i.e.*, I-beam, cross, and square ring) [38, 39]. We create nominal designs by randomly interpolating the signed distance fields of these baselines [40]. As a result, each design is stored as 64×64 level-set values (*i.e.*, $\mathbf{x}_{\text{nom}} \in \mathbb{R}^{64 \times 64}$). We can obtain

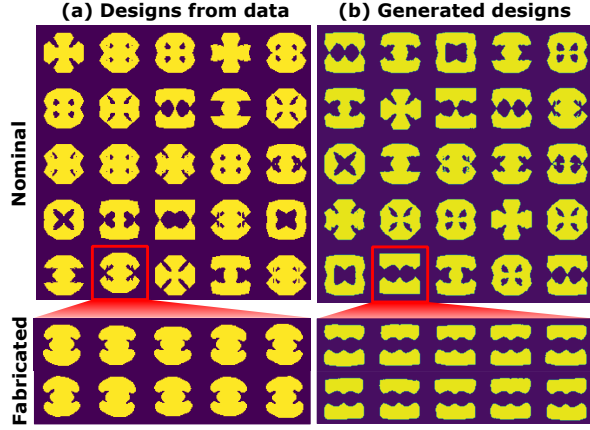


Fig. 7. Visual inspection on generated designs: (a) Metasurface designs randomly drawn from training data; (b) Designs randomly generated from a trained generator.

final designs by thresholding the signed distance fields. Building on a given set of baselines, this shape generation scheme allows a unit cell population that is topologically diverse.

Fabricated design data. Similar to the airfoil design example, we randomly perturb a set of 12×12 FFD control points in both x and y directions with white Gaussian noise that has a standard deviation of 1 pixel. This leads to the distortion of the 64×64 grid coordinates at all the pixels, together with the level-set value at each pixel. We then interpolate a new signed distance field as the fabricated (distorted) design. To account for the limited precision of fabrication, we further apply a Gaussian filter with a standard deviation of 2 to smooth out sharp, non-manufacturable features. The Gaussian filter can also change the topology by, *e.g.*, removing small holes. To demonstrate that our proposed deep generative model is flexible in addressing more complicated uncertainties, we also construct a scenario where fabrication errors contain non-smooth boundary variation and random hole nucleation. We show that our proposed model can also address this more complicated scenario. Please refer to Appendix B for details.

4.2.2 Generative Model Training and Evaluation

As mentioned in the Background section, optimizing designs with varying topology under geometric uncertainty hosts a great challenge. GAN-DUF can handle this problem by modeling the uncertainty using the proposed generative adversarial network. Same as the airfoil example, we set the parent latent vector to have

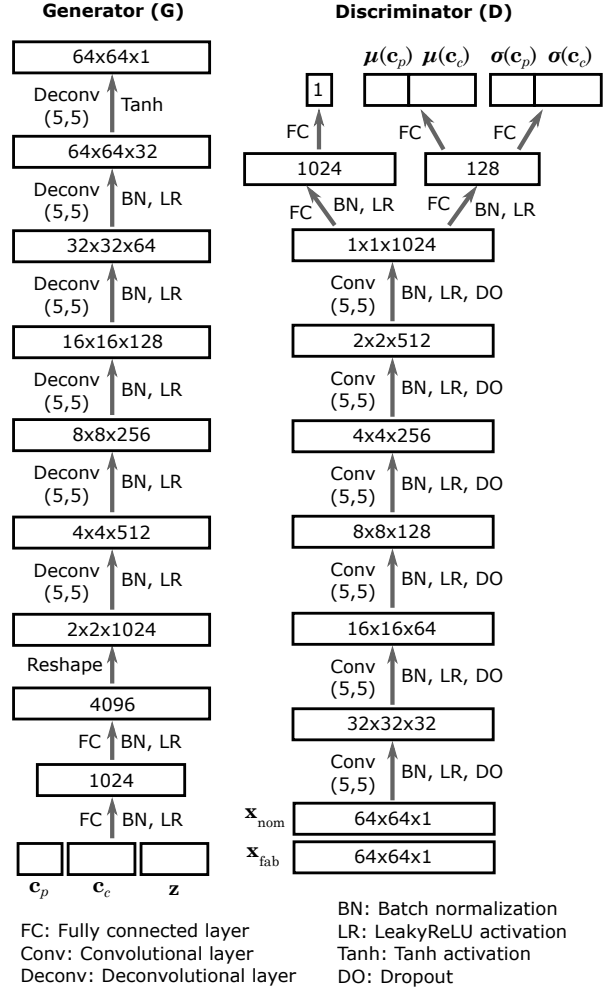


Fig. 8. Generator and discriminator architectures in the metasurface design example.

a uniform prior distribution, while both the child latent vector and the noise have normal prior distributions. Again, we fixed the noise dimension to 10. The generator and the discriminator architectures are shown in Fig. 8. The discriminator predicts both the discriminative distribution $D(x_{nom}, x_{fab})$ and the auxiliary distribution $Q(c_p, c_c | x_{nom}, x_{fab})$. During training, we set both the generator's and the discriminator's learning rate to 0.0001. We trained the model for 50,000 steps with a batch size of 32.

Similar to the airfoil design example, we want to investigate the effect of the parent and the child latent dimensions on the trained GAN's capability of covering the design space and approximating the fabricated de-

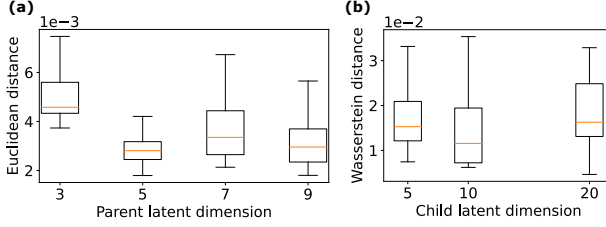


Fig. 9. Parametric study for the metasurface design example: (a) The effect of parent latent dimensions on design space coverage indicated by fitting errors; (b) The effect of child latent dimensions on conditional distribution approximation indicated by Wasserstein distances.

sign performance distributions. To evaluate the design space coverage, we performed a fitting test as described in Sec. 4.1.2. Specifically, we use SLSQP as the optimizer and set the number of random restarts to 3 times the parent latent dimension. Here the fitting error is the Euclidean distance between the signed distance fields of the generated nominal design and a target nominal design sampled from the dataset. Under each parent latent dimension setting, we randomly select 100 target designs. Figure 9a indicates that a parent latent dimension of 5 achieves sufficiently large design coverage, while further increasing the parent latent dimension cannot improve the coverage.

To evaluate the trained GAN’s performance on approximating the fabricated design performance distributions, we use the Wasserstein distance to measure the distributional difference between $P(f(\mathbf{x}_{\text{fab}})|\mathbf{x}_{\text{nom}})$ and $P(f(G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}))|\mathbf{x}_{\text{nom}})$ (see Sec. 4.1.2). Under each child latent dimension setting, we compute the Wasserstein distance between the energy absorbance distributions of “ground truth” fabricated designs and generated fabricated designs. We performed fewer design evaluations than the airfoil design example due to the higher cost of wave analysis. Under each child latent dimension setting, we computed Wasserstein distances for 10 nominal designs, each with 30 “ground truth” fabricated designs and 30 generated fabricated designs. Figure 9b indicates that a child latent dimension of 10 achieves the lowest median and 25%/75% quantiles.

Figure 7b shows nominal and fabricated designs randomly generated from the trained generator with a parent and a child latent dimensions of 5 and 10, respectively.

4.2.3 Design Optimization

During the design optimization stage, we fixed the parent and the child latent dimensions to be 5 and 10, re-

spectively, based on the aforementioned parametric study. The objective is to maximize the mean energy absorbance over the given range of frequencies, 8-9 THz. To achieve broadband functionality, we formulate the objective function as the sum of energy absorbance at individual frequencies (*i.e.*, $f = \sum_{i=1}^{n_\omega} A(\omega_i)$, where n_ω is the number of frequencies at which absorbance is to be observed). The RF Module of COMSOL Multiphysics® [41] is used for evaluation of metasurfaces.

Similar to the airfoil design example, we compared standard optimization with robust design optimization. To investigate the benefit of modeling free-form uncertainty over uniform uncertainty, we also benchmarked the performance of GAN-DUF against the method that models fabrication uncertainty as uniform boundary variation [6, 16, 17]. Specifically, following [6], we first convolve the nominal design pattern $\mathbf{x}_{\text{nom}}(r)$ with a Gaussian filter to obtain a blurred pattern $\tilde{\mathbf{x}}(r)$:

$$\tilde{\mathbf{x}}(r) = \sum_{r_i} \frac{1}{\alpha} \mathbf{x}_{\text{nom}}(r_i) e^{-\frac{(r-r_i)^2}{\sigma^2}}, \quad (14)$$

where r is the spatial coordinate on the pattern and α is a normalization factor defined as

$$\alpha = \sum_{r_i} e^{-\frac{(r-r_i)^2}{\sigma^2}}. \quad (15)$$

The coefficient σ controls the blur radius that corresponds to patterning resolution. In our experiment, we set $\sigma = 2$. We simply threshold the blurred pattern $\tilde{\mathbf{x}}(r)$ to create eroded and dilated patterns $\bar{\mathbf{x}}(r)$:

$$\bar{\mathbf{x}}(r) = \begin{cases} 0, & 0 \leq \tilde{\mathbf{x}}(r) \leq \eta \\ 1, & \eta \leq \tilde{\mathbf{x}}(r) \leq 1 \end{cases}, \quad (16)$$

where η is the threshold with a value between 0 and 1. We can create an eroded pattern by setting $\eta > 0.5$ and a dilated pattern by setting $\eta < 0.5$. The eroded and dilated patterns represent extreme fabrication errors. The edge deviation Δ (*i.e.*, the distance of boundary shift) is related to both η and σ :

$$\frac{1}{2} - \eta = \text{erf} \left\{ \frac{\Delta}{\sigma} \right\}. \quad (17)$$

In this experiment, we set Δ to -0.4 and 0.4 for the eroded and the dilated patterns, respectively. We can compute



Fig. 10. An example of metasurface design with fabrication uncertainty modeled as uniform boundary variation [6, 16, 17].

η based on Eq. (17) and then obtain the eroded/dilated pattern using Eq. 16. Please see [6] for more details. Figure 10 shows an example of resulted patterns. While we manually select σ and Δ as in [6], we can also calibrate this parameter using training data. However, this calibration will require additional study and experimentation, and hence may deviate from our goal of comparing our method to existing ones. Nevertheless, considering data-driven parameter calibration for the existing methods could be an interesting future work.

To account for this uniform boundary variation as the uncertainty in robust design optimization, we set the objective to be $f(\mathbf{x}_{\text{nom}}) + 0.5f(\bar{\mathbf{x}}_{\text{eroded}}) + 0.5f(\bar{\mathbf{x}}_{\text{dilated}})$, where $\bar{\mathbf{x}}_{\text{eroded}}$ and $\bar{\mathbf{x}}_{\text{dilated}}$ denote the eroded and the dilated patterns, respectively, and f computes the overall energy absorbance for a given metasurface pattern. Due to the prohibitive cost of optimizing a high-dimensional design ($\mathbf{x}_{\text{nom}} \in \mathbb{R}^{64 \times 64}$) without analytical gradient, we perform the robust design optimization in the latent space instead, *i.e.*, based on $\mathbf{x}_{\text{nom}} = G(\mathbf{c}_p, \mathbf{0}, \mathbf{0})$. We will use “robust (uniform)” to refer to this optimization scenario.

In each optimization scenario, we performed BO with 15 initial LHS samples and 85 sequentially selected samples based on the acquisition strategy of EI⁸. The quantile of fabricated design performances at each \mathbf{c}_p was estimated from 30 MC samples. We used fewer MC samples than those in the airfoil design case due to the higher cost of evaluating the objective (*i.e.*, performing wave analysis to compute the energy absorbance).

Figure 11 shows the design solutions and the distributions of ground-truth fabricated design performances for these solutions. We observe similar trends as in the airfoil design case, where the standard optimization finds the solution with the highest nominal performance, while the two solutions obtained from robust optimization have higher performances (in general) after fabrication (Fig. 11b). Particularly, by considering free-form uncer-

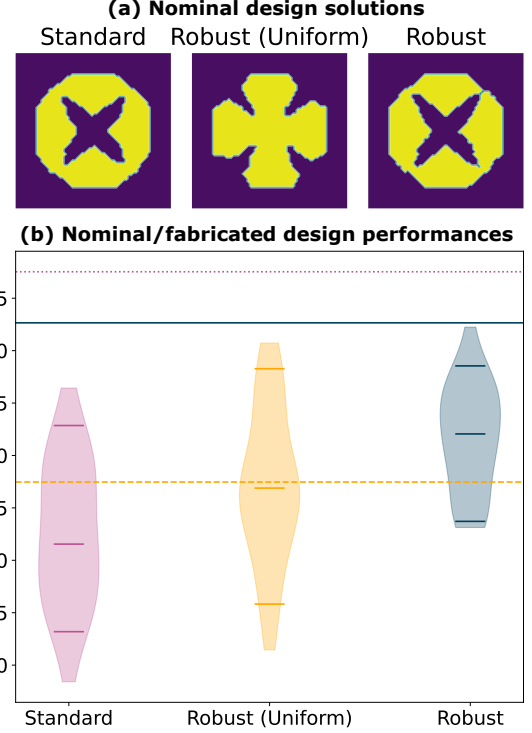


Fig. 11. Solutions for the metasurface design example: (a) Nominal metasurface design solutions obtained by standard optimization and robust design optimization; (b) When considering the manufacturing uncertainty, the robust design solution $\mathbf{x}_{\text{robust}}^*$ shows improved post-fabrication performance distribution compared to the standard design solution $\mathbf{x}_{\text{std}}^*$, even though the nominal performance of $\mathbf{x}_{\text{robust}}^*$ is worse than $\mathbf{x}_{\text{std}}^*$. The short horizontal lines indicate 95% quantiles, medians, and 5% quantiles.

tainty enabled by GAN-DUF, the robust design solution further improves upon the solution obtained under uniform uncertainty, in terms of their actual post-fabrication performances. The low p -values in Table 2 indicate the statistical significance of this improvement. This demonstrates that making simplifying assumptions on the form of uncertainty can lead to sub-optimal solutions. The proposed GAN-DUF does not suffer from this issue since it has the flexibility of learning and considering any form of uncertainty. This result is expected because the ability to accurately model the uncertainties is a prerequisite for finding the true robust design solution that performs better than others when considering uncertainties.

⁸We performed $3d_p$ initial LHS evaluations and $30d_p$ total evaluations in BO, where $d_p = 5$ is the parent latent dimension.

Table 2. The p -values obtained by permutation tests that quantify the statistical significance of the improvement when considering free-form uncertainty enabled by GAN-DUF, in terms of the post-fabrication performances. The test statistics are the improvement of the quantiles and the mean. For example, a low value in row “Median” and column “Standard” indicates that there is a significant improvement of the free-form uncertainty-enabled robust design solution over the standard design solution in terms of the medians of their post-fabrication performance distributions.

Measures of location	p -value	
	Standard	Robust (uniform)
5% quantile	1e-5	0.0031
25% quantile	3e-5	0.00024
Median	1e-5	0.0118
Mean	1e-5	0.0028

5 CONCLUSION

We proposed GAN-DUF to facilitate design under free-form geometric uncertainty. It contains a novel deep generative model that simultaneously learns a compact representation of nominal designs and the conditional distribution of fabricated designs given any nominal design. The proposed framework is generalizable to any geometric design representations (*i.e.*, both shape and topological designs) and can address free-form uncertainties without resorting to any simplifying assumption on the type of uncertainty. We applied GAN-DUF to two real-world engineering design examples (namely aerodynamic shape optimization and metasurface absorber topology optimization) and showed its capability in finding the design solution that is more likely to possess a better performance after fabrication or manufacturing.

In this work, we assume that fabricated designs are independent. In the case where the distribution of fabricated designs depends on the varying fabrication tool condition due to, *e.g.*, tool wear, we can introduce an additional latent vector \mathbf{c}_t that encodes the variation of the fabrication tool. In this way, the proposed generative model can estimate the conditional distribution of fabricated designs given the tool condition. We can also continuously calibrate \mathbf{c}_t using new fabrication data.

Although we only considered fabrication/manufacturing uncertainty when demonstrating the proposed framework, it is also applicable to other sources of geometric uncertainties such as those caused by operational wear or erosion. In addition to robust design optimization demonstrated in this work, we can

also combine the proposed hierarchical generative model with reliability-based design optimization to find designs that are less likely to fail after fabrication or under operational wear/erosion.

GAN-DUF is generalizable to 3D designs with certain adjustments to neural network architectures, on top of the 2D cases demonstrated in this work. For example, if the 3D designs are represented as point clouds, one can adopt the generator/discriminator architecture from [42]. Since 3D data may contain more complex geometric variation, it is likely that we need more layers, higher latent dimensions, and a larger dataset to accommodate the extra complexity.

While in this work we used synthetic fabrication data, for future work, we will collect real fabricated designs as training and test data to validate the effectiveness of GAN-DUF in a completely realistic scenario.

ACKNOWLEDGEMENTS

This work was supported by the NSF CSSI program (Grant No. OAC 1835782) and the Northwestern McCormick Catalyst Award. We thank the anonymous reviewers for their comments.

REFERENCES

- [1] Chen, W., Chiu, K., and Fuge, M. D., 2020, “Airfoil design parameterization and optimization using bézier generative adversarial networks,” *AIAA Journal*, **58**(11), pp. 4723–4735.
- [2] Chen, W., and Ramamurthy, A., 2021, “Deep generative model for efficient 3d airfoil parameterization and generation,” In *AIAA Scitech 2021 Forum*, p. 1690.
- [3] Chen, W., and Ahmed, F., 2021, “Mo-padgan: Reparameterizing engineering designs for augmented multi-objective optimization,” *Applied Soft Computing*, **113**, p. 107909.
- [4] Chen, S., Chen, W., and Lee, S., 2010, “Level set based robust shape and topology optimization under random field uncertainties,” *Structural and Multidisciplinary Optimization*, **41**(4), pp. 507–524.
- [5] Chen, S., and Chen, W., 2011, “A new level-set based approach to shape and topology optimization under geometric uncertainty,” *Structural and Multidisciplinary Optimization*, **44**(1), pp. 1–18.
- [6] Wang, E. W., Sell, D., Phan, T., and Fan, J. A., 2019, “Robust design of topology-optimized metasurfaces,” *Optical Materials Express*, **9**(2), pp. 469–482.

- [7] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., 2014, “Generative adversarial nets,” In *Advances in neural information processing systems*, pp. 2672–2680.
- [8] Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I., and Abbeel, P., 2016, “Infogan: Interpretable representation learning by information maximizing generative adversarial nets,” In *Advances in Neural Information Processing Systems*, pp. 2172–2180.
- [9] Chen, W., and Fuge, M., 2019, “Synthesizing designs with interpart dependencies using hierarchical generative adversarial networks,” *Journal of Mechanical Design*, **141**(11), p. 111403.
- [10] Du, X., Sudjianto, A., and Chen, W., 2004, “An integrated framework for optimization under uncertainty using inverse reliability strategy,” *J. Mech. Des.*, **126**(4), pp. 562–570.
- [11] Maute, K., 2014, “Topology optimization under uncertainty,” In *Topology optimization in structural and continuum mechanics*. Springer, pp. 457–471.
- [12] Chen, W., Allen, J., Tsui, K.-L., and Mistree, F., 1996, “A procedure for robust design: Minimizing variations caused by noise factors and control factors,” *Journal of Mechanical Design, Transactions of the ASME*, **118**(4), pp. 478–485.
- [13] Du, X., and Chen, W., 2004, “Sequential optimization and reliability assessment method for efficient probabilistic design,” *J. Mech. Des.*, **126**(2), pp. 225–233.
- [14] Choi, S.-K., Canfield, R. A., and Grandhi, R. V., 2007, *Reliability-Based Structural Optimization* Springer.
- [15] Baudoui, V., Klotz, P., Hiriart-Urruty, J.-B., Jan, S., and Morel, F., 2012, “Local uncertainty processing (loup) method for multidisciplinary robust design optimization,” *Structural and Multidisciplinary Optimization*, **46**(5), pp. 711–726.
- [16] da Silva, G. A., Beck, A. T., and Sigmund, O., 2019, “Stress-constrained topology optimization considering uniform manufacturing uncertainties,” *Computer Methods in Applied Mechanics and Engineering*, **344**, pp. 512–537.
- [17] Sigmund, O., 2009, “Manufacturing tolerant topology optimization,” *Acta Mechanica Sinica*, **25**(2), pp. 227–239.
- [18] Morris, C., Bekker, L., Haberman, M. R., and Seepersad, C. C., 2018, “Design exploration of reliably manufacturable materials and structures with applications to negative stiffness metamaterials and microstereolithography,” *Journal of Mechanical Design*, **140**(11).
- [19] Wiest, T., Seepersad, C. C., and Haberman, M. R., 2022, “Robust design of an asymmetrically absorbing willis acoustic metasurface subject to manufacturing-induced dimensional variations,” *The Journal of the Acoustical Society of America*, **151**(1), pp. 216–231.
- [20] Lazarov, B. S., Schevenels, M., and Sigmund, O., 2012, “Topology optimization with geometric uncertainties by perturbation techniques,” *International Journal for Numerical Methods in Engineering*, **90**(11), pp. 1321–1336.
- [21] Lazarov, B. S., Schevenels, M., and Sigmund, O., 2012, “Topology optimization considering material and geometric uncertainties using stochastic collocation methods,” *Structural and Multidisciplinary optimization*, **46**(4), pp. 597–612.
- [22] Keshavarzzadeh, V., Fernandez, F., and Tortorelli, D. A., 2017, “Topology optimization under uncertainty via non-intrusive polynomial chaos expansion,” *Computer Methods in Applied Mechanics and Engineering*, **318**, pp. 120–147.
- [23] Kang, Z., and Liu, P., 2018, “Reliability-based topology optimization against geometric imperfections with random threshold model,” *International Journal for Numerical Methods in Engineering*, **115**(1), pp. 99–116.
- [24] Huang, Q., Nouri, H., Xu, K., Chen, Y., Sosina, S., and Dasgupta, T., 2014, “Statistical predictive modeling and compensation of geometric deviations of three-dimensional printed products,” *Journal of Manufacturing Science and Engineering*, **136**(6).
- [25] Huang, Q., Zhang, J., Sabbaghi, A., and Dasgupta, T., 2015, “Optimal offline compensation of shape shrinkage for three-dimensional printing processes,” *Iie transactions*, **47**(5), pp. 431–441.
- [26] Sabbaghi, A., Huang, Q., and Dasgupta, T., 2018, “Bayesian model building from small samples of disparate data for capturing in-plane deviation in additive manufacturing,” *Technometrics*, **60**(4), pp. 532–544.
- [27] Ferreira, R. d. S. B., Sabbaghi, A., and Huang, Q., 2019, “Automated geometric shape deviation modeling for additive manufacturing systems via bayesian neural networks,” *IEEE Transactions on Automation Science and Engineering*, **17**(2), pp. 584–598.
- [28] Pham, T., Kwon, P., and Foster, S., 2021, “Additive manufacturing and topology optimization of magnetic materials for electrical machines—a review,” *Energies*, **14**(2), p. 283.
- [29] Sederberg, T. W., and Parry, S. R., 1986, “Free-

form deformation of solid geometric models,” In Proceedings of the 13th annual conference on Computer graphics and interactive techniques, pp. 151–160.

- [30] Kraft, D., 1988, *A software package for sequential quadratic programming* Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht. Wiss. Berichtswesen d. DFVLR.
- [31] Economon, T. D., Palacios, F., Copeland, S. R., Lukaczyk, T. W., and Alonso, J. J., 2016, “Su2: An open-source suite for multiphysics simulation and design,” *Aiaa Journal*, **54**(3), pp. 828–846.
- [32] McKay, M. D., Beckman, R. J., and Conover, W. J., 2000, “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics*, **42**(1), pp. 55–61.
- [33] Jones, D. R., Schonlau, M., and Welch, W. J., 1998, “Efficient global optimization of expensive black-box functions,” *Journal of Global optimization*, **13**(4), pp. 455–492.
- [34] Fisher, R. A., 1936, “Design of experiments,” *British Medical Journal*, **1**(3923), p. 554.
- [35] Chen, H.-T., Taylor, A. J., and Yu, N., 2016, “A review of metasurfaces: physics and applications,” *Reports on progress in physics*, **79**(7), p. 076401.
- [36] Bukhari, S. S., Vardaxoglou, J. Y., and Whittow, W., 2019, “A metasurfaces review: Definitions and applications,” *Applied Sciences*, **9**(13), p. 2727.
- [37] Liu, X., Fan, K., Shadrivov, I. V., and Padilla, W. J., 2017, “Experimental realization of a terahertz all-dielectric metasurface absorber,” *Optics express*, **25**(1), pp. 191–201.
- [38] Larouche, S., Tsai, Y.-J., Tyler, T., Jokerst, N. M., and Smith, D. R., 2012, “Infrared metamaterial phase holograms,” *Nature materials*, **11**(5), pp. 450–454.
- [39] Azad, A. K., Kort-Kamp, W. J., Sykora, M., Weisse-Bernstein, N. R., Luk, T. S., Taylor, A. J., Dalvit, D. A., and Chen, H.-T., 2016, “Metasurface broadband solar absorber,” *Scientific reports*, **6**(1), pp. 1–6.
- [40] Whiting, E. B., Campbell, S. D., Kang, L., and Werner, D. H., 2020, “Meta-atom library generation via an efficient multi-objective shape optimization method,” *Optics Express*, **28**(16), pp. 24229–24242.
- [41] Multiphysics, C., 1998, “Introduction to comsol multiphysics®,” *COMSOL Multiphysics*, Burlington, MA, accessed Feb, **9**, p. 2018.
- [42] Shu, D. W., Park, S. W., and Kwon, J., 2019, “3d point cloud generative adversarial network based on tree structured graph convolutions,” In Proceedings

of the IEEE/CVF international conference on computer vision, pp. 3859–3868.

APPENDIX A: A SAMPLE SIZE STUDY

In the main text, we fixed the number of fabricated designs per nominal design in the training data. In practice, the sample size of fabricated designs may differ across use cases. There is a trade-off between the cost of acquiring fabrication data and the accuracy of quantifying fabrication uncertainties. To investigate how the sample size affects uncertainty quantification and draw insights into the minimum fabrication data requirement, we perform a sample size study on the fabrication data. We vary the number of fabricated designs per nominal design in the training data while fixing other hyperparameters. After training the deep generative model, we evaluate how well the performance distributions of fabricated designs are approximated by the trained generator. We measure this in the same way as the parametric study over child latent dimensions described in Sec. 4.1.2 and Sec. 4.2.2 — *i.e.*, using the Wasserstein distance to measure the distributional difference between $P(f(\mathbf{x}_{\text{fab}})|\mathbf{x}_{\text{nom}})$ and $P(f(G(\mathbf{c}_p, \mathbf{c}_c, \mathbf{z}))|\mathbf{x}_{\text{nom}})$. Results in Fig. 12 show a general trend of decreased Wasserstein distance or improved performance distribution approximation, as the sample size of the fabricated designs increases. Among the tested cases, we observe the most accurate approximation, in both the airfoil and the metasurface design examples, when there are 7 fabricated designs per nominal design in the training data. Further increasing the number of fabricated designs per nominal design to 10 does not lead to a significant improvement in the approximation accuracy. Our future work will study how GAN-DUF will perform in design under uncertainty under sparse samples of fabricated designs.

APPENDIX B: MORE COMPLICATED UNCERTAINTIES

Due to the flexibility of deep neural networks, our proposed GAN does not assume any specific forms of uncertainties. To help demonstrate this point, we create a scenario for the metasurface design example where fabrication uncertainties include defects of random hole nucleation and non-smooth boundary deviations.

Specifically, compared to the fabricated design data construction mentioned in Sec. 4.2.1, we remove the Gaussian filter, increase the FFD lattice resolution (*i.e.*, the number of FFD control points), and double the random perturbation scale of FFD control points, to create

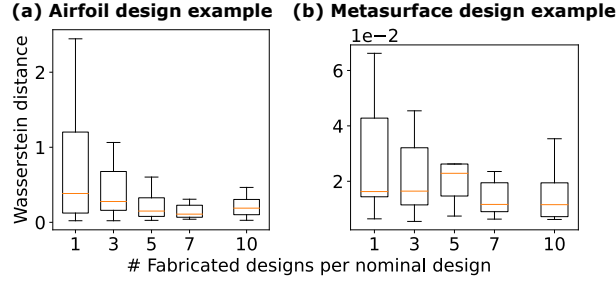


Fig. 12. Sample size study showing how the number of fabricated designs per nominal design affects the approximation of fabricated designs' performance distribution.

non-smooth boundary deviations. We also create a hole with a random shape and size at a random location of each fabricated design⁹. The nominal design data remain unchanged. Examples of fabricated design data are shown in the upper part of Fig. 13.

We show that by some minor adjustments of hyper-parameters, our proposed GAN can capture these more complicated uncertainties, as shown in the lower part of Fig. 13. To adapt to the more complicated fabrication data, we decrease the convolutional/deconvolutional kernel size to (3, 3), increase the child latent dimension to 20, and double the number of filters in the generator's deconvolutional layers. The visual comparison shows that GAN-DUF can generate fabricated designs with random hole nucleation and non-smooth boundary variation as occurred in the data, indicating its ability to capture these more complicated uncertainties.

⁹The hole will not show if its location is sampled outside the metasurface pattern.

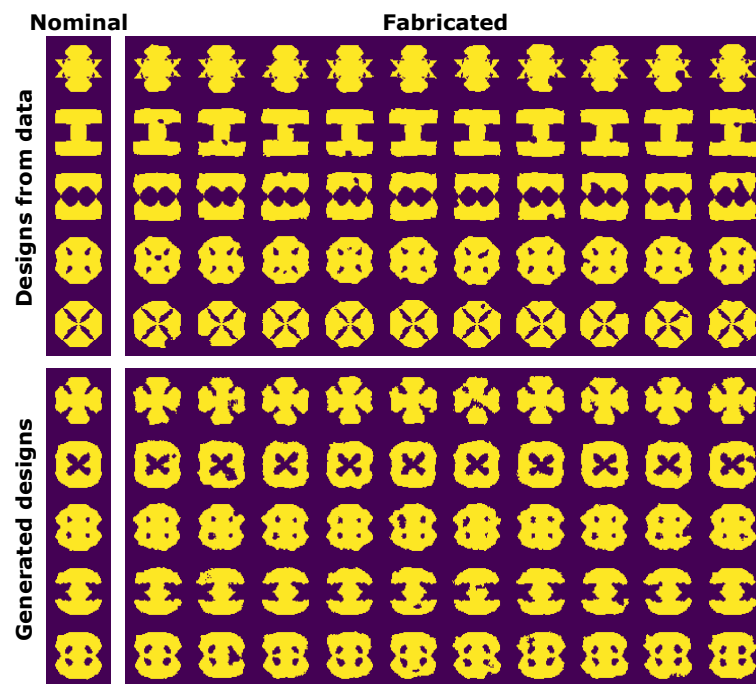


Fig. 13. The proposed GAN can address fabrication uncertainties with non-smooth shape variation and hole nucleation.