

Generative Inverse Design of Metamaterials with Functional Responses by Interpretable Learning

Wei (Wayne) Chen,* Rachel Sun, Doksoo Lee, Carlos M. Portela, and Wei Chen

Metamaterials with functional responses can exhibit varying properties under different conditions (e.g., wave-based responses or deformation-induced property variation). This work addresses rapid inverse design of such metamaterials to meet target qualitative functional behaviors, a challenge due to its intractability and nonunique solutions. Unlike data-intensive and noninterpretable deep-learning-based methods, this work proposes the random-forest-based interpretable generative inverse design (RIGID), a single-shot inverse design method for fast generation of metamaterials with on-demand functional behaviors. RIGID leverages the interpretability of a random forest-based “design → response” forward model, eliminating the need for a more complex “response → design” inverse model. Based on the likelihood of target satisfaction derived from the trained random forest, one can sample a desired number of design solutions using Markov chain Monte Carlo methods. RIGID is validated on acoustic and optical metamaterial design problems, each with fewer than 250 training samples. Compared to the genetic algorithm-based design generation approach, RIGID generates satisfactory solutions that cover a broader range of the design space, allowing for better consideration of additional figures of merit beyond target satisfaction. This work offers a new perspective on solving on-demand inverse design problems, showcasing the potential for incorporating interpretable machine learning into generative design under small data constraints.

1. Introduction

Metamaterials are engineered materials, typically composed of periodically arranged building blocks (i.e., unit cells), that exhibit properties and functionalities beyond their constituent materials.^[1,2] We can achieve certain effective material properties of metamaterials by designing their structure (i.e., geometries of unit cells) rather than chemical composition. Optical,^[3] acoustic,^[4] thermal,^[5] and mechanical metamaterials^[2,6] are metamaterials with unique properties tailored to manipulate specific types of waves or energy. For example, optical metamaterials or metasurfaces (i.e., the 2D form of metamaterials) are designed to control and manipulate electromagnetic waves, which can lead to unusual properties such as a negative refractive index.^[7] Acoustic metamaterials are designed to control how sound (acoustic) waves propagate which enables functionalities such as noise reduction.^[8]

Functional responses refer to varying properties or behaviors under different conditions. A notable example that involves

functional responses is metamaterials, which alter their electromagnetic, acoustic, or elastic wave propagation behaviors depending on the wavelength or frequency.^[9] Another example is metamaterials that exhibit changing properties or functionalities due to deformation in response to external stimuli like temperature^[10] and magnetic fields.^[11] Tailoring functional responses of these metamaterials is of interest to applications such as sound and vibration control, analog computing, medical imaging, sensing, communication, and soft robotics. In many use cases, rather than precisely controlling the complete functional responses, we only care about qualitative behaviors under certain conditions. For example, acoustic metamaterials were usually designed to have bandgaps at specified frequencies to achieve functionalities like wave guiding,^[12,13] focusing,^[14,15] and vibration mitigation.^[16–18] However, it is unnecessary and computationally expensive to design for the whole dispersion relation.^[19–23] Similarly, we may design optical metamaterials to qualitatively manipulate optical properties (e.g., high or low absorption/reflection/transmission) under certain wavelengths, without requiring the entire spectral response to match an exact target.^[24,25]

Identifying metamaterial designs from a given target forms an inverse design problem. Unlike many forward problems where

W. (Wayne) Chen
J. Mike Walker '66 Department of Mechanical Engineering
Texas A&M University
College Station, TX 77843, USA
E-mail: w.chen@tamu.edu

R. Sun, C. M. Portela
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139, USA

D. Lee, W. Chen
Department of Mechanical Engineering
Northwestern University
Evanston, IL 60208, USA

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/aisy.202400611>.

© 2024 The Author(s). Advanced Intelligent Systems published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

DOI: [10.1002/aisy.202400611](https://doi.org/10.1002/aisy.202400611)

one can obtain solutions (e.g., spectral responses or material properties under external stimuli) by modeling the physics or conducting experiments, inverse design problems are usually intractable. Traditionally, these problems are solved by iterative optimization (i.e., minimizing the difference between the actual quantity of interest and the target).^[19,20,22,26,27] This, however, requires repeatedly updating the design solution and solving forward problems. When the design target changes, one needs to rerun the entire optimization process. Thus, inverse design by iterative optimization becomes impractical if solving the forward problem (by simulations or experiments) is time-consuming or if the design target needs to change frequently. For example, in the case of acoustic metamaterial design, one may want to obtain a material waveguide containing spatially varying unit cells with different bandgap ranges at different locations in a tessellation. To accelerate the optimization approach, prior works replaced simulations or experiments with machine learning models.^[28,29] However, the efficiency and quality of final solutions are highly dependent on both the machine learning model and the optimization algorithm. On the other hand, a single run of optimization usually only returns one final solution, although multiple designs might satisfy a given target (i.e., the nonuniqueness of solutions). For example, multiple acoustic metamaterial designs may have bandgaps within the same target frequency range. This nonuniqueness nature of inverse design problems was also shown for optical metasurfaces,^[27,30–32] especially when we only care about qualitative target responses. It is generally advantageous for inverse design approaches to yield a diverse set of solutions, as this allows consideration of additional figures of merit beyond target satisfaction. For example, deriving diverse solutions for acoustic metamaterials will allow us to consider the kinematic compatibility of multiple designs in waveguide design problems. However, optimization approaches cannot explore diverse alternative solutions efficiently.

In this work, we aim to achieve iteration-free, single-shot inverse design for metamaterials—given a qualitative target response, we want to generate multiple satisfying design solutions rapidly, without the need for iteratively evaluating designs (i.e., running simulations or conducting experiments); and after one-time initial training, no additional training or design evaluation is required to generate designs for different targets. This allows fast and comprehensive exploration of the feasible design space under different target responses.

Prior research attempted to realize iteration-free, single-shot inverse design using machine learning. There are three mainstream methods. The most straightforward approach is to learn a direct inverse mapping from the response to design variables. Neural networks are the most commonly used machine learning model for this purpose, due to their high flexibility in approximating arbitrary nonlinear input–output relationships.^[21,33] Despite the simplicity of the direct inverse mapping, its underlying assumption of the response–design mapping being one to one does not hold in many cases due to the nonuniqueness of solutions, as mentioned earlier. Such nonuniqueness will cause conflicting training instances where the same input (response) is associated with distinct outputs (designs), which will destabilize the convergence during neural network training.^[30,34] To avoid this issue, past work has proposed the tandem neural network (T-NN) that cascades an inverse-design network with a pretrained

forward-modeling network to avoid using designs as labels when training the inverse-design network and hence solved the training convergence issue.^[30,35–40] Nonetheless, the original T-NNs still learn a one-to-one response–design mapping and cannot account for the nonuniqueness of design solutions. To fundamentally solve this problem, one needs to learn a one-to-many mapping. For example, Bastek et al.^[38] integrated stochastic sampling into the inverse-design network to allow the generation of multiple feasible solutions. Ha et al.^[41] used multiple T-NNs to predict multiple design solutions given a target response. Wang et al.^[42] adapted the T-NN to mimic the structure of a variational autoencoder that is capable of learning the conditional distributions of designs given target responses. A large body of recent works achieved the goal of learning one-to-many mapping using conditional generative models, typically conditional generative adversarial networks,^[32,43–46] conditional variational autoencoders,^[31,47,48] and conditional diffusion models.^[49–51] These models can generate multiple designs given a target response by learning the distribution of designs conditioned on the response. We include more detailed explanations of these three mainstream methods in Section S1, Supporting Information.

We refer interested readers to Refs. [34,52–57] for comprehensive reviews of existing metamaterial design methods. To the best of our knowledge, almost all the existing iteration-free, single-shot metamaterial inverse design methods are based on deep learning, which has many common issues such as high data demand, exhaustive hyperparameter tuning, slow training, and low interpretability, especially compared to traditional machine learning models like decision trees and random forests. On the other hand, Elzouka et al.^[58] proposed to use the decision tree as a more interpretable model to solve both the forward prediction and inverse design problem. After training a decision tree for forward prediction, one can identify explicit design rules (i.e., feasible regions in the design space) by tracing from target leaf nodes to the root node. This approach also captures the one-to-many mapping nature of inverse design problems since it gives feasible design variable ranges rather than a single solution. However, there remain some limitations. First, for solutions identified by the design rules, the method does not differentiate their likelihood of target satisfaction. Yet in reality, solutions always have different likelihoods due to the uncertainty of model estimation. Second, the method has to train two models: a random forest was trained first to ensure model accuracy and robustness, and then a large decision tree was trained to emulate the performance of the random forest and provide design rules. This is due to the challenge of deriving explicit design rules from an ensemble model like the random forest. Finally, the method was demonstrated on a problem with more than 10^4 training data, while the effectiveness on smaller datasets (i.e., data with orders of magnitude smaller sample sizes) was not studied.

This work aims to address the aforementioned problems by proposing a method called random forest-based interpretable generative inverse design (RIGID). Figure 1 shows an overview of this method. Specifically, we first train a forward prediction random forest. Then given a design target, we can probe the trained random forest to infer the likelihood of any design satisfying the target. To generate new designs tailored to the target, we can sample from the design space according to the likelihood. Compared to the most widely studied neural network-based

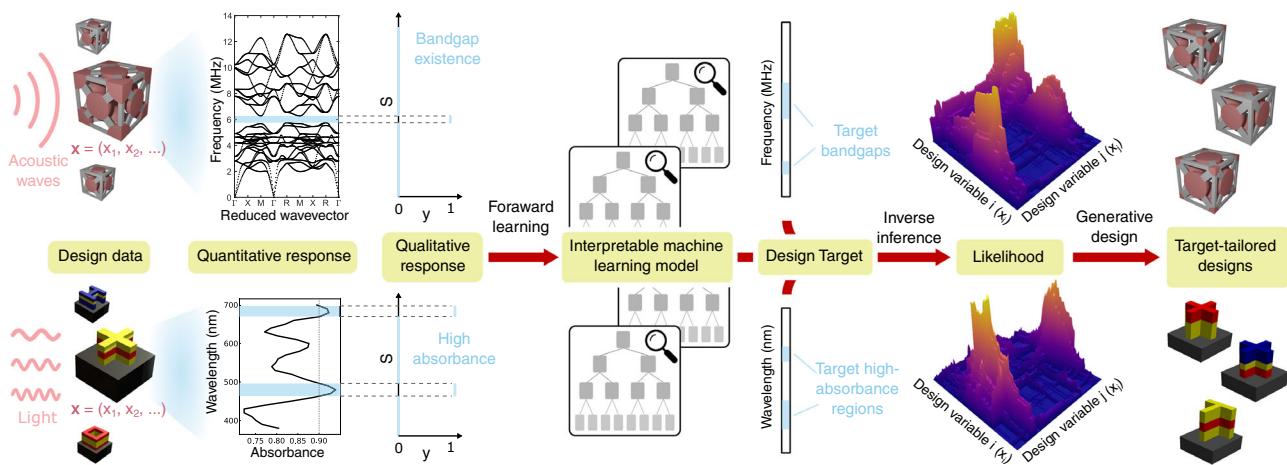


Figure 1. Schematic diagram of the RIGID method. We first train a random forest on a design-response dataset to learn the forward design-response relation—predicting qualitative responses (e.g., bandgap existence at any given wave frequency) of designs. Then given a design target, we can infer the likelihood of any design satisfying the target by probing into the trained random forest. New designs with tailored responses can be generated by sampling the design space based on the likelihood estimation.

methods, RIGID has a much lower cost in training and hyper-parameter tuning and works more robustly on small-size datasets (as random forests are less prone to overfitting). Similar to deep generative models, it can generate a desired number of solutions, allowing one to explore alternative solutions that might have desired properties or functionalities beyond the ones considered as the target. The explicit likelihood estimation also offers an interpretable quantification of a design’s target satisfaction probability and allows an exploitation–exploration trade-off when selecting generated designs. We validate the RIGID method on two metamaterial design examples—an acoustic metamaterial design example, where the goal is to generate metamaterials with specific bandgaps, and an optical metasurface design example, where the goal is to generate metasurfaces with high absorbance at specified wavelengths.

Our contributions are threefold. First, we propose an iteration-free, single-shot inverse design method that is fast, generative, interpretable, and small data compatible. Second, we demonstrate the effectiveness of the proposed method on acoustic and optical metamaterial design examples and propose both qualitative and quantitative ways of assessing our method. Finally, we create two synthetic test cases for fast examination and validation of model performance. These test cases can be used for future benchmarking studies of related methods.

2. Methods

The functional response of metamaterials can be modeled as $y = f(\mathbf{x}, s)$, where \mathbf{x} denotes metamaterial design variables (e.g., materials and geometry parameters), s is an auxiliary variable representing the independent variable (or the “ x -axis”) of the response (e.g., the frequency/wavelength or the external stimuli such as temperature), and y indicates the value of the response associated with our design target. In this article, we assume $y \in \{0, 1\}$ since we only focus on qualitative behaviors at specified frequencies (e.g., for an acoustic metamaterial or an optical

metamaterial design, whether a bandgap exists or whether the energy absorbance is higher than a threshold within a range of frequencies). We leave the more challenging problem of tailoring quantitative behaviors as future work.

Our goal is to solve the inverse problem—find a set of design solutions that satisfy $f(\mathbf{x}, s) = 1$ when s is inside a target domain Ω , that is, $\{\mathbf{x}^* | f(\mathbf{x}^*, s) = 1, \forall s \in \Omega\}$. Here $f(\mathbf{x}, s) = 1$ could mean, for instance, the existence of bandgaps for acoustic metamaterials or high absorption for optical metasurfaces. The domain Ω can be any frequency or wavelength interval(s). Note that this design target only requires the satisfaction of behaviors within the specified domain Ω , while behaviors outside this domain are out of concern in this study, although it is straightforward to adopt our method to this problem setting.

We use a random forest to approximate the function f . A random forest is an ensemble learning method that combines the predictions of multiple decision trees to improve accuracy and reduce overfitting.^[59] The trained random forest serves as a *forward prediction* model that predicts the outcome y given design variables \mathbf{x} and the auxiliary variable s . Compared to the widely used neural networks, the random forest as a forward prediction model 1) offers significantly faster training, 2) has fewer hyperparameters to tune, 3) is less susceptible to overfitting on small data, and 4) has higher interpretability (i.e., the decision-making of each tree in the random forest is transparent). More importantly, this interpretability also allows us to realize inverse design without training a separate inverse model.

Figure 2 shows how, by probing the trained random forest, one can estimate a likelihood distribution for target satisfaction of solutions over the entire design space and sample (generate) new designs based on this likelihood distribution. Since we target qualitative (binary) behaviors at specified s (e.g., a bandgap in 3–4 MHz frequency or high absorption at a wavelength of 400–500 nm), we first identify the leaf nodes (on each decision tree in the random forest) that are relevant to the s in the target (Figure 2, Step 1). We do this by tracing down each tree, checking only the nodes that use s as the splitting feature, and pruning the

Target \mathcal{T} : Generate acoustic metamaterials with bandgap(s) in 3-4 and 6-7 MHz frequency ranges

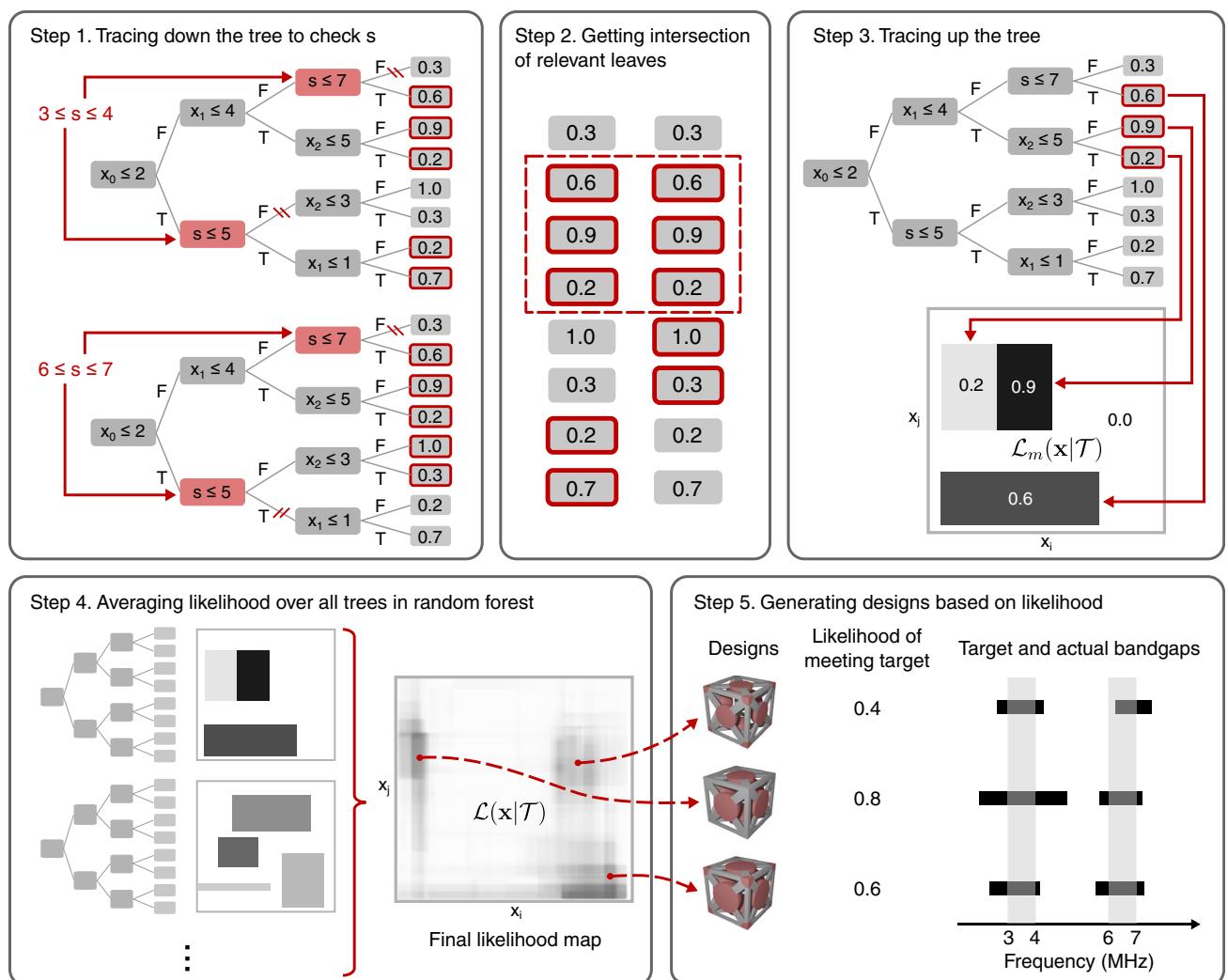


Figure 2. The inverse design pipeline of the proposed RIGID method (using the inverse design of acoustic metamaterials as an example). Given design parameters \mathbf{x} and the auxiliary variable s (e.g., wave frequency), a trained random forest predicts the probability of the qualitative response y (e.g., bandgap existence). Each tree in the random forest splits the joint space of \mathbf{x} and s into regions, each associated with a specific prediction (shown on leaf nodes). The splitting criteria are encoded in tree nodes. “T” means meeting a criterion and “F” means not meeting it. RIGID first identifies leaf nodes that are relevant to the considered range of auxiliary variable s by checking splitting criteria related to s and pruning tree branches that are irrelevant (Step 1). If the considered range of s has multiple parts, we repeat this step for each part and take the intersection of relevant leaves (Step 2). Each relevant leaf node corresponds to a decision path, indicating a region in the design space, as well as a predicted probability of target satisfaction, which is a score we assign to the corresponding design space region (Step 3). With multiple trees in a random forest, we can average the scores predicted by each tree and use the average score as our likelihood estimation (Step 4). We can then sample from the design space based on the likelihood distribution to generate new designs tailored to the target (Step 5). Note that the 2D likelihood maps are only for visualization purposes. The actual dimension will be the same as the design space dimension (i.e., the number of design variables).

branches that are irrelevant to the s in the target. For example, as shown in Figure 2, there are two tree nodes using s as the splitting feature, with splitting criteria at $s \leq 5$ and $s \leq 7$. Given the target frequency range of $3 \leq s \leq 4$, we can remove the right branches of both nodes as these branches are only relevant to $s > 5$ and $s > 7$, respectively, which conflicts with the target range of $3 \leq s \leq 4$. After pruning these branches, we end up with a set of leaves relevant to the target (highlighted in Figure 2, Step 1). When we have a combined target (e.g., bandgaps in both 3-4 and 6-7 MHz, as

shown in Figure 2), we need to get the intersection of all the sets of relevant leaves and use that as the final set of relevant leaves (highlighted in Figure 2, Step 2). Note that a combined target includes cases where there are multiple nonadjacent target ranges (e.g., 3-4 and 6-7 MHz) or when a target range is split by a tree node (e.g., a target range of 4-6 MHz can be split by the node “ $s \leq 5$ ”, thus we need to consider it as the combination of two target ranges—4-5 and 5-6 MHz). A more detailed discussion of this step is in Section S2, Supporting Information.

The next step is to trace up the tree from the N relevant leaves, obtained by Step 2, to the root node (Figure 2, Step 3). This will result in N decision paths, along which are nodes indicating splitting criteria for design variables \mathbf{x} . Thus, each decision path represents a set of design variable ranges or in other words, a region in the design space. We assign each region a score equal to the predicted probability at each corresponding leaf. This probability is learnt from the training data and equals the proportion of positive data in a leaf. It indicates the tree's belief in the probability of a design \mathbf{x} satisfying the target $\mathcal{T} = \{f(\mathbf{x}^*, s) = 1 | \forall s \in \Omega\}$ ($\Omega = [3, 4] \cup [6, 7]$ in this example) if the design falls in the design space region corresponding to the leaf. Therefore, with a single-decision tree m , we already have the map of likelihood $\mathcal{L}_m(\mathbf{x}|\mathcal{T}) = \mathbb{P}_m(\mathcal{T}|\mathbf{x})$ for target satisfaction: each of the N regions has a uniformly distributed likelihood equal to the predicted probability at the corresponding leaf, and the rest of the design space has a likelihood of 0 (Figure 2, Step 3).

Since a single decision tree usually lacks accuracy, robustness, and a way to quantify estimation uncertainty, we still want to take advantage of the random forest as an ensemble model for inverse design. We use Steps 1–3 to derive the likelihood distribution for each of the M trees in the random forest and simply use the average of these M likelihood distributions as the final likelihood for target satisfaction, $\mathcal{L}(\mathbf{x}|\mathcal{T}) = \sum_{m=1}^M \mathcal{L}_m(\mathbf{x}|\mathcal{T})/M$, which is a more complex and smooth function (Figure 2, Step 4). If more trees believe a design \mathbf{x} has a higher likelihood of satisfying the target, then the design will have a higher likelihood $\mathcal{L}(\mathbf{x}|\mathcal{T})$. Finally, to generate new designs, we can sample from $\mathcal{L}(\mathbf{x}|\mathcal{T})$ using Markov chain Monte Carlo (MCMC) methods such as Metropolis–Hastings^[60] (Figure 2, Step 5).

We can also derive the posterior of a design \mathbf{x} given the target if we know the prior $p(\mathbf{x})$ based on Bayes' theorem.

$$p(\mathbf{x}|\mathcal{T}) \propto \mathcal{L}(\mathbf{x}|\mathcal{T})p(\mathbf{x}) \quad (1)$$

In this work, we assume a uniform prior, making the posterior directly proportional to the likelihood $\mathcal{L}(\mathbf{x}|\mathcal{T})$, and sampling from the posterior is equivalent to sampling from the likelihood.

Compared to prior works, RIGID provides the following unique benefits. 1) It is effective on small data problems as the random forest is less susceptible to overfitting. 2) The training is fast (in seconds of wall time) and does not require computationally demanding hyperparameter tuning. Once the training is done, no further training or iterative optimization is required to generate designs for different targets. 3) The method estimates the explicit likelihood of target satisfaction for every possible solution in the design space. Given a design target of specific functional behavior, we can generate an unlimited number of solutions based on the likelihood, allowing us to explore alternative solutions that might have desired properties or functionalities beyond the ones considered as the target. 4) The method offers a high level of transparency as one can easily probe the trained model to understand its reasoning behind any decision-making (i.e., why a design has a high/low likelihood). 5) When generating design solutions, one can use a single parameter—the sampling threshold—to easily tune the trade-off between exploitation (i.e., generated designs have higher chances of satisfying the target) and exploration (i.e., generated designs

cover a broader area of the design space), as we will demonstrate in Results.

3. Results

We demonstrate our RIGID method on an acoustic metamaterial design problem, an optical metasurface design problem, and two synthetic design problems. Based on a recent review article by Lee et al.^[34] and other related works (e.g., ref. [38]), existing iteration-free, single-shot inverse design methods were demonstrated on training data size ranging from 10^3 to 10^6 in scale. Here we show that our method can work with much smaller-scale datasets (i.e., fewer than 250 training samples for all the case studies in this work).

For all the test problems, we used the same random forest hyperparameter settings and did not perform hyperparameter tuning. Specifically, each random forest contains 1000 trees with a minimum of two samples required to split an internal node and a minimum of one sample required to be at a leaf node. Gini impurity^[61] was used as the splitting criterion at tree nodes. The train-test split ratio was 4:1. Since the positive/negative training data can be highly imbalanced (e.g., the frequency ranges with bandgaps are much narrower than those without), we used the synthetic minority oversampling technique^[62] to oversample the positive class. For all the case studies, the random forest training took less than 10 s on an Intel Core i5-9300 H CPU 2.4 GHz and 8 GB memory. After training, we generate new designs by sampling from the resulting likelihood distribution using Metropolis–Hastings. In practice, Metropolis–Hastings can generate identical samples, which provides no benefits for design purposes. Thus in this work, we reject the designs identical to previous ones during sampling.

We compare RIGID to a commonly used low-cost generative design strategy—surrogate modeling combined with a genetic algorithm (GA).^[63–67] We choose this baseline according to the following criteria to ensure that its efficiency and application scenario are comparable to those of RIGID: 1) the baseline method should generate design solutions quickly (i.e., no additional training or design evaluation is required given different design targets), 2) the baseline method can generate multiple satisfying solutions to the inverse design problem, and 3) the baseline method can handle small data. These criteria exclude sequential design strategies like Bayesian optimization (as they violate the first criterion) and any existing single-shot inverse design methods mentioned in Section 1 (as direct inverse mapping and T-NN violate the second criterion and conditional generative model-based methods usually violate the last criterion).

GANs are optimization methods that iteratively evolve a population of candidate solutions, based on biologically inspired operators such as mutation, crossover, and selection, toward better solutions.^[68] As we usually need to evaluate a large population of candidate solutions at each iteration (referred to as a generation), a surrogate model (usually a machine learning model) is often constructed to accelerate design response prediction, thereby speeding up the optimization process. For the surrogate model in the baseline method, we use the same trained random forest as in RIGID for consistency, eliminating differences due to

forward prediction accuracy and focusing on comparing the inverse design methods. We set the objective of GA to maximize the aggregated probability of target satisfaction within the target domain Ω , that is, $\max_x \sum_{s_i \in \Omega} f'(\mathbf{x}, s_i)$, where $\{s_i\}_{i=1}^n$ are discretized auxiliary variable values (e.g., discrete values of frequencies or wavelengths) and f' represents the trained random forest. We include the detailed configurations of GA in Section 6. After optimization, we rank the solutions generated throughout the optimization process based on their objective values and select the top solutions as final designs. We compare these designs to those generated by RIGID. From this point, we will call designs generated by RIGID and GA as RIGID designs and GA designs, respectively.

3.1. Applying RIGID to Design Acoustic Metamaterials with Target Bandgaps

Here we consider acoustic metamaterials that can control elastic wave propagation at ultrasound (MHz) frequencies. Varying the microscale geometries of acoustic metamaterials changes the dynamic properties of a material, such as bandgaps^[18] (i.e., forbidden frequency ranges of a material) and wave propagation direction.^[12] These materials promise applications in waveguides,^[12,13] lenses,^[14,15] and vibration mitigation.^[17] Designing acoustic metamaterials with target bandgaps is challenging, as many 3D architectures do not naturally have full bandgaps. We present the braced cubic design framework (Figure 3A,B) as a

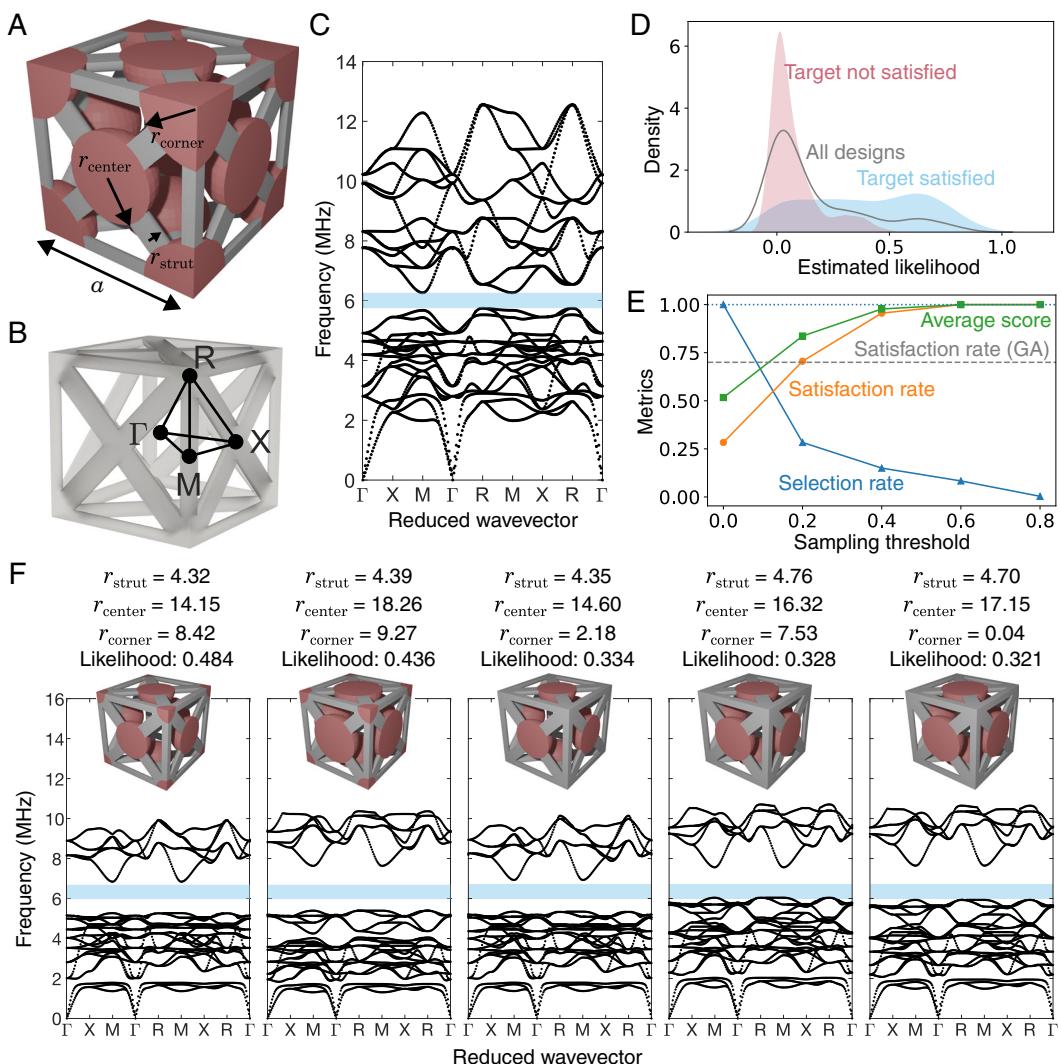


Figure 3. Acoustic metamaterial design problem configuration and results. A) Design variables of center and corner mass radii (r_{center} and r_{corner}) and strut radius (r_{strut}). B) High-symmetry points of the cubic irreducible Brillouin zone. C) A sample dispersion relation and bandgap (marked by the highlighted zone). The design objective is to generate new acoustic metamaterial designs with target bandgaps. D) KDE of the estimated likelihood for generated designs. E) Satisfaction rates, average scores, and selection rates for RIGID designs under varying sampling thresholds (solid lines), in comparison to the satisfaction rate of GA designs (horizontal dashed line). The horizontal dotted line indicates 100% satisfaction. F) Geometries and corresponding dispersion relations of five RIGID designs with the highest likelihood of satisfying a specified target bandgap (marked as highlighted frequency regions). All the radii (r_{strut} , r_{center} , and r_{corner}) have unit of μm . Here only the fourth design fails to meet a small portion (at around 6 MHz) of the target bandgap, whereas the others meet it. Generated designs for other targets are shown in Appendix, Figure S2–S6, Supporting Information.

method to tune the size and location of bandgaps (Figure 3C). In particular, spherical micro-inertia are added to the center and corner of a braced cubic unit cell with strut radius r_{strut} . Microinertia placed at the center of the brace has radius r_{center} while microinertia placed at the corner of the cubic unit cell has radius r_{corner} .

We randomly created 284 sets of geometric parameters $\mathbf{x} = (r_{\text{strut}}, r_{\text{center}}, r_{\text{corner}})$ with $4 \leq r_{\text{strut}} \leq 6.41$, $0 \leq r_{\text{center}} \leq 20$, and $0 \leq r_{\text{corner}} \leq 20$ (unit: μm). The unit cell size was set at $a = 60 \mu\text{m}$. For each of these designs, we performed Bloch-wave analysis to compute its acoustic dispersion relation. Bandgap location and width were extracted for each design based on its dispersion relation.

Out of the 284 sets of design variables and bandgap data, we used 227 samples as training data. We first discretized the entire frequency range into 100 intervals and trained a random forest to predict bandgap existence $y \in \{0, 1\}$ at a specific interval s for a given design \mathbf{x} . The trained model has a test F1 score of 0.82. The resulting confusion matrix on test data is shown in Appendix Table S1, Supporting Information.

To test the inverse design capability of RIGID, we randomly created ten targets, each containing 1–2 frequency ranges in which bandgap(s) should exist. We generated 30 designs for each target by sampling from the resulting likelihood distribution over the design space (Note that it is possible for the likelihood to be zero everywhere in the design space when the model believes the target is unachievable. We ignore these cases as it is meaningless and impossible to sample designs from such likelihood distribution). Bandgaps were identified from dispersion relations computed using Bloch-wave analysis.

Figure 3D shows the kernel density estimation (KDE) for the likelihood of the 300 RIGID designs, conditioned on their target satisfaction. We use \mathcal{D} and $\mathcal{D}_{\text{feas}}$ to represent the complete set of generated designs and the set of generated designs that actually satisfy the target, respectively. Then $\mathcal{D} \setminus \mathcal{D}_{\text{feas}}$ denotes the set of generated designs that cannot fulfill the target in reality. In an ideal scenario, all solutions in \mathcal{D} would satisfy the target, which means $\mathcal{D} = \mathcal{D}_{\text{feas}}$, and their density profiles should coincide. However, this ideal scenario is impossible in reality due to both limited model accuracy and uncertainty. As RIGID generates designs by sampling the entire design space based on the likelihood values, it is expected for nonsatisfactory designs to be generated as long as the likelihood values for these designs are nonzeros. Therefore, the fact that a design is generated does not mean that the model is certain about the design satisfying the target. We have to look at the estimated likelihood to know how likely the target will be satisfied and make decisions from there. RIGID thus incorporated the model's confidence into the inverse design process. For a reasonable model, most designs in $\mathcal{D} \setminus \mathcal{D}_{\text{feas}}$ should have low estimated likelihood values. Consequently, the density of $\mathcal{D}_{\text{feas}}$'s likelihood is a result of shifting some of \mathcal{D} 's density from left (low likelihood) to right (high likelihood). This expectation aligns with the observation in Figure 3D.

When sampling new designs or selecting solutions from generated designs, we can put a sampling threshold $\tau \in (0, 1)$ on the likelihood values to filter out “less promising” solutions. To further examine model behavior and quantify how τ affects the inverse design outcome, we define the following metrics.

$$\begin{aligned} \text{Selection Rate} &= \frac{|\mathcal{D}_{\phi \geq \tau}|}{|\mathcal{D}|} \\ \text{Satisfaction Rate} &= \frac{|\mathcal{D}_{\phi \geq \tau} \cap \mathcal{D}_{\text{feas}}|}{|\mathcal{D}_{\phi \geq \tau}|} \\ \text{Average Score} &= \frac{1}{|\mathcal{D}_{\phi \geq \tau}|} \sum_{i=1}^{|\mathcal{D}_{\phi \geq \tau}|} q_i \end{aligned} \quad (2)$$

where $\mathcal{D}_{\phi \geq \tau}$ is the set of generated designs with the likelihood of at least τ (i.e., the selected designs) and q_i denotes the percentage overlap between the target and the actual behavior of selected designs. The satisfaction rate evaluates how many selected designs satisfy the target based on a binary criterion (i.e., whether or not a design satisfies the complete target), whereas the average score provides a soft measure where partial satisfaction is also counted. The average score is lower bounded by the satisfaction rate.

As shown in Figure 3E, the selection rate decreases when τ increases since more solutions are filtered out. On the other hand, both the satisfaction rate and the average score increase with τ , which indicates a high correlation between the estimated likelihood of a solution and its probability of actually achieving the target. As τ reaches 0.6, the satisfaction rate and the average score reach 1, indicating that all generated designs satisfy their targets. When sampling or selecting new solutions, we can use the sample threshold τ to tune the trade-off between exploitation and exploration—a low τ favors exploration as sampled solutions will cover a larger area of the design space, while a high τ favors exploitation as sampled solutions will have a higher chance of satisfying the target.

In the comparative study, we generated the same number of designs for the same targets by applying GA. We computed the satisfaction rate of GA designs and found that the value is close to the satisfaction rate of RIGID designs when setting τ to 0.2. This means that if we set τ to be above 0.2, RIGID will most likely achieve a higher satisfaction rate than GA. The distributions of generated satisfactory solutions shown in Figure 4 indicate that GA designs are highly localized while RIGID can capture the high diversity of the inverse design solutions and sometimes discover solutions very different from the original feasible solutions from data. Besides, Figure 4 also shows an example where GA fails to find any satisfactory solution while RIGID can still generate a very diverse set of satisfactory solutions.

Figure 3F visualizes the geometries and dispersion relations of RIGID designs generated based on a randomly created bandgap target. Only the top five designs with the highest likelihood values are shown. In this example, our method generates geometrically different designs that have a high probability of achieving target bandgaps, each yielding a slightly different dispersion relation. This is promising in design applications requiring other material properties, such as dynamic wave velocity or quasistatic stiffness, in which the user can select from a menu of designs with the same target bandgap but other varying properties. Generated designs based on the other nine bandgap targets can be found in Appendix Figure S2–S6, Supporting Information.

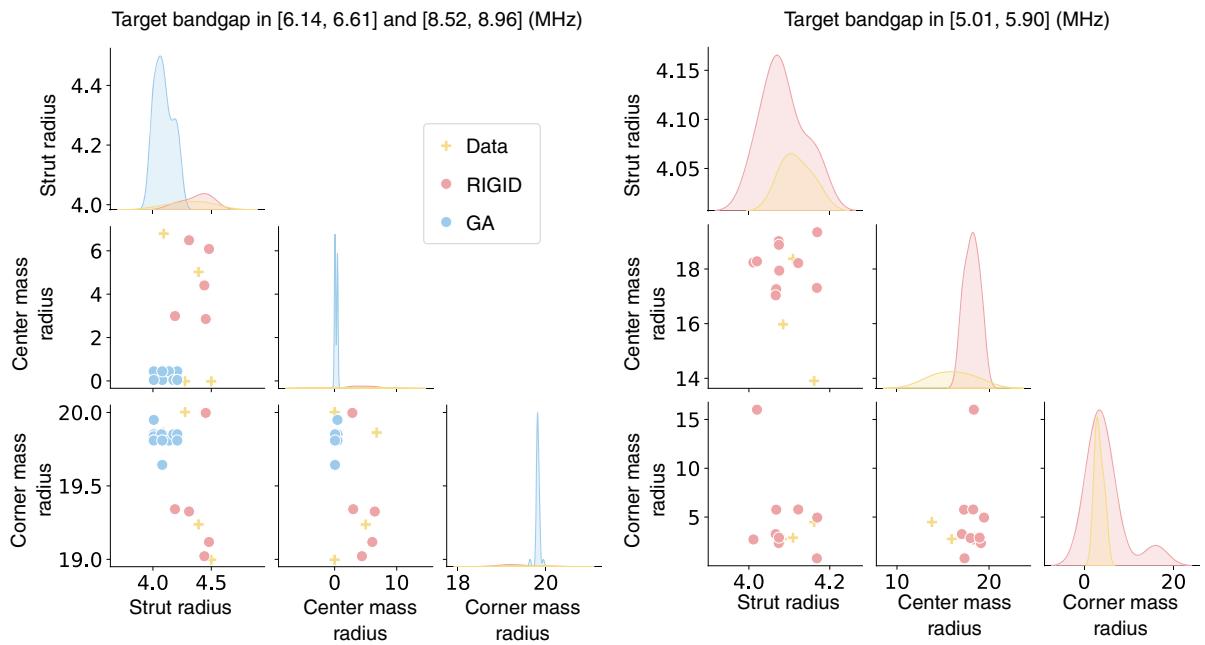


Figure 4. Distributions of satisfactory solutions for two bandgap targets. The off-diagonal plots show the pairwise bivariate distributions of design variables, and the diagonal plots show the marginal distributions of the data in each column. The left panel shows that GA designs are highly localized while RIGID can lead to diverse solutions. The right panel indicates that none of the GA designs satisfy the target, while satisfactory RIGID designs are diverse and can be very different from feasible designs from data. Solutions from data include feasible designs in both training and test data.

3.2. Applying RIGID to Design Optical Metasurfaces with Target High-Absorbance Wavelengths

Optical metasurfaces are artificially engineered systems that can support exotic light propagation building on subwavelength inclusions.^[7,69–73] Among a diverse array of applications, metamaterial absorbers^[74–80] have been intensely studied for medical imaging, sensing, and wireless communications.

In this case study, we consider four types of cross sections ($c \in \{1, 2, 3, 4\}$) chosen from the literature (Figure 5B). It is assumed that a 3D geometric instance is composed of a stack of three layers of prismatic unit cells, each of which is vertically extruded and stacked (Figure 5A). The geometries constructed in this way can be regarded as an instantiation of multilayered metasurfaces,^[81–85] which offer richer design freedom than the single-layer counterpart. The thickness of each layer (h_l , $l = 1, 2, 3$) is allowed to continuously vary between 50 and 150 nm. Herein we do not consider parametric variations of a given type of unit cell cross section; yet those can be trivially incorporated in the proposed design framework if necessary.

We also design the material of each layer (m_l , $l = 1, 2, 3$). Three dielectric materials of interest, each of which is assigned to a different color in Figure 5A, are Ti (red), Si (blue), and Ge (yellow). The challenge associated with this design problem is its mixed-variable design space containing three continuous variables (i.e., layer thicknesses) and four categorical variables (i.e., material choices at each layer and the geometry type), which may lead to potentially complicated underlying design-response relations (e.g., those across different geometry or material types). In general, a dielectric material is characterized through a complex

refractive index $\tilde{n} \in \mathbb{C}$ defined as $\tilde{n} = n + jk$, where $j = \sqrt{-1}$ is the imaginary unit, $n \in \mathbb{R}$ involves the speed at which the light propagates through the material, and $k \in \mathbb{R}$ is the extinction coefficient that dictates the energy loss due to the material. Within the frequency regime of interest, those exhibit nonlinear dispersion; both the real and imaginary terms in general are a nonanalytic function of excitation wavelength s , that is, $n(s)$ and $k(s)$. In addition, the impact of the same material choice on the spectral response $A(s)$ varies depending on the layer location at which the material is placed.

Based on the above configuration, we randomly sampled 258 sets of design variables $\mathbf{x} = (c, h_1, h_2, h_3, m_1, m_2, m_3)$ and computed their corresponding absorbance spectra using wave analysis. We set $t = 0.9$ as the absorbance threshold, so that “high” absorbance means the absorbance $A(s)$ is no less than 0.9. We trained a random forest on 206 training data (i.e., 80% of the 258 designs and corresponding absorbance spectra) to predict whether “high” absorbance is presented (i.e., the binary indicator $y = 1$) at a wavelength s for a given design \mathbf{x} . The trained random forest has a test F1 score of 0.83. The confusion matrix on test data is shown in Appendix Table S2, Supporting Information.

Note that this problem involves inverse design with both continuous and categorical variables, which common optimization and generative modeling-based inverse design cannot handle well without special treatment.^[86–88] On the other hand, our random forest-based method can naturally address such mixed-variable problems without any issues.

Similar to the acoustic metamaterial design problem, we use ten randomly created targets to evaluate the inverse design performance of RIGID, except that here a target is represented as the

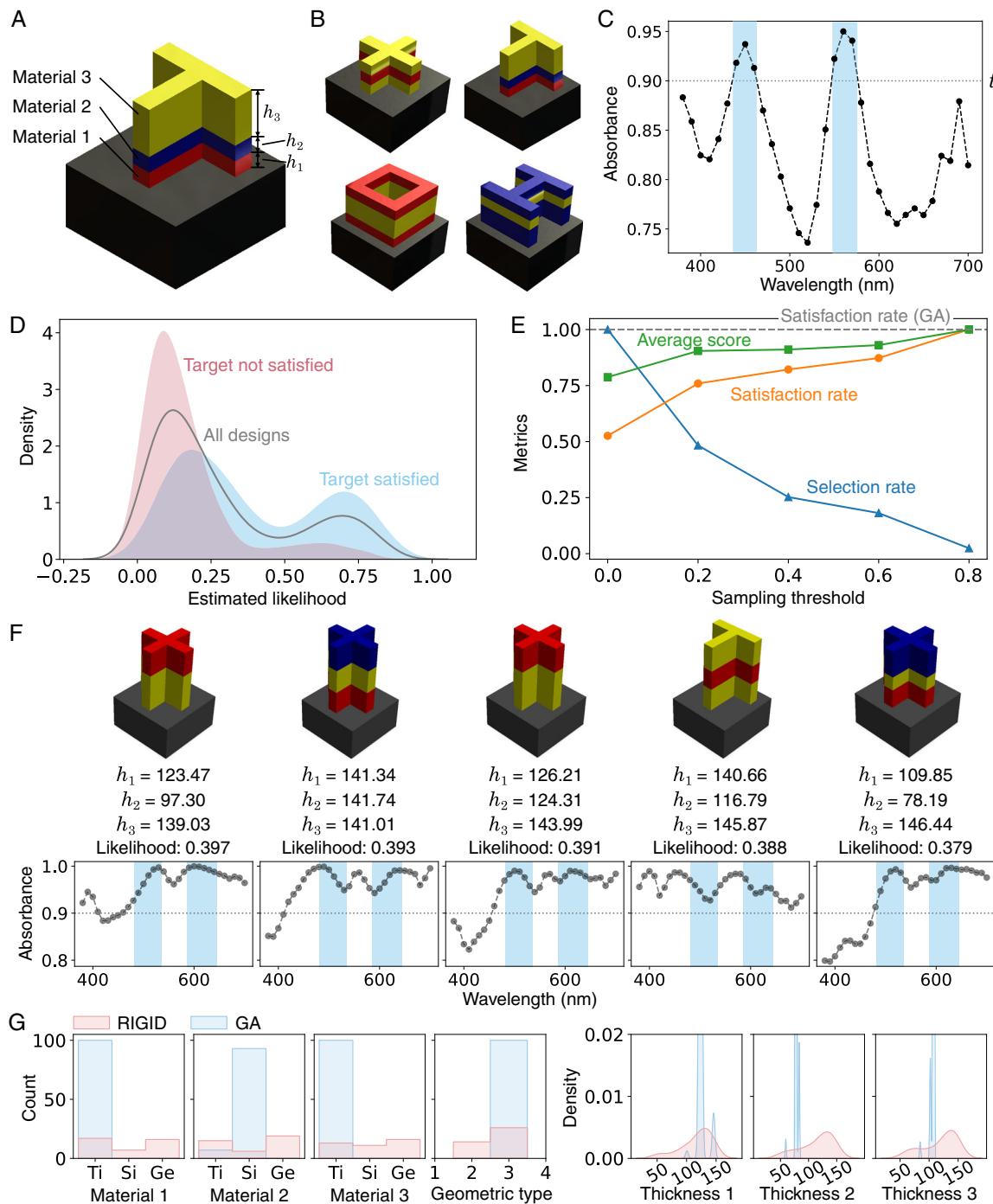


Figure 5. Optical metasurface design problem configuration and results. A,B) Design variables (materials, layer thicknesses, and cross-section geometry types). C) A sample absorbance spectrum and the wavelength intervals (highlighted wavelength regions) corresponding to absorbance above the threshold t . The design objective is to generate new optical metasurface designs that exhibit higher absorbance than a threshold t at the user-defined wavelength interval(s). D) KDE of the estimated likelihood for generated designs. E) Satisfaction rates, average scores, and selection rates for RIGID designs under varying sampling thresholds (solid lines), in comparison to the satisfaction rate of GA designs (horizontal dashed line). F) Designs (geometries and material selections) and corresponding absorbance spectra of five metasurfaces generated by RIGID. These five solutions are generated designs with the highest likelihood of satisfying specified target high-absorbance regions (marked as highlighted wavelength regions). All the layer thicknesses ($h_i, i = 1, 2, 3$) have a unit of nm. Here all five designs satisfy the target. Generated designs for other targets are shown in Appendix Figure S7–S9, Supporting Information. G) Distributions of design variables for satisfactory solutions generated by RIGID and GA (for the same target defined in Panel F). GA designs are highly localized and lack diversity compared to RIGID designs.

wavelength range(s) within which absorbance should be at least 0.9. We generated 100 designs for each target by sampling from the estimated likelihood distribution. Among the 1000 generated solutions, we successfully conducted wave analysis for 911 designs and obtained their absorbance spectra. Figure 5D shows the KDE for the likelihood of these 911 designs, conditioned on their target satisfaction. The densities share similar behavior as in the acoustic problem (Figure 3D)—unsatisfied/infeasible designs $\mathcal{D} \setminus \mathcal{D}_{\text{feas}}$ are concentrated at low likelihood regions, which cause the likelihood density of satisfied/feasible designs $\mathcal{D}_{\text{feas}}$ to be a result of shifting some of \mathcal{D} 's density from left (low likelihood) to right (high likelihood). The sampling threshold and metrics relation shown in Figure 5E also follow the same trend as in the acoustic problem (Figure 3E), which again demonstrates a strong positive correlation between the estimated likelihood and the probability of generated designs actually achieving their targets.

We also generated the same amount of designs under the same targets using GA. We found that GA designs can generate 100% satisfaction rate (Figure 5E). However, similar to the acoustic metamaterial design example, the generated designs are localized compared to RIGID designs which cover much wider feasible region(s) in the design space. This can be reflected by the design variable distributions of generated feasible solutions shown in Figure 5G, which correspond to generated designs for a randomly created target (the target is shown as the highlighted wavelength regions in Figure 5F). For example, all the GA designs have the same cross-section geometric type while RIGID generated satisfactory designs with different geometric types. It is easy to see that, without considering solution diversity, achieving a high target satisfaction rate is trivial—as long as we discover one satisfactory solution, we can add sufficiently small perturbations to that solution to generate an infinite number of alternative solutions that have similar responses and thus are also likely to satisfy the target. Note that compared to the acoustic problem, this optical problem is less challenging as there is an average of 28.5% data already satisfying the ten targets (in contrast to an average of 9.3% satisfactory designs from data in the acoustic problem). Therefore, finding a few satisfactory solutions for the optical problem is relatively easy; the challenge lies in discovering all feasible solutions, where RIGID excels by identifying a wide range of them. We can also see that when the sampling threshold is 0.8, RIGID achieves a satisfaction rate of 100%, demonstrating its ability to trade off the diversity of generated designs for higher feasibility.

Figure 5F shows generated RIGID designs with the top five likelihood estimations for a randomly created target. While the materials, cross-section geometries, and layer thicknesses of generated designs can be different, all the designs satisfy the target. Generated designs based on the other nine targets can be found in Appendix Figure S7–S9, Supporting Information.

3.3. Synthetic Design Problems for Rapid Validation and Visualization

While the above metamaterial design problems represent practical use cases, the validation study is time-consuming due to the expensive computation of metamaterials' responses such as

dispersion relations and absorbance spectra. To allow fast validation of the proposed method and easier inspection of the estimated likelihood in the design space, we create two synthetic case studies. Both problems have 2D “design spaces” that allow easy visualization.

3.3.1. SqExp Problem

To construct the first synthetic problem, we used a *squared exponential function* with tunable parameters a and b to mimic the quantitative functional response of metamaterials. The qualitative response (e.g., “high”- or “low”-energy absorption at a wavelength) is defined as

$$I(a, b; s) = \begin{cases} 0, & \text{if } z = \exp\left(-\left(\frac{s-a}{0.3b+0.1}\right)^2\right) < t \\ 1, & \text{otherwise} \end{cases} \quad (3)$$

where z represents quantitative response and t is a threshold that converts z into a qualitative response $I(a, b; s)$. Specifically, $I(a, b; s) = 1$ can mean the existence of a bandgap or high absorbance at a frequency s . Then $\{s | I(a, b; s) = 1\}$ represents a range of s that mimics our interested material behavior, such as the frequency range of the bandgap or the wavelength range of high absorbance. We call this the *positive range*. By varying a and b , we can produce different synthetic responses and positive ranges. Therefore, we can use a and b as synthetic design variables. There is a clear relation between these design variables and the positive range that Equation (3) creates— a and b control the center location and the width of the positive range, respectively.

In this design problem, we sampled 100 sets of a and b uniformly at random. We set t as 0.9. Based on Equation (3), we obtained the corresponding responses (Figure 6A). These sets of a , b , and responses constitute a dataset for training and testing our model.

3.3.2. SupSin Problem

Another synthetic design problem was constructed by replacing the squared exponential function in the SqExp problem with a *superposed sine function*. Given synthetic design variables a and b , we can produce qualitative responses using the following equation.

$$I(a, b; s) = \begin{cases} 0, & \text{if } z = \sin(2\pi(s+a)) + \sin(3\pi(s+b)) < t \\ 1, & \text{otherwise} \end{cases} \quad (4)$$

Same as in the SqExp problem, we set $t = 0.9$ and created a dataset with 100 sets of synthetic design variables and corresponding ranges derived from synthetic responses (Figure 6B). Unlike the squared exponential function, the superposed sine function can be multimodal, which means it can result in multiple synthetic ranges to mimic, for example, multiple bandgaps. The bandgap locations are controlled by a and b .

For each synthetic example, we split the data into 80 training data and 20 test data. We trained a random forest with the same hyperparameter settings as the other problems, to predict the binary indicators $I(a, b; s)$. The F1 scores are 0.85 and 0.86 for the SqExp and the SupSin problems, respectively. The resulting

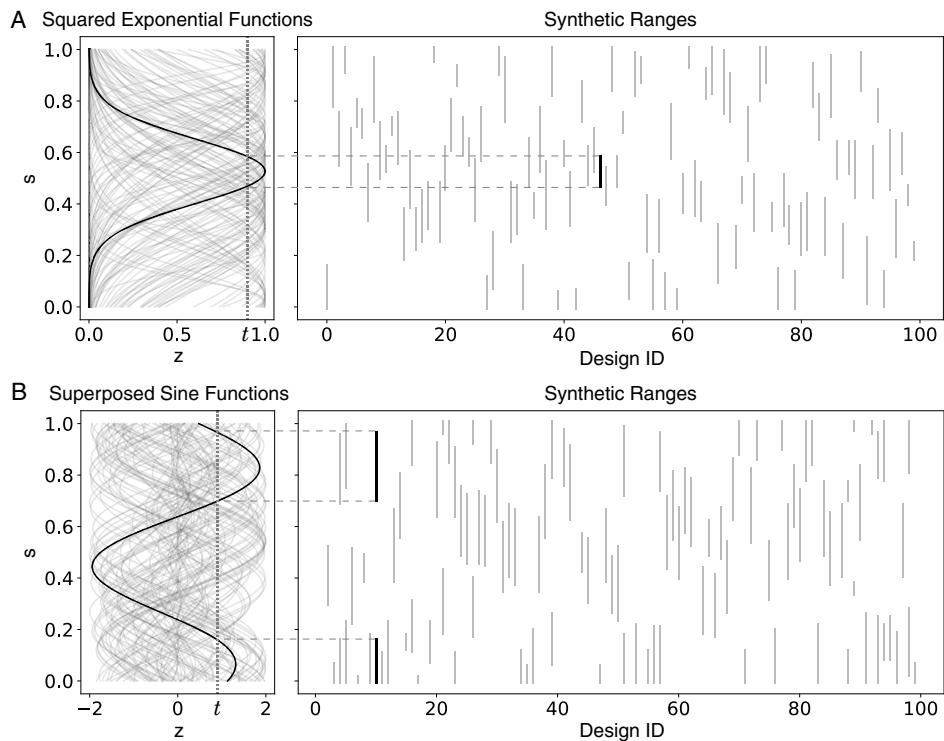


Figure 6. Synthetic data creation for A) the SqExp problem and B) the SupSin problem. For each problem, the left panel shows 100 functions with randomly sampled parameters a and b . We treat a and b as synthetic design variables and the corresponding functions as quantitative responses (e.g., absorbance spectra of optical metasurfaces). The right panel shows qualitative responses (e.g., high-absorbance wavelength ranges or bandgaps) are simulated by synthetic ranges, derived by thresholding the 100 functions (Equation (3) and (4) with threshold $t = 0.9$).

confusion matrices are shown in Appendix, Table S3 and S4, Supporting Information. We evaluated the inverse design performance with the trained models.

Due to the fast evaluation of Equation (3) and (4), we can exhaust all the possible solutions in the design space to obtain the ground-truth feasible region(s) for a target. **Figure 7A** shows the estimated likelihood values and the ground-truth feasible regions under randomly created targets. In general, high-likelihood regions match actual feasible regions well, which further demonstrates the effectiveness of RIGID. We can also observe that feasible regions in the SqExp and the SupSin problems follow distinct patterns. In the SqExp problem, a and b control the center location and the width of the positive range, respectively. Therefore, the position of the feasible region along the a -axis moves with the location of the target range, while the feasible region gradually shrinks as b decreases since the decrease of b (i.e., positive range width) causes the choice of a (i.e., positive range center location) to be more restricted to fit the target range. In the SupSin problem, there might be multiple positive ranges appearing at the peaks of the superposed sine function in Equation (4). Design variables a and b control positive range locations by translating each sine function. Due to the sine function's periodicity, we can obtain multiple feasible regions along both a - and b -axes. **Figure 7A** shows that the likelihood estimation by RIGID successfully captured the above-mentioned patterns of feasible regions.

We also visualize the distributions of satisfactory designs generated by both RIGID and GA in **Figure 7A**. Again, we observe a

much broader coverage of the feasible design space by RIGID designs, while GA designs are merely minor perturbations of one or two feasible solutions.

Figure 7B demonstrates how the estimated likelihood varies when increasing the number of trees in a random forest. With a single-decision tree, the estimated likelihood function is almost a binary function and highly inaccurate. The likelihood in the SqExp case is even zero everywhere, which makes it impossible to sample designs based on the likelihood. As the number of trees increases, the likelihood function becomes smoother and eventually converges.

Besides these qualitative visual inspections, we also calculated the metrics proposed in Equation (2), as shown in **Figure 7C**. For each of the two synthetic problems, these metrics were computed on 500 designs generated by giving five random target ranges. Again, the satisfaction rate and the average score increase with the sampling threshold, indicating a strong correlation between the sampling threshold and the probability of generated designs actually achieving their targets. In both problems, all the selected designs satisfy their targets (i.e., the satisfaction rates and average scores reach 1) when the sampling threshold reaches 0.8.

4. Discussion

In this section, we discuss the benefits and necessities of RIGID in terms of its interpretability, data demand, the ability to deal

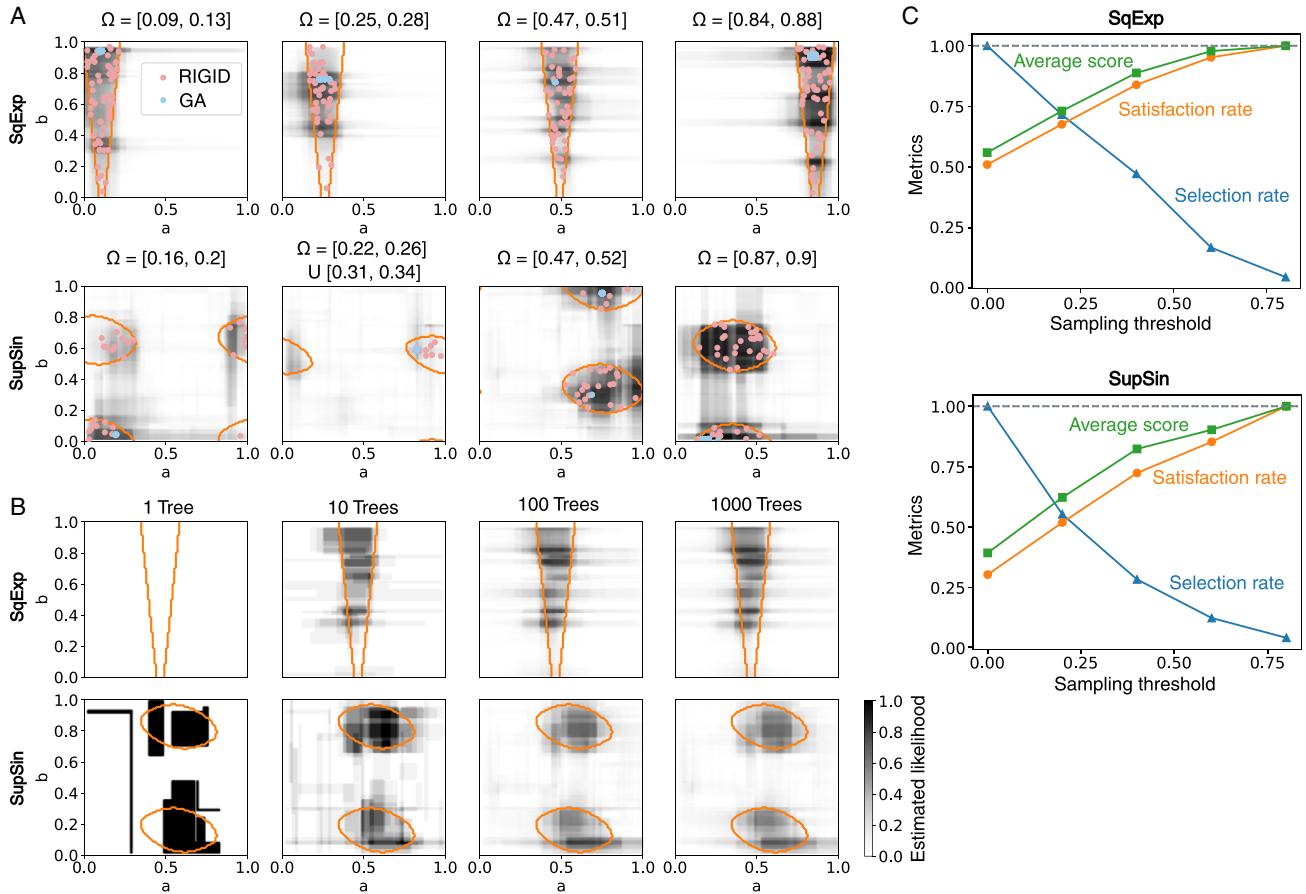


Figure 7. Visualization of estimated likelihood and validation metrics for synthetic problems. A) Likelihood function values for randomly created design targets. Orange lines show boundaries of actual feasible regions associated with the targets $\mathcal{T} = \{I(a, b; s) = 1 \mid \forall s \in \Omega\}$. Points show satisfactory RIGID designs and GA designs. B) Likelihood function values estimated by random forests with varying numbers of decision trees. The design target is set as $\mathcal{T} = \{I(a, b; s) = 1 \mid \forall s \in [0.45, 0.48]\}$ for the SqExp problem and $\mathcal{T} = \{I(a, b; s) = 1 \mid \forall s \in [0.63, 0.68] \cup [0.69, 0.71]\}$ for the SupSin problem. C) Validation metrics for inverse design generation.

with discrete design variables, and the diversity of generated designs.

4.1. Interpretability

RIGID offers two levels of interpretability. First, RIGID is more transparent than deep learning models. The transparency of the random forest as a forward model helps us understand how the predictions are made. The inverse design process based on likelihood estimation and sampling is also transparent. The transparency of both forward and inverse processes offers insights into the decision-making process and diagnostic capabilities. Second, owing to the interpretability of the forward prediction model, we can derive the likelihood, which the inverse design relies on, by probing into each decision path of trees in the random forest. The likelihood is an interpretable metric to help us understand the probability of a generated design satisfying the given target and thus aid the decision-making process when selecting final solutions. The results (Figure 3D,E, 5D,E, and 7C) have demonstrated a high correlation between the likelihood and actual probability of target satisfaction. The likelihood also serves as an

indicator of the need for adjusting the design space. When the likelihood is low across the entire design space, the model believes the target is almost impossible to achieve, which may indicate a need for expanding the design space (i.e., extending design space bounds or adding new design freedom) to allow wider coverage of the response space and easier discovery of satisfying solutions. Please see more discussion in Appendix, Section S4, Supporting Information. Another benefit of having the explicit likelihood as an interpretable metric is that we can adjust the likelihood threshold to trade-off between exploration and exploitation—higher (lower) threshold means more focus on exploitation (exploration). Both levels of interpretability were absent from most prior works of iteration-free, single-shot inverse design methods.

4.2. Data Demand

RIGID is particularly useful when data collection is expensive, as in many cases where high-fidelity simulation or experimental data are needed. Despite the advancement of surrogate solvers and hardware-accelerated simulation, a persistent trade-off

between computational resources and accuracy remains, exacerbated by challenges in accessing high-performance infrastructure and collecting high-fidelity data. With most of the prior inverse design works based on deep learning with a high demand for data and computational resources, it is useful to create a method that works under scenarios at the other end of the data/computational resource requirement spectrum. Additionally, when the only way to acquire high-fidelity data is through experiments, data deficiency will become a pressing challenge for deep learning-based methods. While there is a large collection of deep learning methods to address single-shot inverse design problems with large datasets, the setting of small data seems to be understudied by prior works, despite this setting's practicality under constrained budgets. RIGID fills this gap by offering a much more data-efficient and interpretable alternative to deep-learning-based single-shot inverse design methods.

4.3. Diversity Versus Feasibility of Generated Designs

The MCMC in RIGID can sample new designs in the entire design space based on the estimated likelihood, which is not limited to local optimality and has the potential to discover multiple very different satisfying solutions (if any) in one shot. This is clearly demonstrated in the SupSin example where there are multiple disconnected feasible regions in the design space and the likelihood estimated by RIGID successfully coincides with these feasible regions (Figure 7A). The ability to generate multiple satisfying solutions allows us to consider other figures of merit, such as manufacturability, compatibility, and cost, in addition to the main objective of satisfying the target response. Besides enabling flexibility of design choice, generating a comprehensive solution set can also bring more insights into the structure–property relation associated with the target. For example, after obtaining a diverse set of designs satisfying the same target, we can investigate the differences among these designs to identify critical features that lead to the satisfaction of the target.

For all the four problems demonstrated in this work, when the sampling threshold is set to 0.8, RIGID achieves a satisfaction rate of 100%. Although GA can also lead to a high target satisfaction rate for relatively simple inverse design problems (e.g., the optical metasurface design problem in Section 3.2), the generated solutions lack diversity. In contrast, RIGID provides a unique way to control the trade-off between the diversity and the feasibility of generated designs by tuning the sampling threshold. It is important to note that, without considering diversity, a high target satisfaction rate becomes trivial, as we can simply apply sufficiently small perturbations to one feasible solution to generate an infinite number of similar solutions that have similar responses and hence are also likely to meet the target. In many inverse design problems where solutions are nonunique, finding one solution can be simple, but the real challenge lies in discovering feasible solutions that cover a broad range of the design space—a problem where RIGID will outperform optimization-based methods as demonstrated by our results.

4.4. Discrete Design Variables

The use of a random forest in RIGID allows easy consideration of discrete design variables. For example, our optical metasurface

design problem has the geometry type and the material choice for each layer as categorical design variables. In contrast, these discrete variables will pose challenges for other inverse design methods based on some machine learning models (e.g., conditional generative models) or gradient-based optimization.

5. Summary and Outlook

We proposed RIGID, an iteration-free, single-shot inverse design method that generates metamaterials to satisfy qualitative behaviors of functional responses. Such qualitative behaviors are important design targets in many applications such as tailoring bandgaps of acoustic metamaterials for wave-guiding, focusing, and vibration mitigation or tailoring the absorption level of optical metasurfaces at certain wavelengths for medical sensing, imaging, and communication applications. Unlike most existing machine learning-based inverse design methods that require training an inverse model to map targets to designs, the RIGID method takes advantage of the random forest's interpretability and derives the likelihood of target satisfaction by probing the trained forward model. Incorporated with MCMC, one can sample a desired number of new designs based on the estimated likelihood. Therefore, RIGID functions as a generative model that can capture the conditional distribution of satisfying designs given a target or in other words, the one-to-many mapping from the target to satisfying designs. Using both real-world and synthetic design problems, we demonstrated that RIGID is efficient and effective on datasets with training sample sizes smaller than 250. We used both qualitative and quantitative approaches to validate the proposed method. The quantitative results revealed a strong correlation between the estimated likelihood of a solution and its probability of actually achieving the target, which demonstrated the effectiveness of the likelihood estimation. Due to the fast evaluation of output responses and the transparency of ground-truth solutions, the proposed synthetic problems can be used for future benchmarking studies of metamaterial design problems.

While we address qualitative design targets in this study, a similar idea has the potential to generalize to quantitative targets. Such problems can be, for example, generating optical metasurface designs with specific optical spectra,^[31,89] generating functional materials with target nonlinear constitutive relations,^[49,90] or generating programmable metamaterials with prescribed functional responses.^[91,92] It is also straightforward to adjust the target to achieve multifunctionality (e.g., negative/positive Poisson's ratio under low/high compression rate^[93]).

Although this study only demonstrates the RIGID method on parametric design (i.e., designs are represented by geometric and/or material parameters), the method also applies to shape or topological design problems where the shape or topology of designs can vary without being restricted to a limited number of geometric parameters.^[31,43,45,89,94] In those cases, as valid designs only lie on a lower-dimensional manifold of the design space, the likelihood of target satisfaction will be zero almost everywhere in the original design space. Thus before applying RIGID, we need to obtain a latent representation that compactly captures the manifold of valid designs^[95,96] and use the latent representation as design variables for inverse design.

6. Experimental Section

Configurations for GA: The initial population for GA was generated by randomly sampling solutions within the design variable bounds. At each generation, we selected individuals to reproduce for a new generation by randomly picking three individuals from the population, selecting the best one, and repeating this process 300 times. We set the probability for crossover and mutation to be 0.5 and 0.2, respectively. Once an individual mutated, there was a 5% chance for each design variable in this individual to mutate. For a continuous design variable, we defined the mutation operation as adding Gaussian noise with a mean of 0 and a standard deviation equal to 1% of the variable's range. For a categorical design variable, mutation is defined as randomly reassigning the category of the variable. We ran GA for ten generations, each with a population of 300. Finally, we selected a set of solutions with the highest objective values among all the individuals created during GA and used these selected solutions as generated designs.

Computation of Acoustic Dispersion Relations: We performed Bloch wave analysis in COMSOL Multiphysics to compute the dispersion relations of acoustic metamaterials. Poisson's ratio of 0.49, Young's modulus of 2.7 GPa, and density of 1170 kg m^{-3} were set as material properties with $\approx 1.5 \times 10^4$ mesh elements per unit cell. We used Floquet–Bloch periodic boundary conditions to obtain the first 30 eigenfrequencies along all symmetry domains of the cubic irreducible Brillouin zone (Figure 3B) for all lattices, thus generating a dispersion relation.

Computation of Optical Absorbance Spectra: We computed the absorbance spectra for optical metasurfaces using wave analysis inspired by Zhang et al.^[85] The RF Module of COMSOL Multiphysics^[97] was used to evaluate the spectral response of concern, which is the energy absorbance $A(s)$ in the visible regime (380–700 nm). An absorbance spectrum was computed with respect to 33 wavelength components s_k that were uniformly discretized over the specified range. An incident plane wave was assumed to be given from the port, located at the top face of the analysis domain. We set the periodicity of the analysis domain as 400 nm. The periodic boundary condition on electromagnetic fields was imposed on the lateral faces of the analysis domain. A substrate made of SiO_2 was placed right below a given unit cell instance (the black layers in Figure 5A,B). With full electric fields computed through the wave analysis, the energy absorbance at a single wavelength s was quantified as $A(s) = 1 - |S_{11}(s)|^2$, where S_{ij} is the component of the S-parameter matrix that specifies energy transfer between ports. We used the data presented in Ref. [98] to set the material dispersion of the dielectric.

Supporting Information

Supporting Information is available from the Wiley Online Library or from the author.

Acknowledgements

This work was supported by the startup funds from the J. Mike Walker '66 Department of Mechanical Engineering at Texas A&M University, the National Science Foundation (NSF) BRITE Fellow program (CMMI 2227641), the NSF CSSI program (OAC 1835782), the Kansas City National Security Campus (PDRD #705288), and NSF CAREER Award (CMMI-2142460). R.S. acknowledges financial support from the NSF Graduate Research Fellowship Program. This material is based on work supported by the National Science Foundation Graduate Research Fellowship Program under grant no. 2141064. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Wei (Wayne) Chen: Conceptualization: (lead); Data curation: (supporting); Formal analysis: (lead); Funding acquisition: (equal); Investigation: (lead); Methodology: (lead); Project administration: (lead); Supervision: (lead); Validation: (lead); Visualization: (lead); Writing—original draft: (lead); Writing—review & editing: (lead). **Rachel Sun:** Data curation: (equal); Formal analysis: (supporting); Investigation: (supporting); Validation: (supporting); Visualization: (equal); Writing—original draft: (equal); Writing—review & editing: (equal). **Doksoo Lee:** Conceptualization: (supporting); Data curation: (equal); Formal analysis: (equal); Investigation: (supporting); Methodology: (supporting); Validation: (supporting); Visualization: (supporting); Writing—original draft: (equal); Writing—review & editing: (equal). **Carlos M. Portela:** Funding acquisition: (equal); Project administration: (equal); Supervision: (equal); Writing—review & editing: (equal). **Wei Chen:** Funding acquisition: (equal); Project administration: (equal); Supervision: (equal); Writing—review & editing: (equal).

Data Availability Statement

The data that support the findings of this study are openly available in GitHub at <https://github.com/DIGITLab23/RIGID>, Doi: <http://dx.doi.org/10.5281/zenodo.14014653>.

Keywords

functional responses, generative design, interpretable machine learning, inverse design, metamaterials

Received: July 19, 2024

Revised: September 29, 2024

Published online:

- [1] M. Kadic, T. Bückmann, R. Schittny, M. Wegener, *Rep. Prog. Phys.* **2013**, *76*, 126501.
- [2] K. Bertoldi, V. Vitelli, J. Christensen, M. Van Hecke, *Nat. Rev. Mater.* **2017**, *2*, 1.
- [3] C. M. Soukoulis, M. Wegener, *Nat. Photonics* **2011**, *5*, 523.
- [4] S. A. Cummer, J. Christensen, A. Alù, *Nat. Rev. Mater.* **2016**, *1*, 1.
- [5] R. Schittny, M. Kadic, S. Guenneau, M. Wegener, *Phys. Rev. Lett.* **2013**, *110*, 195901.
- [6] J. Christensen, M. Kadic, O. Kraft, M. Wegener, *MRS Commun.* **2015**, *5*, 453.
- [7] J. B. Pendry, *Phys. Rev. Lett.* **2000**, *85*, 3966.
- [8] N. Gao, Z. Zhang, J. Deng, X. Guo, B. Cheng, H. Hou, *Adv. Mater. Technol.* **2022**, *7*, 2100698.
- [9] S. Jiang, X. Liu, J. Liu, D. Ye, Y. Duan, K. Li, Z. Yin, Y. Huang, *Adv. Mater.* **2022**, *34*, 2200070.
- [10] J. W. Boley, W. M. Van Rees, C. Lissandrello, M. N. Horenstein, R. L. Truby, A. Kotikian, J. A. Lewis, L. Mahadevan, *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 20856.
- [11] C. Ma, Y. Chang, S. Wu, R. R. Zhao, *ACS Appl. Mater. Interfaces* **2022**, *14*, 33892.
- [12] F. Casadei, T. Delpero, A. Bergamini, P. Ermanni, M. Ruzzene, *J. Appl. Phys.* **2012**, *112*, 064902.
- [13] G.-S. Liu, Y. Zhou, M.-H. Liu, Y. Yuan, X.-Y. Zou, J.-C. Cheng, *Sci. Rep.* **2020**, *10*, 981.
- [14] G. Kim, C. M. Portela, P. Celli, A. Palermo, C. Daraio, *Extreme Mech. Lett.* **2021**, *49*, 101499.
- [15] Y. Xie, Y. Fu, Z. Jia, J. Li, C. Shen, Y. Xu, H. Chen, S. A. Cummer, *Sci. Rep.* **2018**, *8*, 16188.

[16] K. H. Matlack, A. Bauhofer, S. Krödel, A. Palermo, C. Daraio, *Proc. Natl. Acad. Sci. USA* **2016**, *113*, 8386.

[17] S. Krödel, N. Thomé, C. Daraio, *Extreme Mech. Lett.* **2015**, *4*, 111.

[18] A. Bayat, S. Gaitanaros, *J. Appl. Mech.* **2018**, *85*, 011004.

[19] H. Ronellenfitsch, N. Stoop, J. Yu, A. Forrow, J. Dunkel, *Phys. Rev. Mater.* **2019**, *3*, 095201.

[20] H. Goh, L. F. Kallivokas, *Comput. Methods Appl. Mech. Eng.* **2020**, *370*, 113263.

[21] X. Li, S. Ning, Z. Liu, Z. Yan, C. Luo, Z. Zhuang, *Comput. Methods Appl. Mech. Eng.* **2020**, *361*, 112737.

[22] X. Zhang, J. Xing, P. Liu, Y. Luo, Z. Kang, *Extreme Mech. Lett.* **2021**, *42*, 101126.

[23] Z. Chen, A. Ogren, C. Daraio, L. C. Brinson, C. Rudin, *Extreme Mech. Lett.* **2022**, *57*, 101895.

[24] R. Vismara, N. O. Länk, R. Verre, M. Käll, O. Isabella, M. Zeman, *Opt. Express* **2019**, *27*, A967.

[25] Y. Cheng, F. Chen, H. Luo, *Nanoscale Res. Lett.* **2021**, *16*, 1.

[26] Z. Liu, D. Zhu, K.-T. Lee, A. S. Kim, L. Raju, W. Cai, *Adv. Mater.* **2020**, *32*, 1904790.

[27] D. Lee, L. Zhang, Y. Yu, W. Chen, *Adv. Opt. Mater.* **2024**, *12*, 2303087.

[28] S. Inampudi, H. Mosallaei, *Appl. Phys. Lett.* **2018**, *112*, 24.

[29] M. V. Zhelyeznyakov, S. Brunton, A. Majumdar, *ACS Photonics* **2021**, *8*, 481.

[30] D. Liu, Y. Tan, E. Khoram, Z. Yu, *Acs Photonics* **2018**, *5*, 1365.

[31] W. Ma, F. Cheng, Y. Xu, Q. Wen, Y. Liu, *Adv. Mater.* **2019**, *31*, 1901111.

[32] S. An, B. Zheng, H. Tang, M. Y. Shalaginov, L. Zhou, H. Li, M. Kang, K. A. Richardson, T. Gu, J. Hu, C. Fowler, H. Zhang, *Adv. Opt. Mater.* **2021**, *9*, 1.

[33] I. Malkiel, M. Mrejen, A. Nagler, U. Arieli, L. Wolf, H. Suchowski, *Light: Sci. Appl.* **2018**, *7*, 1.

[34] D. Lee, W. Chen, L. Wang, Y.-C. Chan, W. Chen, *Adv. Mater.* **2024**, *36*, 2305254.

[35] S. An, C. Fowler, B. Zheng, M. Y. Shalaginov, H. Tang, H. Li, L. Zhou, J. Ding, A. M. Agarwal, C. Rivero-Baleine, K. A. Richardson, *ACS Photonics* **2019**, *6*, 3196.

[36] S. Kumar, S. Tan, L. Zheng, D. M. Kochmann, *NPJ Comput. Mater.* **2020**, *6*, 1.

[37] C. Yeung, J. M. Tsai, B. King, B. Pham, D. Ho, J. Liang, M. W. Knight, A. P. Raman, *Nanophotonics* **2021**, *10*, 1133.

[38] J.-H. Bastek, S. Kumar, B. Telgen, R. N. Glaesener, D. M. Kochmann, *Proc. Natl. Acad. Sci. USA* **2022**, *119*, e2111505119.

[39] C. Xie, H. Li, C. Cui, H. Lei, Y. Sun, C. Zhang, Y. Zhang, H. Dong, L. Zhang, *Appl. Phys. Lett.* **2023**, *123*, 18.

[40] K. Mahesh, S. K. Ranjith, R. Mini, *Eng. Comput.* **2024**, *40*, 279.

[41] C. S. Ha, D. Yao, Z. Xu, C. Liu, H. Liu, D. Elkins, M. Kile, V. Deshpande, Z. Kong, M. Bauchy, X. Zheng, *Nat. Commun.* **2023**, *14*, 5765.

[42] Z.-W. Wang, A. Chen, Z.-X. Xu, J. Yang, B. Liang, J.-C. Cheng, *Sci. China Phys. Mech. Astron.* **2023**, *66*, 224311.

[43] J. Jiang, D. Sell, S. Hoyer, J. Hickey, J. Yang, J. A. Fan, *ACS Nano* **2019**, *13*, 8872.

[44] S. So, J. Rho, *Nanophotonics* **2019**, *8*, 1255.

[45] F. Wen, J. Jiang, J. A. Fan, *ACS Photonics* **2020**, *7*, 2098.

[46] C. Gurbuz, F. Kronowetter, C. Dietz, M. Eser, J. Schmid, S. Marburg, *J. Acoust. Soc. Am.* **2021**, *149*, 1162.

[47] W. Ma, Y. Xu, B. Xiong, L. Deng, R.-W. Peng, M. Wang, Y. Liu, *Adv. Mater.* **2022**, *34*, 2110022.

[48] Z.-D. Lei, Y.-D. Xu, C. Lei, Y. Zhao, D. Wang, *Photonics Res.* **2024**, *12*, 123.

[49] A. J. Lew, M. J. Buehler, *Mater. Today* **2023**, *64*, 10.

[50] Z. Zhang, C. Yang, Y. Qin, H. Feng, J. Feng, H. Li, *Nanophotonics* **2023**, *12*, 3871.

[51] J.-H. Bastek, D. M. Kochmann, *Nat. Mach. Intell.* **2023**, *5*, 1466.

[52] S. So, J. Mun, J. Park, J. Rho, *Adv. Mater.* **2023**, *35*, 2206399.

[53] X. Zheng, X. Zhang, T.-T. Chen, I. Watanabe, *Adv. Mater.* **2023**, *35*, 2302530.

[54] Y. Jin, L. He, Z. Wen, B. Mortazavi, H. Guo, D. Torrent, B. Djafari-Rouhani, T. Rabczuk, X. Zhuang, Y. Li, *Nanophotonics* **2022**, *11*, 439.

[55] W. Ma, Z. Liu, Z. A. Kudyshev, A. Boltasseva, W. Cai, Y. Liu, *Nat. Photonics* **2021**, *15*, 77.

[56] J. Jiang, M. Chen, J. A. Fan, *Nat. Rev. Mater.* **2021**, *6*, 679.

[57] Y. Xu, B. Xiong, W. Ma, Y. Liu, *Prog. Quantum Electron.* **2023**, *89*, 100469.

[58] M. Elzouka, C. Yang, A. Albert, R. S. Prasher, S. D. Lubner, *Cell Rep. Phys. Sci.* **2020**, *1*, 12.

[59] L. Breiman, *Mach. Learn.* **2001**, *45*, 5.

[60] W. Hastings, *Biometrika* **1970**, *57*, 97.

[61] L. Breiman, J. Friedman, C. J. Stone, R. A. Olshen, in *Classification and Regression Trees*, Chapman and Hall/CRC, Boca Raton, FL **1984**.

[62] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, *J. Artif. Intell. Res.* **2002**, *16*, 321.

[63] Z. Liu, L. Raju, D. Zhu, W. Cai, *IEEE J. Emerg. Select. Top. Circuits Syst.* **2020**, *10*, 126.

[64] A. P. Garland, B. C. White, S. C. Jensen, B. L. Boyce, *Mater. Des.* **2021**, *203*, 109632.

[65] J. Zhang, G. Wang, T. Wang, F. Li, *ACS Appl. Mater. Interfaces* **2021**, *13*, 7792.

[66] S. C.-y. Shen, M. J. Buehler, *Commun. Eng.* **2022**, *1*, 37.

[67] S. Lee, Z. Zhang, G. X. Gu, *Mater. Horiz.* **2022**, *9*, 952.

[68] M. Mitchell, in *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, MA **1998**.

[69] N. Yu, P. Genet, M. A. Kats, F. Aieta, J.-P. Tetienne, F. Capasso, Z. Gaburro, *Science* **2011**, *334*, 333.

[70] A. V. Kildishev, A. Boltasseva, V. M. Shalaev, *Science* **2013**, *339*, 1232009.

[71] T. J. Cui, M. Q. Qi, X. Wan, J. Zhao, Q. Cheng, *Light: Sci. Appl.* **2014**, *3*, e218.

[72] S. S. Bukhari, J. Y. Vardaxoglou, W. Whittow, *Appl. Sci.* **2019**, *9*, 2727.

[73] J. Hu, S. Bandyopadhyay, Y.-h. Liu, L.-y. Shao, *Front. Phys.* **2021**, *8*, 586087.

[74] N. I. Landy, S. Sajuyigbe, J. J. Mock, D. R. Smith, W. J. Padilla, *Phys. Rev. Lett.* **2008**, *100*, 207402.

[75] X. Liu, T. Starr, A. F. Starr, W. J. Padilla, *Phys. Rev. Lett.* **2010**, *104*, 207403.

[76] J. Hao, J. Wang, X. Liu, W. J. Padilla, L. Zhou, M. Qiu, *Appl. Phys. Lett.* **2010**, *96*, 25.

[77] C. M. Watts, X. Liu, W. J. Padilla, *Adv. Mater.* **2012**, *24*, OP98.

[78] Y. Cui, Y. He, Y. Jin, F. Ding, L. Yang, Y. Ye, S. Zhong, Y. Lin, S. He, *Laser Photonics Rev.* **2014**, *8*, 495.

[79] X. Liu, K. Fan, I. V. Shadrivov, W. J. Padilla, *Opt. Express* **2017**, *25*, 191.

[80] G.-Y. Lee, G. Yoon, S.-Y. Lee, H. Yun, J. Cho, K. Lee, H. Kim, J. Rho, B. Lee, *Nanoscale* **2018**, *10*, 4237.

[81] Y. Zhou, I. I. Kravchenko, H. Wang, J. R. Nolen, G. Gu, J. Valentine, *Nano Lett.* **2018**, *18*, 7529.

[82] Y. Li, Q. Cao, Y. Wang, *IEEE Antennas Wireless Propag. Lett.* **2018**, *17*, 1314.

[83] G. Marino, D. Rocco, C. Gigli, G. , K. Pantzas, S. Suffit, P. Filloux, I. Sagnes, G. Leo, C. De Angelis, *Nanophotonics* **2021**, *10*, 1837.

[84] S. C. Malek, A. C. Overvig, A. Alù, N. Yu, *Light: Sci. Appl.* **2022**, *11*, 246.

[85] Y. Zhang, Y. Yi, W. Li, S. Liang, J. Ma, S. Cheng, W. Yang, Y. Yi, *Coatings* **2023**, *13*, 531.

[86] Y. Zhang, D. W. Apley, W. Chen, *Sci. Rep.* **2020**, *10*, 4924.

[87] C. Ma, S. Tschiatschek, R. Turner, J. M. Hernández-Lobato, C. Zhang, *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 11237.

[88] L. Xu, K. Veeramachaneni, *arXiv:1811.11264*, **2018**.

[89] Z. Liu, D. Zhu, S. P. Rodrigues, K.-T. Lee, W. Cai, *Nano Lett.* **2018**, *18*, 6570.

[90] N. N. Vlassis, W. Sun, *Comput. Methods Appl. Mech. Eng.* **2023**, *413*, 116126.

[91] W. Li, F. Wang, O. Sigmund, X. S. Zhang, *Proc. Natl. Acad. Sci. USA* **2022**, *119*, e2120563119.

[92] X. Lin, F. Pan, Y. Ma, Y. Wei, K. Yang, Z. Wu, J. Guan, B. Ding, B. Liu, J. Xiang, Y. Chen, *Proc. Natl. Acad. Sci. USA* **2023**, *120*, e2305380120.

[93] A. Bossart, D. M. Dykstra, J. Van der Laan, C. Coulais, *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2018610118.

[94] S. An, B. Zheng, H. Tang, M. Y. Shalaginov, L. Zhou, H. Li, M. Kang, K. A. Richardson, T. Gu, J. Hu, C. Fowler, H. Zhang, *Adv. Opt. Mater.* **2021**, *9*, 2001433.

[95] L. Wang, Y.-C. Chan, F. Ahmed, Z. Liu, P. Zhu, W. Chen, *Comput. Methods Appl. Mech. Eng.* **2020**, *372*, 113377.

[96] W. Chen, D. Lee, O. Balogun, W. Chen, *J. Mech. Des.* **2023**, *145*, 011703.

[97] COMSOL AB, *Comsol Multiphysics (version 5.6)*, <https://www.comsol.com>, **2020**, Stockholm, Sweden.

[98] D. E. Aspnes, A. Studna, *Phys. Rev. B* **1983**, *27*, 985.