An Indirect Design Representation for Topology Optimization Using Variational Autoencoder and Style Transfer

Tinghao Guo*, Danny J. Lohan*, James T. Allison†
University of Illinois at Urbana-Champaign, Urbana IL, 61801

Ruijin Cang ‡, Yi Ren§
Arizona State University, Tempe AZ, 85281

In this paper we propose an indirect low-dimension design representation to enhance topology optimization capabilities. Established topology optimization methods, such as the Solid Isotropic Material with Penalization (SIMP) method, can solve large-scale topology optimization problems efficiently, but only for certain problem formulation types (e.g., those that are amenable to efficient sensitivity calculations). The aim of the study presented in this paper is to overcome some of these challenges by taking a complementary approach: achieving efficient solution via targeted design representation dimension reduction, enabling the tractable solution of a wider range of problems (e.g., those where sensitivities are expensive or unavailable). A new data-driven design representation is proposed that uses an augmented Variational Autoencoder (VAE) to encode 2D topologies into a lower-dimensional latent space, and to decode samples from this space back into 2D topologies. Optimization is then performed in the latent space as opposed to the image space. Established topology optimization methods are used here as a tool to generate a data set for training by changing problem conditions systematically. The data is generated using problem formulations that are solvable by SIMP, and are related to (but distinct from) the desired design problem. We further introduce augmentations to the VAE formulation to reduce unrealistic scattering of small material clusters during topology generation, while ensuring diversity of the generated topologies. We compare computational expense for solving a heat conduction design problem (with respect to the latent design variables) using different optimization algorithms. The new non-dominated points obtained via the VAE representation were found and compared with the known attainable set, indicating that use of this new design representation can simultaneously improve computational efficiency and solution quality.

I. Introduction

Over the past few decades, topology optimization methods have been applied to a wide range of domains for the purpose of material layout optimization. Examples include the early seminal work based on the homogenization approach,[1] as well as various notable extensions for engineering problems with different governing equations and responses.[2] In the domain of heat transfer, topology optimization has been implemented for problems in conduction,[3,4] convection,[5,6] and conjugate heat transfer.[7] For a recent review of heat transfer topology optimization, readers are referred to Dbouk,[8] where numerous examples of successful topological optimization studies are presented.

*Graduate Student, Department of Industrial and Enterprise Systems Engineering, 104 S. Mathews Ave, Urbana, IL, 61801
†Assistant Professor, Department of Industrial and Enterprise Systems Engineering, 104 S. Mathews Ave, Urbana, IL 61801, AIAA Member
‡Graduate Student, Department of Mechanical Engineering, 501 E. Tyler Mall, Tempe AZ, 85281
§Assistant Professor, Department of Mechanical Engineering, 501 E. Tyler Mall, Tempe AZ, 85281

While topology optimization methods are becoming mature in many applications, some important limitations remain. Some notable shortcomings are reviewed in Sigmund et al.[2] and Deaton et.al.[9] For instance, solution performance typically depends on the choice of the starting material distribution, the spatial discretization, and the filtering mechanism.[10,11,2] In addition, efficient solution using established methods is often limited to specific problem types (e.g., objective functions that lend well to efficient sensitivity calculations). These and other limitations motivate additional complementary strategies for homogeneous system topology optimization.

Solid Isotropic Material with Penalization (SIMP), an established density-based topology optimization strategy, capitalizes on problem structure and elegant sensitivity calculations to support efficient solution of large-scale problems. An alternative strategy to achieve computational efficiency, while expanding the set of solvable problem types, is to use indirect design representations. Instead of optimizing material distribution directly, we optimize parameters that determine material distribution through a particular mapping. If the indirect representation has lower dimension than the original direct representation, reduced optimization problem dimension can help reduce solution time. A drawback of this strategy is that reduced-dimension representations may limit the set of designs that can be accessed, placing importance on identifying or creating mappings that provide targeted coverage of important design space regions.

Indirect design representations have been used recently for structural and thermal system optimization, using generative algorithms as a basis for design representation mapping. Khetan, Lohan, and Allison introduced a design abstraction for truss topology and geometry optimization using a cellular division algorithm based on map L-systems.[12] Lohan and Allison have proposed solutions to problems of scalability in design representation and localized design dependency.[13,10,14,15,12] Lohan et al. presented a generative design algorithm (GDA) as the design representation for generating dendritic topologies.[10] These early examples are promising, but all rely on creative identification or construction of an effective generative design representation. Reliance on expertise, intuition, and existing candidate representations limits generalizability. What happens in the case where established methods (such as SIMP) cannot solve the desired problem, and an effective indirect design representation cannot be readily identified? Here we propose a data-driven strategy where new indirect design representations are constructed based on design data, independent of intuition or existing representations.

Significant progress has been made toward extracting knowledge from design data. For example, Matthews et al.[16] developed a method for extracting and verifying design knowledge from design databases, including relationships between components. Fuge et al.[17] introduced a model to predict successful design approaches from features of design problems. Ren et al. discussed the potential to learn either a constrained design space,[18] or a more efficient optimization algorithm[19] from crowdsourced design solutions. These previous efforts, however, are largely based on historical design data, and do not address the problem of constructing effective indirect design representations. Here we focus on design of potentially unprecedented systems (i.e., no historical data available), and on how to use data-driven methods to construct effective indirect design representations.

Strategies based on historical data are descriptive in nature. For example, knowledge or heuristics derived from historical design data may reveal how expert designers reason through design decisions, but cannot provide new insights into alternative designs that may perform better. Historical data is fixed; we cannot design experiments to generate new data that could provide better insights. Here we propose a strategy where a systematically-specified set of design optimization problems is used to generate design data for further analysis. This frees us from the limitations connected to data derived from existing designs, allowing us to address unprecedented systems, and supports a process that is more normative in nature (i.e., how *should* the system be designed, not how has it been designed). This strategy also supports the use of designed experiments to generate design data, opening up more possibilities for analysis.

The demonstration example used here involves design of a heat conduction system in a fixed volume domain.The desired design problem is to maximize power density while observing maximum temperature constraints. A related problem is solved efficiently using SIMP (minimize thermal compliance, subject to conductive material volume constraints) with a range of different boundary conditions and volume constraints to produce a wide variety of conductive material topologies. These form a manifold in the topology space that is governed implicitly by the optimality conditions. When the number of parameters that control the boundary conditions is limited, one may expect the manifold to be low-dimensional in comparison with that of the topology space. An analytical characterization of the manifold will therefore be expected to reduce optimization solution expense by circumventing the scalability issue common to the calculation of design

sensitivities with respect to a meshed topology.

While a direct characterization is difficult, an approximation of the manifold can be derived based on a collection of optimal designs. This paper implements this concept using an augmented Variational Autoencoder (VAE), which *automatically* learns to generate topologies from a low-dimensional latent space. More specifically, this purely data-driven generative model attempts to find a two-way mapping from the sample topologies to a set of normally distributed points in the latent space. By constructing this mapping, new topologies can be generated by sampling the latent space. Training samples for the VAE model are collected by applying SIMP to a set of problems with a distribution of boundary conditions. We further introduce augmentations to the VAE formulation to prevent unrealistic scattering of small material clusters during topology generation, while ensuring diversity of the generated topologies. Improvement in solution efficiency is demonstrated using a multi-objective optimization problem for heat conduction. We were able to identify the non-dominated solutions on the Pareto frontier. It indicates the proposed design representation is capable of exploring the design space effectively via the latent space.

II. Related Work

II.A. Variational Autoencoder

A variational autoencoder (VAE) [20] is an extension of the standard autoencoder.[21] Both models are composed of an encoder $\mathbf{z} = f(\mathbf{x})$ that converts the input $\mathbf{x}$ to a latent vector $\mathbf{z}$, and a decoder $\hat{\mathbf{x}} = g(\mathbf{z})$ that produces a reconstruction $\hat{\mathbf{x}}$ from $\mathbf{z}$. A schematic of an autoencoder is shown in Fig. 1. The model is trained by minimizing the reconstruction loss between a set of inputs and the corresponding outputs. Variants of the autoencoder (e.g., denoising,[22] sparse ,[23] and contractive[24]) have been developed to learn concise representations from high-dimensional input data, and are adopted widely for data compression,[25] network pre-training,[26] and feature extraction.[27,28]

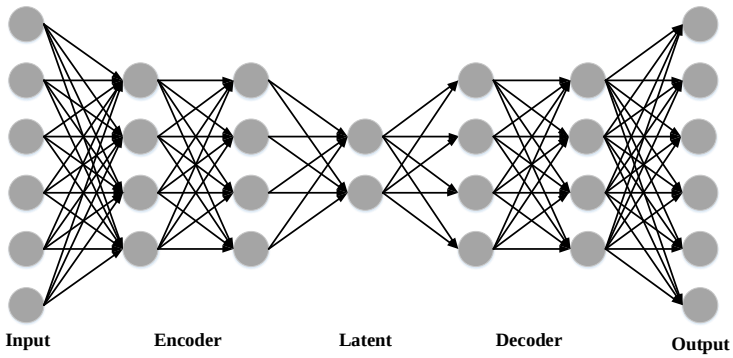

Figure 1. A schematics of an autoencoder

Autoencoders, however, do not ensure that the model distribution (the distribution of model outputs) is congruent with the input data distribution, and thus may create outputs with drastically different topologies from the input data. VAEs address this challenge by enforcing the distribution of the encoded inputs to match the sampling distribution in the latent space.

We present a brief derivation of the VAE loss below, which is a function to be minimized with respect to encoder parameters ϕ and decoder parameters θ. Let the dataset be $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$, which defines the data distribution $p_{\text{data}}(\mathbf{x})$. Let the decoder define the probability of an output, $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$, conditioned on the latent vector $\mathbf{z}$. Matching the model distribution with the data distribution is equivalent to maximizing the marginal likelihood $\Pi_i^N p(\mathbf{x}_i)$, where

$$p(\mathbf{x}) = \int_{\mathbf{z}} p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} = E_{\mathbf{z}\sim p(\mathbf{z})} p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}). \tag{1}$$

In general, the computation of Eq. (1) and its gradient with respect to network parameters is expensive, due to numerical integration and decoder network complexity. It is noted, however, that for a particular $\mathbf{x}$, most $\mathbf{z}$ will lead to $p_\theta(\mathbf{x}|\mathbf{z}) \approx 0$, and thus have limited contribution to $p(\mathbf{x})$. Therefore, it is reasonable to

sample only those $\mathbf{z}$ that are likely to produce $\mathbf{x}$, and use them to compute $p(\mathbf{x})$. To do so, a new function $q(\mathbf{z}|\mathbf{x})$ (the encoder) is introduced, which takes $\mathbf{x}$ and outputs a distribution of $\mathbf{z}$. Ideally, the space of $\mathbf{z}$ that are likely under q will be much smaller than that under the prior $p(z)$, so that the marginal $\mathbb{E}_{\mathbf{z}\sim q}\mathbf{p}_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$ becomes inexpensive to compute. Below we evaluate the difference between the approximation ($\mathbb{E}_{\mathbf{z}\sim q}\mathbf{p}_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$) and the target ($p(\mathbf{x})$): We start by deriving the Kullback-Leibler divergence between two distributions in the latent space: the encoder distribution $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ and the posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$:

$$D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{\mathbf{z}\sim q}\log q_{\boldsymbol{\phi}}(\mathbf{z}) - \log p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}). \tag{2}$$

By applying Bayes' rule to $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$ and noticing that $\log p(\mathbf{x})$ is independent from $\mathbf{z}$, we have:

$$\log p(\mathbf{x}) - D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{\mathbf{z}\sim q}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] - D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p(\mathbf{z})). \tag{3}$$

The left hand side has the quantity we want to maximize: $\log p(\mathbf{x})$, and the KL-divergence term $D_{\mathrm{KL}}(\cdot)$ that ideally reaches 0. The right hand side of the equation contains the reconstruction likelihood ($\mathbb{E}_{\mathbf{z}\sim q}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})]$) and the KL-divergence between the encoder distribution and the prior distribution for sampling the latent space ($D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))$). These RHS terms can be maximized via stochastic gradient descent. Specifically, we model $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}|\mu(\mathbf{x};\boldsymbol{\phi}), \Sigma(\mathbf{x};\boldsymbol{\phi}))$ as a normal distribution. The mean $\mu(\cdot,\boldsymbol{\phi})$ and the variance-covariance matrix $\Sigma(\cdot,\boldsymbol{\phi})$ comprise the encoder network. Similarly, we model the decoder outputs to follow $p_{\boldsymbol{\theta}(\mathbf{x}|\mathbf{z})} = \mathcal{N}(\mathbf{x}|f(\mathbf{z};\boldsymbol{\theta}), \sigma^2\mathbf{I})$ with mean $f(\mathbf{z};\boldsymbol{\theta})$ and variance σ^2. The function $f(\cdot;\boldsymbol{\theta})$ is thus the decoder network. σ determines the importance of the reconstruction of $\mathbf{x}$ during the training of a generative model, and is set to 1 in the proposed model.

To summarize, the training of a VAE maximizes:

$$L(\theta,\phi,\mathbf{x}) = -\mathbb{E}_{\mathbf{z}\sim q}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] + D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})), \tag{4}$$

which represents a lower bound of the marginal $p(\mathbf{x})$.

II.B. Style Transfer Convolutional Neural Network

Through a preliminary study, we observed that the outputs from the VAE contain topologies with small disconnected phases. Disconnected conductive material reduces power density, and generally does not aid heat extraction, so it is desirable to avoid these features. As a strategy to prevent implicitly the generation of topologies with disconnected conductive material, we introduce an augmentation to the VAE model to force the resulting random generations to follow the topological style from the samples. This approach of implicitly meeting design requirements via design representation is similar in spirit to the technique used by Khetan et al.[12] to guarantee generation of structurally stable trusses. The style transfer network was proposed originally to address the problem of texture transfer, i.e., to transfer image styles from source images to target contents. Synthesis of content and style historically has been a challenge in image processing,[29,30,31,32] but is now solved more easily since the recent development of deep convolutional neural networks (CNNs),[33,34] due to their ability to extract high-level semantic information from images. Gatys et al.[35] proposed a CNN-based style transfer network that can separate and recombine image contents and style information, and then generate new images with the target content and styles. To be more specific, given a source image $\mathbf{s}$ containing a prescribed style and a target image $\mathbf{t}$, the network recovers an image $\mathbf{x}$ with content similar to $\mathbf{t}$ and texture from $\mathbf{s}$. The recovered image $\mathbf{x}$ can be obtained by solving a nonlinear least-squares problem:

$$\min_{\mathbf{x}} \|F(\mathbf{x}) - F(\mathbf{t})\|^2 + \|C(\mathbf{x}) - C(\mathbf{s})\|^2, \tag{5}$$

where $\|C(\mathbf{x}) - C(\mathbf{s})\|^2$ is the content loss, and content features $C(\cdot)$ are represented by activations of deep hidden layers. Here $\|F(\mathbf{x}) - F(\mathbf{t})\|^2$ is the style loss, where style features $F(\cdot)$ are represented as the covariances among channel-wise hidden layer activations.

III. Proposed Method

In this section, we present the proposed method for solving heat conduction system topology design using the latent representation given by the VAE. This method involves two phases, as illustrated in Fig. 2. The first phase is to train a VAE using training topologies; the second phase is to perform multi-objective optimization with respect to the latent representation $\mathbf{z}$ learned through the VAE. Genetic, gradient-based, and hybrid algorithms are tested here for performing optimization in the latent space.

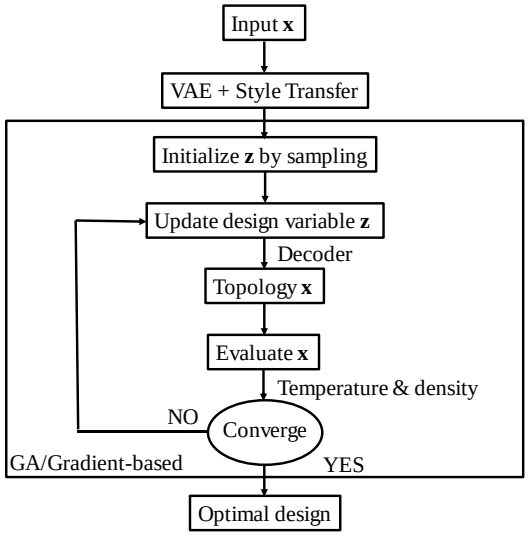

Figure 2. Flowchart of the optimization scheme with respect to z

III.A. Data Collection

To generate data for VAE training, a density-based topology optimization approach is utilized. The topology optimization problem solved is:

$$\begin{aligned} \min_{\mathbf{x}} \quad & C(\mathbf{x}) = \mathbf{U}^{\mathbf{T}}\mathbf{P} \\ \text{s.t} \quad & V(\mathbf{x}) \leq V_{\min} \\ & R(\mathbf{x}) \geq R_{\min} \\ & \mathbf{KU} = \mathbf{P} \\ & \mathbf{0} \leq \mathbf{x} \leq \mathbf{1}, \end{aligned} \tag{6}$$

where thermal compliance, $C(\mathbf{x})$, is minimized subject to a material volume constraint, minimum radius constraint, and Fourier's law for heat conduction. $\mathbf{U}$ and $\mathbf{P}$ represent the global displacement and force vectors, respectively; $\mathbf{K}$ is the global stiffness matrix.[36] The design variable, $\mathbf{x}$, represents material density. Each element of $\mathbf{x}$ specifies the material density of a corresponding voxel, and each element is bounded between 0 (void material properties) and 1 (solid material properties). To bias element densities towards a binary distribution, a power penalization (SIMP) is used. To enforce the minimum radius constraint, a density-based filter is implemented. Interested readers are referred to relevant references for more detail regarding topology optimization in heat transfer.[13,10,14] While 3D topology formulation was also discussed by Liu and Tovar,[37] this paper focuses on 2D thermal system topologies.

A set of problem parameters, including volume and radius bounds, as well as heat sink locations, are applied to Prob. (6). These problems are solved to obtain a set of topologies, each optimal with respect to the corresponding problem. The optimization process may produce infeasible solutions, but these were removed from the data set. Only the feasible and optimal designs were use for training the VAE. When using the SIMP method, each element of the spatial mesh has a corresponding design variable that varies continuously between 0 and 1. The penalty factor biases $\mathbf{x}$ values toward 0 or 1 during solution, but most values after convergence are close to 0 or 1 (not exactly binary). This necessitates a post-processing step where a filter threshold value is used to convert each element into a binary digit. The resulting data set was split into training and test sets. Validation was performed by using the test data set to check image reconstruction quality.

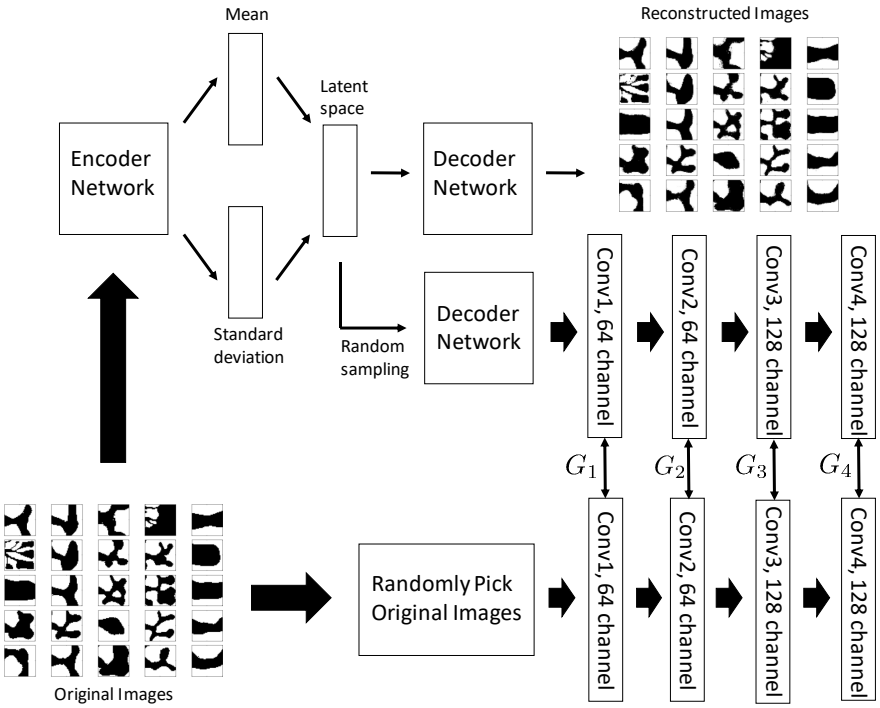

Figure 3. The VAE model augmented by a style loss

III.B. The VAE and Style Transfer Network

Here multilayer perceptron (MLP) networks are used to construct the encoder and decoder of a VAE. The encoder contains two fully-connected layers with hidden layer sizes 1,000 and 100, respectively. The decoder architecture is symmetric to the encoder. The output dimension of the encoder is set to 5, 10, 15, 20, and 25, for the purpose of testing how latent space dimensionality affects topology optimization results. This comparison will be presented in the next section. Figure 3 illustrates an overview of the proposed model.

During training, a batch of 40 random samples are generated via the latent space. The style transfer network has four layers, and the style loss is calculated with respect to the Gram matrices. The loss function for training contains the reconstruction, KL divergence, style, and mode-collapse losses: $L_{total} = L_{recon} + L_{KL} + L_{collapse} + L_{style}$. Reconstruction and KL-divergence losses are standard in VAE implementations;[20] the style loss measures how well the generated images match the training data with respect to image style;[35] the mode-collapse loss prevents the VAE model from only producing similar samples,[38] which is important for wide design space exploration. This loss is defined as:

$$f_{collaps}(S) = \frac{1}{N(N-1)} \sum_i \sum_{j \neq i} \left(\frac{S_i^{lT} S_j^l}{\|S_i^l\| \|S_j^l\|} \right)^2, \tag{7}$$

where S denotes a batch of samples taken from the lth style transfer network.

III.C. Heat Conduction Design Using Latent Variables

Here we discuss a topology design problem similar to Prob. (6), but with two key differences. First, instead of searching in the original topology space, we do so in the latent space derived from the VAE. Second, instead of compliance, we consider two objectives that are more closely aligned with underlying design intent: minimizing the maximum temperature $T_{\max}$, and maximizing the power density ρ. These objectives are conflicting: increasing power density typically increases maximum temperature.Thus, we formulate a multi-objective optimization problem where the objectives $T_{\max}(\mathbf{z})$ and $-\rho(\mathbf{z})$ are to be minimized with

respect to the latent variables $\mathbf{z}$.

$$\min_{\mathbf{z}\in\mathcal{Z}} \quad \{T_{\max}(\mathbf{z}), -\rho(\mathbf{z})\} \tag{8}$$

The problem can be solved with gradient-based or genetic algorithms. The initial guesses for gradient-based methods and the initial populations for genetic algorithms are drawn from the prior $p(\mathbf{z})$ defined by the VAE model.

The solution to a multi-objective optimization problem is a set of non-dominated (Pareto-optimal) solutions that quantify the tradeoff between objective functions. Four approaches for solving this multi-objective optimization problem are tested. The first, the weighted-sum method, is to sum the two objectives as shown in Prob. (9), with a weight w. We parametrically vary w from 0 to 1 to produce and solve a family of single-objective problems using a gradient-based method. This produces a set of solutions that belong to the Pareto set (non-dominated designs).

$$\min_{\mathbf{z}\in\mathcal{Z}} \quad w \cdot T_{\max}(\mathbf{z}) + (1-w)\cdot(-\rho(\mathbf{z})) \tag{9}$$

In the second solution strategy, we select $-\rho(\mathbf{z})$ to use as a single objective function, and convert the temperature objective to a constraint with bound T_{allow}, as shown in Prob. (10). The upper bound T_{allow} is varied parametrically to produce a family of optimization problems, the solution of which produces a set of non-dominated solutions. This approach is sometimes referred to as the ϵ-constraint method.[39]

$$\begin{aligned} \min_{\mathbf{z}\in\mathcal{Z}} \quad & -\rho(\mathbf{z}) \\ & T_{\max} \leq T_{\text{allow}} \end{aligned} \tag{10}$$

The third method is a hybrid implementation of the epsilon-constraint method. More specifically, for each value of T_{allow} considered, Prob. (10) is solved first using a genetic algorithm (GA) to locate an approximate solution that is likely to be near the global optimum. The GA result is then used as a starting point for a gradient-based method, which then hones in rapidly on a precise local solution. On their own, gradient-based methods produce locally-optimal solutions, which may underperform globally-optimal solutions. Hybrid strategies such as this improve the chances of, but do not guarantee, identification of global optima.

Finally, Problem 8 can also be solved directly using a multi-objective genetic algorithm (MOGA)[40] due to its population-based nature. While only one problem needs to be solved in this case instead of a set of problems, genetic algorithms tend to be computationally expensive. The Pareto-optimal solutions and frontier obtained using all four methods will be presented in the next section.

IV. Numerical Results

We prepared a total of 15,000 topologies for training the VAE model and another 2,870 for validation. Five VAE models are trained with latent dimensions 5D, 10D, 15D, 20D, and 25D. For the multi-objective optimization, hyper-parameters (e.g. latent space dimension, initial population, etc.) were varied parametrically to obtain comprehensive results.

IV.A. Reconstruction and generation of topologies

We first show that the VAE is capable of data reconstruction. Figure 4 compares the original and reconstructed data for the test data set. Here a latent space of dimension 20 is used, and a sample of 30 topologies is visualized. We observed some isolated and blurry parts in the reconstructed data, indicating that the loss design of the VAE still has room for improvement.

In addition to reconstructing known topologies, the VAE can be used to generate new ones. Figure 5 reveals some topologies created from random samples in the latent space. It is interesting to note that some of the generated topologies exhibit unexpected patterns. For instance, sample #45 has several separate chunks, whereas samples #73 and #76 resemble dendritic structures. For a particular generated topology, 2 out of 20 latent variables were selected and tuned within the range of [-3,3]. The results of this parametric study are illustrated in Fig. 6. The results show that significant, yet smooth, topology changes can be achieved by adjusting the VAE latent variables.

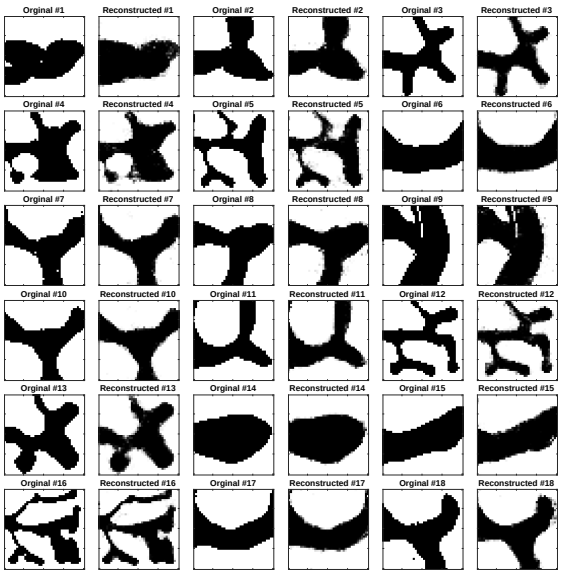

Figure 4. A sample of the reconstructed data

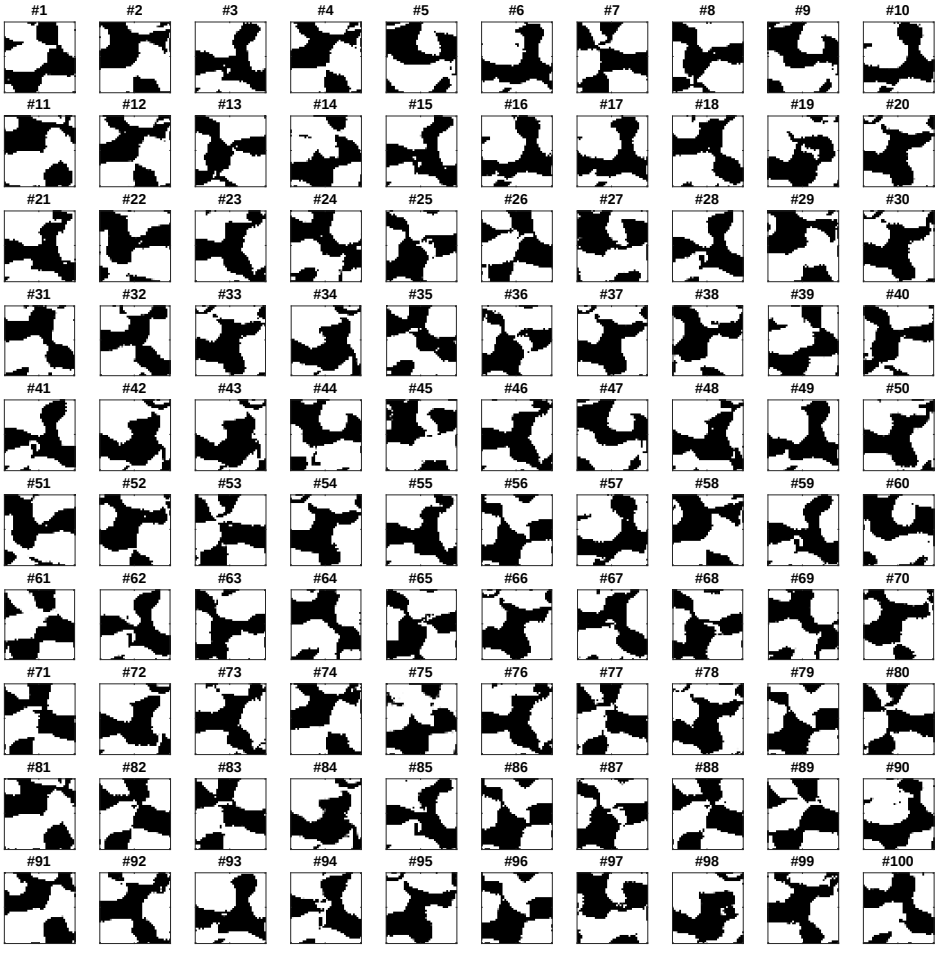

Figure 5. A sample of randomly generated topologies sampled from the latent space

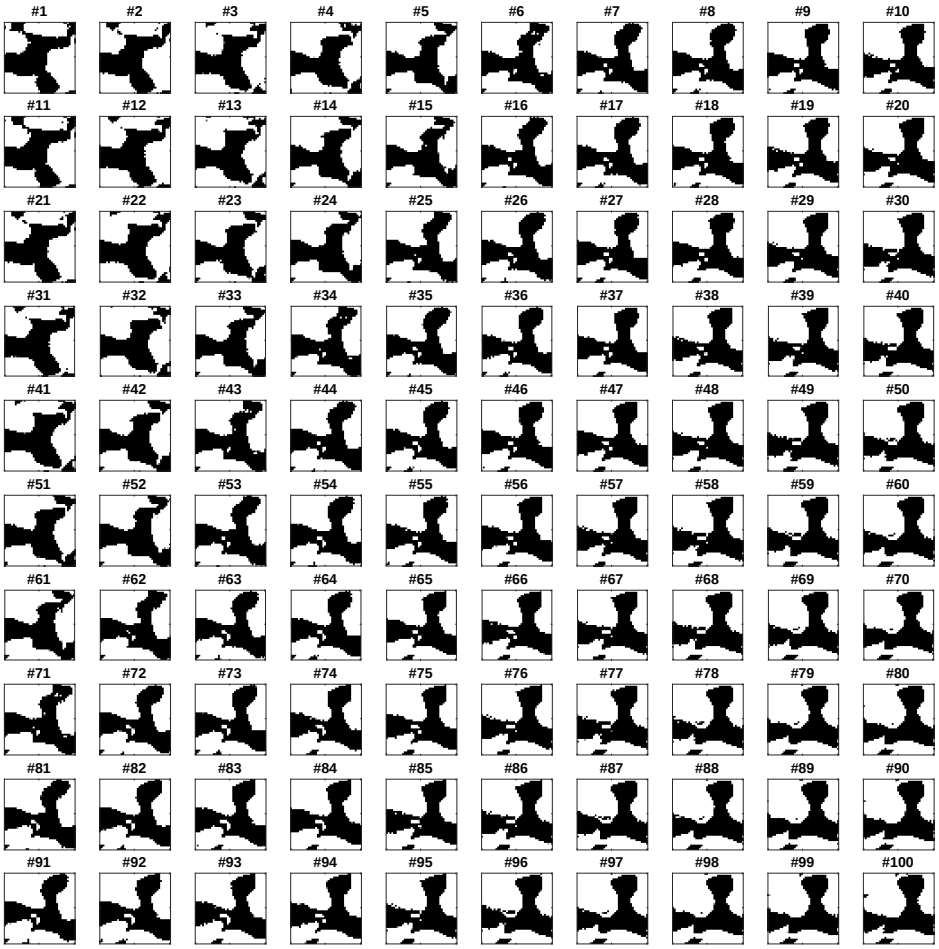

Figure 6. The parametric study for 2 out of 20 latent variables

IV.B. Pareto Frontiers

IV.B.1. The Weighted Sum Method

A multistart strategy was used to find the pareto-optimal solutions to Prob. 9. This improves the probability of finding the global optimum. Figure 7 shows the training data points along with the optimal solutions given by the weighted sum method. The vertical axis is the squared root of power density $\sqrt{\rho(z)}$, and the horizontal axis is the maximum temperature reached by the design. Many of the points overlap on the figure (i.e., more individual problems were solved than the number of non-training points appear in the figure). The legend indicates results from different latent space dimensions. The weighted-sum method did not perform well, as it did not identify any non-dominated points. A sample of the local optimal solutions (dominated solutions) can be found in Appendix Fig. 17.

IV.B.2. The ϵ Constraint Method

The results from applying the ϵ constraint method are shown in Fig. 8. It is clearly much more effective, as it can access not only non-dominated solutions from the training data, but also new non-dominated points. The majority of the optimal solutions are around the center of the training data cluster. A few points with red stars and black crosses are on the Pareto frontier. Among these local solutions, eight non-dominated designs are illustrated in Fig. 9.

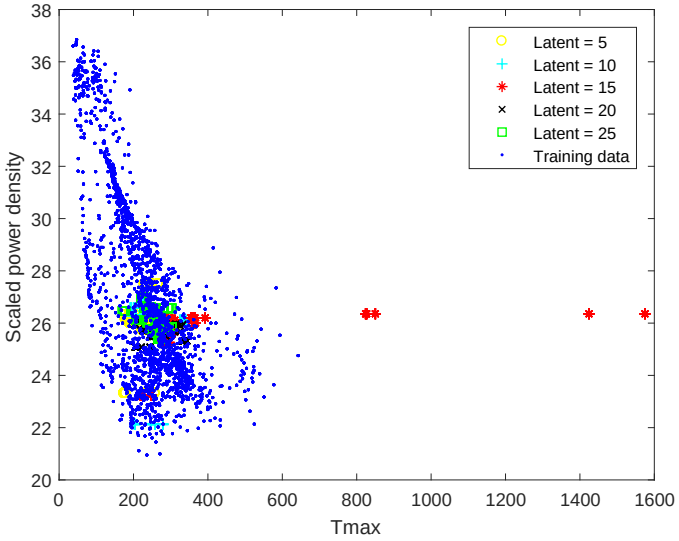

Figure 7. Local optimal solutions for the weighted sum method

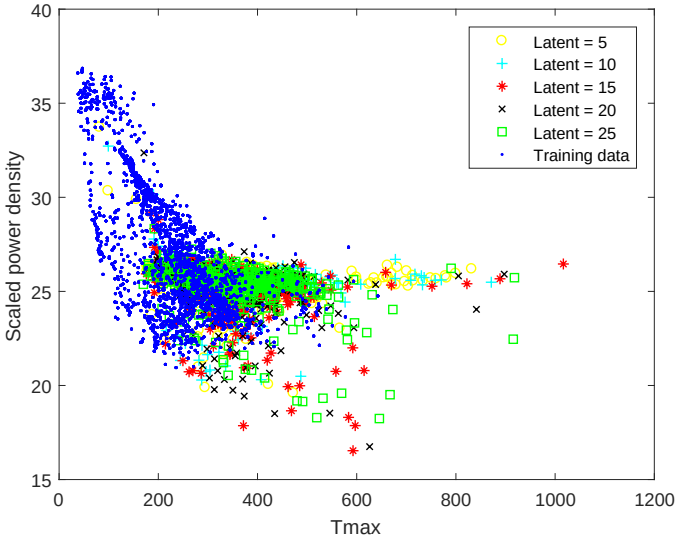

Figure 8. Local optimal solutions for the ϵ constraint method

IV.B.3. The Hybrid Method

The hybrid method uses a genetic algorithm (GA) with loose tolerances to find an approximate solution (ideally in the neighborhood of the global optimum) to use as a starting point for a gradient-based method. A initial population size of 200 was specified, and VAE models with different latent space dimensions were compared. Figure 10 displays the results of the hybrid method. Compared with the ϵ-constraint method, the hybrid method tends to identify more Pareto-optimal solutions; in particular, several optimal solutions with $T_{\max}$ greater than 600 (shown using yellow circles and black crosses) were identified. Six Pareto-optimal solutions are shown in Fig. 11. While these points lie on the Pareto frontier, not every solution is feasible or useful. For instance, solution #4 appears to be a reasonable design, but solution #3 has several isolated material chunks.

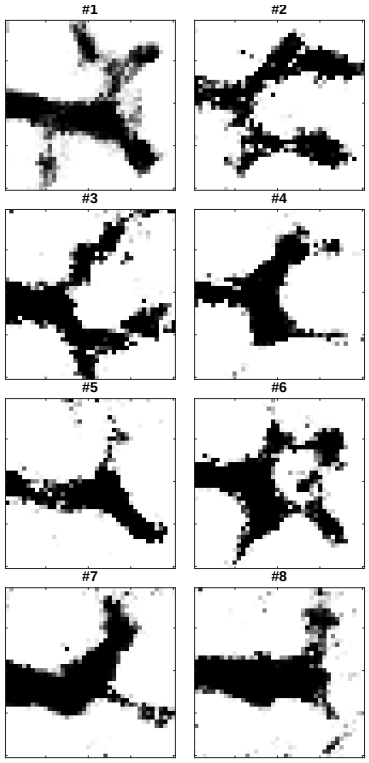

Figure 9. Pareto-optimal solutions for the ϵ constraint method

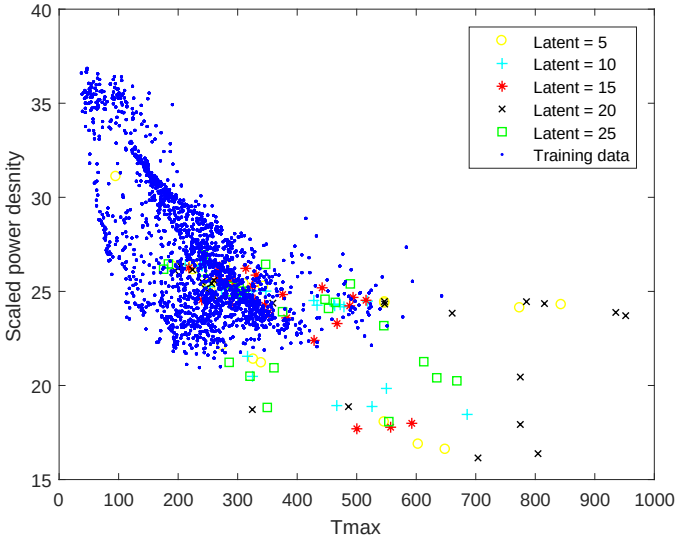

Figure 10. Optimal solutions given for the hybrid method

IV.B.4. Multi-Objective Genetic Algorithm

The multiobjective genetic algorithm (MOGA) supports direct solution of multiobjective problems. The implementation here uses an elitist strategy, a variant of NSGA-II.[40] The algorithm terminates if a metric, called *spread*, is less than a pre-defined value. Here the *spread* is a measure of the movement of solutions on the Pareto frontier between the two most recent optimization iterations.[41]

Two elements were explored using the MOGA method. Different latent variable dimensions were tested

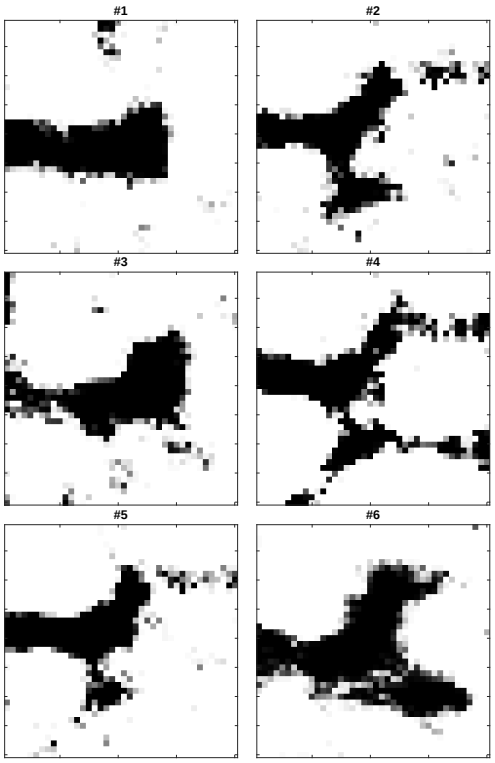

Figure 11. Pareto-optimal solutions for the hybrid method

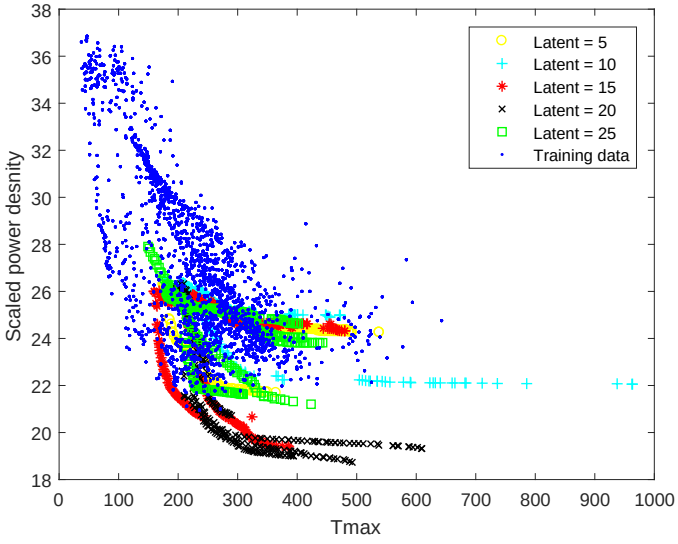

Figure 12. Optimal solutions for the MOGA method (initial population = 200)

using the same initial population size (200), reported in Fig. 12. A total of 96 Pareto optimal solutions are shown in Fig. 13. They appear to have are very similar to each other, but have some minor differences (such as material density). They appear to have three main solutions clusters: #1–#36, #37–#83, and #84–#96. The solution cluster #84–#96 has a number of isolated elements in the right corner of the topologies.

Using the fixed latent variable size 20, one can compare the system performance of MOGA with different initial populations. In Fig. 14, the solutions with initial population sizes 600, 800, and 1000 are shown. The latent design representation indeed covers the optimal design space with a total of 258 Pareto-optimal

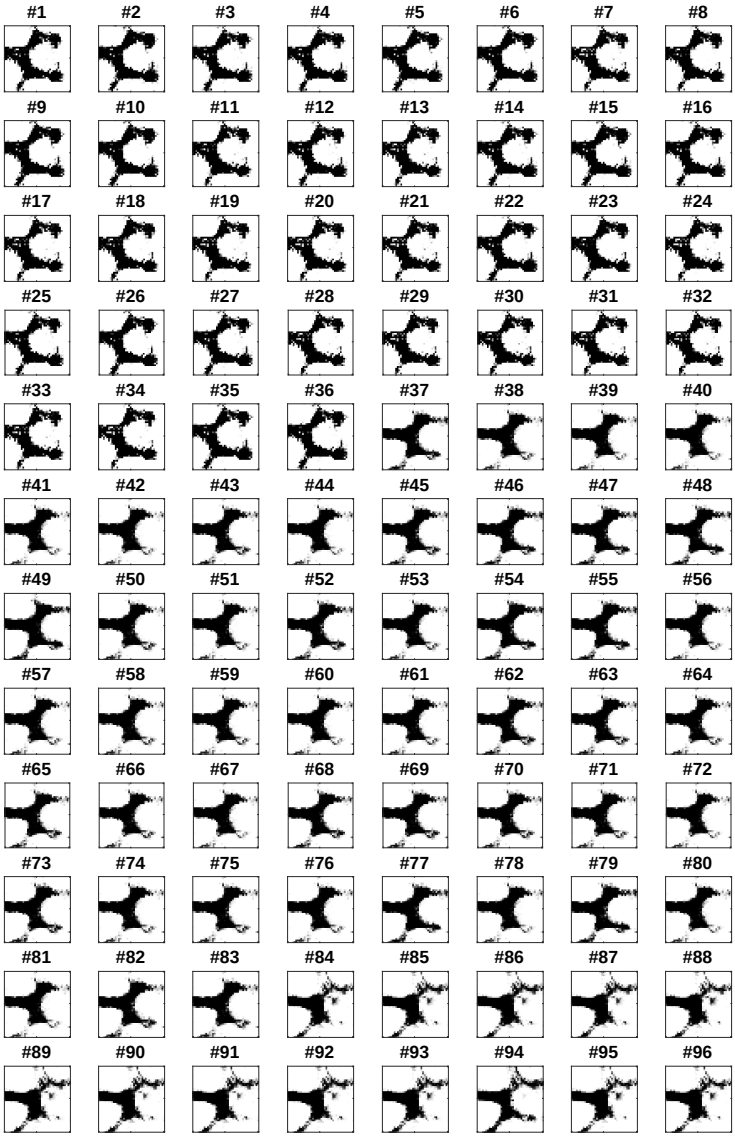

Figure 13. Select Pareto-optimal solutions given by the MOGA method (initial population = 200)

solutions. Four optimal solutions are reported here, representing the basic topology structure; the other optimal topologies are shown in the Appendix in Figs. 18–20. In Fig. 15, #5 and #118 were found by MOGA; neither of these solutions were identified by other methods. It is interesting to note that a loop is formed in these solution. Solution #43 is a very dense solution. The structure of Solution #44 is highly similar to those produced by the other three methods.

IV.C. Comparison among the Four Methods

Not all four methods were able to identify all known non-dominated points (points on the Pareto frontiers). Among them, MOGA seems to perform the best. We have included the Pareto frontier and the Pareto-optimal solutions for the MOGA method in the Appendix. Here we present a comparison of the four methods using a latent space dimension of 20 in Fig. 16. An important result is that many new non-dominated designs were found that go well beyond the attainable set estimated by the training data. In particular, many new non-dominated points were found using the indirect design representation (toward the lower-right portion of the plot). In other words, using the VAE indirect representation enabled identification of new solutions that for some design conditions are preferable to those that are reachable using conventional methods. In

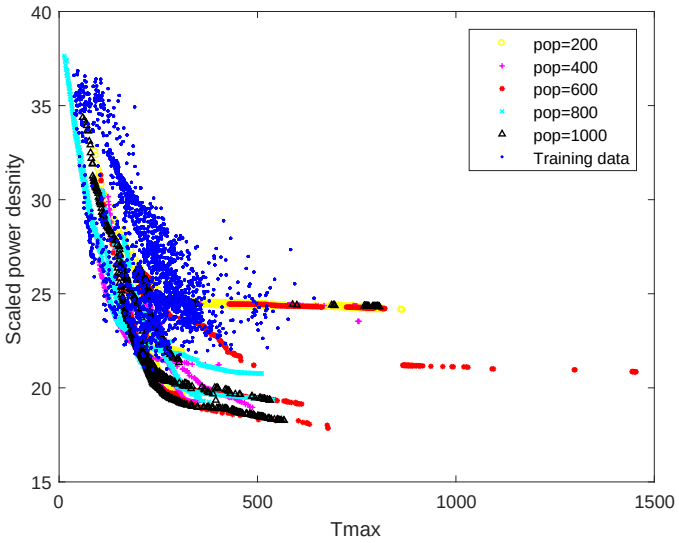

Figure 14. Optimal solutions for the MOGA method (latent = 20)

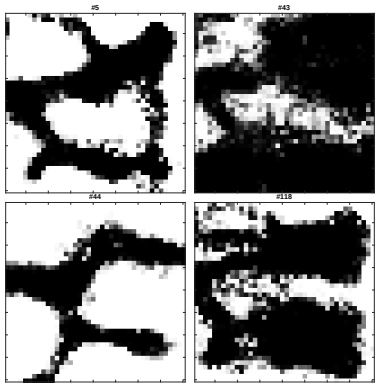

Figure 15. Four Pareto-optimal solutions for the MOGA method

addition, design problem dimension is reduced. Note that MOGA successfully identified the majority of the Pareto optimal solutions, while other methods achieved them partially. Note as well that some solutions with $T_{\max} > 400$ (black stars) were found by the hybrid method that were not identified using the MOGA.

The computational expense for the different methods is compared in Table IV.C. The *function count* was used as the metric here, and refers to the cumulative number of times the physics-based analysis was performed. These results are based on a latent space dimension of 20 and an initial population 200. The ϵ-constraint method required somewhat fewer function evaluations than the weighted sum method, but was successful in finding non-dominated solutions. Because the hybrid method utilizes both the gradient and heuristic algorithms, it is computationally more expensive than MOGA. However, it may be possible to balance GA and gradient-based algorithm parameters to reduce computational expense. MOGA may be worth the additional computational expense, as it is capable of identifying a large number of non-dominated designs, many of which intuitively appear to be physically reasonable (e.g., fewer disconnected chunks).

Table 1. Computational expense comparison

	Weighted sum	ϵ constraint	Hybrid	MOGA
F-count	4351	4303	75905	30601

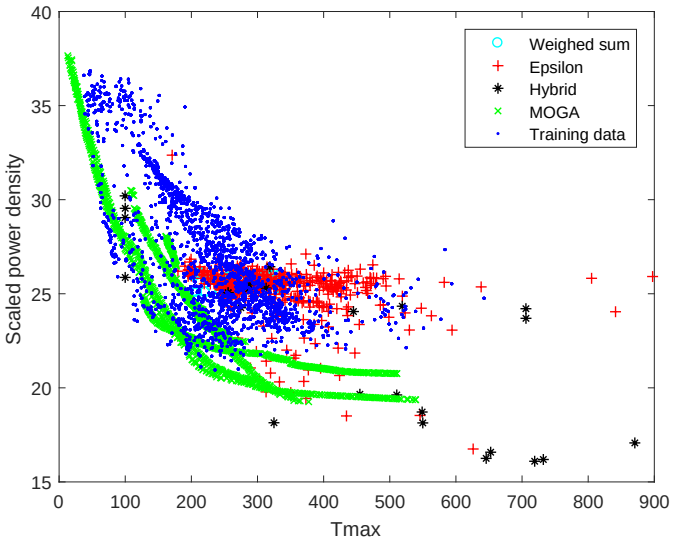

Figure 16. Comparison of Pareto optimal solutions for the four methods (latent = 20)

V. Discussion

In this study, deep convolutional neural networks were used both as the encoder and the decoder. For instance, a deep convolutional VAE with 16 layers and a style transfer network with 4 activation layers was first introduced. The filter size of the conventional layers for the encoder was 4×4, and the channel sizes were $32, 32, 64, 64, 96$, and 96 for the six layers, respectively. A max-pooling layer was inserted after each of the two convolution layers to reduce the number of hidden units. Two fully-connected layers were used to encode the input $\mathbf{x}$ into the latent space. The decoder architecture was symmetric to the encoder. While this deep convolutional VAE worked well in conjunction with the style transfer network in the training phase, it increased the computational expense of optimization during the second phase. The preliminary results indicated infeasibility in the solutions for certain formulations. As a result, the multilayer perceptions was finally chosen as the VAE. It is important to strike a balance between a simpler (less-expensive) and a more complicated (more accurate) models.

The proposed design representation may be limited in some circumstances. Each pixel of the optimal solution is real-valued rather than binary. Because the encoder maps a real-valued sample from the latent space to the original image space using the real-valued weights and biases, it is possible for the resulting output to be real-valued. A cut off for each pixel could be used as a post-processing step, but such a filter may not be well-suited for a gradient-based optimization algorithm in the second phase. This issue could be addressed in the future to help move design optimization results closer to practical topologies. Improved sampling techniques for obtaining training data may also increase method utility. Specifically, understanding how to plan systematic exploration of design problems for the generation of design data for unprecedented systems is an important question for this emerging approach. Designing our strategy for obtaining training data is an option not available when relying on historical data, and could be an important advantage in using design optimization to generate design data for use with machine learning.

The VAE model tends to produce blurry images, and not all details are preserved in the reconstructed or generated images due to the choice of the VAE loss function.[42] Improved models, such as sequential VAEs[43] and the modified VAE based on adversarial training,[42] have been proposed and may help generate more realistic images. Generative adversarial networks (GANs) could be another option in the first phase to generate sharper images.[44]

The data set was obtained by specifying several levels of boundary conditions, volume, and radius constraints (a full-factorial experiment was performed). Each training data point obtained using Eq. 6 is the result of a single SIMP solution. Because the volume constraint was varied, complete Pareto sets are contained in the training data shown in the figures above for the SIMP solutions for each boundary condition specified. An important result here is that some of the VAE results were successful in pushing down the

Pareto frontier in the objective function space, i.e., new non-dominated points were found. The gap between the (initially) non-dominated training points and the non-dominated VAE points is relatively large in many of the figures. The training data may correspond to local optima, so it may be possible to narrow the gap between training data results and VAE results.

VI. Conclusion

In this paper, an indirect design representation based on a VAE was proposed for topology optimization. The notion of designed experiments to generate data was first presented. The VAE combining a style transfer neural network was trained using the design data. A description of the multi-objective optimization problem for the heat conduction, including the variants of the formulation, was discussed. The Pareto frontiers were presented using the proposed design representation. Using MOGA with the VAE representation was particularly effective at identifying a large majority of non-dominated solutions (when combining all design results together). Along the Pareto frontier, several types of solutions were found (see the Appendix for details). The VAE representation enables identification of new designs, including new non-dominated designs, that are fundamentally different from the training data designs. Optimizing in the latent space was demonstrated to be successful in terms of: 1) accessing new designs, 2) improving solution quality (new non-dominated designs), and 3) using training data from a related (but easy-to-solve problem) to construct a design representation that enables efficient solution of the desired problem.

This is a new perspective on how machine learning strategies may be utilized effectively in engineering design. The proposed method is data-driven, and that has the potential to make practical the solution of topology optimization problems that cannot be solved efficiently using established density-based methods. This could become an important new strategy for normative data-driven design methods, especially for design of unprecedented systems without historical data.

References

[1]Bendsøe, M. P., "Optimal Shape Design as a Material Distribution Problem," *Structural and Multidisciplinary Optimization*, Vol. 1, No. 4, 1989, pp. 193–202.

[2]Sigmund, O. and Maute, K., "Topology Optimization Approaches," *Structural and Multidisciplinary Optimization*, Vol. 48, No. 6, 2013, pp. 1031–1055.

[3]Gersborg-Hansen, A., Bendsoe, M. P., and Sigmund, O., "{Topology Optimization of Heat Conduction Problems Using the Finite Volume Method}," *Structural and Multidisciplinary Optimization*, Vol. 31, mar 2006, pp. 251–259.

[4]Zhuang, C. and Xiong, Z., "Temperature-Constrained Topology Optimization of Transient Heat Conduction Problems," *Numerical Heat Transfer, Part B: Fundamentals*, Vol. 68, No. 4, 2015, pp. 366–385.

[5]Iga, A., Nishiwaki, S., Izui, K., and Yoshimura, M., "Topology optimization for thermal conductors considering design-dependent effects, including heat conduction and convection," *International Journal of Heat and Mass Transfer*, Vol. 52, No. 11-12, 2009, pp. 2721–2732.

[6]"Topology optimization of convection-dominated, steady-state heat transfer problems," *International Journal of Heat and Mass Transfer*, Vol. 50, No. 15-16, 2007, pp. 2859–2873.

[7]Alexandersen, J., Sigmund, O., and Aage, N., "Large scale three-dimensional topology optimisation of heat sinks cooled by natural convection," *International Journal of Heat and Mass Transfer*, Vol. 100, 2016, pp. 876–891.

[8]Dbouk, T., "A review about the engineering design of optimal heat transfer systems using topology optimization," *Applied Thermal Engineering*, Vol. 112, 2017, pp. 841–854.

[9]Deaton, J. D. and Grandhi, R. V., "A Survey of Structural and Multidisciplinary Continuum Topology Optimization: Post 2000," *Structural and Multidisciplinary Optimization*, Vol. 49, No. 1, 2014, pp. 1–38.

[10]Lohan, D. J., Dede, E. M., and Allison, J. T., "Topology Optimization for Heat Conduction Using Generative Design Algorithms," *Structural and Multidisciplinary Optimization*, Vol. 55, No. 3, March 2017, pp. 1063–1077.

[11]Svanberg, K. and Svärd, H., "Density Filters for Topology Optimization Based on the Pythagorean Means," *Structural and Multidisciplinary Optimization*, Vol. 48, No. 5, 2013, pp. 859–875.

[12]Khetan, A., Lohan, D. J., and Allison, J. T., "Managing Variable-Dimension Structural Optimization Problems Using Generative Algorithms," *Structural and Multidisciplinary Optimization*, Vol. 50, No. 4, Oct. 2015, pp. 695–715.

[13]Lohan, D. J., Dede, E., and Allison, J. T., "Topology Optimization Formulations for Circuit Board Heat Spreader Design," *Proceedings of 17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2016, p. 3669.

[14]Lohan, D. J. and Allison, J. T., "Temperature Constraint Formulations for Heat Conduction Topology Optimization," *ISSMO 12th World Congress of Structural and Multidisciplinary Optimisation, to appear*, Braunschweig, Germany, June 2017.

[15]Khetan, A. and Allison, J. T., "Large-Scale Topology Optimization Using Parametrized Boolean Networks," *ASME 2014 International Design Engineering Technical Conferences*, No. DETC2014-34256, Buffalo, NY, USA, Aug. 2014, p. V02AT03A006.

[16]Matthews, P. C., Blessing, L. T., and Wallace, K. M., "The introduction of a design heuristics extraction method," *Advanced Engineering Informatics*, Vol. 16, No. 1, 2002, pp. 3–19.

[17]Fuge, M., Peters, B., and Agogino, A., "Machine Learning Algorithms for Recommending Design Methods," *Journal of Mechanical Design*, Vol. 136, No. 10, 2014, pp. 101103.

[18]Ren, Y., Bayrak, A. E., and Papalambros, P. Y., "EcoRacer: Game-Based Optimal Electric Vehicle Design and Driver Control Using Human Players," *Journal of Mechanical Design*, Vol. 138, No. 6, 2016, pp. 061407.

[19]Sexton, T. and Ren, M. Y., "Learning an Optimization Algorithm through Human Design Iterations," *arXiv preprint arXiv:1608.06984*, 2016.

[20]Kingma, D. P. and Welling, M., "Auto-encoding Variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[21]Bengio, Y. et al., "Learning Deep Architectures for AI," *Foundations and Trends® in Machine Learning*, Vol. 2, No. 1, 2009, pp. 1–127.

[22]Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P.-A., "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, Vol. 11, No. Dec, 2010, pp. 3371–3408.

[23]Makhzani, A. and Frey, B., "K-sparse Autoencoders," *arXiv preprint arXiv:1312.5663*, 2013.

[24]Rifai, S., Vincent, P., Muller, X., Glorot, X., and Bengio, Y., "Contractive auto-encoders: Explicit invariance during feature extraction," *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 833–840.

[25]Le, Q. V. et al., "A Tutorial on Deep Learning Part 2: Autoencoders, Convolutional Neural Networks and Recurrent Neural Networks," 2015.

[26]Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H., "Greedy Layer-wise Training of Deep Networks," *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06, MIT Press, Cambridge, MA, USA, 2006, pp. 153–160.

[27]Xing, C., Ma, L., and Yang, X., "Stacked Denoise Autoencoder Based Feature Extraction and Classification for Hyperspectral Images," *Journal of Sensors*, Vol. 2016, 2015.

[28]Rifai, S., Vincent, P., Muller, X., Glorot, X., and Bengio, Y., "Contractive auto-encoders: Explicit invariance during feature extraction," *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 833–840.

[29]Efros, A. A. and Leung, T. K., "Texture Synthesis by Non-parametric Sampling," *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, Vol. 2, IEEE, 1999, pp. 1033–1038.

[30]Efros, A. A. and Freeman, W. T., "Image Quilting for Texture Synthesis and Transfer," *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM, 2001, pp. 341–346.

[31]Kwatra, V., Schödl, A., Essa, I., Turk, G., and Bobick, A., "Graphcut Textures: Image and Video Synthesis Using Graph Cuts," *ACM Transactions on Graphics (ToG)*, Vol. 22, ACM, 2003, pp. 277–286.

[32]Wei, L.-Y. and Levoy, M., "Fast Texture Synthesis Using Tree-structured Vector Quantization," *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 2000, pp. 479–488.

[33]Krizhevsky, A., Sutskever, I., and Hinton, G. E., "Imagenet Classification with Deep Convolutional Neural networks," *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[34]Simonyan, K. and Zisserman, A., "Very Deep Convolutional Networks for Large-scale Image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[35]Gatys, L. A., Ecker, A. S., and Bethge, M., "A Neural Algorithm of Artistic Style," *arXiv preprint arXiv:1508.06576*, 2015.

[36]Sigmund, O., "A 99 Line Topology Optimization Code Written in Matlab," *Structural and Multidisciplinary Optimization*, Vol. 21, No. 2, 2001, pp. 120–127.

[37]Liu, K. and Tovar, A., "An Efficient 3D Topology Optimization Code Written in Matlab," *Structural and Multidisciplinary Optimization*, Vol. 50, No. 6, 2014, pp. 1175–1196.

[38]Zhao, J., Mathieu, M., and LeCun, Y., "Energy-based Generative Adversarial Network," *arXiv preprint arXiv:1609.03126*, 2016.

[39]Miettinen, K., *Nonlinear Multiobjective Optimization*, Vol. 12, Springer Science & Business Media, 2012.

[40]Deb, K., *Multi-objective Optimization using Evolutionary Algorithms*, Vol. 16, John Wiley & Sons, 2001.

[41]MATLAB, "Find Pareto Front of Multiple Fitness Functions Using Genetic Algorithm," 2017, https://www.mathworks.com/help/gads/gamultiobj.html.

[42]Dosovitskiy, A. and Brox, T., "Generating Images with Perceptual Similarity Metrics based on Deep Networks," *Advances in Neural Information Processing Systems 29*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Curran Associates, Inc., 2016, pp. 658–666.

[43]Zhao, S., Song, J., and Ermon, S., "Towards Deeper Understanding of Variational Autoencoding Models," *arXiv preprint arXiv:1702.08658*, 2017.

[44]Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y., "Generative Adversarial Nets," *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

A. Appendix Figures

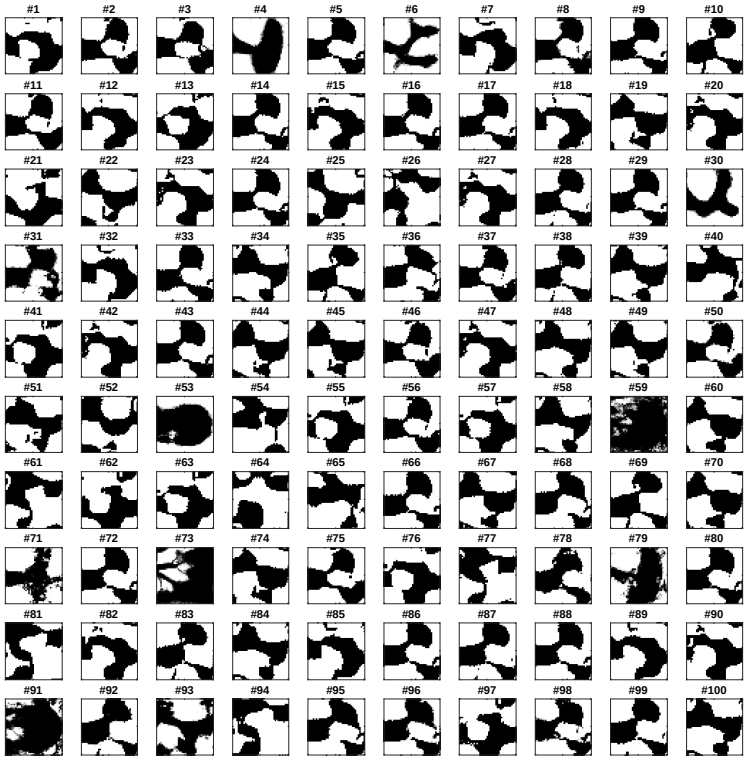

Figure 17. A sample of dominated points for the weighted sum method

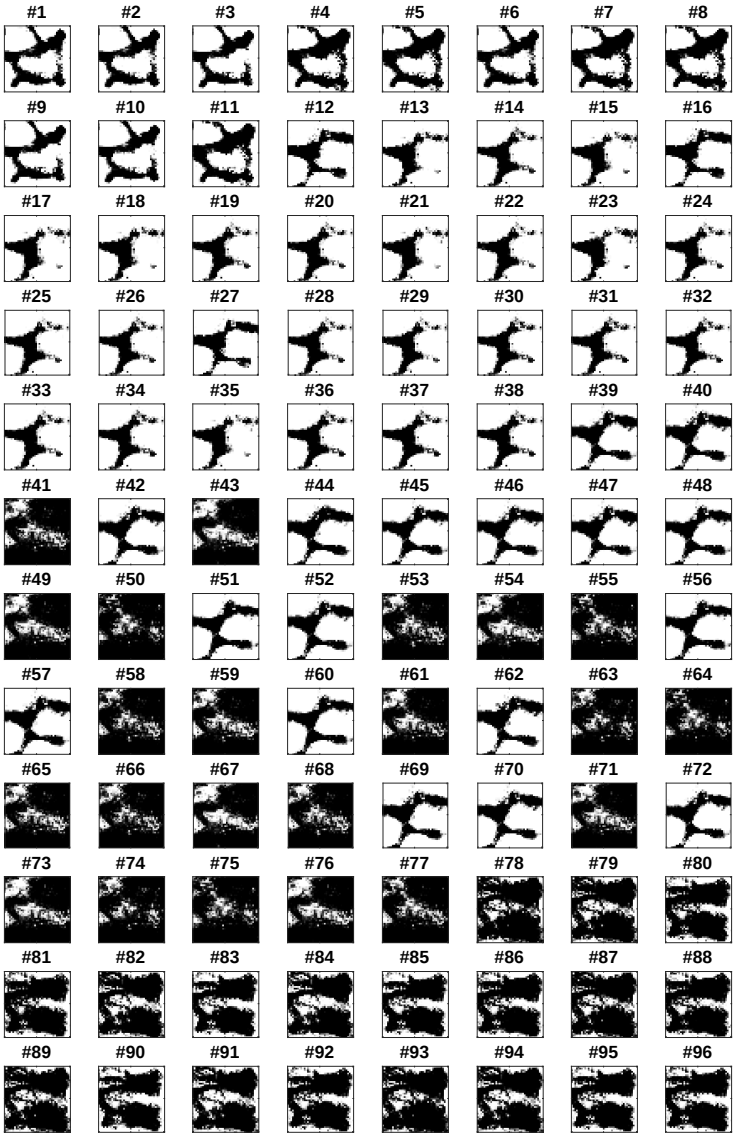

Figure 18. Pareto-optimal solutions (#1 $\sim$ #100) for the MOGA method (latent = 20)

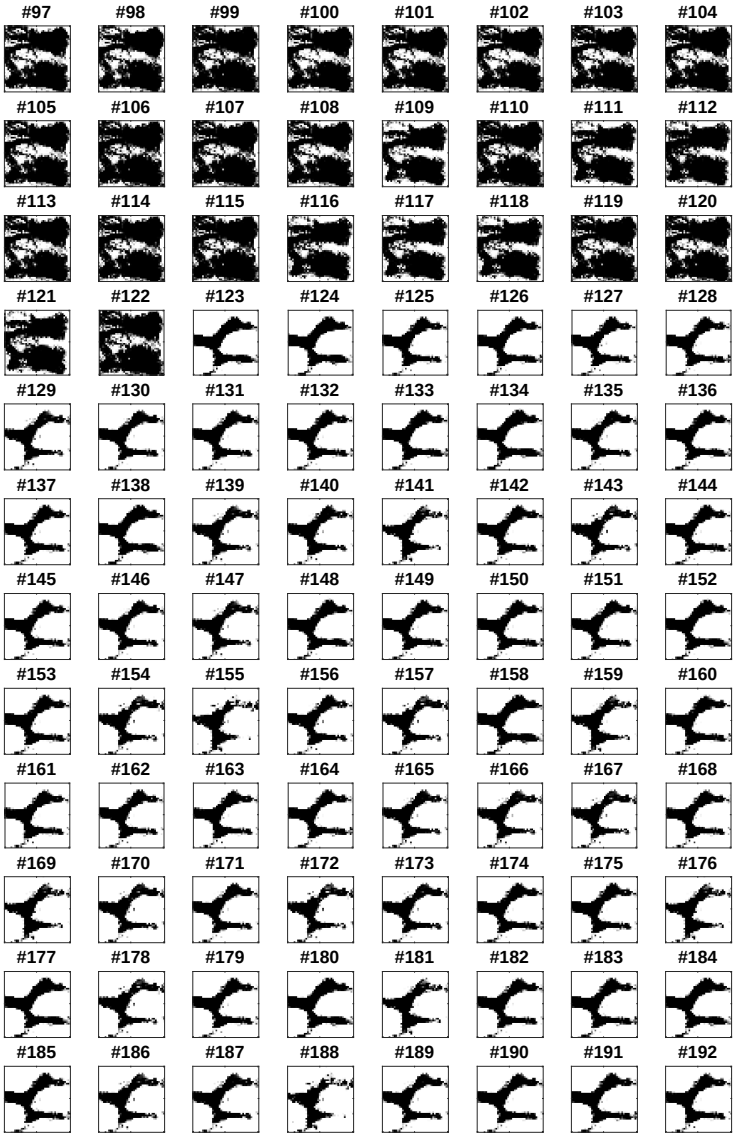

Figure 19. Pareto-optimal solutions (#101 $\sim$ #200) for the MOGA method (latent = 20)

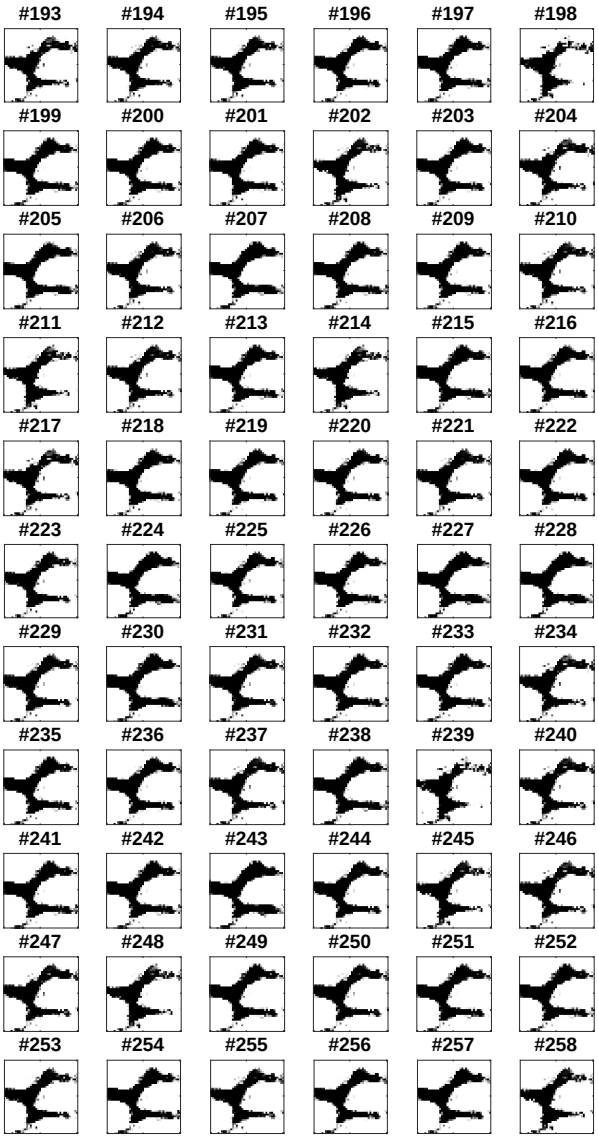

Figure 20. Pareto-optimal solutions (#201 $\sim$ #258) for the MOGA method (latent = 20))