



HAL
open science

Structure and appearance optimization for controllable shape design

Jonàs Martínez, Jérémie Dumas, Sylvain Lefebvre, Li-Yi Wei

► **To cite this version:**

Jonàs Martínez, Jérémie Dumas, Sylvain Lefebvre, Li-Yi Wei. Structure and appearance optimization for controllable shape design. *ACM Transactions on Graphics*, 2015, 34 (6), pp.12. 10.1145/2816795.2818101 . hal-01240642

HAL Id: hal-01240642

<https://inria.hal.science/hal-01240642>

Submitted on 9 Dec 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Structure and Appearance Optimization for Controllable Shape Design

Jonàs Martínez
INRIA

Jérémie Dumas
Université de Lorraine, INRIA

Sylvain Lefebvre
INRIA

Li-Yi Wei
University of Hong Kong

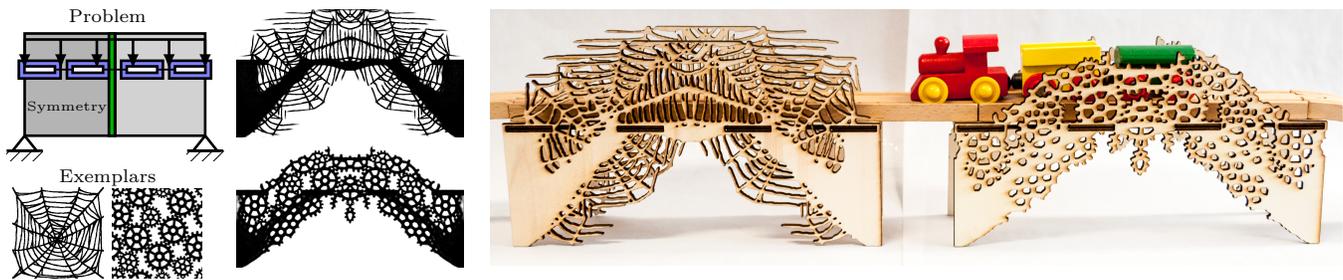


Figure 1: Our technique automatically generates rigid shapes answering a specific loading scenario and resembling an input exemplar pattern, while using a user-specified quantity of material. **Top left:** Loading scenario; in this case the synthesized shape is anchored to the ground by its bottom left/right corners, while supporting a road through four attachments. Each attachment contains an empty region (white) surrounded by a solid boundary (blue), serving as a socket to plug in the road plank. **Bottom left:** Two exemplars defining the desired appearance. **Second column:** Two synthesized bridges answering the loading scenario but each using a different exemplar. **Photograph:** Fabricated objects using the synthesized shapes.

Abstract

The field of topology optimization seeks to optimize shapes under structural objectives, such as achieving the most rigid shape using a given quantity of material. Besides optimal shape design, these methods are increasingly popular as design tools, since they automatically produce structures having desirable physical properties, a task hard to perform by hand even for skilled designers. However, there is no simple way to control the appearance of the generated objects.

In this paper, we propose to optimize shapes for *both* their structural properties *and* their appearance, the latter being controlled by a user-provided pattern example. These two objectives are challenging to combine, as optimal structural properties fully define the shape, leaving no degrees of freedom for appearance. We propose a new formulation where appearance is optimized as an objective while structural properties serve as *constraints*. This produces shapes with sufficient rigidity while allowing enough freedom for the appearance of the final structure to resemble the input exemplar.

Our approach generates rigid shapes using a specified quantity of material while observing optional constraints such as voids, fills, attachment points, and external forces. The appearance is defined by examples, making our technique accessible to casual users. We demonstrate its use in the context of fabrication using a laser cutter to manufacture real objects from optimized shapes.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—[Physically based modeling]; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—[Color, shading, shadowing, and texture]; J.6 [Computer Applications]: Computer-aided Engineering—[Computer-aided manufacturing (CAM)];

Keywords: manufacturing, modeling, texture synthesis, topology optimization

1 Introduction

Recent years have witnessed a significant spread of rapid manufacturing technologies, such as 3D printing and laser cutting. In principle, these techniques empower casual users with the ability to create tangible objects from their virtual counterparts. In practice, it remains extremely difficult to design objects which are aesthetically pleasing and at the same time structurally sound for real world constraints, such as being rigid enough to perform their intended function.

An important effort towards simplifying the creation of complex yet functional objects emerged from the field of topology optimization [Bendsøe 1989; Sigmund 2009; Brackett et al. 2011]. In this field, the primary consideration is to design lightweight structures that are as rigid as possible. That is, optimizing for the most rigid shape using a prescribed amount of material. This is a key engineering problem as material use and weight are directly related to cost and efficiency. These techniques are a perfect match to additive manufacturing technologies as they typically produce complex geometries impossible to manufacture otherwise.

However, these approaches only consider rigidity as an optimization objective, and the appearance of the final object cannot be controlled besides explicit constraints such as avoiding regions of space or enforcing symmetries [Kosaka and Swan 1999]. In this work we propose to jointly optimize for the rigidity *and* the appearance of the structure, as defined by a user-specified exemplar pattern. This is different from after-the-facts reinforcement of the final result [Stava et al. 2012], and from synthesizing uniform, man-

ufacturable patterns [Dumas et al. 2015] (Figure 2): the optimized shape is obtained as the result of a *single optimization problem* integrating both appearance and rigidity, and operates under a *constrained material budget*. It is easy to use via by-example specification and simple constraints such as solid and void regions to enforce, as well as external forces. These constraints are general and allow us to optimize for appearance, mechanical strength, and material cost.

A natural intuition is to combine topology optimization [Sigmund 2009] and by-example texture synthesis [Wei et al. 2009] to satisfy both structural and appearance objectives. However, these are challenging to combine, preventing the algorithm to properly converge. This is confirmed experimentally (see Figure 5) as finding good compromises with a simple combination of these two objectives requires tedious parameter tuning, if possible at all.

Contributions. Our contributions are:

- A new formulation in which appearance is optimized as an objective while rigidity is understood as a *constraint*.
- Controls that are powerful yet easy to understand: single parameter for the appearance-rigidity tradeoff (α), volume usage bounds (v_{min}, v_{max}), and appearance specified by example.
- First order derivatives for the appearance objective, enabling gradient descent under non-linear constraints.

Scope. In this paper, we target the synthesis of flat (and possibly curved) shapes that are manufactured with laser cutting. Our formulation directly translates to volumes (see Section 5.3), but results in large problems which are too slow to solve via our current implementation. Scalability to higher dimensions is an important direction of future work.

2 Related Work

Our work is at the crossing of two fields: by-example texture synthesis and topology optimization.

By-example texture synthesis. The problem of synthesizing a new image resembling an exemplar image is a long-standing problem in Computer Graphics [Wei et al. 2009]. We focus here only on works most related to ours.

The initial methods for by-example texture synthesis are based on Markov Random Fields [Efros and Leung 1999; Wei and Levoy 2000]: a probabilistic model is defined by sampling neighborhoods from the exemplar. These approaches are not trivially amenable to our context due to the stochastic nature of the optimization process. Kwatra et al. [2005] proposed a different point of view of the problem which is based on formulating an energy — the pixels of the output image being the variables. By optimizing this energy a new image is synthesized which resembles the example. We adopt this point of view to define the appearance energy relating the produced shape to the exemplar.

Few works have considered both texture synthesis and fabrication. Zhou et al. [2014] synthesize patterns along curves while precisely controlling their topology. The results can then be laser-cut or 3D printed as they form singly-connected objects. The software *Magics* by *Materialize* contains libraries of structures that can be tiled inside a 3D object. To the best

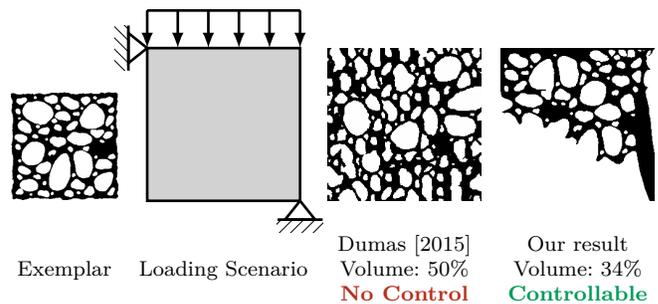


Figure 2: Comparison with [Dumas et al. 2015]. **From left to right:** input exemplar; loading scenario, attachment points and optimization domain (gray square); result of [Dumas et al. 2015]; our result. Regardless of the loading scenario Dumas et al. always seek to produce a structure that fills a given region, while we generate a rigid shape using the user-specified quantity of material and resembling the input exemplar.

of our knowledge, these structures are however not designed from examples nor optimized for a specific loading scenario. In concurrent work, Dumas et al. [2015] propose a technique synthesizing a uniform stochastic pattern *covering* a surface, while ensuring that it is printable. However, the application is different: our approach generates a shape under a *prescribed material budget* and will generally not fill a domain. Instead, it seeks for the optimal compromise between appearance and rigidity while distributing material in space. Figure 2 highlights the differences: while [Dumas et al. 2015] targets the equivalent of uniform texturing, we target controllable shape design.

Style transfer. Style transfer is a variant of by-example synthesis where the process is guided to enrich existing content with details. Hertzmann et al. [2001] pioneered this idea by proposing to transfer details specified by a pair of images A:A' to produce an image B' from a different source image B. Recent approaches have explored how to exchange styles within and across collections of shapes [Xu et al. 2010; Xu et al. 2012; Li et al. 2013; Han et al. 2015; Ma et al. 2014]. These approaches typically require a collection or a pair defining style by analogy, which are not available in our context.

Our work jointly optimizes appearance and structural objectives, instead of transferring style after the facts. The global structure therefore emerges from the details of the pattern, which becomes an intrinsic part of the final shape.

Topology optimization. Topology optimization is the process of determining the optimal distribution of material inside a domain [Bendsøe and Kikuchi 1988; Deaton and Grandhi 2014], so as to optimize various objectives such as rigidity. We discuss these approaches in more details later in the text, as they are an important component of our work.

Different manufacturing constraints for topology optimization have been investigated [Sigmund 2009], however, there exists little work regarding aesthetics. Paulino et al. [2015] recently proposed to perform topology optimization over a 2D polygonal tessellated domain, for artistic purposes. Kosaka et al. [1999] constrain the process to enforce symmetries, which can be used for aesthetic considerations. Christiansen et al. [2015] demonstrate the use of topology optimization

for modeling purposes. However, none of these methods can handle general exemplar patterns.

Fabrication. Our work targets modeling for fabrication. Several recent approaches considering structural properties have been proposed in this context, for instance, to reinforce fragile 3D models [Stava et al. 2012], to analyze rigidity prior to fabrication [Zhou et al. 2013; Umetani and Schmidt 2013], to achieve strong but lightweight parts [Wang et al. 2013; Lu et al. 2014], or to fabricate objects with prescribed mechanical behavior [Panetta et al. 2015; Schumacher et al. 2015]. However, these methods consider mechanical properties only, but not appearance as our method does.

3 Our Formulation

Our method combines two fundamental ingredients. The first is the notion of appearance as defined by neighborhood similarities between a synthesized structure and an example pattern. The second comes from mechanical engineering and is the notion of *compliance*. Figure 3 provides an overview of our method.

The basic problem in which compliance appears is the prediction of the mechanical behavior of a structure when it is subjected to precise boundary conditions — that is, a set of attachment points and external loads applied to the structure. In this work we consider small deformations for which the behavior of the structure can be characterized by linear elasticity. We consider isotropic materials described by their Young’s modulus and Poisson’s ratio. The compliance is the

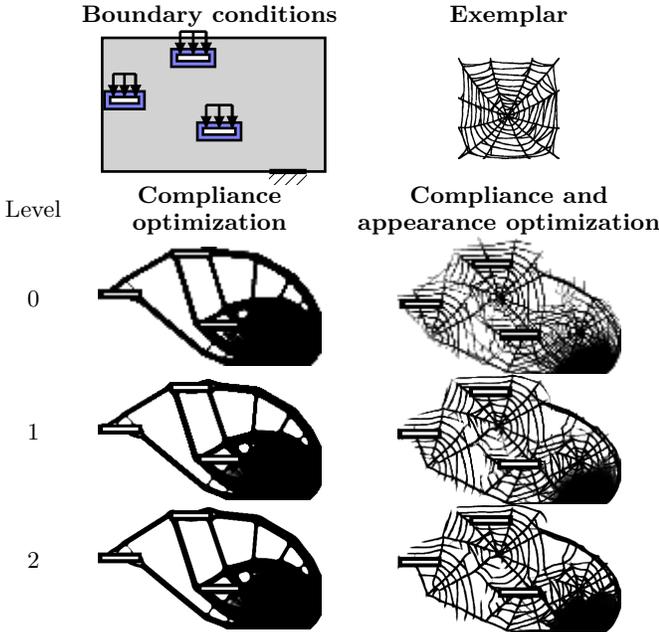


Figure 3: Overview of our multi-resolution optimization approach. Given an input exemplar (upper right) and output boundary conditions (upper left), our method optimizes the corresponding output in a multi-resolution fashion (lower rows). The boundary conditions can include support, solid, and void. In this example, the volume is constrained to 35% of the overall output domain, and the relaxation factor of the compliance with respect to the optimal is set to $\alpha = 1.2$.

work exerted by the forces on the structure, i.e. the sum of the dot product of forces and displacements, as illustrated in Figure 4. A low compliance implies that forces produce only small displacements, which characterizes a high rigidity.

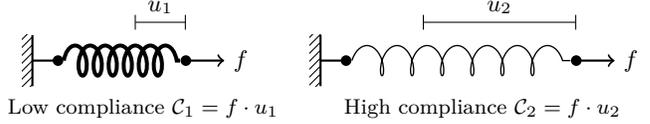


Figure 4: Two 1D linear elements (springs) with a single degree of freedom. A same force f generates a displacement of respectively u_1 and u_2 . The compliances reflect that the stronger spring (leftmost) is more rigid: $C_1 < C_2$. The same concept translates to elements of higher dimensions.

We model the shape in an n -dimensional grid of square elements denoted by \mathbf{x} , each having 2^n corner nodes shared with their neighbors. Each element e in \mathbf{x} receives a *density* x_e which through optimization has to converge towards void ($= 0$) or solid ($= 1$), thus defining an interior and exterior. In practice, intermediate values remain after optimization, and we apply a thresholding after convergence.

We formulate our goal as a multi-objective optimization problem that minimizes both an appearance energy $\mathcal{A}_I(\mathbf{x})$ and the structural compliance $\mathcal{C}(\mathbf{x})$:

$$\bar{\mathbf{x}} = \arg \min_{\mathbf{x}} (\mathcal{A}_I(\mathbf{x}), \mathcal{C}(\mathbf{x}))$$

, where I is the input exemplar — a black and white pattern defining void (pixel = 0) and solid (pixel = 1) regions; $\bar{\mathbf{x}}$ is the outcome — densities defining a shape in the grid — computed through our optimization procedure. The user has to specify at least one attachment point for the problem to be well-posed. She can optionally impose additional conditions, such as regions of void or fill, symmetry, and external forces; see results in Figure 1 and Section 6.

Previous works exist to optimize each of these energies in isolation (Section 2). Therefore, a straightforward approach would optimize for a linear blend of both energies, that is:

$$\mathcal{A}_I(\mathbf{x}) + \lambda \mathcal{C}(\mathbf{x}) \tag{1}$$

with $\lambda > 0$ allowing to explore the tradeoff. Unfortunately, such a simple scheme does not produce reliable results: the values of λ that can produce a reasonable output differ widely between exemplars, boundary conditions, and domain size, when they exist at all. Figure 5 illustrates this issue and compares to our formulation.

We therefore propose to modify the formulation of the problem. We note that the goal is not necessarily to obtain the most rigid structure, but rather a structure with *sufficient rigidity*, i.e. which does not yield under the given loads. Thus, our insight is that rigidity should be considered as a constraint, which can be relaxed to allow more freedom for the appearance objective. Thus, our goal is now to minimize $\mathcal{A}_I(\mathbf{x})$ such that the structural compliance is below a threshold C_{max} and the volume is bounded:

$$\begin{aligned} \arg \min_{\mathbf{x}} : & \mathcal{A}_I(\mathbf{x}) \\ \text{subject to :} & \mathcal{C}(\mathbf{x}) \leq C_{max} \\ & v_{min} \leq \sum_e x_e \leq v_{max} \\ & \forall e \ 0 \leq x_e \leq 1 \end{aligned}$$

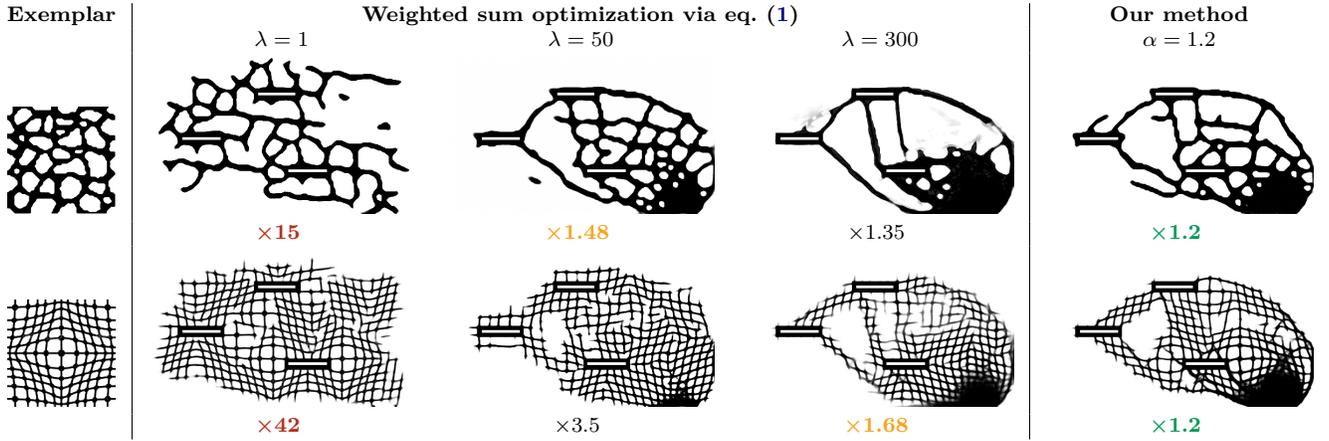


Figure 5: Comparison of a straightforward weighted sum approach and our formulation. All results use the same parameters and a volume constrained to 30% of the entire domain. We give below each result the ratio between its compliance and the compliance of the shape optimized without appearance objective (C_{opt}); e.g. $\times 1.2$ implies the result is within 20% of the computed optimum. λ weights the importance of rigidity versus appearance. On the left hand side, a low λ gives results with good appearance but mediocre compliance. On the right hand side, a large λ produces more rigid results but degraded appearance. We show in orange the values of λ producing reasonable compromises, and in green and red the best and worst compliance ratios, respectively. Note that these differ significantly between both exemplars. Our method (rightmost column) does not need any specific setting besides the threshold from the computed optimum (20%). The boundary conditions are the same as in Figure 3.

The volume bounding constraint is important. The weight of the structure is often negligible compared to external forces, in which case the most rigid shape would tend towards a full block of material. v_{max} prevents this naive solution to exist. On the contrary, when only considering the weight of the structure — i.e. solving a *self-weight problem* — the naive solution is an empty shape. v_{min} prevents it. The volume constraint is also a natural control for the user. Combined to appearance it allows changing the overall size of the structure (see Figure 6). For the sake of clarity we express volume constraints as a percentage of the design domain. We often use only external forces or only self-weight in which case we respectively set $v_{min} = 0\%$ or $v_{max} = 100\%$. In such cases we report only the non-trivial bound.

A meaningful value of the C_{max} constraint is crucial to ensure a feasible solution. We describe how the threshold is computed in Section 3.1, and describe the appearance objective in Section 3.2. We discuss our solver and numerical scheme in Section 4 and extensions in Section 5.

3.1 Compliance Constraint

We determine the compliance threshold C_{max} by first computing a solution C_{opt} to the problem considering compliance alone. We adopt the well-established Solid Isotropic Material Penalization (SIMP) topology optimization method [Bendsøe and Kikuchi 1988]. Implementation details of this method can be found in [Sigmund 2001; Andreassen et al. 2011].

3.1.1 Topology Optimization for Determining C_{opt}

The SIMP method seeks to minimize compliance by assigning densities to each element in \mathbf{x} given a constrained total material budget $v_{min} \leq \sum_e x_e \leq v_{max}$. The elastic structure is simulated with the Finite Element Method (FEM).

Each square element e receives a continuous scalar density $0 \leq x_e \leq 1$, and its Young’s modulus is defined as:

$$E_e = E_{min} + (x_e)^p (E_0 - E_{min})$$

, where E_0 is the material Young’s modulus, $E_{min} > 0$ is a small value to prevent numerical instabilities, and $p = 3$ is the standard SIMP penalization factor which penalizes intermediate values in the solution.

The compliance of the output is measured by summing the compliance of each individual element. Let us denote:

- \mathbf{u} Global displacement vector.
- \mathbf{u}_e Element displacement vector.
- \mathbf{f} Global force vector.
- \mathbf{K} Global stiffness matrix.
- \mathbf{K}_0 Element stiffness matrix with unit Young’s modulus.
- t Element thickness.

Compliance minimization is formulated as follows:

$$\begin{aligned} C_{opt} &= \min_{\mathbf{x}} : \mathcal{C}(\mathbf{x}) \\ \text{subject to : } & v_{min} \leq \sum_e x_e \leq v_{max} \\ & \forall e \ 0 \leq x_e \leq 1 \end{aligned}$$

, where the compliance term is given by the equation:

$$\begin{aligned} \mathcal{C}(\mathbf{x}) &= \mathbf{u}^T \mathbf{f} = t \sum_e E_e \mathbf{u}_e^T \mathbf{K}_0 \mathbf{u}_e \\ \text{subject to : } & \mathbf{K} \mathbf{u} = \mathbf{f} \end{aligned}$$

We use the Method of Moving Asymptotes (MMA) [Svanberg 1987] to minimize $\mathcal{C}(\mathbf{x})$ and obtain a solution C_{opt} .

Note that when the boundary conditions include large external forces, the weight of the structure tends to have a negligible influence, and can be ignored. We discuss in Section 5.1 how self-weight can be taken into account.

3.1.2 Setting the Threshold C_{max}

We first perform standard topology optimization to compute a lower bound on the achievable minimal compliance C_{opt} . We then set $C_{max} = \alpha C_{opt}$, where $\alpha \geq 1$. As we increase the value of α , we obtain results having a higher compliance, but that are more similar to the exemplar. This is a simple and predictable parameter controlling how much freedom is allowed to appearance, as can be seen in Figure 6.

3.2 Appearance Objective

Our method can be used with any appearance energy that has first order derivatives available, as required by the solver (Section 4). Our formulation builds upon the work of Kwatra et al. [2005] but adapts it to facilitate the computation of first order derivatives with respect to \mathbf{x} .

In the following, when using the notation x_e , e is meant as a coordinate within the grid of elements. We denote by $N = 2\delta + 1$ the texture synthesis neighborhood size (typically 15×15 , i.e. $\delta = 7$). We denote by $m(e)$ the *coordinate* of the matching pixel in I whose neighborhood \mathbf{z}_e is the most similar to the current neighborhood of x_e in the result. The matching process is described in Section 4. We denote by $z_c^e = I[m(c) + (e - c)]$ the value at coordinate e from the best matching neighborhood of c , where $I[z]$ accesses the pixel value (0 or 1) at coordinate z in I . The appearance energy contributed by an element $x_e \in \mathbf{x}$ is:

$$\mathcal{A}_I(x_e) = \sum_{c, \|c-e\|_\infty \leq \delta} |x_e - z_c^e|^r \quad (2)$$

The total appearance energy is $\mathcal{A}_I(\mathbf{x}) = \sum_e \mathcal{A}_I(x_e)$. We use $r = 1.2$, set experimentally as in [Kwatra et al. 2005].

4 Solver

We are now ready to solve for the global optimization problem, that is to minimize appearance under the compliance constraint (Section 3). This is a challenging optimization as all terms are non-linear and are subject to inequality constraints. In addition, the appearance objective contains combinatorial terms: the best matching neighborhoods.

We thus optimize for the density \mathbf{x} in an iterative block coordinate descent scheme alternating between finding the best matching neighborhoods coordinates $m(e)$ and the densities x_e for each element e .

Initialization. We set $x_e = \min(\frac{v_{max}}{|\mathbf{x}|}, 1)$ for all elements and select random coordinates for best matches $m(e)$. One can choose a different initial guess, but a better convergence is typically observed when starting from a uniform gray; see Figure 2 in [Sigmund and Maute 2013].

Neighborhood matching. Given the densities, the coordinates of the most similar neighborhoods are computed with PatchMatch [Barnes et al. 2009] and the random walk of [Busto et al. 2010]. For a neighborhood at coordinate e in \mathbf{x} and a neighborhood at coordinate e' in I the similarity used for matching is defined as:

$$d(e, e') = \sum_{c, \|c-e\|_\infty \leq \delta} |x_c - I[e' + (c - e)]|^r + \lambda_{occ} \frac{O[e']}{N^2} \cdot \frac{|I|}{|\mathbf{x}| N^2}$$

The first term compares pairwise densities throughout neighborhoods and corresponds to the appearance energy (eq. (2)). The second term is used to increase the spatial uniformity of the appearance energy [Kopf et al. 2007; Kaspar et al. 2015]. O is an occurrence map storing how many times each exemplar pixel is used in the different closest neighborhoods. It is computed from the previous iteration. $\lambda_{occ} \geq 0$ controls the amount of enforced spatial uniformity. $\frac{|I|}{|\mathbf{x}| N^2}$ is a normalizing factor making λ_{occ} independent of synthesis resolution. In our implementation $\lambda_{occ} = 20$.

Optimizing densities. Optimizing the appearance objective (eq. (2)) given the best matching neighborhoods is more challenging. We are facing a non-linear, non-convex optimization problem for both objective and constraints. In addition, evaluating the compliance constraint is computationally expensive as it requires solving for the FEM equation. For these reasons we rely on the Globally Convergent Method of Moving Asymptotes (GCMMA) [Svanberg 1995]. GCMMA converges in fewer iterations than augmented Lagrangian methods, reducing the number of required FEM solutions. It iteratively solves subproblems that are convex approximations of the original problem, and rely on the gradient of both objective and constraints functions to do so. GCMMA therefore requires the first order derivatives of the objective and constraint functions.

Derivatives. The derivative of the volume constraint is 1. The compliance derivative is [Bendsøe and Sigmund 2003]:

$$\frac{\partial \mathcal{C}}{\partial x_e} = 2\mathbf{u}_e^T \frac{\partial \mathbf{f}}{\partial x_e} - \mathbf{u}_e^T \frac{\partial \mathbf{K}}{\partial x_e} \mathbf{u}_e$$

The term $\frac{\partial \mathbf{f}}{\partial x_e}$ is null when the weight of the structure is neglected. For self-weight problems (Section 5.1) this term will however influence the result.

The derivatives are typically filtered to prevent numerical issues, and to control the optimization quality. We consider the extensively used smoothing operator described by [Sigmund 1997] to filter the derivative of the function \mathcal{C} :

$$\widehat{\frac{\partial \mathcal{C}}{\partial x_e}} = \frac{1}{\max(\epsilon, x_e) \sum_i w_{e,i}} \sum_i w_{e,i} x_i \frac{\partial \mathcal{C}}{\partial x_i}$$

, where $w_{e,i} = \max(0, \gamma - \text{dist}(e, i))$ is a weight factor (convolution), controlled by a parameter γ (filtering radius), and ϵ (10^{-3}) is a small coefficient avoiding division by zero.

At this step, all z_c^e are assumed to be constants. Therefore, the derivative of $\mathcal{A}_I(\mathbf{x})$ can be expressed as:

$$\frac{\partial \mathcal{A}_I}{\partial x_e} = \sum_{c, \|c-e\|_\infty \leq \delta} r \frac{|x_e - z_c^e|^r}{(x_e - z_c^e)}$$

The derivative is not defined when $x_e = z_c^e$. Thus, using a small $\epsilon > 0$, we regularize $\mathcal{A}_I(\mathbf{x})$ as follows:

$$\tilde{\mathcal{A}}_I(x_e) = \sum_{c, \|c-e\|_\infty \leq \delta} ((x_e - z_c^e)^2 + \epsilon)^{r/2}$$

, and it follows that the derivative is:

$$\frac{\partial \tilde{\mathcal{A}}_I}{\partial x_e} = \sum_{c, \|c-e\|_\infty \leq \delta} r (x_e - z_c^e) ((x_e - z_c^e)^2 + \epsilon)^{r/2-1}$$

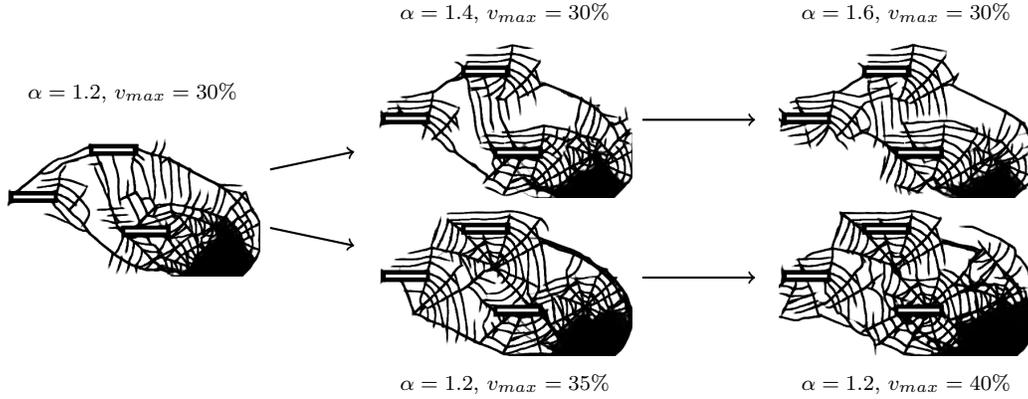


Figure 6: Progressive relaxation of the maximum compliance (top row) and maximum volume (bottom row) constraints.

Multi-resolution. For improved performance and quality we optimize through a multi-resolution scheme. The process starts from downsampled versions of the grid x and exemplar I . The resolution is iteratively doubled, using the previous result to initialize the next finer resolution by bilinear up-sampling. The process is illustrated in Figure 3.

Our algorithm optimizes three resolution levels. The compliance relaxation parameter α remains constant throughout the process. The exemplar is downscaled to match the resolution of each level. The strain-displacement and constitutive material matrices are also changed according the resolution.

We use the same multi-resolution scheme to compute the compliance solution (Section 3.1) and obtain C_{max} for each resolution level. In practice, we observe that the value of C_{max} is remarkably stable across resolutions.

Convergence. The optimization process ends when $\frac{\|\nabla \mathcal{A}\|}{\|\mathcal{A}\|}$ is below a small threshold (we use 0.001), or when a maximum number of iterations is reached (in our implementation we use 40, 20, 10 on the three successive resolution levels).

5 Extensions

Optional constraints can be added to our basic formulation. We describe two important ones. The first is to consider the weight of the structure itself during optimization (Section 5.1). This is useful when there is no external force besides gravity applied to the structure. The second is to consider symmetry constraints, which are useful for aesthetics purposes but also to reduce computation time when the solution is known to have symmetries (Section 5.2). We also describe how to optimize for 3D outputs even though our method is dimension agnostic (Section 5.3).

5.1 Self-weight

We optionally take into account the weight of the structure and the forces it generates under gravity. Note that on self-weight problems — i.e. no external forces — the complete void is a trivial optimal. We therefore impose $v_{min} > 0$ in such cases.

Forces due to the structure weight are modeled by a vertical force acting on each grid node q as follows:

$$q_x = 0, \quad q_y = -g \sum_{e, q \in \mathbf{q}_e} x_e \frac{m_e}{|\mathbf{q}_e|}$$

, where \mathbf{q}_e is the set of nodes belonging to element e (as defined in Section 3), m_e is the element mass, and g is the absolute value of the gravitational acceleration. Let us emphasize that this force depends on the current densities of the elements x_e .

When using the SIMP formulation on a problem taking into account the weight of the structure, the displacements might become unbounded for low density regions, resulting in numerical issues [Bruyneel and Duysinx 2005]. We therefore use a modified formulation of material stiffness as suggested by [Pedersen 2000] to overcome this problem:

$$E_e = \begin{cases} E_{min} + (x_e)^p (E_0 - E_{min}) & \mu < x_e \leq 1 \\ E_{min} + x_e \mu^{p-1} (E_0 - E_{min}) & 0 < x_e \leq \mu \end{cases} \quad (3)$$

In our experiments we set $\mu = 0.25$. This switches to a linear stiffness model in regions of low densities. The derivatives are updated accordingly. The non-differentiable point where the switch between models occurs does not have a detrimental impact in practice [Bruyneel and Duysinx 2005].

We observe that on self-weight problems the volume bounds v_{min}, v_{max} have to allow for some freedom to achieve convergence. Indeed, the optimized shape is a subtle tradeoff: adding matter makes some regions more rigid but also adds stress to others through gravity. Figure 7 illustrates results obtained on self-weight problems.

5.2 Symmetry

Symmetry plays an important role in aesthetics for shape design. We adopt the symmetry reduction approach of [Kosaka and Swan 1999] for topology optimization. The design domain \mathbf{x} is partitioned into $S > 1$ subdomains \mathbf{x}^i . We define a mapping between \mathbf{x}^i and an imaginary domain \mathbf{x}^* , and only optimize for \mathbf{x}^* . The derivatives are:

$$\frac{\partial \mathcal{C}}{\partial x_e^*} = \frac{1}{S} \sum_i \frac{\partial \mathcal{C}}{x_e^i}$$

That is, we compute the derivatives for the design domain \mathbf{x} , and optimize \mathbf{x}^* according the averaged derivatives given by

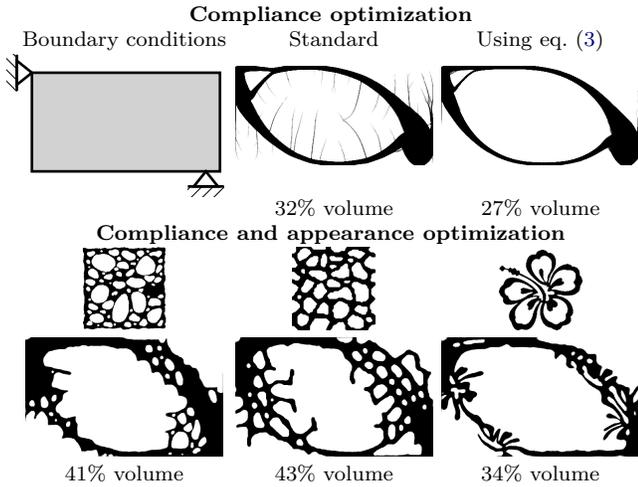


Figure 7: Optimizing self-weight problems (no external forces). **Top:** A different material stiffness model is required to avoid degeneracies in low density regions. **Bottom:** All results use the same parameters $\alpha = 2.2$, $v_{min} = 20\%$, $v_{max} = 100\%$. Self-weight problems are more challenging to optimize (see text), and therefore the bounds are relaxed to let the optimizer converge. Note how different volumes are obtained depending on the exemplar.

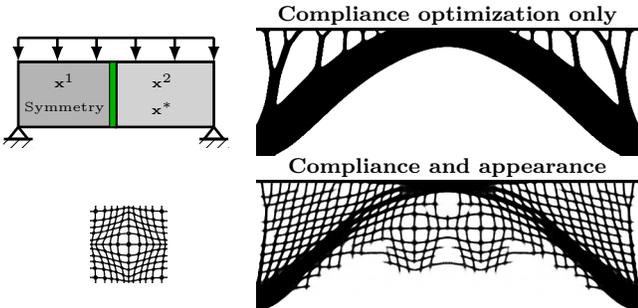


Figure 8: Constrained symmetry, with boundary conditions defined on \mathbf{x}^* . Maximum volume is constrained to 45%.

the mapping. Note that even though all x_e^i variables map to the same x_e^* , their individual gradients on the right-hand side may differ due to asymmetric loading scenarios.

5.3 Optimizing 3D Structures

Our formulation is amenable to 3D, adding a third dimension and using a grid of cubic (hexahedral) elements. While this would provide a full volume synthesis, such an approach is computationally expensive and requires a 3D exemplar as input (schemes using several 2D exemplars to define a volume could be adapted [Wei et al. 2009]).

Instead, we propose to optimize structures along several interleaved planes in 3D, as illustrated in Figure 9. This is different from independently optimizing 2D shapes: the 3D hexahedral elements at the crossing of several planes are shared, and stresses and appearance propagate across the different planes. The results in Figure 9 show how pattern features are able to flow from one plane to another.

6 Results

Most of our results are obtained by laser cutting from the synthesized shapes. We then assemble objects by gluing several planks together. In most cases, we compute and assemble independent 2D results, under the assumption that forces remain in a plane — which works well in practice for most scenarios. We however also investigated full 3D solutions, as described in Section 5.3.

Before presenting our results in more details Section 6.2, we describe Section 6.1 how we obtain the final curves for laser cutting. We discuss performance in Section 6.3 and validation tests in Section 6.4.

6.1 Contour Extraction

Contours for laser cutting are extracted in a few simple steps. The optimized shape \bar{x} is first thresholded (0.5) to snap values which are between 0–1 to void or solid. In rare cases, this results in the creation of small disconnected components. We filter these by keeping the connected components anchored to attachment points. Disconnected components are further discussed in Section 6.5. Finally, the filtered grid is upsampled by bilinear filtering (x2 in our implementation), and paths for laser cutting are extracted along the isovalue 0.5.

6.2 Fabricated Objects

We created several objects using our approach. In all cases, the user only specified the attachment points, external forces, target volume and example pattern. The algorithm automatically synthesizes the structure. Thus, many results can be easily produced using a variety of patterns: the algorithm deals with the complex task of generating the intricate details of the final structure. While we only laser cut miniatures, industrial cutters could be employed to fabricate large-scale objects in a variety of materials.

Besides attachment points and external forces, the user may also rely on *passive elements*, which can represent non-designable parts with a fixed density. The inset figure illustrates the use of passive elements to optimize a structure around the SIGGRAPH logo. This is obtained by constraining the value of x_e , that is $x_{min} \leq x_e \leq x_{max}$. If desired, other passive element properties can be predefined, such as the material volumetric mass density and stiffness.



In Figure 10 different bridge sides supporting a road are obtained by changing the loading scenario. A symmetry constraint is also used, but this is optional. Results using different patterns on the same set of conditions are shown Figure 1. In Figure 11 a set of shelves is produced. They are designed to be fixed on the ground and to support several shelves offset from the attachment point: the weight is entirely supported by the sides. Yet, the structure remains visually similar to the exemplar pattern. In Figure 12 we apply the same principle to produce phone stands. Figure 13 shows a variety of tables obtained by interleaving three planks. Using different patterns immediately changes the look and feel of the results. The tables are very strong and can support large weights.

Finally, we show in Figure 9 3D results where the structures along each plane are optimized jointly in an interleaved 3D problem. This allows the pattern to flow from one plane

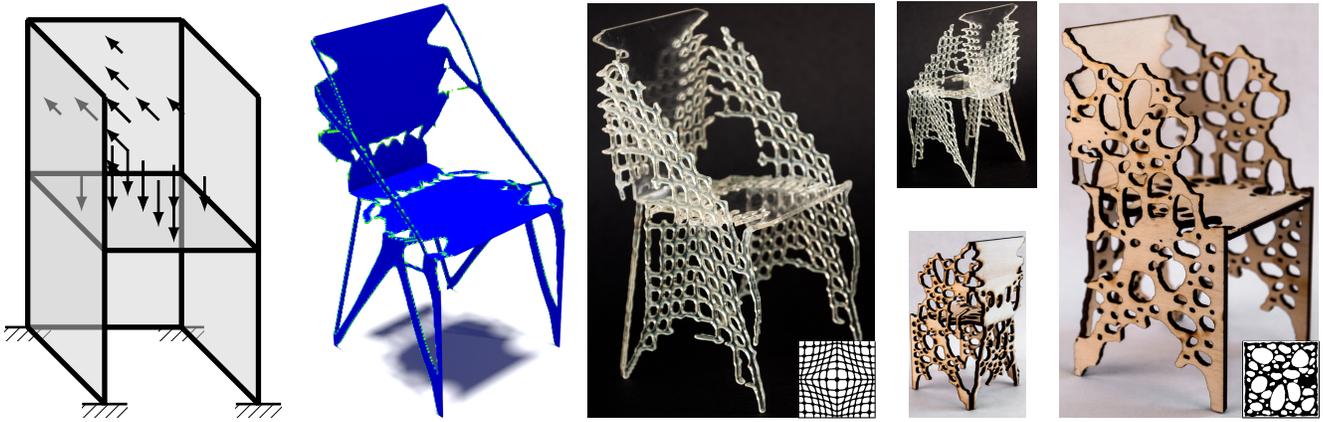


Figure 9: 3D synthesis. *From left to right: Boundary conditions, result optimized without appearance, two fabricated chairs from our optimized results using different patterns.*

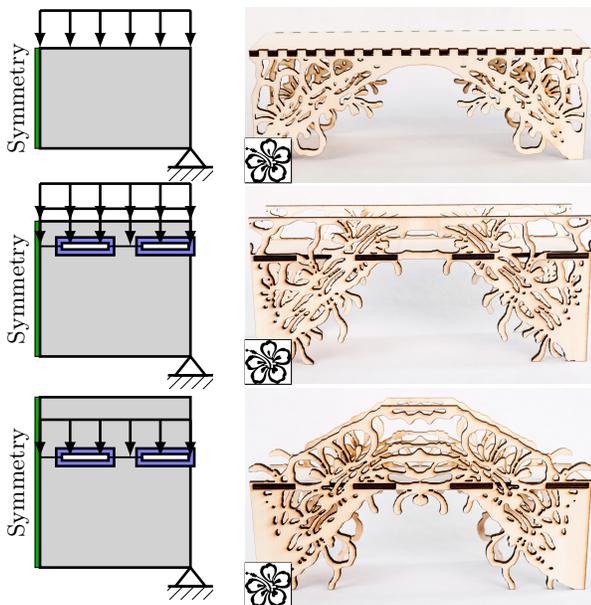


Figure 10: Bridges obtained from the same exemplar using different loading scenarios. Symmetry is used, which is why the boundary conditions are shown for only half of the problem. **Top:** Forces are applied at the top, supporting only the road. **Middle:** Forces are applied below the top for the road, and at the top to create a handrail. **Bottom:** Forces are applied in the middle. Combined with the passive elements this produces an arch.

to another, while in previous results the pattern features could be interrupted between different planks. Also, note how the result optimized without appearance looks much less appealing in 3D, while our results produce intricate, visually interesting details.

6.3 Performance

We implemented our approach with Python and use the GCMMA implementation of the NLopt [Johnson 2007] optimization library. We measured the execution time on an Intel® Core™ i7-4770K @ 3.50GHz, 16 GB RAM. Table 1



Figure 11: A set of shelves, meant to be fixed on the ground. Two exemplars are used on the same problem, producing results of very different styles but the same purpose.



Figure 12: Phone stands fabricated from optimized structures using two different exemplar patterns.

summarizes performance for the main results. On average, in 2D, 75% of the time is spent on the FEM computation, 20% on the appearance gradient computation, and 5% on the GCMMA optimization.

As performance was not our focus, our reference implementation uses a single thread and takes in the order of minutes to converge. Nevertheless, the multi-resolution approach allows the user to preview the result being computed.

As can be seen in Table 1, the 3D result of Figure 9 takes roughly three times longer to compute than a 2D problem with the same number of elements.



Figure 13: Three tables produced with our system, using three planks and structures optimized independently. The two tables on the left use the same external conditions but different patterns. The right table weights 76g and is supporting a filled cup of 762g (10× heavier).

	# elements	exemplar	time	# iter
Figure 1	202K	Web 240 × 240	8'33"	120
Figure 2	62K	Sponge 200 × 200	1'56"	77
Figure 5	150K	Cells 210 × 210	4'20"	112
Figure 7	80K	Cells 200 × 200	1'04"	56
Figure 8	120K	Grid 160 × 160	5'24"	118
Figure 9	114K	Sponge 104 × 104	14'20"	113
Figure 10*	162K	Flower 176 × 169	5'45"	106
Figure 12	230K	Grid 330 × 330	7'38"	89
Figure 14	90K	Cells 105 × 105	3'34"	119

Table 1: Performance on our main results. The exemplar resolutions are shown as width×height. Times are formatted as min'sec'. * Figure 10 middle result.

6.4 Structural Properties

We first verify that the structural properties optimized by our system can be observed after fabrication. We show in Figure 14 two bridge structures using the same volume but optimized with a different force in the center. As can be seen, for the case where the force is small the optimizer grows features on the top of the bridge, as this does not violate the compliance constraint. On the contrary, when the force is large the optimizer concentrates material below the arch, for reinforcement. When a heavy load is applied to both, the bridge optimized for a small load collapses, whereas the bridge optimized with the correct force withstands it easily.

Compliance reflects the global rigidity of a shape but does not consider local stresses. Therefore, there is a concern that shapes of low compliance but high local stresses could be produced, resulting in local failures under loads. In practice the results usually exhibit low local stresses, but for a few specific places such as sharp corners. This is illustrated in Figure 15 for the classical L-Beam test. Our approach inherits this limitation from compliance-based methods. It is however worth noting that our method produces results having comparable local stresses to those optimized without appearance, as shown in Figure 15.

Note that in the field of topology optimization compliance is widely used due to its smoother behavior (see e.g. §2.4 in [Deaton and Grandhi 2014], §6.10 in [Sigmund and Maute 2013]) making it amenable to efficient gradient descent mini-

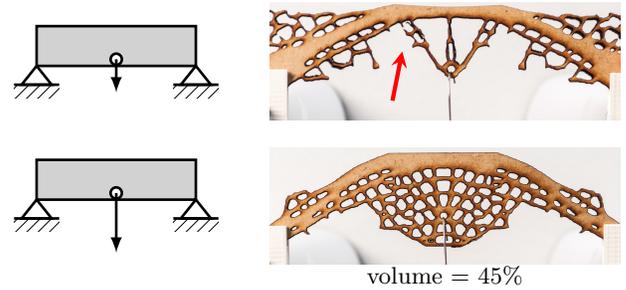


Figure 14: Two structures in MDF wood optimized for different forces but with the same volume constraint. The ratio between the two forces is 10 : 1. The arrow indicates where the top structure starts to rupture when loaded with the heavier weight intended for the bottom structure (please refer to the accompanying video).

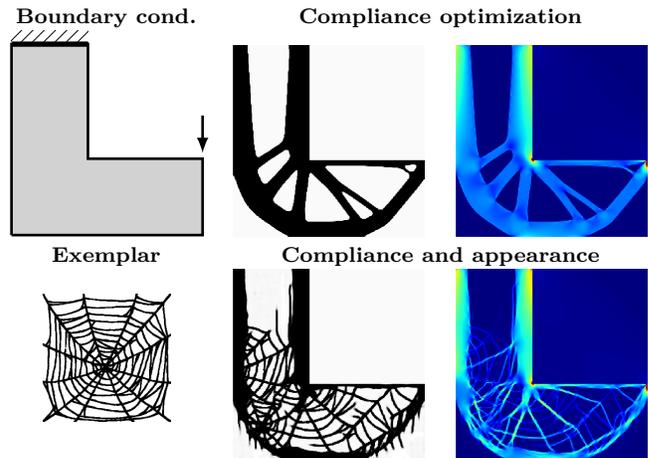


Figure 15: Comparison of Von Mises stresses. $v_{max} = 40\%$, $\alpha = 1.3$. We use the classical L-Beam boundary conditions for stress analysis [Duysinx and Bendsoe 1998] (see top left image). Stress colors use the same normalization of both rows. Our structure (bottom row) has a higher compliance ($\alpha = 1.3$), but local stresses are comparable to the ones found in the result without appearance (top row).

mization. Optimizing local stresses is still an active research topic [Paris et al. 2005; Lee et al. 2012; Holmberg et al. 2013].

6.5 Limitations, Future Work

Connectivity and convergence. While the SIMP method does not explicitly prevent disconnected components from appearing, they are typically not present in an optimized design. Whenever the compliance constraint is violated, matter is redistributed in weaker regions to reduce the compliance. Combined with the volume constraint this discourages the existence of disconnected components. In all our results, the disconnected elements represent less than 3% of the total volume. However, in cases where the exemplar has many small disconnected components and where the user allows for a high compliance threshold, the appearance objective is free to generate disconnected components (Figure 16, col.1–2).

Self-weight (Section 5.1) explicitly penalizes disconnected components: unsupported matter produces high compliance.

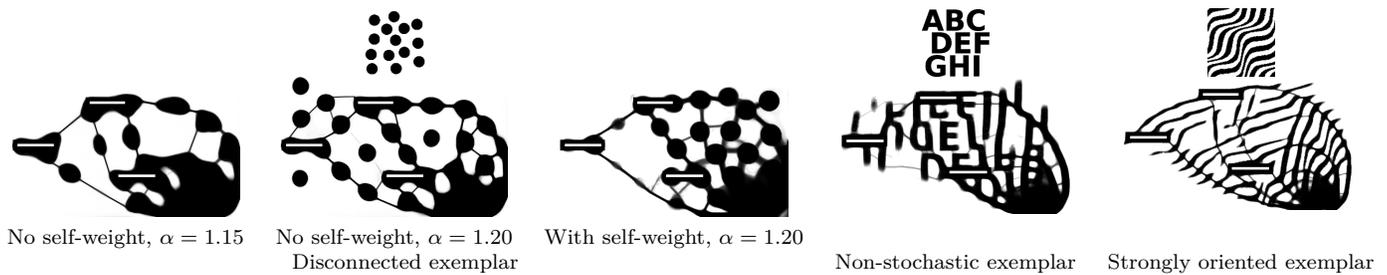


Figure 16: Some challenging exemplars.

Therefore, using self-weight encourages well connected shapes (Figure 16, col.3). However, combining self-weighting with external forces generally leads to very challenging optimization problems and the resulting shapes are not perfectly converged. In addition, self-weight require relaxing the compliance bound (see Figure 7). This typically requires the user to test a few different parameters. Strictly enforcing connectivity thus remains an open direction of future work.

Note that there are assumptions about the exemplars that are inherited from texture synthesis. They should exhibit an overall stochastic, homogeneous appearance, and have features roughly the size of the selected neighbourhood size. The appearance of organized patterns is not properly reproduced (Figure 16, col.4). Strongly oriented patterns make it challenging to create a rigid structure when the features do not align with stress directions (Figure 16, col.5).

Thickness constraints. While we did not impose a minimum length-scale on our designs, previous works exist to control the minimum thickness and hole size in the SIMP framework [Sigmund 2009; Zhou et al. 2015]. Combining these approaches to ours is left as future work.

7 Conclusion

Our work enables a novel way to design shapes that are rigid under a set of external conditions. It offers an unprecedented control over appearance through the specification of an exemplar. Rigidity is understood as a constraint affording for a simple and predictable control on the tradeoff between appearance and structural properties.

We envision that expert users will use our technique to quickly produce initial designs serving as a starting point, while non-expert users will explore a large variety of appearances for objects having the same mechanical purpose.

There are several avenues of future work. First, our technique does not scale well to dense 3D problems which are almost impractical due to their large computational cost. Second, some patterns have appearance that complicates the task of achieving a rigid structure, as illustrated in Figure 16. While this is currently a limitation we believe that integrating the pattern orientation as an optimization variable will help on the rightmost exemplar, allowing features to align with the local directions of stress. Finally, we are looking forward to explore design tools exploiting our technique.

We hope to bring a novel modeling tool that will empower users — experts or otherwise — to produce shapes that are unique, visually appealing, and yet structurally sound for a given usage scenario.

Acknowledgements

We would like to thank anonymous reviewers for their valuable comments. This work was supported by ERC grant ShapeForge (StG-2012-307877) and general research fund *Dynamic Element Textures* (HKU 717112E).

References

- ANDREASSEN, E., CLAUSEN, A., SCHEVENELS, M., LAZAROV, B., AND SIGMUND, O. 2011. Efficient topology optimization in Matlab using 88 lines of code. *Struct. Multidiscip. Optim.* 43, 1, 1–16.
- BARNES, C., SHECHTMAN, E., FINKELSTEIN, A., AND GOLDMAN, D. B. 2009. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* 28, 3, 24:1–24:11.
- BENDSØE, M. P., AND KIKUCHI, N. 1988. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering* 71, 2, 197–224.
- BENDSØE, M. P., AND SIGMUND, O. 2003. *Topology Optimization: Theory, Methods and Applications*.
- BENDSØE, M. P. 1989. Optimal shape design as a material distribution problem. *Structural Optimization* 1, 192–202.
- BRACKETT, D., ASHCROFT, I., AND HAGUE, R. 2011. Topology optimization for additive manufacturing. In *Proc. of the 24th Solid Freeform Fabrication Symposium*, 6–8.
- BRUYNEEL, M., AND DUYSINX, P. 2005. Note on topology optimization of continuum structures including self-weight. *Struct. Multidiscip. Optim.* 29, 4, 245–256.
- BUSTO, P. P., EISENACHER, C., LEFEBVRE, S., AND STAMMINGER, M. 2010. Instant texture synthesis by numbers. In *Proc. of the VMV Workshop*, 81–85.
- CHRISTIANSEN, A. N., BÆRENTZEN, J. A., NOBEL-JØRGENSEN, M., AAGE, N., AND SIGMUND, O. 2015. Combined shape and topology optimization of 3D structures. *Computers & Graphics* 46, 25 – 35.
- DEATON, J., AND GRANDHI, R. 2014. A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Struct. Multidiscip. Optim.* 49, 1, 1–38.
- DUMAS, J., LU, A., LEFEBVRE, S., WU, J., AND DICK, C. 2015. By-Example Synthesis of Structurally Sound Patterns. *ACM Trans. Graph.* 34, 4, 137:1–137:12.

- DUYSINX, P., AND BENDSØE, M. P. 1998. Topology optimization of continuum structures with local stress constraints. *Int. J. Numer. Methods. Eng.* 43, 8, 1453–1478.
- EFROS, A. A., AND LEUNG, T. K. 1999. Texture synthesis by non-parametric sampling. In *Proceedings of the International Conference on Computer Vision*, 1033–1038.
- HAN, Z., LIU, Z., HAN, J., AND BU, S. 2015. 3D shape creation by style transfer. *Vis. Comput.* 31, 9, 1147–1161.
- HERTZMANN, A., JACOBS, C. E., OLIVER, N., CURLESS, B., AND SALESIN, D. H. 2001. Image analogies. In *Proc. of SIGGRAPH 2001*, 327–340.
- HOLMBERG, E., TORSTENFELT, B., AND KLARBRING, A. 2013. Stress constrained topology optimization. *Struct. Multidiscip. Optim.* 48, 1, 33–47.
- JOHNSON, S. G., 2007. The NLOpt nonlinear-optimization package.
- KASPAR, A., NEUBERT, B., LISCHINSKI, D., PAULY, M., AND KOPF, J. 2015. Self tuning texture optimization. *Computer Graphics Forum* 34, 2, 349–359.
- KOPF, J., FU, C.-W., COHEN-OR, D., DEUSSEN, O., LISCHINSKI, D., AND WONG, T.-T. 2007. Solid texture synthesis from 2D exemplars. *ACM Trans. Graph.* 26, 3.
- KOSAKA, I., AND SWAN, C. C. 1999. A symmetry reduction method for continuum structural topology optimization. *Computers & Structures* 70, 1, 47 – 61.
- KWATRA, V., ESSA, I., BOBICK, A., AND KWATRA, N. 2005. Texture optimization for example-based synthesis. *ACM Trans. Graph.* 24, 3, 795–802.
- LEE, E., JAMES, K. A., AND MARTINS, J. R. 2012. Stress-constrained topology optimization with design-dependent loading. *Struct. Multidiscip. Optim.* 46, 5, 647–661.
- LI, H., ZHANG, H., WANG, Y., CAO, J., SHAMIR, A., AND COHEN-OR, D. 2013. Curve style analysis in a set of shapes. *Computer Graphics Forum* 32, 6, 77–88.
- LU, L., SHARF, A., ZHAO, H., WEI, Y., FAN, Q., CHEN, X., SAVOYE, Y., TU, C., COHEN-OR, D., AND CHEN, B. 2014. Build-to-last: Strength to weight 3D printed objects. *ACM Trans. Graph.* 33, 4, 97:1–97:10.
- MA, C., HUANG, H., SHEFFER, A., KALOGERAKIS, E., AND WANG, R. 2014. Analogy-driven 3D style transfer. *Computer Graphics Forum* 33, 2, 175–184.
- PANETTA, J., ZHOU, Q., MALOMO, L., PIETRONI, N., CIGNONI, P., AND ZORIN, D. 2015. Elastic textures for additive fabrication. *ACM Trans. Graph.* 34, 4, 135:1–135:12.
- PARÍS, J., MUÍÑOS, I., NAVARRINA, F., COLOMINAS, I., AND CASTELEIRO, M. 2005. A minimum weight FEM formulation for structural topological optimization with local stress constraints. In *VI World Congress on Structural and Multidisciplinary Optimization*.
- PAULINO, G. H., AND GAIN, A. L. 2015. Bridging art and engineering using Escher-based virtual elements. *Struct. Multidiscip. Optim.* 51, 4, 867–883.
- PEDERSEN, N. 2000. Maximization of eigenvalues using topology optimization. *Struct. Mult. Optim.* 20, 1, 2–11.
- SCHUMACHER, C., BICKEL, B., RYS, J., MARSCHNER, S., DARAIO, C., AND GROSS, M. 2015. Microstructures to control elasticity in 3D printing. *ACM Trans. Graph.* 34, 4, 136:1–136:13.
- SIGMUND, O., AND MAUTE, K. 2013. Topology optimization approaches: A comparative review. *Struct. Multidiscip. Optim.* 48, 6, 1031–1055.
- SIGMUND, O. 1997. On the design of compliant mechanisms using topology optimization. *Mechanics of Structures and Machines* 25, 4, 493–524.
- SIGMUND, O. 2001. A 99 line topology optimization code written in Matlab. *Struct. Mult. Optim.* 21, 2, 120–127.
- SIGMUND, O. 2009. Manufacturing tolerant topology optimization. *Acta Mechanica Sinica* 25, 2, 227–239.
- STAVA, O., VANEK, J., BENES, B., CARR, N., AND MĚCH, R. 2012. Stress relief: Improving structural strength of 3D printable objects. *ACM Trans. Graph.* 31, 4, 48:1–48:11.
- SVANBERG, K. 1987. The method of moving asymptotes—a new method for structural optimization. *Int. J. Numer. Methods. Eng.* 24, 2, 359–373.
- SVANBERG, K. 1995. A globally convergent version of MMA without linesearch. In *Proc. of the first world congress of structural and multidisciplinary optimization*, 9–16.
- UMETANI, N., AND SCHMIDT, R. 2013. Cross-sectional structural analysis for 3D printing optimization. In *SIGGRAPH Asia 2013 Technical Briefs*, 5:1–5:4.
- WANG, W., WANG, T. Y., YANG, Z., LIU, L., TONG, X., TONG, W., DENG, J., CHEN, F., AND LIU, X. 2013. Cost-effective printing of 3D objects with skin-frame structures. *ACM Trans. Graph.* 32, 6, 177:1–177:10.
- WEI, L.-Y., AND LEVOY, M. 2000. Fast texture synthesis using tree-structured vector quantization. In *Proc. of SIGGRAPH 2000*, 479–488.
- WEI, L.-Y., LEFEBVRE, S., KWATRA, V., AND TURK, G. 2009. State of the art in example-based texture synthesis. In *Eurographics 2009, State of the Art Report*.
- XU, K., LI, H., ZHANG, H., COHEN-OR, D., XIONG, Y., AND CHENG, Z. 2010. Style-content separation by anisotropic part scales. *ACM Trans. Graph.* 29, 6, 184:1–184:10.
- XU, K., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2012. Fit and diverse: Set evolution for inspiring 3D shape galleries. *ACM Trans. Graph.* 31, 4, 57:1–57:10.
- ZHOU, Q., PANETTA, J., AND ZORIN, D. 2013. Worst-case structural analysis. *ACM Trans. Graph.* 32, 4, 137:1–137:12.
- ZHOU, S., JIANG, C., AND LEFEBVRE, S. 2014. Topology-constrained synthesis of vector patterns. *ACM Trans. Graph.* 33, 6, 215:1–215:11.
- ZHOU, M., LAZAROV, B. S., WANG, F., AND SIGMUND, O. 2015. Minimum length scale in topology optimization by geometric constraints. *Computer Methods in Applied Mechanics and Engineering* 293, 266 – 282.