Freely orientable microstructures for designing deformable 3D prints

THIBAULT TRICARD, Université de Lorraine, CNRS, Inria, LORIA
VINCENT TAVERNIER, Université Grenoble Alpes, CNRS, Laboratoire Jean Kuntzmann, Inria
CÉDRIC ZANNI, Université de Lorraine, CNRS, Inria, LORIA
JONÀS MARTÍNEZ, Université de Lorraine, CNRS, Inria, LORIA
PIERRE-ALEXANDRE HUGRON, Université de Lorraine, CNRS, Inria, LORIA
FABRICE NEYRET, Université Grenoble Alpes, CNRS, Laboratoire Jean Kuntzmann, Inria
SYLVAIN LEFEBVRE, Université de Lorraine, CNRS, Inria, LORIA

ACM Reference Format:
Thibault Tricard, Vincent Tavernier, Cédric Zanni, Jonàs Martínez, Pierre-Alexandre Hugron, Fabrice Neyret, and Sylvain Lefebvre. 2020. Freely orientable microstructures for designing deformable 3D prints. *ACM Trans. Graph.* 39, 6, Article 211 (December 2020), 8 pages. https://doi.org/10.1145/3414685.3417790

In this supplemental material we provide additional explanations and derivations for several of the equations provided in paper. A summary of the notation used is given in Table 2. Section 1 provides an analysis of the self-supportability of our structure, Section 2 describes our implementation of the phasor noise, Section 3 describes singularities and how to detect and quantify them, Section 4 adds more detail to the singularities filtering process, Section 5 provides an analysis of the singularity reduction process, and Section 6 provides an analysis of the mechanical properties of the generated structure.

supplemental sections	referenced article section
Sec 1 Self-supportability analysis	Sec 3.2.3
Sec 2 Phasor noise implementation	Sec 3.2.1
Sec 3 Singularity energy	Sec 4.2.1
Sec 4 Filtering singularities	Sec 4.1 and 4.2.2
Sec 5 Results of singularity reduction	Sec 4.3
Sec 6 Mechanical analysis	Sec 5

Table 1. List of the section of the supplemental and the part of the article they refer to.

Symbol	Definition	Introduced
$\mathbf{d_x}$	direction at $\mathbf{x}$	
$f_\mathbf{x}$	frequency at $\mathbf{x}$	
$\mathbf{f_x}$	combined direction and frequency at $\mathbf{x}$	
$\Gamma_\mathbf{x}$	angular spread at $\mathbf{x}$	
$\mathcal{G}(\mathbf{x})$	complex Gabor noise	Eqn. (2)
$\phi(\mathbf{x})$	phasor noise	Eqn. (1)
$p_{\mathbf{v},\mathbf{w}}(\mathbf{x})$	phasor sawtooth	Eqn. (3)
$\varphi(\mathbf{x})$	phase field	Eqn. (5)
$\omega_{\mathbf{f},\mathbf{c}}(\mathbf{x})$	complex expontential oscilator	Eqn. (4)
k_j	complex Gabor kernel j	Eqn. (2)
b	bandwith of the noise and Gaussian	Eqn. (2)
$\mathbf{x}_j$	center of kernel j	Eqn. (2)
f_j	frequency of kernel j	Eqn. (2)
$\mathbf{d}_j$	direction of anisotropy of kernel j	Eqn. (2)
φ_j	phase of kernel j	Eqn. (2)
$\mathcal{N}(k_i)$	set of kernels interacting with kernel i	Alg. (1)
$\times$	cross product	
$\circledast$	convolution product	
$\cdot$	dot product	

Table 2. Summary of the main notations.

1 SELF-SUPPORTABILITY ANALYSIS

Even though our implementation does not consider it, there is, in fact, additional freedom under overhang constraints.

Any given frame (hence extruded diamonds) can be described by the application of three rotations around fixed axis x, y, z to an initial base frame $(\mathbf{e}_x(1\,0\,0), \mathbf{e}_y(0\,1\,0), \mathbf{e}_z(0\,0\,1))$ corresponding to a diamond in the xy-plane, extruded along y and printed in the direction z. Rotation are first *roll* of angle θ around the y-axis, then *pitch* of angle α around the x-axis and finally *yaw* of angle γ around the z-axis.

Only the two first rotations (roll and pitch) affect the self-supportability of the extruded diamond structures. Indeed the yaw rotation does not change the z coordinate of rotated vector. Furthermore, if the roll rotation is ignored the structure will always be self-supported; however, this does reduce the degrees of freedom provided to the

Authors' addresses: Thibault Tricard, Université de Lorraine, CNRS, Inria, LORIA; Vincent Tavernier, Université Grenoble Alpes, CNRS, Laboratoire Jean Kuntzmann, Inria; Cédric Zanni, Université de Lorraine, CNRS, Inria, LORIA; Jonàs Martínez, Université de Lorraine, CNRS, Inria, LORIA; Pierre-Alexandre Hugron, Université de Lorraine, CNRS, Inria, LORIA; Fabrice Neyret, Université Grenoble Alpes, CNRS, Laboratoire Jean Kuntzmann, Inria; Sylvain Lefebvre, Université de Lorraine, CNRS, Inria, LORIA.

0730-0301/2020/12-ART211 $15.00
https://doi.org/10.1145/3414685.3417790

user as the mechanical behavior of the structure is not fully isotropic in the plane orthogonal to the direction of extrusion.

In order to represent all possible extruded diamond structures, it is possible to constraint rotation angles to the following ranges:

- constraining yaw rotation to $]-\pi, \pi]$
- constraining pitch rotation α to $[0, \frac{\pi}{2}]$
- constraining roll rotation θ to $[-\frac{\pi}{4}, \frac{\pi}{4}]$

We denote vector rotation around a given axis as:

$$R_{\mathbf{e}}(\gamma)\mathbf{u} = (\mathbf{e}^T\mathbf{u})\mathbf{e} + \cos\gamma((\mathbf{e}\times\mathbf{u})\times\mathbf{e}) + \sin\gamma(\mathbf{e}\times\mathbf{u}) \qquad (1)$$

Let us derive the maximum roll angle θ in function of the pitch rotation angle α. In order to do so, we need to derive a formula for the normal of the walls of the structure after application of the two first rotations.

Before rotation, supporting normal vectors are:

$$\mathbf{n}_\pm = \frac{\sqrt{2}}{2}(\pm 1, 0, -1)$$

which become:

$$R_{\mathbf{e_x}}(\alpha)\, R_{\mathbf{e_y}}(\theta)\, \mathbf{n}_\pm$$

after rotation (due to range of angle previously defined for "roll" angle θ and "pitch" angle α). Note that the constraint on self-supportability coming from $\mathbf{n}_-$ is the same as the one coming from $\mathbf{n}_+$ with the rotation angle $-\theta$. Let us denote $\mathbf{n} = \mathbf{n}_+$ for simplicity.

Hence, in order to satisfy self-supportability, the following inequality should be verified:

$$\min(\, \mathbf{e_z}\cdot R_{\mathbf{e_x}}(\alpha)R_{\mathbf{e_y}}(\theta)\,\mathbf{n}, \mathbf{e_z}\cdot R_{\mathbf{e_x}}(\alpha)R_{\mathbf{e_y}}(-\theta)\,\mathbf{n}\,) > \epsilon \qquad (2)$$

with a limit $\epsilon \in [-1, 0[$ and with:

$$R_{\mathbf{e_x}}(\alpha)R_{\mathbf{e_y}}(\theta)\mathbf{n} = \frac{\sqrt{2}}{2}\begin{pmatrix} \cos\theta - \sin\theta \\ (\cos\theta + \sin\theta)\sin\alpha \\ -(\cos\theta + \sin\theta)\cos\alpha \end{pmatrix}$$

Equation (2) is equivalent to:

$$\frac{\sqrt{2}}{2}\min\left(-(\cos\theta + \sin\theta)\cos\alpha, -(\cos -\theta + \sin -\theta)\cos\alpha\right) > \epsilon$$

and, given the range of value for α (which guarantee $\cos\alpha \geq 0$), to:

$$\max(\cos\theta + \sin\theta, \cos\theta - \sin\theta) < -\sqrt{2}\frac{\epsilon}{\cos\alpha}$$

$$\cos\theta + |\sin\theta| < -\sqrt{2}\frac{\epsilon}{\cos\alpha}$$

Given that all interesting ϵ limit values are negatives and given the range of value for θ (which guarantee $\cos\theta \geq 0$), the inequality is equivalent to:

$$(\cos\theta + |\sin\theta|)^2 < 2\frac{\epsilon^2}{\cos^2\alpha}$$

$$1 + 2\cos\theta|\sin\theta| < 2\frac{\epsilon^2}{\cos^2\alpha}$$

$$2\cos\theta|\sin\theta| < 2\frac{\epsilon^2}{\cos^2\alpha} - 1$$

and on the range of interest:

$$|\sin 2\theta| < 2\frac{\epsilon^2}{\cos^2\alpha} - 1$$

Finally, if $\cos^2\alpha > \epsilon^2$ the constraint becomes:

$$|\theta| < \frac{1}{2}\arcsin\left(2\frac{\epsilon^2}{\cos^2\alpha} - 1\right)$$

otherwise the constraint is always verified.

Now, if we rotate the diamond cell around the horizontal diagonal of the base diamond shape, the wall angles with respect to the horizontal plane actually increase (see Figure 7, right). Thus, any such rotated cell remains well within overhang constraints. (Of course, the bottom edge along the extrusion direction is unsupported in this discussion; however in the final structure it is supported from below by another diamond cell.)

By composition of the two previous types of rotations, all possible directions for the transverse axis can be achieved. We obtain the two diamond diagonal directions as $\mathbf{v} = \mathbf{z}\times\mathbf{u}$ and $\mathbf{w} = (\mathbf{z}\times\mathbf{u})\times\mathbf{u}$.

2 PHASOR NOISE IMPLEMENTATION

We implement our phasor noises as follows. The positions of the kernels ($\mathbf{x}_j$) are obtained from a random sampling providing a dense spatial coverage of the domain. In practice, we use a grid approach as in [**?**], with a constant number of kernels per cell (8). The position of a kernel is initialized with a uniform random value within the virtual grid cell. For each kernel, f_j and $\mathbf{d}_j$ are sampled from the frame field F (using $\mathbf{v}$ or $\mathbf{w}$). f_j is given by the inverse scales of respectively $\mathbf{v}$, $\mathbf{w}$. The rotation angle for $p_\mathbf{v}$ is uniformly sampled in $[-\Gamma_x, \Gamma_x]$. φ_j is initially set to 0. In a phasor noise the size of the kernels is defined by the bandwidth of the noise. The choice of bandwidth impacts the size of the kernels in the spatial domain, with a larger bandwidth resulting in smaller kernels. We choose the bandwidth large enough so that the kernels are small enough to properly reproduce rapid spatial changes of the user fields. However, we keep the kernels large enough to not introduce changes smaller than the structure feature size, which would break the pattern. Using a large bandwidth would also induce an increase in the total number of kernels to fill the same target volume, thus increasing the computational cost of the method.

The frequency being a direct indicator of the final structure feature size, we use the minimal scale to determine the bandwidth. In practice, we use $b = \frac{c}{F_{min}}$ with $c = 4.5$ which gave the best results based on our experiments.

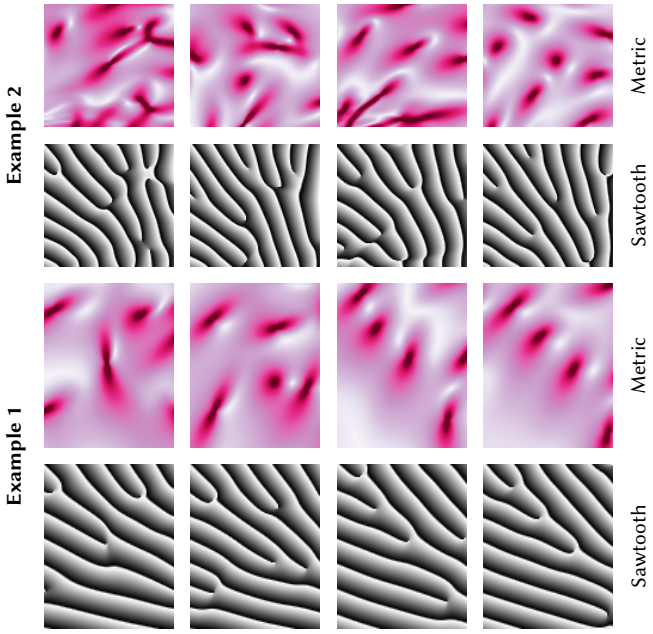
3 SINGULARITY ENERGY

We define the *local singularity measure* $\mathcal{S}(\mathbf{x})$ at location $\mathbf{x}$ as:

$$\mathcal{S}(\mathbf{x}) = \frac{||\nabla \operatorname{Arg}(\varphi_\mathbf{x}(\boldsymbol{\delta}))||}{2\pi F(\mathbf{x})} \qquad (3)$$

Note that the gradient operator is applied to the local phasor field around $\mathbf{x}$, differentiating with respect to the offset variable $\boldsymbol{\delta}$. The singularity measure is dimensionless and represents the relative phase change speed, as it is normalized by the local frequency $F(\mathbf{x})$, following Remark 2. In our implementation — which is used only for analysis — the gradient operator is computed using finite differences on the generated images of $\operatorname{Arg}(\varphi_\mathbf{x}(\boldsymbol{\delta}))$ (taking into account the 2π wrapping).

$\mathcal{S}(\mathbf{x})$ provides a strong spatial indicator of the two different families of singularities, points and curves, as shown in Figure 1.



(a) Original (b) Filtered (c) Phase alignment (d) Both

Fig. 1. Singularity measure $\mathcal{S}(\mathbf{x})$. Darker spots in pink metric images indicate areas with a high $\mathcal{S}(\mathbf{x})$ value. Aligning kernel phases through regularization outright removes curve singularities and some point singularities (c). The filter reduces the extent of remaining singularities to turn them as much as possible into point singularities with a smooth phasor field gradient (b). The best results are obtained by combining both approaches (d).

In order to objectively measure the number of singularities (point or curves) in a result pattern, we introduce for an image domain Ω the *singularity energy*:

$$E_{\mathcal{S}}(q) = \frac{|\{\mathcal{S}(\mathbf{x}) > q, \mathbf{x} \in \Omega\}|}{|\Omega|} \tag{4}$$

The *singularity energy* effectively counts how much surface area of a given image is covered by singularities. The threshold q is to be determined visually depending on how much profile distortion should be considered a singularity. In all our examples we chose a demanding threshold of $q = 2$. In other words, all points whose phase shift is twice the frequency of the noise are considered part of a singularity.

Equipped with this measure, we can now analyze the effect of our regularization and filter.

4 FILTERING SINGULARITIES

4.1 Algorithm

At each iteration, the algorithm adjusts the phase ϕ_j of each kernel such that it complies better with the phases of its neighbors. As we manipulate complex numbers, for a kernel k_j this amounts to evaluating the oscillator of each neighboring kernel at $\mathbf{x}_j$. Aligning phases between kernels having large differences in orientation is not sound. Thus, the average is weighted by the dot product of orientation vectors between k_j and its neighbor, clamped to zero. This weighting scheme accommodates in particular for the varying angular spread Γ. We then take the phase (complex argument) of the average oscillator.

The algorithm for one iteration is given in Algorithm 1 in the main material. It trivially maps to an efficient GPU implementation, with one thread per kernel. The neighborhood $\mathcal{N}(k_j)$ of kernels interacting with k_j is efficiently retrieved from the kernel grid, as is done when evaluating a Gabor (or phasor) noise [**?**]. In addition, this neighborhood is the same for all k_j in a same grid cell, enabling an efficient pre-fetching in shared memory of the set of neighbor kernels.

4.2 Closed-form filter derivation

In order to provide an efficient computation of the filter used to regularize a signal defined by a complex Gabor noise, we derive here the closed-form expression of the result of the filter.

We start from Equation (9) of the paper:

$$\begin{aligned}\mathscr{F}(\mathbf{x}) &= (g_a \circledast \varphi_{\mathbf{x}})(\mathbf{x}) \\ &= \left(g_a \circledast \frac{\mathcal{G}}{\omega_{\mathbf{f}_{\mathbf{x}},\mathbf{x}}}\right)(\mathbf{x})\end{aligned} \tag{5}$$

with $\mathcal{G}$ a complex Gabor noise defined as:

$$\mathcal{G}(\mathbf{x}) = \sum_j e^{-b\|\mathbf{x}-\mathbf{x}_j\|^2} e^{2i\pi \mathbf{f}_j \cdot (\mathbf{x}-\mathbf{x}_j) + i\varphi_j} \tag{6}$$

with Gaussians defined as :

$$g_a(\mathbf{x}) = e^{-a\|\mathbf{x}-\mathbf{x}_j\|^2}$$

and oscillators defined as:

$$\omega_{\mathbf{f},\mathbf{c}}(\mathbf{x}) = e^{2i\pi \mathbf{f}\cdot(\mathbf{x}-\mathbf{c})}$$

For the sake of compactness, we denote here frequencies and directions of anisotropy as single two-dimensional parameters ($\mathbf{f}_{\mathbf{x}}$, $\mathbf{f}_j$ and $\mathbf{f}$ in above formulas).

Our objective is to reformulate the filtered noise as a convolution noise. For this purpose, let us first re-write the right parameter of the convolution product of equation (5) as a convolution with a set of Dirac delta functions $\{\delta_{\mathbf{x}_j}\}_j$. Given any point $\mathbf{y}$ in $\mathbb{R}^3$, we have:

$$\begin{aligned}\frac{\mathcal{G}}{\omega_{\mathbf{f}_{\mathbf{y}},\mathbf{y}}}(\mathbf{x}) &= e^{-2i\pi \mathbf{f}_{\mathbf{y}}\cdot(\mathbf{x}-\mathbf{y})} \sum_j e^{-b\|\mathbf{x}-\mathbf{x}_j\|^2} e^{2i\pi \mathbf{f}_j\cdot(\mathbf{x}-\mathbf{x}_j)} e^{i\varphi_j} \\ &= \sum_j e^{-2i\pi \mathbf{f}_{\mathbf{y}}\cdot((\mathbf{x}-\mathbf{x}_j)+(\mathbf{x}_j-\mathbf{y}))}\, e^{-b\|\mathbf{x}-\mathbf{x}_j\|^2}\, e^{2i\pi \mathbf{f}_j\cdot(\mathbf{x}-\mathbf{x}_j)}\, e^{i\varphi_j} \\ &= \sum_j e^{2i\pi(\mathbf{f}_j-\mathbf{f}_{\mathbf{y}})\cdot(\mathbf{x}-\mathbf{x}_j)}\, e^{-b\|\mathbf{x}-\mathbf{x}_j\|^2}\, e^{-2i\pi \mathbf{f}_{\mathbf{y}}\cdot(\mathbf{x}_j-\mathbf{y})}\, e^{i\varphi_j} \\ &= \sum_j \omega_{\mathbf{f}_j-\mathbf{f}_{\mathbf{y}},0}(\mathbf{x}-\mathbf{x}_j)\, g_b(\mathbf{x}-\mathbf{x}_j)\, \omega_{\mathbf{f}_{\mathbf{y}},\mathbf{x}_j}(\mathbf{y})\, e^{i\varphi_j} \\ &= \sum_j \left(\left(g_b\, \omega_{\mathbf{f}_j-\mathbf{f}_{\mathbf{y}},0}\right) \circledast \left(\omega_{\mathbf{f}_{\mathbf{y}},\mathbf{x}_j}(\mathbf{y})\, e^{i\varphi_j} \delta_{\mathbf{x}_j}\right)\right)(\mathbf{x})\end{aligned} \tag{7}$$

By injecting equation (7) in equation (5), we obtain :

$$\mathscr{F} = g_a \circledast \left(\sum_j \left(g_b \, \omega_{\mathbf{f}_j - \mathbf{f}_y, 0} \right) \circledast \left(\omega_{\mathbf{f}_y, \mathbf{x}_j}(\mathbf{y}) \, e^{\mathrm{i}\varphi_j} \delta_{\mathbf{x}_j} \right) \right)$$

Applying first distributivity then associativity of the convolution product, we obtain:

$$\mathscr{F} = \sum_j \left(g_a \circledast \left(g_b \, \omega_{\mathbf{f}_j - \mathbf{f}_y, 0} \right) \right) \circledast \left(\omega_{\mathbf{f}_y, \mathbf{x}_j}(\mathbf{y}) \, e^{\mathrm{i}\varphi_j} \delta_{\mathbf{x}_j} \right) \tag{8}$$

which is a sum of convolution between a new family of parametrized kernel and a set of Dirac delta functions weighted by complex numbers. The next step is to verify that a closed-form expression for the following kernel can be derived :

$$G_j = g_a \circledast \left(g_b \, \omega_{\mathbf{f}_j - \mathbf{f}_y, 0} \right) \tag{9}$$

Let us take a look at the expanded expression of G_j:

$$\begin{aligned} G_j(\mathbf{x}) &= \iiint_{\mathbb{R}^3} \left(g_b \, \omega_{\mathbf{f}_j - \mathbf{f}_y, 0} \right)(\mathbf{x} - \mathbf{u}) \; g_a(\mathbf{u}) \; d\mathbf{u} \\ &= \iiint_{\mathbb{R}^3} g_b(\mathbf{x} - \mathbf{u}) \; g_a(\mathbf{u}) \; \omega_{\mathbf{f}_j - \mathbf{f}_y, 0}(\mathbf{x} - \mathbf{u}) \; d\mathbf{u} \end{aligned} \tag{10}$$

The product of the two Gaussians can be reformulated as :

$$g_b(\mathbf{x} - \mathbf{u}) \; g_a(\mathbf{u}) = g_{\frac{ab}{a+b}}(\mathbf{x}) \; g_{a+b}(\mathbf{u} - \frac{b}{a+b}\mathbf{x}) \tag{11}$$

hence, we have:

$$G_j(\mathbf{x}) = g_{\frac{ab}{a+b}}(\mathbf{x}) \iiint_{\mathbb{R}^3} g_{a+b}(\mathbf{u} - \frac{b}{a+b}\mathbf{x}) \; \omega_{\mathbf{f}_j - \mathbf{f}_y, 0}(\mathbf{x} - \mathbf{u}) \; d\mathbf{u}$$

applying a change of variable $\mathbf{u} := \mathbf{u} - \frac{b}{a+b}\mathbf{x}$, we obtain:

$$\begin{aligned} G_j(\mathbf{x}) &= g_{\frac{ab}{a+b}}(\mathbf{x}) \iiint_{\mathbb{R}^3} g_{a+b}(\mathbf{u}) \; \omega_{\mathbf{f}_j - \mathbf{f}_y, 0}(\frac{a}{a+b}\mathbf{x} - \mathbf{u}) \; d\mathbf{u} \\ &= g_{\frac{ab}{a+b}}(\mathbf{x}) \omega_{\mathbf{f}_j - \mathbf{f}_y, 0}(\frac{a}{a+b}\mathbf{x}) \iiint_{\mathbb{R}^3} g_{a+b}(\mathbf{u}) \; \omega_{\mathbf{f}_y - \mathbf{f}_j, 0}(\mathbf{u}) \; d\mathbf{u} \end{aligned}$$

In order to compute the remaining integral term, we only need to reformulate the 3D Gaussian as a product between a 1D Gaussian in the direction of the oscillator and a 2D Gaussian in the orthogonal plane:

$$\begin{aligned} I &= \iiint_{\mathbb{R}^3} g_{a+b}(\mathbf{u}) \; \omega_{\mathbf{f}_y - \mathbf{f}_j, 0}(\mathbf{u}) \; d\mathbf{u} \\ &= \int_{\mathbb{R}} \iint_{\mathbb{R}^2} g_{a+b}(\mathbf{v}) \; g_{a+b}(r) \; \omega_{\|\mathbf{f}_y - \mathbf{f}_j\|, 0}(r) \; d\mathbf{v} \, dr \\ &= \iint_{\mathbb{R}^2} g_{a+b}(\mathbf{v}) \; d\mathbf{v} \int_{\mathbb{R}} g_{a+b}(r) \; \omega_{\|\mathbf{f}_y - \mathbf{f}_j\|, 0}(r) \; dr \end{aligned}$$

Both integrals have closed-form expressions:

$$\iint_{\mathbb{R}^2} g_c(\mathbf{v}) \; d\mathbf{v} = \frac{\pi}{c}$$

$$\int_{\mathbb{R}} e^{-cx^2 + \mathrm{i}Jx} \; dx = \left(\frac{\pi}{c} \right)^{\frac{1}{2}} e^{-\frac{J^2}{4c}}$$

Hence, we obtain:

$$I = \left(\frac{\pi}{a+b} \right)^{\frac{3}{2}} e^{-\pi^2 \frac{\|\mathbf{f}_y - \mathbf{f}_j\|^2}{a+b}}$$

Therefore the formula for the new kernel family is:

$$\begin{aligned} G_j(\mathbf{x}) &= \left(\frac{\pi}{a+b} \right)^{\frac{3}{2}} g_{\frac{ab}{a+b}}(\mathbf{x}) \; \omega_{\mathbf{f}_j - \mathbf{f}_y, 0}(\frac{a}{a+b}\mathbf{x}) \; e^{-\pi^2 \frac{\|\mathbf{f}_y - \mathbf{f}_j\|^2}{a+b}} \\ &= \left(\frac{\pi}{a+b} \right)^{\frac{3}{2}} e^{-\frac{ab}{a+b}\|\mathbf{x}\|^2} e^{2\mathrm{i}\pi(\mathbf{f}_j - \mathbf{f}_y) \cdot (\frac{a}{a+b}\mathbf{x})} e^{-\pi^2 \frac{\|\mathbf{f}_y - \mathbf{f}_j\|^2}{a+b}} \end{aligned} \tag{12}$$

Let us define $C = \left(\frac{\pi}{a+b} \right)^{\frac{3}{2}}$. If we inject equation (12) in equation (8) of the filtered noise, we obtain:

$$\begin{aligned} \mathscr{F}(\mathbf{x}) &= \sum_j G_j \circledast \left(\omega_{\mathbf{f}_y, \mathbf{x}_j}(\mathbf{y}) e^{\mathrm{i}\varphi_j} \delta_{\mathbf{x}_j} \right) \\ &= \sum_j \left(C \, e^{-\frac{ab}{a+b}\|\mathbf{x}\|^2} \, e^{2\mathrm{i}\pi(\mathbf{f}_j - \mathbf{f}_y) \cdot (\frac{a}{a+b}\mathbf{x})} \, e^{-\pi^2 \frac{\|\mathbf{f}_y - \mathbf{f}_j\|^2}{a+b}} \right) \\ &\quad \circledast \left(\omega_{\mathbf{f}_y, \mathbf{x}_j}(\mathbf{y}) \, e^{\mathrm{i}\varphi_j} \, \delta_{\mathbf{x}_j} \right) \\ &= C \sum_j \left(e^{-\frac{ab}{a+b}\|\mathbf{x} - \mathbf{x}_j\|^2} \, e^{2\mathrm{i}\pi(\mathbf{f}_j - \mathbf{f}_y) \cdot (\frac{a}{a+b}(\mathbf{x} - \mathbf{x}_j))} \, e^{-\pi^2 \frac{\|\mathbf{f}_y - \mathbf{f}_j\|^2}{a+b}} \right) \\ &\quad \cdot \left(e^{2\mathrm{i}\pi \mathbf{f}_y \cdot (\mathbf{y} - \mathbf{x}_j)} \, e^{\mathrm{i}\varphi_j} \right) \\ &= C \sum_j \; e^{-\frac{ab}{a+b}\|\mathbf{x} - \mathbf{x}_j\|^2} \, e^{-\pi^2 \frac{\|\mathbf{f}_y - \mathbf{f}_j\|^2}{a+b}} \\ &\quad \cdot e^{2\mathrm{i}\pi(\mathbf{f}_j - \mathbf{f}_y) \cdot (\frac{a}{a+b}(\mathbf{x} - \mathbf{x}_j))} \, e^{2\mathrm{i}\pi \mathbf{f}_y \cdot (\mathbf{y} - \mathbf{x}_j) + \mathrm{i}\varphi_j} \end{aligned} \tag{13}$$

In practice the filter is always evaluated at $\mathbf{x} = \mathbf{y}$, therefore we have:

$$\begin{aligned} \mathscr{F}(\mathbf{x}) &= C \sum_j \; e^{-\frac{ab}{a+b}\|\mathbf{x} - \mathbf{x}_j\|^2} e^{-\pi^2 \frac{\|\mathbf{f}_x - \mathbf{f}_j\|^2}{a+b}} \\ &\quad \cdot e^{2\mathrm{i}\pi(\mathbf{f}_j - \mathbf{f}_x) \cdot (\frac{a}{a+b}(\mathbf{x} - \mathbf{x}_j))} e^{2\mathrm{i}\pi \mathbf{f}_x \cdot (\mathbf{x} - \mathbf{x}_j) + \mathrm{i}\varphi_j} \\ &= C \sum_j e^{-\frac{ab}{a+b}\|\mathbf{x} - \mathbf{x}_j\|^2} \, e^{-\pi^2 \frac{\|\mathbf{f}_x - \mathbf{f}_j\|^2}{a+b}} \, e^{2\mathrm{i}\pi(\frac{a}{a+b}\mathbf{f}_j - \frac{b}{a+b}\mathbf{f}_x) \cdot (\mathbf{x} - \mathbf{x}_j) + \mathrm{i}\varphi_j} \end{aligned} \tag{14}$$

As we are only interested by the argument of this value, we can evaluate the filtered noise by summing the following kernels:

$$\mathcal{K}_j(\mathbf{x}) = e^{-\frac{ab}{a+b}\|\mathbf{x} - \mathbf{x}_j\|^2} \, e^{-\pi^2 \frac{\|\mathbf{f}_x - \mathbf{f}_j\|^2}{a+b}} \, e^{2\mathrm{i}\pi(\frac{a}{a+b}\mathbf{f}_j - \frac{b}{a+b}\mathbf{f}_x) \cdot (\mathbf{x} - \mathbf{x}_j) + \mathrm{i}\varphi_j}$$

which is equivalent to Equation (11) of the paper:

$$\mathcal{K}_j(\mathbf{x}) = e^{-\frac{ab}{a+b}\|\mathbf{x} - \mathbf{x}_j\|^2 - \frac{\pi^2 \|\Delta_{jx}\|^2}{a+b}} \, e^{2\mathrm{i}\left(\pi f_j \mathbf{d}_j \cdot (\mathbf{x} - \mathbf{x}_j) + \frac{a}{a+b}\Delta_{jx} \cdot (\mathbf{x} - \mathbf{x}_j)\right) + \mathrm{i}\varphi_j}$$

with

$$\Delta_{j\mathbf{x}} = f_j \mathbf{d}_j - f_x \mathbf{d}_x$$

4.3 Attenuation Factor

In this section we provide our formulation for the attenuation factor $A(x)$ we added to Δ_{jx}.

$$A(\mathbf{x}) = \operatorname{argmax} \left(\max \left(1 - \frac{\Gamma_y}{\pi/2}, \mathbf{d}_x \cdot \mathbf{d}_y \right) \right)$$

For all y where $\|x - y\|$ is inferior to size of a kernel. Note that we do not evaluate $A(x)$ for each pixel of each slice, as it would cost a lot and slow down the method, instead we pre-compute $A(x)$ and store it in a low resolution 3D texture (64^3).

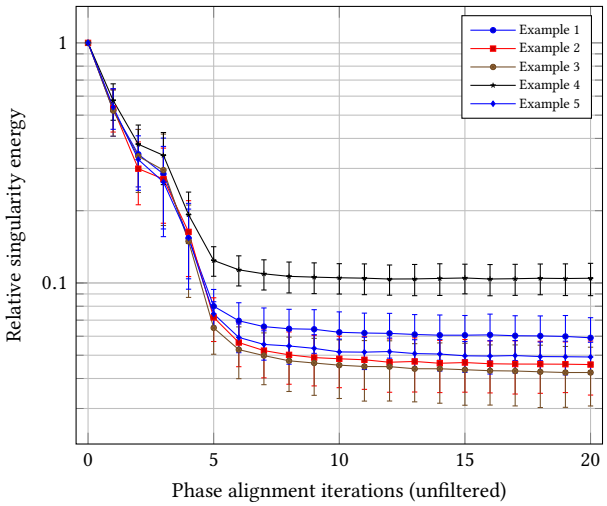


Fig. 2. Relative change in singularity energy as the iterative phase alignment converges. As the number of iterations increases, the singularity energy can be reduced by up to 95%. Each line uses a different set of parameters for generating the control fields, with each point being averaged over 50 different random seeds. This allows smoothing out variations due to specific instances in order to evaluate the average behavior of the phase alignment. The error bars show the standard deviation around the mean value among those seeds. The effect of the phase alignment is stable over varying parameters (examples have similar average behavior), and varying seeds (the standard deviation is small). Note the log scale.

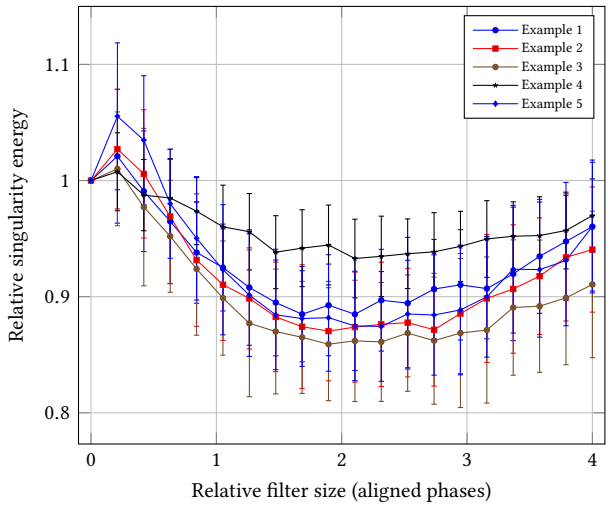


Fig. 3. Relative change of singularity energy as the filter size increases. The filter radius is defined relative to the noise kernel radius. All points are computed with 20 phase alignment iterations. As in Figure 2, results are averaged over 50 samples to reduce noise. The energy reduction reaches a minimum value at a relative filter size of 2 (the filter bandwidth is $1/\sqrt{2}$ the noise kernel bandwidth) on all examples. When the filter size increases beyond the tipping point, we tend toward a unique phase shift per direction of oscillation. Such a global choice of phase is not valid for a varying direction field, reducing the filter performance. The relative reduction in energy is weaker than what the regularization achieves (at best 15% on these examples) however this still improves the visual result significantly.

5 RESULTS OF SINGULARITY REDUCTION

In this section we have a closer look at the reduction of singularities achieved on various examples by each approach separately iterated phase alignment and filtering and then combined (oscillation regularization). We use the singularity energy (Section 3) to analyze the results.

5.1 Regularization convergence and filter size

There is a lower bound to the singularity energy for each example we consider, since there is no combination of phase shifts that could possibly align all kernels given a varying orientation field, in the general case. Figure 2 analyzes the reduction in singularity energy with more iterations of the phase alignment algorithm. We clearly see the plateau appear after a few iterations. Visual inspection of examples in Figure 14 of the paper reveals that a large number of singularities have been removed, and those that remain have shrunk toward points.

Even though the result is greatly improved by the phase alignment, there is still a noticeable amount of remaining distortion. By smoothing the phasor field using our filter, most of the remaining defects are removed and the area covered by singularities further shrinks.

Note that the energy reduction of the filter is smaller than what the regularization achieves, due to the choice of the threshold in the definition of F_S.

Figure 3 analyzes the filter effect for varying bandwidths a. This reveals a U-shaped curve. Figure 2 and Figure 3 allow us to choose

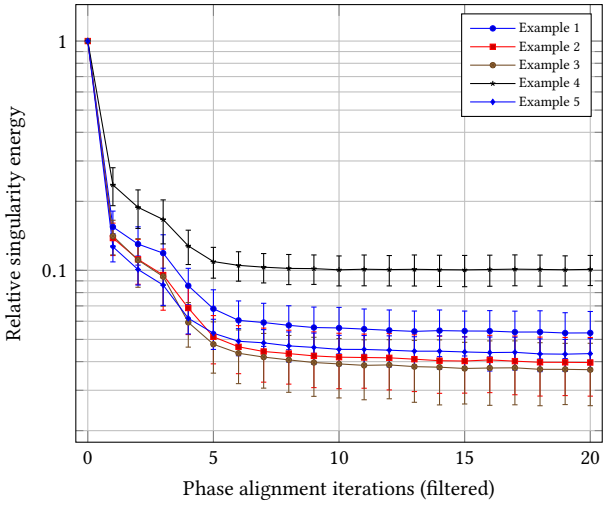


Fig. 4. Relative change of singularity energy as the phase alignment converges, while the evaluation filter bandwidth is set to its optimal value. The results are similar to Figure 2, but the singularity energy reduction converges to 96% of the initial value.

the number of regularization iterations and filtering bandwidth. We use 20 iterations in all our examples. Since the filter smoothes discontinuities in the phasor field, its bandwidth should be determined by the spatial dimensions of singularities. However the filter should

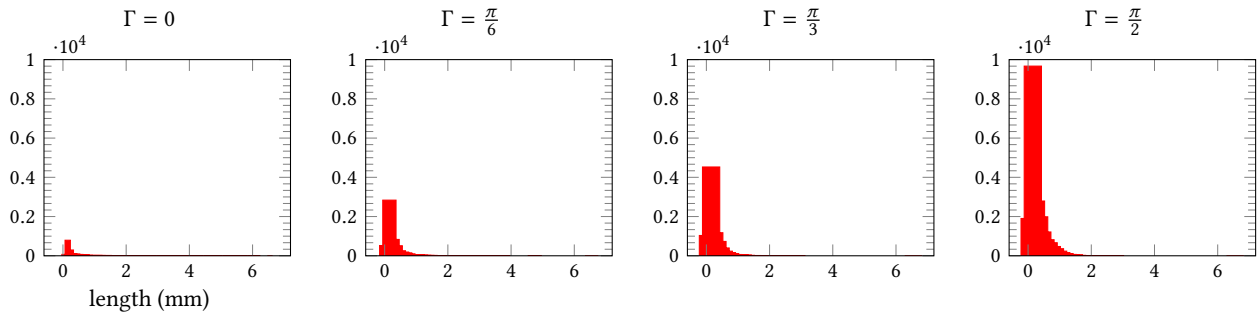


Fig. 5. Overhang statistics for 35 mm cubes sliced at 0.2 mm thickness printing with a 0.4 mm nozzle, filled at 25% density for varying value of Γ. The histograms count the number of segments of some length exceeding the overhang constraint (less than 40% of the extrusion disc supported from below). The graphs are clamped after the first 95% of segment counts for readability. These results were computed on a cube at 25% density.

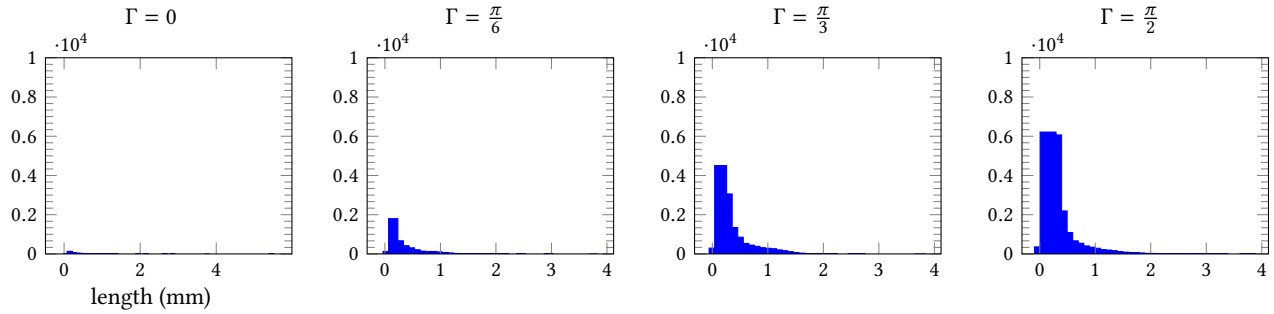


Fig. 6. Same as Figure 5 for overlap statistics. A segment is counted when the overlap along it exceeds 10% of the nozzle diameter. The graphs are clamped after the first 95% of segment counts for readability.

not be too large relative to variations in the control fields, or it would break the assumptions made in order to recover the local phasor field φ_c. Thus, we pick the filtering bandwidth to be $1/\sqrt{2}$ the unfiltered noise kernel bandwidth (i.e. the filter kernel radius is twice the radius of the noise kernel).

6 MECHANICAL ANALYSIS

In this section we study the macroscopic mechanical behavior produced by our microstructures. They are meant to be used under large deformations, beyond the linear regime. Before moving to actual samples, we first provide a numerical analysis of the linear elastic behavior of the diamond structure in Figure 7. This confirms the transverse-rigid nature of the structure (u is significantly stiffer). However, it also shows that directions exactly aligned with the diamond walls are stiffer than any other direction in the vw plane. This is not ideal but is a trade-off we had to make to accommodate for AM constraints. Interestingly, this behavior is less problematic under large deformations: Figure 8 considers a specimen with maximal rigidity contrast ($\Gamma = 0$), testing for compression along u, v, w as well as $u+w$ (strain in the direction of the diamond structure walls). For this latter measure, we use a second cubic specimen with the diamond walls aligned with the specimen sides. We observe collapsing (explaining the plateau) in the measures along $u+w$, followed by a progressive approach to the v and w curves (almost identical for 30% strain). Thus, under large deformations, the collapsing produces a response that is roughly equivalent to v and w, especially when compared with the much more rigid u axis.

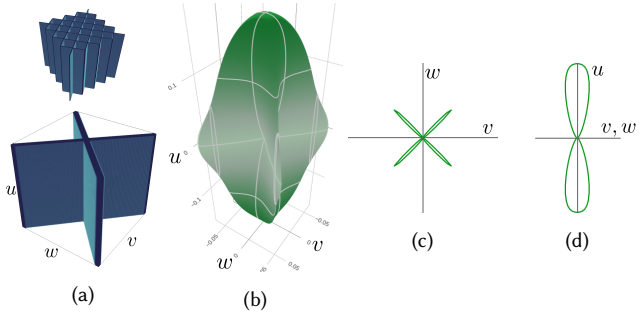


Fig. 7. Numerical simulation (small deformations, i.e. linear elasticity) of the periodic diamond structure. **(a)** Periodic domain used for the homogenization [?] (grid with resolution 100^3, solid phase with Poisson's ratio 0.3 and Young's modulus 1). The solid phase (in blue) has a density of 17%. The homogenized linear elasticity tensor [?] is computed with the software CrAFT [?]. **(b)** Visualization of the directional Young's modulus given by ELATE [?]. The minimal Young's modulus is achieved in the axes v and w, while the maximal one is achieved in the axis u. **(c)** Young's modulus in the vw plane, the central point corresponds to the origin. There is a sharp increase in the Young's modulus for directions closely aligned with the diamond walls, as illustrated by the cross. **(d)** Young's modulus in the uv and uw plane.

We report additional stress-strain curves measured on printed samples in Figure 11, for varying frequencies scales and angular spreads. Note that regardless of the parameter Γ all structures of a given scale have the same density (up to 2% variations). We selected

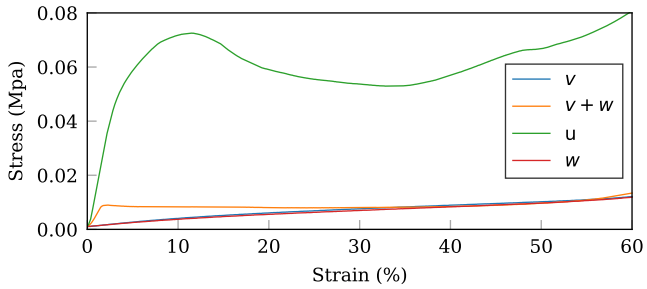


Fig. 8. Compression test with two specimens, having equal density $\rho = 10\%$ and angular spread $\Gamma = 0$. In this case, the printing direction is along the axis u. One of the specimens has been rotated 45° degrees along the u axis. Please refer to text for details.

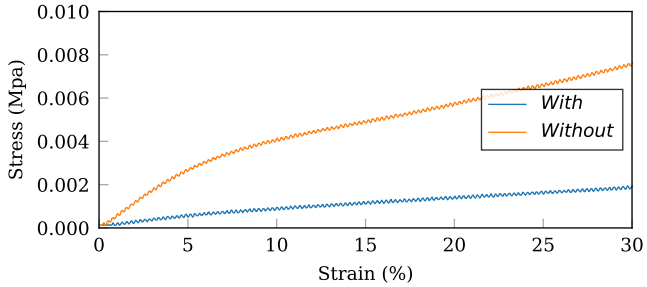


Fig. 9. Effect of the filtering process on the mechanical properties. This measure shows that the singularities add unwanted rigidity in the material. This measure was made on two cube of 40mm with a measured density of 10%

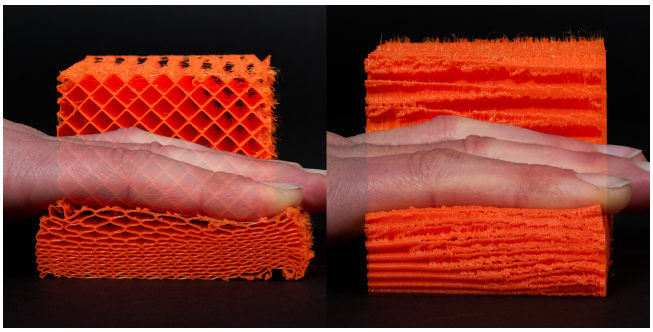


Fig. 10. Basic structure at rest and under full compression along w. **Left**: v, w face. **Right**: u, w face. Note how the structure does not expand along u, even when w is fully compressed.

the scale to match densities of 10%, 20% and 30%. As expected, a lower density results in an overall less rigid structure. We observe three other major experimental trends in Figure 11, across the three different densities. First, as we increase the angular spread Γ the elastic response in u becomes fairly close to v and w. The w axis ends up being less rigid than u and v, which is mainly explained by the fact that the angular spread is solely occurring in the u, v plane. While we could imagine compensating for this by changing the frequency of $p_{\mathbf{w}}$, overhang constraints make this difficult. Nevertheless, the rigidity contrast is significantly reduced compared to the case of $\Gamma = 0$ verifying our original assumption. Second trend, for $\Gamma = 0$ and $\Gamma = \frac{\pi}{6}$ there is buckling in the u axis. However, this buckling is much stronger in the case of $\Gamma = 0$, indicating that a small degree of angular spread allows to reduce it. We do not consider buckling along u as a problem for our use cases: the amount of stress required to trigger it is very significantly above the stress required to fully compress the v or w axes. Third trend, for $\Gamma = 0$, there exists a small but noticeable difference between the response of v and w, while ideally, both should have the same response. This is primarily explained by the anisotropy introduced by the layered printing along the w axis, used for this particular test.

Figure 10 in the paper shows the behavior of the basic case ($\Gamma = 0$, D constant) when w is fully compressed. As can be seen, the lateral expansion is very dissimilar in u and v: it is large along v and nonexistent along u (the same occurs for w, u compressing along v, by symmetry). This behavior relates to Poisson's ratios, which, in linear elasticity is effectively zero for pairs u, v and u, w while being close to one for pairs v, w and w, v. This is particularly interesting for our application as compression along w (resp. v) triggers a large expansion along only v (resp. w). Through the orientation field D we control the v, w plane where these expansions occur (recalling the u is aligned with D).

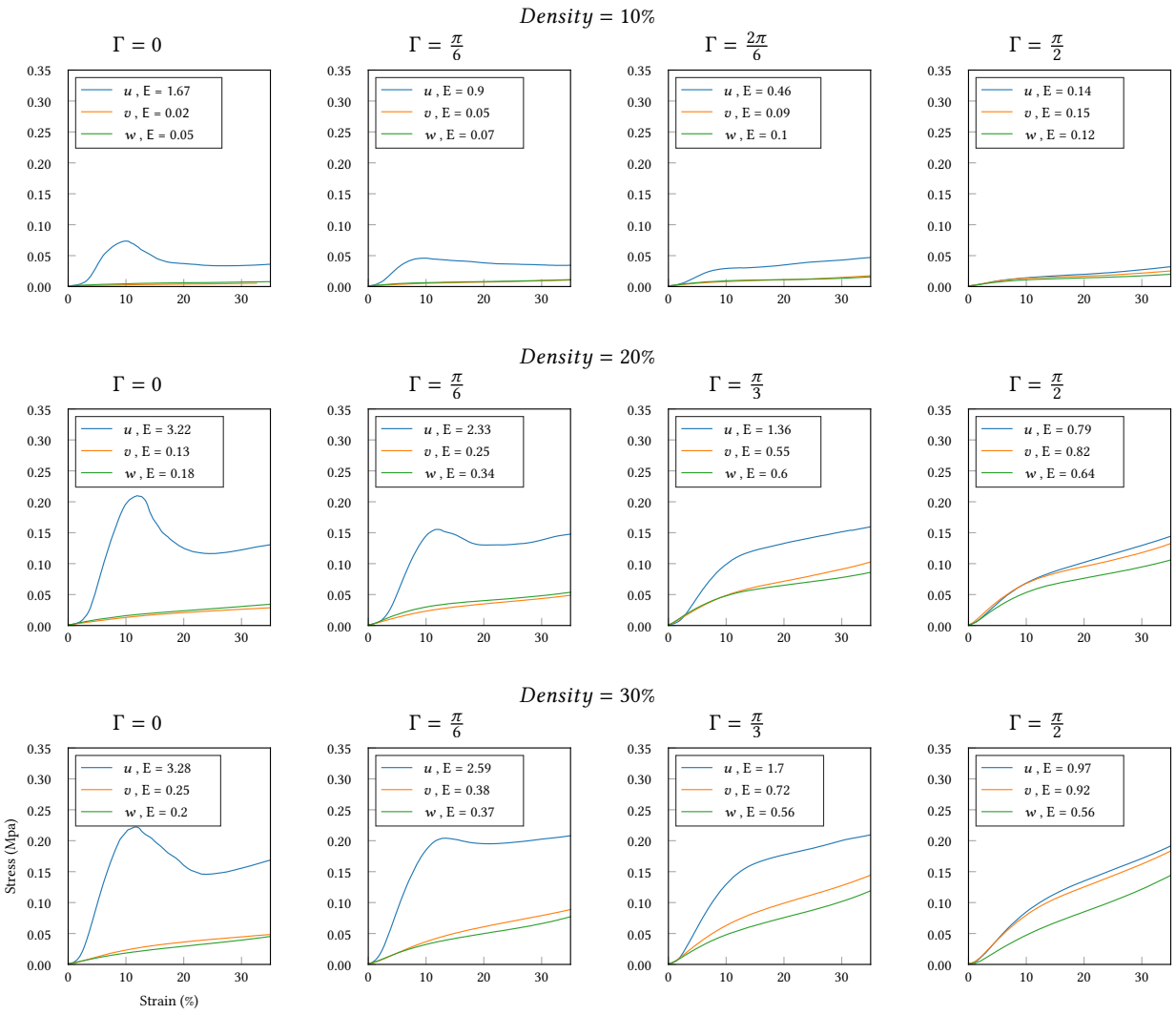


Fig. 11. Compression test with 12 different specimens (3D printed cubes) with varying density ρ (columns) and amount of angular spread Γ (rows), from 0 to isotropy ($\pi/2$). For each specimen, we measure the stress-strain response [?] in the three orthogonal directions u, v, w denoted with different colors. The Young's modulus E is in Mpa. The printing direction is along the axis w. Please refer to text for discussion.