



**HAL**  
open science

## Extrusion-Based Ceramics Printing with Strictly-Continuous Deposition

Jean Hergel, Kevin Hinz, Sylvain Lefebvre, Bernhard Thomaszewski

► **To cite this version:**

Jean Hergel, Kevin Hinz, Sylvain Lefebvre, Bernhard Thomaszewski. Extrusion-Based Ceramics Printing with Strictly-Continuous Deposition. ACM Transactions on Graphics, 2019, 10.1145/3355089.3356509 . hal-02363424

**HAL Id: hal-02363424**

**<https://inria.hal.science/hal-02363424v1>**

Submitted on 14 Nov 2019

**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.

# Extrusion-Based Ceramics Printing with Strictly-Continuous Deposition

JEAN HERGEL, Université de Montréal

KEVIN HINZ, Université de Montréal

SYLVAIN LEFEBVRE, National Institute for Research in Computer Science and Automation (INRIA)

BERNHARD THOMASZEWSKI, Université de Montréal

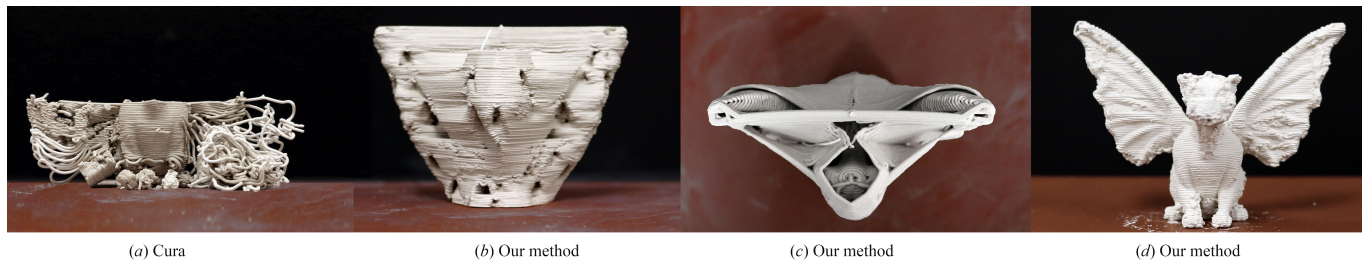


Fig. 1. Existing path planning methods for thermoplastic materials rely on transfer moves that lead to catastrophic failure when printing this Gargoyle model in clay (a). To eliminate transfer moves, our method generates a continuous, self-supported deposition path (a, b) whose non-model parts can be removed to reveal an intact model (d). Our approach thus enables successful 3D printing of geometries that were previously impossible to realize in clay.

We propose a method for integrated tool path planning and support structure generation tailored to the specific constraints of extrusion-based ceramics printing. Existing path generation methods for thermoplastic materials rely on transfer moves to navigate between different print paths in a given layer. However, when printing with clay, these transfer moves can lead to severe artifacts and failure. Our method eliminates transfer moves altogether by generating deposition paths that are continuous within and across layers. Our algorithm is implemented as a sequential top-down pass through the layer stack. In each layer, we detect points that require support, connect support points and model paths, and optimize the shape of the resulting continuous path with respect to length, smoothness, and distance to the model. For each of these subproblems, we propose dedicated solutions that take into account the fabrication constraints imposed by printable clay. We evaluate our method on a set of examples with multiple disconnected components and challenging support requirements. Comparisons to existing path generation methods designed for thermoplastic materials show that our method substantially improves print quality and often makes the difference between success and failure.

CCS Concepts: • **Computing methodologies** → **Computer graphics**.

Additional Key Words and Phrases: 3d Printing, Optimization

## ACM Reference Format:

Jean Hergel, Kevin Hinz, Sylvain Lefebvre, and Bernhard Thomaszewski. 2019. Extrusion-Based Ceramics Printing with Strictly-Continuous Deposition. *ACM Trans. Graph.* 38, 6, Article 194 (November 2019), 11 pages. <https://doi.org/10.1145/3355089.3356509>

Authors' addresses: Jean Hergel, Université de Montréal; Kevin Hinz, Université de Montréal; Sylvain Lefebvre, National Institute for Research in Computer Science and Automation (INRIA); Bernhard Thomaszewski, Université de Montréal.

© 2019 Association for Computing Machinery.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3355089.3356509>.

## 1 INTRODUCTION

As one of the oldest human-made materials, ceramics are valued for their excellent hardness, resistance to corrosion and abrasion, as well as their unique appearance. Besides their many technical applications, ceramics are ubiquitous in crockery, tableware, and home decor. The predominant manufacturing technologies to date are based on slip-casting, pressing, and extruding clay in different consistencies [Händle 2007]. While well understood and largely automated, existing processes either impose limitations on shape or they require expensive tooling which makes small series or one-off productions impractical.

The advance of digital manufacturing into ceramic materials now enables automatic fabrication of unique designs with unprecedented complexity [Zocca et al. 2015]. Our work is motivated by the increasing availability of affordable consumer-level clay printers based on paste extrusion, opening the possibilities of digital ceramics to artists, craftsmen, hobbyists, and average end-users. But although shape complexity is, in principle, only limited by print resolution, it is in practice restricted by an important fabrication constraint that is best illustrated by comparison to thermoplastic materials.

For thermoplastics, the deposited material solidifies rapidly after extrusion; it attains its final rigidity almost immediately, while material at the nozzle is still liquid. Moving the print head without extruding—so called *transfer moves*—will hence immediately detach the filament from the model. This may lead to some minor artifacts, but the structural integrity of the model is generally not a risk. For clay, however, there is no such stiffness grading between model and extruder since clay maintains the same soft consistency during the entire printing process. The attachment strength between structure and freshly deposited material is therefore just as strong as between nozzle and extruded filament. Moving the print head without extruding will thus pull on the structure and induce unwanted

deformations that amplify over time and, eventually, lead to failure. This issue generally appears for shapes that, once sliced, exhibit multiple disconnected components per layer. Its severity increases with model complexity and is especially acute for the support structures generated by existing methods optimized for thermoplastics. As shown by our experiments, attempts at printing such structures in clay will lead from disappointing results to complete failures; see, e.g., the leftmost image in Fig. 1. Consequently, transfer moves should be avoided at all cost to ensure reliable deposition—ideally, there should be only one continuous print path per layer.

In this work, we propose a method for integrated tool path planning and support structure generation tailored to the specific constraints of paste extrusion printing. As the core of our approach, we guarantee a single continuous print path for the entire model and thus eliminate transfer moves and their corresponding perils altogether. Our method traverses the layer stack of a sliced input model in one top-down pass, solving a discrete path planning and a continuous path optimization problem in each layer. For the discrete part, we cast the problem of finding a continuous path that sufficiently supports structure in the layer above into a form that can be solved efficiently using approximate shortest path algorithms. To minimize interference with the model, we introduce a model-avoiding metric and model-shielding proxy geometry to ensure that the generated path connects to the model only where necessary. For the continuous part, we further optimize path length and curvature in each layer such as to reduce material usage and print time. Combined with coupling constraints between adjacent layers, this continuous optimization leads to smoother and less dense non-model structures that are easier to print and remove.

We evaluate our method and the impact of our design decisions on a set of challenging examples. Our results indicate that our method leads to good-quality prints even for complex models with multiple disconnected components and advanced support requirements. We compare our method to a state-of-the-art path generation method and show that continuous deposition not only substantially improves print quality; it often makes it possible to manufacture parts that otherwise *consistently fail to print*.

## 2 RELATED WORK

*Computational Fabrication.* The graphics community has in recent years started to embrace the challenge of making 3D printing more accessible to end-users through non-expert design systems and various computer-assisted content generation and processing tools. One particular line of work aimed at predicting and improving the resilience of geometric models during printing, cleaning, and use [Langlois et al. 2016; Lu et al. 2014; Stava et al. 2012; Umetani and Schmidt 2013; Zhou et al. 2013]. While this kind of analysis is beyond the scope of our present work, adapting these computational fabrication techniques to the peculiar material properties of clay is an interesting direction for research.

*Tool Path Planning.* Creating efficient tool paths is an important problem for many digital manufacturing processes. Commonly used strategies for filling regions within a layer are zig-zag [Ding et al. 2014] and contour-parallel [Yang et al. 2002] patterns as well as space-filling curves [Cox et al. 1994]. Unlike other methods from

the latter category, the connected Fermat spirals proposed by Zhao et al. [2016], and later extended to CNC machining of curved surfaces [Zhao et al. 2018], has the advantage of producing a single low-curvature path per filled region. While these properties are promising for clay printing, Fermat spirals do not offer a solution to connect different paths in a given layer, nor to generate continuous support structures. Rather than alternative solutions, Fermat spirals and our method are thus complementary to each other. On a more technical level, our method shares the overall strategy of splitting the design of smooth, continuous paths into discrete and continuous subproblems. However, both of these subproblems are very different: instead of generating space-filling curves, we create approximately shortest paths that connect sets of support points and model paths without crossing them; and instead of smoothing the paths in a single layer, we simplify supporting and connecting structure through smoothing and shortening subject to cross-layer compatibility constraints.

In the context of paste extrusion, Fleming et al. [2017] consider reordering the infill paths generated by a standard slicer to minimize spurious depositions when crossing over the part. While this work points to transfer moves as a source of concern for extrusion-based manufacturing, the proposed technique does not consider generating supports nor the case of multiple components within a layer.

Hergel and Lefebvre [2014] improve on the various artifacts that arise in the context of multi-material fused-filament printing with multiple extruders. Similar to our method, they also solve an obstacle avoiding path planning problem, with the goal of preventing inactive print heads from moving over already printed regions. However, since their setting does not impose continuous deposition constraints, their problem formulation and solution are quite different from ours.

*Support Structures.* A common restriction of many additive manufacturing processes is that material can only be deposited where there is already structure below. If the model to be printed violates this constraint, additional support structure has to be inserted.

The first step in support structure generation is to detect those regions of the model that require support. There are two main standard approaches to this end: the mesh-based approach flags downward-facing polygons as requiring support if their vertical angle falls below a given threshold value [Allen and Dutta 1995]; the slice-based approach operates on boolean differences between adjacent layers [Chalasan and Roscoe 1995].

Once regions that require support are determined, supporting geometry has to be generated. The simplest method is to extrude faces that require support downwards until they intersect either model or ground. This simple approach does, however, lead to substantial support volumes that can easily dominate the overall material cost and print time. Moreover, dense support structures with many connections to the model are difficult to remove and degrade the surface quality of the printed model.

For these reasons, there has been significant work aimed at reducing or optimizing support structure in various ways. A common approach is to optimize the orientation of the model in order to minimize the amount of required support structure [Allen and Dutta

1995; Vanek et al. 2014]. A different, perceptually motivated approach in this context is due to Zhang et al. [2015], who optimize orientation to minimize the visual impact of marks remaining after support removal. While we do not consider print orientation optimization in this work, any of the techniques in this space could, in principle, be combined with our method.

Another general strategy for reducing support volume is to use sparse cellular [Strano et al. 2013], truss-like [Wang et al. 2013], or branching structures [Dumas et al. 2014; Schmidt and Umetani 2014; Vanek et al. 2014]. A generalization of these concepts is the method by Mezzadri et al. [2018] who cast support structure generation as a topology optimization problem. All of these approaches effectively reduce support volume by generating lower-dimensional support structures. Due to their highly fragmented nature, however, none of those strategies are suitable for clay extrusion with its continuous deposition constraint.

Yet another approach to reduce support requirements is to use multi-axis printing setups in order to change the effective material deposition direction during printing. Earlier work in this direction focused on wire frame printing using three [Mueller et al. 2014], five [Wu et al. 2016a], and six-axis setups [Huang et al. 2016]. Targeting general solid models, Wu et al. [2017] proposed a 6-DOF system that re-orientates the model during printing such as to reduce or avoid support structure requirements. Using a similar setup, Dai et al. [2018] decompose the model into curved layers and use continuous re-orientation to increase print head accessibility, thus further reducing or even eliminating support requirements. Apart from such robotic systems being significantly more expensive than consumer-level FFF printers, considering the limited load-bearing capacity of freshly printed clay, it is unclear whether re-orientation could be applied to our setting without inducing model collapse. Finally, while robotic systems have been explored for printing clay and concrete at larger scales [Buswell et al. 2018; Gosselin et al. 2016], continuous deposition and support generation was not a focus.

*In-Fill Structures.* A related problem to support structure design is the generation of in-fill structure that fills the interior of a given model. Standard software typically offers a discrete set of in-fill ratios and patterns for the user to choose from. One focus of research has been on generating self-supporting in-fill structures with optimal strength-to-weight ratio for given loads [Wu et al. 2018, 2016b]. Another focus has been the efficient generation of as-small as possible internal supports through iterated carving [Wang et al. 2018] and top-bottom sweeps [Hornus and Lefebvre 2018]. Tricard et al. [2019] produce tree-like deposition paths in each slice, together forming inner vaults that simplify as they grow downwards. Our algorithm follows a similar top-down simplification strategy, optimizing for self-supported, strictly continuous deposition paths that eliminate transfer moves.

*Digital Ceramics Manufacturing.* Ceramics are appreciated for their excellent mechanical, chemical, and electrical properties, but manufacturing of complex or custom geometries is difficult with conventional technologies. Digital manufacturing of ceramics is therefore receiving increasing attention from research and industry. While there are many different technologies [Chen et al. 2018],

our primary motivation for focusing on paste-extrusion is accessibility. There is a range of extrusion-based approaches includes Fused Deposition Ceramics, Freeze-form Extrusion Fabrication, and Robocasting; see also [Armani et al. 2016] for a detailed taxonomy. While most of these processes require rather expensive hardware, Fused Deposition Ceramics uses thermoplastic filament filled with ceramics particles and can, in principle, be implemented on top of conventional FFF printers [Vaidyanathan et al. 2000]. However, the high fill-rate needed to obtain good ceramics quality leads to extremely brittle filament that requires special pre-heating stages and makes printing overall challenging. Instead of using thermoplastic filament, our work focuses on direct extrusion of moldable clay with a water content of approximately %30. Several commercial printers based on this process are available at consumer-level prices. While the entry barrier for artists and hobbyists is thus comparatively low, the range of geometries that can be printed with existing methods is limited. We know of no work to address the problem of tool path planning and support structure generation for ceramics printing with direct clay extrusion.

### 3 GENERATING CONTINUOUS DEPOSITION PATHS

The goal of our method is to generate deposition paths that enable successful extrusion-based ceramics printing of general models. To this end, the generated deposition paths

- must not involve any transfer moves that displace the print head without extruding,
- must be self-supporting,
- should avoid connecting or crossing through the model whenever possible to minimize impact on visual quality and facilitate support removal, and
- should be economical in terms of material and printing time.

Before describing our method in detail, we present the main observations and ideas underlying our approach below.

#### 3.1 Overview

The basic strategy of our method is to traverse the layer stack from top to bottom, maintaining the invariant that each layer is a continuous, cyclic path that provides sufficient support for the layer above. In each layer, we first detect points requiring support (Sec. 3.2), then compute continuous and model-avoiding deposition paths (Sec. 3.3), and finally optimize the result with respect to length, smoothness, and downward collision avoidance (Sec. 3.4). The result is a self-supporting structure that prints continuously and provides support for the model where required. Our method builds on the following observations and ideas.

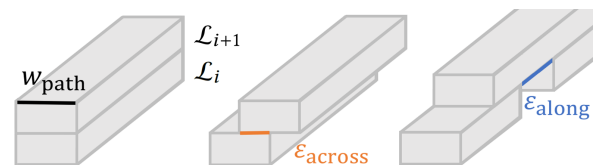


Fig. 2. Paths from neighboring layers without offset (left), with maximum lateral offset (middle), and maximum bridging distance (right).

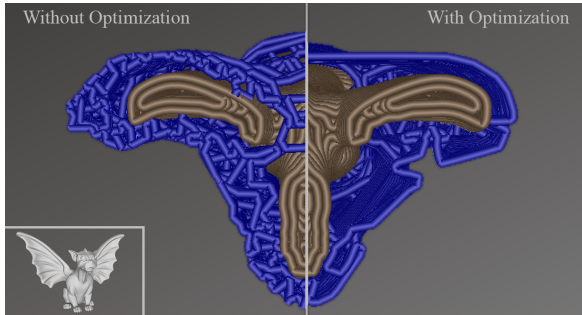


Fig. 3. Effect of path optimization. *Left*: without path optimization, the non-model structure (blue) is dense and exhibits larger curvature. *Right*: path optimization leads to shorter, smoother, and sparser non-model structure that is faster to print and easier to remove.

*Bridges and Overhangs.* Although to a lesser extent than for thermoplastic materials [Dumas et al. 2014], clay nevertheless allows us to horizontally offset print paths across layers and thus create *overhangs*. Moreover, we can also print *bridges* that span a certain distance without any structure underneath; see Fig. 2. The ability to exploit these tolerances is at the core of algorithm and essential for generating paths that print reliably, facilitate removal of non-model structures, and are economical in terms of print times and material use.

*Continuous Supporting Curves.* In order to generate self-supporting deposition paths, our algorithm optimizes in each layer  $\mathcal{L}_i$  a continuous path  $C_i$  that strictly covers the model paths  $\mathcal{M}_i$  in the layer while its non-model part  $C_i \setminus \mathcal{M}_i$  provides *sufficient support* for the path  $C_{i+1}$  in the layer above. Indeed, the freedom to create slight overhangs and bridges allows us to position the supporting curve within an envelope as opposed to strictly below. Consequently, the path  $C_i$  may be significantly different from that in  $C_{i+1}$  as long as it satisfies support requirements. We exploit this freedom to manage structure complexity and to avoid re-connecting to the model.

*Managing Structure Complexity.* Overhanging regions require dense support in the layer just below. To this end, our method will produce a continuous path that folds back and forth below the area to be supported. Simply extending these paths downwards, however, would result in dense non-model structures that are slower to print and harder to remove. Our method exploits the overhang tolerances to produce simpler curves in each subsequent layer, thus reducing the geometric complexity of the non-model structures below the supported areas. The effect is shown in Fig. 3.

*Avoiding the Model.* When generating deposition paths from top to bottom, the non-model structures tend to extend vertically and can easily collide with the model below. In many cases, this unwanted effect can be prevented by exploiting the overhang angle to incline the non-model structure away from the model before collisions occur. To this end, we introduce proxy geometry that *safeguards* the model against such collisions. Mirroring the maximum admissible slope for the non-model structure, this safeguard effectively provides a collision look-ahead: by keeping their distance



Fig. 4. Side view of safeguard. *Left*: generating paths from top to bottom, the non-model structure (red) cannot be displaced enough to avoid the model (grey) once the collision is detected. *Right*: using the safeguard (blue), our algorithm can anticipate the collision with model and exploit the admissible horizontal offset (green) to displace the non-model structure early enough to avoid collisions.

to the safeguard, new deposition paths are progressively pushed away from the model *before* the collision becomes unavoidable. This principle is illustrated in Fig. 4.

### 3.2 Generating Support Points

We assume as input a sliced version of a water-tight geometric model. Every layer  $\mathcal{L}_i$  is composed of one or several disconnected model components, each represented by one or several closed *model paths*  $\mathcal{M}_i^j$  with the number depending on the desired wall strength. We assume that each path for each component per layer is cyclic.

Our algorithm traverses the layer stack from top to bottom and, for each layer  $\mathcal{L}_i$ , generates a continuous cyclic path  $C_i$  that connects all models paths  $\mathcal{M}_i^j$  in layer  $\mathcal{L}_i$  and provides sufficient support for the layer  $\mathcal{L}_{i+1}$  above. To maintain the invariant of sufficient support, we must determine which parts of  $C_{i+1}$  in the layer above are not supported by model paths  $\mathcal{M}_i^j$  in the current layer and insert additional support structure if required.

In order to determine which parts of a given deposition path require support, we first have to establish the precise conditions for a point to be sufficiently supported. Let  $\mathbf{p} = C_{i+1}(s)$  denote the position of a point on the continuous path in layer  $\mathcal{L}_{i+1}$  corresponding to arc length parameter  $s$ , and let  $P_{\downarrow}(\mathbf{p})$  denote its projection onto layer  $\mathcal{L}_i$ . We define the support distance of  $\mathbf{p}$  as

$$\min_{\Delta s, t, j} |\Delta s| \quad \text{s.t.} \quad \|P_{\downarrow}(C_{i+1}(s + \Delta s)), \mathcal{M}_i^j(t)\|_2 < \varepsilon_{\text{across}}, \quad (1)$$

or  $\|P_{\downarrow}(C_{i+1}(s + \Delta s)), \mathcal{S}_i^j\|_2 < \varepsilon_{\text{across}},$

where  $\mathcal{S}_i$  is the set of *support points* for layer  $\mathcal{L}_i$ . If this problem has a solution with  $|\Delta s| < \frac{1}{2}\varepsilon_{\text{along}}$ , we consider the point  $\mathbf{p}$  sufficiently supported; see also Fig. 5.

This support distance criterion suggests the following strategy for generating self-supported, continuous deposition paths. Going through the layers from top to bottom, we traverse the continuous

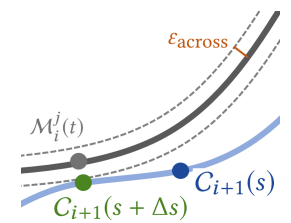


Fig. 5. Support distance  $\Delta s$  for point  $C_{i+1}(s)$ .

path  $C_{i+1}$  using a parameter stride length of  $\epsilon_{\text{along}}$  and compute the support distance of each such sample point using Eq. (1). If a given sample point is not sufficiently supported, we add it to the list of support points  $S_i$ . We then connect all support points  $S_i$  and model paths  $\mathcal{M}_i$  into the continuous deposition path  $C_i$  as described in the next section and repeat the process in the layer below.

### 3.3 Path Planning

Having generated all support points for a given layer  $\mathcal{L}_i$ , the goal is now to compute a single continuous path  $C_i$  that connects all support points  $S_i$  and model paths  $\mathcal{M}_i$ .

A direct approach would be to generate an approximately shortest path through all points in  $S_i$  and  $\mathcal{M}_i$  using a traveling-salesman problem (TSP) solver. However, solving a TSP on the complete set of points would be intractably slow. While reducing the resolution of the curves can mitigate this problem to some extent, it amplifies another problem: instead of following model paths as one continuous segment, the TSP solver may choose to cross through the model multiple times to connect with support points if it reduces the overall length of the path; see Fig. 6a. To avoid these problems by construction, we introduce two important modifications to the direct approach.

*Model Connections.* Instead of using all points in  $\mathcal{M}_i$  for the TSP, we select only one connection point  $\mathbf{m}_i^j$  per model path  $\mathcal{M}_i^j$ . The result  $\mathcal{T}_i$  obtained by applying the TSP solver to this reduced set of points  $T_i = \cup_j \mathbf{m}_i^j \cup S_i$  allows us to construct a short and strictly-continuous path through all points in the current layer: since  $\mathcal{T}_i$  and all  $\mathcal{M}_i^j$  are cyclic, we can start traversal anywhere on  $\mathcal{T}_i$ . Once we encounter a model connection point  $\mathbf{m}_i^j$ , we branch out to visit all points of  $\mathcal{M}_i^j$  sequentially, and then continue along  $\mathcal{T}_i$ .

We chose the connection points such as to minimize the distance to the set of support points  $S_i$ . This strategy provides a low-cost opportunity for the deposition path to connect to the model when it passes through the close-by support points. For layers without any support points such as the top most layer, we greedily select the point on a given model path that minimizes the distance to all other model paths. While there is no guarantee that this specific choice of connection points will lead to the shortest possible path, we have not observed clearly sub-optimal results using this strategy.

*Model-Avoiding Distance.* Representing model paths with a single connection point allows us to print each model path as one continuous segment. However, the TSP solver may still choose to cross through the interior of model paths as shown in Fig. 6b. We address this problem by modifying the costs of crossing edges to reflect the length of a corresponding model-avoiding path.

To compute model-avoiding paths and their corresponding costs, we start by introducing *safeguard* geometry around each model path; see Fig. 6c for an illustration and Sec. 3.3 for details on how safeguards are constructed. This proxy geometry allows the deposition paths to navigate around the model while maintaining a close but safe distance. We then construct a graph  $G(V, E)$  whose vertices  $V$  include the points of all safeguards as well as all support points.

The edges  $E$  are computed using Delaunay triangulation on  $V$ , constrained to include all edges of each safeguard. We discard any edge that crosses through the interior of a given model path.

With the graph set up in this way, we use Dijkstra's algorithm to determine the shortest path and, consequently, the model-avoiding distance between all pairs of points in  $T_i$ . Having determined pairwise model-avoiding distances in this way, we use an off-the-shelf TSP solver [Google 2018] to compute an approximately shortest path  $\mathcal{T}_i$ . Finally, we replace all edges in  $\mathcal{T}_i$  that cross through the model with their corresponding obstacle-avoiding paths as illustrated in Fig. 6e.

*Nested Support.* The algorithm described above extends readily to a recursive version that handles cases with support points inside model paths; see, e.g., Fig. 7. To this end, we apply the algorithm as described above to all support points and model paths inside a given enclosing path  $\mathcal{M}_i^j$ , adding the single connection point  $\mathbf{m}_i^j$  to the set of points passed to the TSP solver. When traversing  $\mathcal{M}_i^j$ , we simply branch out to visit the interior path which starts and terminates at  $\mathbf{m}_i^j$ .

*Cross-Layer Coherency.* Our formulation intrinsically favors cross-layer coherency and the TSP solver will typically produce similar routings for neighboring layers. However, if there are alternative routes with similar lengths, the TSP solver may choose a less coherent but shorter path, even if savings are very small. This can lead to an undesirable break of cross-layer coherency. To prevent this effect and its attendant impact on print quality, we give preference to spatial coherence over length: the weights for all edges in the distance graph  $G$  that are sufficiently close to the path from the layer above are multiplied by a discount factor  $0 < \alpha_D < 1.0$  (we use  $\alpha_D = 0.9$ ). As can be seen by the coherent paths obtained through layers (best revealed in the accompanying video), this strategy effectively eliminates spurious routing changes.

Instead of encouraging coherency in the height direction, a non-model path could, in principle, also be supported by a meandering path in the layer below. However, providing support only in crossing points would make the non-model structure less stable overall.

It should be stressed that, in addition to coherency within each layer, moving from one layer to the next is strictly continuous: our method ensures that the extruder will always start at a point that was present in the layer just below. This is true by construction, since this point is either part of the model or will be supported.

### 3.4 Path Optimization

Applying the algorithms for support point generation and path planning sequentially to each layer in a top-down fashion yields a single continuous, self-supporting deposition path. There are, however, two remaining problems that must still be resolved. First, the non-model structure  $C_i \setminus \mathcal{M}_i$  that was inserted in a given layer to provide support will itself need to be supported in the layers below. Simply propagating these structures downwards without modification will lead to increasingly complex and dense non-model structure (from top to bottom).

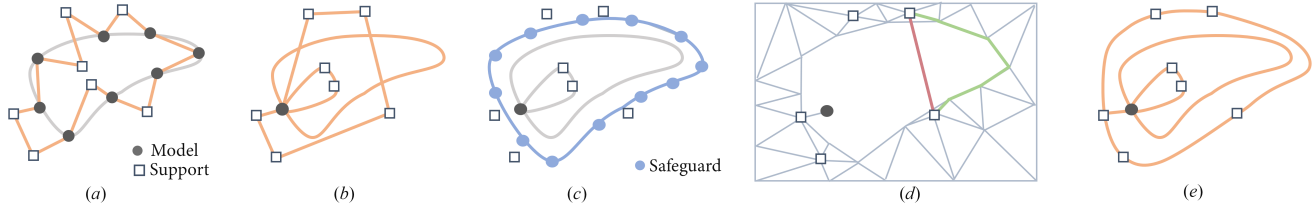


Fig. 6. Overview of path planning algorithm. Passing the entire set of model and support points to the TSP solver leads to non-continuous model paths (a). Representing model paths by a single connection point solves this problem, but the generated path now crosses through the model (b). Proxy geometry around the model path (c) is used to construct a graph (d) that allows us to map the lengths of model-crossing segments (red) to their model-avoiding counterparts (green). Passing these distances to the TSP solver yields the desired result (e).

Second, as the non-model structure propagates downward, it can reconnect to the model, complicating its removal and degrading surface quality.

Both these issues can be resolved by optimizing the shape of the non-model structure, exploiting the freedom to offset support points horizontally as long as their displacement remains below the threshold value  $\epsilon_{\text{across}}$ . Taken together, these per-point displacements define an envelope in which we can deform the original path without violating the support requirements of the layer above. This allows us to gradually smooth and simplify the paths, and to repel too close parts from the structure before they intersect the model.

To this end, we formulate a continuous energy minimization problem in which we seek to compute adjusted positions  $\mathbf{x}_i \in \mathbb{R}^2$  for all non-model vertices  $\bar{\mathbf{x}}_i \in \mathbb{R}^2$  of the current path  $C_i$  such as to optimize a weighted combination of the design objectives introduced below.

**Length and Curvature.** Shorter and smoother paths print faster and are less likely to produce artifacts or failures. They also lead to a less dense non-model structure that is easier to remove. We therefore define two objectives to gradually decrease the length and curvature of non-model structure, using an elastic curve metaphor similar to Zehnder et al. [2016]. The first one aims at reducing the total length of the curve by a given target ratio  $0 \leq \alpha_l \leq 1$  (we use  $\alpha_l = 0.9$ ). We implement this length reduction objective using a sum of per-edge energies

$$E_{\text{length}}(\mathbf{x}) = \mu_{\text{length}} \sum_k \frac{1}{2\alpha_l L_k} (l_k - \alpha_l L_k)^2, \quad (2)$$

where  $l_k = \|\mathbf{x}_k - \mathbf{x}_{k-1}\|$  and  $L_k = \|\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_{k-1}\|$ . To further encourage smoothness, we penalize curvature using a simplified version of the bending energy proposed in [Bergou et al. 2010],

$$E_{\text{bend}} = \mu_{\text{bend}} \sum_k \frac{2}{L_k + L_{k-1}} \left| \frac{\mathbf{t}_{k-1} \times \mathbf{t}_k}{1 + \mathbf{t}_{k-1} \cdot \mathbf{t}_k} \right|^2, \quad (3)$$

where  $\mathbf{t}_k$  and  $\mathbf{t}_{k-1}$  are 3D embeddings of the normalized 2D edge vectors for edges  $k$  and  $k-1$ , respectively.

**Clearance Distance.** The non-model structure necessarily has to connect to the model where support is needed. To maximize print quality, however, we want to prevent the non-model structure from interfering with the model in layers below. We therefore define a penalty energy that encourages support structure to maintain a

given minimum clearance distance  $\epsilon_{\text{model}}$  (we use  $\epsilon_{\text{model}} = 2w_{\text{path}}$ ). To obtain sufficient smoothness, we use the implicit moving least squares formulation of Öztireli et al. [2009],

$$f_{\text{imls}}(\mathbf{y}) = \frac{\sum_k \bar{\mathbf{n}}_k \cdot (\mathbf{y} - \bar{\mathbf{x}}_k) \varphi_k(\mathbf{y})}{\sum_k \varphi_k(\mathbf{y})}, \quad (4)$$

where  $\varphi_k$  are smooth, radially-symmetric functions centered around  $\bar{\mathbf{x}}_k$ . On this basis, we formulate an exponential clearance energy,

$$E_{\text{clear}} = \mu_{\text{clear}} \sum_k e^{-\beta(f_{\text{imls}}(\mathbf{x}_k) - \epsilon_{\text{model}})}, \quad (5)$$

that strongly penalizes small distances, but rapidly decreases away from the model with a rate determined by the sharpness coefficient  $\beta > 0$ . We found  $\beta = 0.5$  to be a good compromise between locality and strictness.

This formulation effectively prevents support paths from drifting into the model during optimization. However, since the distance field considers only model geometry within a given layer, it cannot anticipate problems occurring in layers below. An example of this case is shown in Fig. 4 where parts of the non-model structure that had sufficient clearance in a given layer collides with the model in the layer below. While the path cannot be sufficiently displaced in this example without violating support integrity, this reconnection with the model could have been avoided had the path been displaced early enough in the layer stack. In order to anticipate this type of problem without having to consider multiple layers during optimization, we use a *safeguard* structure. These self-supporting, model enclosing structures are typically used to catch oozing filament in the context of FFF printing; see, e.g., [Hornus et al. 2016]. We instead use it as a proxy geometry that encloses the model such as to guarantee that, whenever a support point is outside the safeguard in a given layer, it can be sufficiently displaced to avoid the model in all layers below, if necessary.

We construct the safeguard incrementally in a pre-process, starting with all outer-most model paths in layer  $\mathcal{L}_0$ . For a given layer, we first transfer the safeguard polygons from the layer below (except for  $\mathcal{L}_0$ ) and erode them by  $\epsilon_{\text{across}}$ . We then dilate the outermost model paths of the current layer and compute the union between both sets of polygons. The safeguard can have multiple disconnected polygons; see, e.g. the Spire model shown in Fig. 7. To encourage non-model structure to stay outside the safeguard, we simply replace the vertices  $\bar{\mathbf{x}}_i$  in (5) with those of the proxy geometry. This

strategy allows us to avoid many of the cases in which the non-model structure would otherwise have reconnected to the model. It is worth noting that this safeguard is the same proxy geometry that we use during path planning to allow deposition paths to navigate around the model.

*Bounded Path Deviation.* While optimizing the shape of the path according to the above objectives, we must make sure that the deformed path stays within the support envelope around  $\bar{\mathbf{x}}_i$ . Ideally, we would like to impose no resistance to horizontal motion until the vertices reach the bound of the envelope. To implement this unilateral constraint, we introduce log-barrier penalty energies of the form

$$E_{\text{dev}}(\mathbf{x}) = \mu_{\text{dev}} \sum_k -\log([\bar{\mathbf{n}}_k \cdot (\mathbf{x}_k - \bar{\mathbf{x}}_k)]^2 - \varepsilon_{\text{across}}^2), \quad (6)$$

where  $\bar{\mathbf{n}}_k$  is the outward normal of the candidate path at  $\bar{\mathbf{x}}_k$  computed by averaging and renormalizing the normals of the two incident edges.

*Implementation.* All of the energy terms described above have continuous first and second order derivatives, allowing us to minimize their weighted sum using an efficient Newton-type solver. Since some terms are highly nonlinear, we use line search and adaptive regularization to ensure progress even in difficult terrain. The coefficients for the individual objectives are set to  $\mu_{\text{length}} = 10.0$ ,  $\mu_{\text{bend}} = 0.01$ ,  $\mu_{\text{dev}} = 0.001$ , and  $\mu_{\text{clear}} = 0.01$  for all examples shown in this work. Although there is no guarantee that this specific choice of parameter values will always be optimal, we have obtained satisfying results for a diverse set of models with varying complexity and sizes of up to 20cm in height.

As illustrated in Fig. 3 and best seen in the accompanying video, the impact of our path optimization scheme is substantial: it leads to significantly smoother and less dense non-model structure that is faster to print and easier to remove.

## 4 PRINTING AND CLEANING

To facilitate reproduction of the results presented in Sec. 5, we provide details on the manufacturing process that we followed.

*Printing Setup.* We use a Delta-style printer (WASP 2040) with an Auger-based paste extruder. Unlike pressure-based variants that suffer from fluctuation in flow rate, the internal screw of the Auger extruder allows for quasi-constant flow rates and is relatively robust to residual air bubbles due to mixing.

*Clay Recipe.* Clay starts as a finely-grained powder composed of various minerals, metal oxides, and organic substances. When mixed with the appropriate amounts of water, a smooth paste is obtained that is soft enough for extrusion, yet stiff enough to maintain its shape once extruded. We use commercially available Laguna clay powder to which we add 33% of water (dry base). After kneading, the clay is put in a pressurized tank (we use 3.4 bar) from where it feeds into the extruder.

*Printing Parameters.* We use a nozzle diameter of 0.6mm and a layer height of 0.5mm. The feed rate for the extruder, and consequently the flow rate of the material, is set such as to create a desired amount of layer compaction, resulting in a path width of

$w_{\text{path}} = 1.5\text{mm}$ . This choice is primarily governed by structural considerations as under-extrusion will lead to less robust results with weaker inter-layer adhesion, whereas over-extrusion can induce model deformations and even collapse. We use the same value across all examples for both our method and the existing slicers with which we experimented. Having fixed the printing parameters and the clay recipe, we experimentally determine the offset thresholds  $\varepsilon_{\text{across}}$  and  $\varepsilon_{\text{along}}$ . To this end, we perform test prints with different values for the two thresholds on simple geometries with varying overhang angles and gap sizes. From these experiments, we determined the maximum values that gave sufficiently good print quality as  $\varepsilon_{\text{across}} = 0.3w_{\text{path}}$  and  $\varepsilon_{\text{along}} = 1.6w_{\text{path}}$ .

*Non-Model Structure Removal.* Removing the non-model structure from the model is a manual process best done while the model has not completely dried, as the material's brittleness increases with decreasing moisture content. The non-model structures generated by our method typically exhibit large, thin regions that detach readily. More careful manipulation is advised where the non-model structure connects to the model. Our instrument of choice for cutting the small connections is a sharp blade that, if the material is sufficiently humid, will cut readily through the sparse connections; see the accompanying video. If the stability of the model is of concern, it can first be dried completely and then selectively wetted with a water spray bottle or paint brush to remove support.

*Drying and Post-Processing.* The printed model has to dry before it can be further processed. Although controlled environments lead to faster and more uniform drying, we use ambient air conditions for simplicity. We have not monitored moisture content, but drying times for the models we tested range roughly between 3 and 8 hours. After the final model has dried, it can be fired in a kiln at temperatures up to 1400C. This process will induce vitrification, i.e., the transformation of clay into ceramics. We have fired most of the examples that we show in this work, and while the models undergo some shrinking, we have not observed fracture or uneven deformations due to firing.

## 5 RESULTS

We evaluated our method on a diverse set of example shapes, all of which are freely available from online shape repositories. For each of them, we generate two g-code files: one using our approach and another using a reference slicer. As reference, we choose the open-source state-of-the-art slicer Cura, which generated the best results among several slicers that we tested; see also Fig. 10. We use a specific profile optimized for clay deposition, following printer manufacturer recommendations. In particular, we disabled retraction and move the print head up and down between travel paths. We print both files and compare their behavior with respect to printing, support removal, and final model quality.

In considering these results, it is important to keep in mind that clay printing is significantly more challenging than standard fused filament fabrication. There is much less know-how and experience, and most hardware—including ours—is still not quite mature. Therefore, even in ideal conditions, the print quality cannot match that





Fig. 7. The *Spire* model with multiple disconnected components printed using the reference slicer (*right*) and our method (*left*).

of 3D printing with thermoplastic materials. We elaborate further on these limitations in Sec. 6.

*Spire*. One of the main observation underlying our work is that transfer moves—moving the print head without extruding—are a substantial problem for clay printing. This is particularly true for shapes that exhibit slender structures which give rise to multiple disconnected model paths per layer. One such example is the *Spire* model shown in Fig. 7 with its multiple columns that converge and merge while ascending. The comparison between our method and the reference slicer is revealing. The result of the reference slicer exhibits artifacts that amplify with increasing height. These flaws are due to substantial horizontal displacements induced while the print head performs transfer moves between columns. The result obtained with our continuous deposition method exhibits almost no deformation during printing and produces a comparatively clean layer structure after removing support.

The *Two Cats* model prints reliably without any support using plastic materials. When sliced for clay, the print of the reference slicer fails catastrophically as one part of the model (the right cat’s neck) cannot withstand the loads induced by the transfer moves. As our approach performs no transfer moves at all, it exerts much less forces on this weak area. In addition, the single connected path and overall continuous structure strengthens the model during fabrication.

The *Rodin* model is an example that requires additional support structure. The results obtained with the reference slicer and ours are shown in Fig. 9. Although we were able to print the entire model with the paths generated by Cura, the severe artifacts can be observed at the hand supporting the chin and around the ribs. Furthermore, substantial over-extrusion is visible, e.g., in the head and neck regions. With our method, wrist and ribs print without problems and the model generally exhibits a much cleaner layer structure with fewer artifacts.

The *Gargoyle* model is challenging to print due to the support requirements and thin features in its feet and wings. Despite several attempts, we could not complete the print using any of the existing methods that we tested, see Fig. 1a and Fig. 10. With our approach, the model prints successfully and, after cleanup, exhibits an intact structure and comparatively clean surface.



Fig. 8. The *Two Cats* model printed using our continuous deposition method (*left*) and the reference slicer (*right*).

The *Wrestlers* are perhaps the most challenging model in our example set, due to the many small cavities that require support. The paths generated by the reference slicer print without dramatic failures, but the model is fused into a dense block of support structure that is impossible to remove without destroying the model. Using our continuous deposition method, the model prints without problems and, although more challenging than for our other examples, the non-model structure can be removed to reveal an overall intact model.

For the *Stanford Bunny*, the reference slicer places solid columns under the muzzle and ears. These columns extend downward until they connect to either the model or the ground. Except for the few layers in which they attach to the model, these columns result in isolated paths that require transfer moves for printing. These transfer moves damage the model in particular around the ears, which are significantly deformed; see Fig. 12, *right*. Although not free from imperfections, the continuous deposition path generated by our method substantially alleviates these artifacts compared to the reference slicer.

Due to the large and dense connections with the model, removing the support structure generated by the reference slicer proved difficult. On several attempts we ripped out parts of the surface upon removal as can be seen Fig. 12, *top right*. In comparison, our structure is relatively easy to remove, with most parts coming off readily once sufficiently dry.



Fig. 9. Comparison between our continuous deposition method (*left*) and the reference slicer (*right*) on the *Rodin* model. The reference slicer produces severe defects around the wrist (*top right*) and ribs, as well as many over-extrusion artifacts. Our method produces an intact wrist (*top left*) and generally cleaner layer structure with much fewer artifacts.

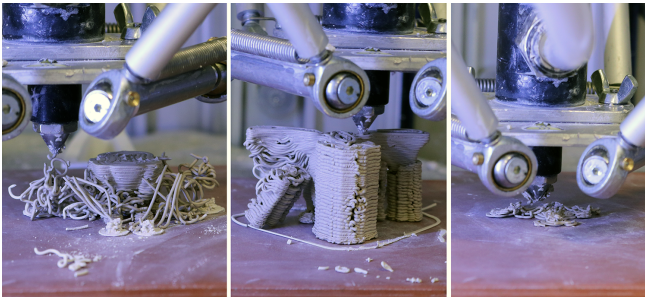


Fig. 10. Results when using print paths generated by existing methods for the Gargoyle model: IceSL (*left*), Slic3r (*middle*), and Meshmixer (*right*). The result for Cura is shown in Fig. 1a.

### 5.1 Timings

We provide computation times for the different parts of our method in Table 1. We implemented our algorithms in C++ and used the freely available TSP solver [Google 2018] for path planning. All computations were done on a 3.2GHz CPU using a single thread.

The computation of the model-avoiding distances via Dijkstra’s algorithm is the most time-intensive part in our current implementation. For the Gargoyle model, this step takes roughly 10 minutes

Model	#Layers	Path Plan.	Path Opt.	Mat. Ratio
Gargoyle	117	12m00s	29s	1.4
Bunny	122	4min48s	14s	0.26
Spire	120	61min00s	85s	0.75
Wrestler	102	35min30s	42s	1.3
Thinker	125	5min06s	19s	1.0
Cats	235	8min32s	28s	1.0

Table 1. Computations times for the three parts of our algorithm for the models presented in this section. The fourth column shows the ratio between non-model and model material usage.

for a total processing time of 12.5 minutes. The solver for continuous path optimization uses the Eigen library for linear algebra and takes between 15 and 20 iterations to converge on average, with an average problem size of roughly 200 points.

## 6 LIMITATIONS & FUTURE WORK

We presented a method for generating strictly-continuous, self-supporting deposition paths for extrusion-based clay printing. Our results indicate that, compared to conventional slicers optimized for thermoplastic materials, our continuous deposition approach leads to more robust printing with fewer artifacts and failure cases.

The main limitation of our work is that we do not consider the physics and deformations of the material. Consequently, there are a number of failure modes that we are not able to predict: model collapses due to increasing self-weight, model collapse due to severe imbalance, or model breaking during support removal. Each of these failure modes are possible directions for future work, noting that some solutions exist already for the second issue [Dumas et al. 2014]. Existing work on analyzing and improving structural soundness would be a good starting point for the third problem [Stava et al. 2012; Umetani and Schmidt 2013; Zhou et al. 2013]. As demonstrated by our examples, our method leads to substantially improved print quality compared to the reference slicer (Figs. 1, 7, 8, 9) and expands the set of objects that can be printed in clay. While we believe that this quality is acceptable for many models and applications, incorporating physics into the path planning and optimization processes might further improve the range of models that can be printed successfully.

Although our method makes clay printing more robust, prints might still sometimes fail due to extrusion problems. There remains some randomness in the printing process itself, especially as clay printing is a less mature technique than thermoplastic printing. One difficulty that we identified and could not resolve completely are occasional fluctuations in extrusion rate, resulting in slightly varying path thickness. We also noticed the deposition head moving with uneven velocity, even though the g-code did not contain corresponding instructions. Finally, compaction of clay becomes a problem when the reservoir starts running low, leading to flow rate variations. It should be noted that these problems all relate to hardware and will occur regardless of the slicer used. Nevertheless, the results that we present in this work are stable and validated: we printed each of them multiple times and obtained, in the vast majority of the cases, equivalent results.

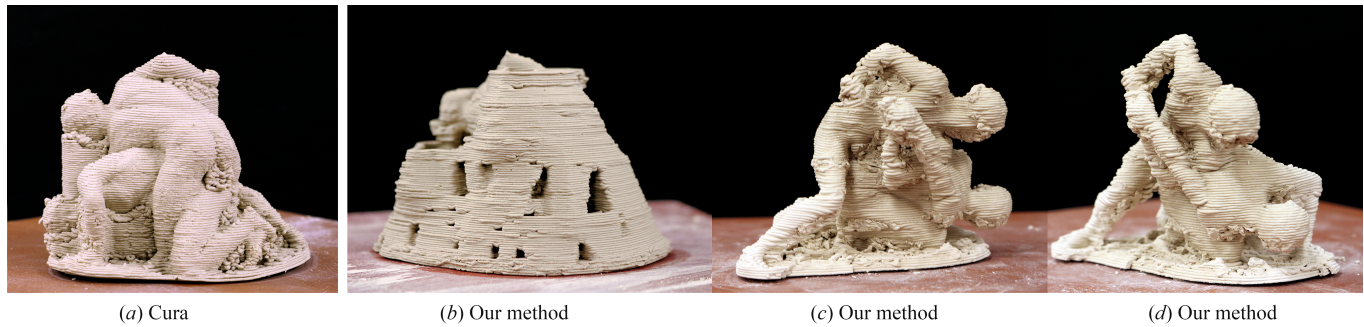


Fig. 11. When using an existing path generation method, a model of two wrestlers in clinch is fused into a dense block of support structure (a), impossible to remove without breaking the model. Our method creates continuous deposition paths (b) whose supporting parts can be removed to reveal an intact model (c, d).

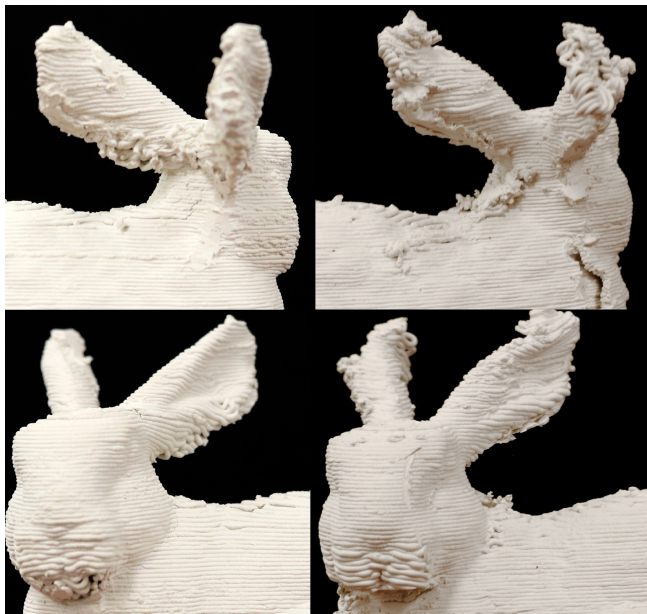


Fig. 12. When using the reference slicer on the *Stanford Bunny*, the transfer moves between the ears lead to artifacts (right). These artifacts are reduced using our method (left).

Our method has higher material requirements than other support generation strategies for thermoplastic materials. However, none of these existing strategies lead to acceptable results for clay printing; see also Fig. 10. We argue that the increase in quality and the range of models that can be printed with acceptable quality is well worth the additional material. Moreover, the excess material can be reused without expensive machines or hazardous chemicals.

For some models, there is a trade off between leaving support removal marks and accepting the artifacts that would result from transfer moves. Predicting such cases is difficult, and the user has to resort to experience. However, our technique significantly increases reliability. Considering the cost of a failed print—wasted

time, wasted material, and cleanup—it is often preferable to stay on the reliable side of the trade-off.

As with existing techniques for generating (non-soluble) supports, removing the non-model structure requires some training and takes time. Thanks to support paths being well separated from the model, this is often an easy task, but it can become more challenging for complex or fragile models.

Overall, compared to existing slicing software, our continuous deposition method substantially improves on the range of models that can be printed in clay. In addition, we feel confident to say from our many experiments that we have not encountered a single case in which the standard method led to better results than our approach. By proposing a solution to the so far unexplored problem of strictly-continuous deposition, our work takes a step towards making clay printing more reliable and useful for end users.

Beyond ceramics, there are many other materials of practical relevance that lend themselves to a paste extrusion process, including concrete [Buswell et al. 2018], silicone [Zehnder et al. 2017], and edible substances [Sun et al. 2018]. While our ceramics-based setup is not suitable for these other materials, experimenting with our continuous support structures on a broader range of paste materials is an interesting direction for future work.

## REFERENCES

- Stephen Allen and Debojyoti Dutta. 1995. Determination and evaluation of support structures in layered manufacturing. *JOURNAL OF DESIGN AND MANUFACTURING* 5 (1995), 153–162.
- Amir Armani, Wenbin Li, Ming Leu, and Gregory Hiltmas. 2016. A novel extrusion-based additive manufacturing process for ceramic parts. In *Proceedings of the 27th Annual Conference on Solid Freeform Fabrication*. 1519–1529.
- Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. 2010. Discrete Viscous Threads. *ACM Trans. Graph.* 29, 4, Article 116 (July 2010), 10 pages. <https://doi.org/10.1145/1778765.1778853>
- Richard A. Buswell, Wilson Ricardo Leal da Silva, Scott Z. Jones, and Justin Dirrenberger. 2018. 3D printing using concrete extrusion: A roadmap for research. *Cement and Concrete Research* 112 (2018), 37–49. <https://doi.org/10.1016/j.cemconres.2018.05.006> SI : Digital concrete 2018.
- Kumar Chalasani and Larry Roscoe. 1995. Support Generation for Fused Deposition Modeling. In *Proc. of Solid Freeform Fabrication Symposium '95*. 229–241.
- Zhangwei Chen, Ziyong Li, Junjie Li, Chengbo Liu, Changshi Lao, Yuelong Fu, Changyong Liu, Yang Li, Pei Wang, and Yi He. 2018. 3D printing of ceramics: A review. *Journal of the European Ceramic Society* (2018). <https://doi.org/10.1016/j.jeurceramsoc.2018.11.013>
- Jordan J. Cox, Yasuko Takezaki, Helaman R.P. Ferguson, Kent E. Kohkonen, and Eric L. Mulkey. 1994. Space-filling curves in tool-path applications. *Computer-Aided Design*

- 26, 3 (1994), 215 – 224. [https://doi.org/10.1016/0010-4485\(94\)90044-2](https://doi.org/10.1016/0010-4485(94)90044-2) Special Issue: NC machining and cutter-path generation.
- Chengkai Dai, Charlie C. L. Wang, Chenming Wu, Sylvain Lefebvre, Guoxin Fang, and Yong-Jin Liu. 2018. Support-free Volume Printing by Multi-axis Motion. *ACM Trans. Graph.* 37, 4, Article 134 (July 2018), 14 pages. <https://doi.org/10.1145/3197517.3201342>
- Donghong Ding, Zengxi (Stephen) Pan, Dominic Cuiuri, and Huijun Li. 2014. A tool-path generation strategy for wire and arc additive manufacturing. *The International Journal of Advanced Manufacturing Technology* 73, 1 (01 Jul 2014), 173–183. <https://doi.org/10.1007/s00170-014-5808-5>
- J r mie Dumas, Jean Hergel, and Sylvain Lefebvre. 2014. Bridging the Gap: Automated Steady Scaffoldings for 3D Printing. *ACM Trans. Graph.* 33, 4, Article 98 (July 2014), 10 pages. <https://doi.org/10.1145/2601097.2601153>
- Chlo e Fleming, Stephanie Walker, Callie Branyan, Austin Nicolai, G. Hollinger, and Yigit Meng . 2017. *Toolpath Planning for Continuous Extrusion Additive Manufacturing*. Technical Report. Oregon State University.
- Google. 2018. Google OR Tools. <https://developers.google.com/optimization/>.
- Clement Gosselin, Romain Duballet, Philippe Roux, Nadja Gaudilliere, Justin Dirrenberger, and Philippe Morel. 2016. Large-scale 3D printing of ultra-high performance concrete – a new processing route for architects and builders. *Materials and Design* 100 (2016), 102 – 109. <https://doi.org/10.1016/j.matdes.2016.03.097>
- Frank H ndle. 2007. *Extrusion in Ceramics* (1 ed.). Springer. <https://doi.org/10.1007/978-3-540-27102-4>
- Jean Hergel and Sylvain Lefebvre. 2014. Clean color: Improving multi-filament 3D prints. *Computer Graphics Forum* 33, 2 (2014), 469–478. <https://doi.org/10.1111/cgf.12318> arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12318>
- Samuel Hornus and Sylvain Lefebvre. 2018. Iterative Carving for Self-supporting 3D Printed Cavities. In *Proceedings of the 39th Annual European Association for Computer Graphics Conference: Short Papers (EG)*. Eurographics Association, Goslar Germany, Germany, 41–44. <http://dl.acm.org/citation.cfm?id=3308470.3308484>
- Samuel Hornus, Sylvain Lefebvre, J r mie Dumas, and Fr d ric Claux. 2016. Tight Printable Enclosures and Support Structures for Additive Manufacturing. In *Proceedings of the Eurographics Workshop on Graphics for Digital Fabrication (GraD-Fab '16)*. Eurographics Association, Goslar Germany, Germany, 11–21. <https://doi.org/10.2312/gdf.20161074>
- Yijiang Huang, Juyong Zhang, Xin Hu, Guoxian Song, Zhongyuan Liu, Lei Yu, and Ligang Liu. 2016. FrameFab: Robotic Fabrication of Frame Shapes. *ACM Trans. Graph.* 35, 6, Article 224 (Nov. 2016), 11 pages. <https://doi.org/10.1145/2980179.2982401>
- Timothy Langlois, Ariel Shamir, Daniel Dror, Wojciech Matusik, and David I. W. Levin. 2016. Stochastic Structural Analysis for Context-aware Design and Fabrication. *ACM Trans. Graph.* 35, 6, Article 226 (Nov. 2016), 13 pages. <https://doi.org/10.1145/2980179.2982436>
- Lin Lu, Andrei Sharf, Haisen Zhao, Yuan Wei, Qingnan Fan, Xuelin Chen, Yann Savoye, Changhe Tu, Daniel Cohen-Or, and Baoquan Chen. 2014. Build-to-last: Strength to Weight 3D Printed Objects. *ACM Trans. Graph.* 33, 4, Article 97 (July 2014), 10 pages. <https://doi.org/10.1145/2601097.2601168>
- Francesco Mezzadri, Vladimir Bouriakov, and Xiaoping Qian. 2018. Topology optimization of self-supporting support structures for additive manufacturing. *Additive Manufacturing* 21, April (2018), 666–682. <https://doi.org/10.1016/j.addma.2018.04.016>
- Stefanie Mueller, Sangha Im, Serafima Gurevich, Alexander Teibrich, Lisa Pfisterer, Fran ois Guimbretiere, and Patrick Baudisch. 2014. WirePrint: 3D Printed Previews for Fast Prototyping. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 273–280. <https://doi.org/10.1145/2642918.2647359>
- Cengiz Oztireli, Ga l Guennebaud, and Markus Gross. 2009. Feature Preserving Point Set Surfaces based on Non-Linear Kernel Regression. *Computer Graphics Forum* (2009). <https://doi.org/10.1111/j.1467-8659.2009.01388.x>
- Ryan Schmidt and Nobuyuki Umetani. 2014. Branching support structures for 3D printing. In *SIGGRAPH Studio*. ACM, 9:1.
- Ondrej Stava, Juraj Vanek, Bedrich Benes, Nathan Carr, and Radomir M ch. 2012. Stress Relief: Improving Structural Strength of 3D Printable Objects. *ACM Trans. Graph.* 31, 4, Article 48 (July 2012), 11 pages. <https://doi.org/10.1145/2185520.2185544>
- Giorgio Strano, Liang Hao, Richard M. Everson, and Ken E. Evans. 2013. A new approach to the design and optimisation of support structures in additive manufacturing. *The International Journal of Advanced Manufacturing Technology* 66, 9 (01 Jun 2013), 1247–1254. <https://doi.org/10.1007/s00170-012-4403-x>
- Jie Sun, Weibiao Zhou, Liangkun Yan, Dejian Huang, and Lien ya Lin. 2018. Extrusion-based food printing for digitalized food design and nutrition control. *Journal of Food Engineering* 220 (2018), 1 – 11. <https://doi.org/10.1016/j.jfoodeng.2017.02.028>
- Thibault Tricard, Fr d ric Claux, and Sylvain Lefebvre. 2019. Ribbed Support Vaults for 3D Printing of Hollowed Objects. *Computer Graphics Forum* 0, 0 (2019). <https://doi.org/10.1111/cgf.13750> arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13750>
- Nobuyuki Umetani and Ryan Schmidt. 2013. Cross-sectional Structural Analysis for 3D Printing Optimization. In *SIGGRAPH Asia 2013 Technical Briefs (SA '13)*. ACM, New York, NY, USA, Article 5, 4 pages. <https://doi.org/10.1145/2542355.2542361>
- Ranji. Vaidyanathan, Joseph Walish, John L. Lombardi, Sridhar Kasichainula, Paul W. Calvert, and Kenneth Cooper. 2000. The extrusion freeforming of functional ceramic prototypes. *JOM* 52, 12 (01 Dec 2000), 34–37. <https://doi.org/10.1007/s11837-000-0066-4>
- Juraj Vanek, Jorge Galica Galicia, and Benes Benes. 2014. Clever Support: Efficient Support Structure Generation for Digital Fabrication. In *Proceedings of the Symposium on Geometry Processing (SGP '14)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 117–125. <https://doi.org/10.1111/cgf.12437>
- Weiming Wang, Yong-Jin Liu, Jun Wu, Shengjing Tian, Charlie C. L. Wang, Ligang Liu, and Xiuping Liu. 2018. Support-Free Hollowing. *TVCG* 24, 10 (2018), 2787–2798.
- Weiming Wang, Tuanfeng Y. Wang, Zhouwang Yang, Ligang Liu, Xin Tong, Weihua Tong, Jiansong Deng, Falai Chen, and Xiuping Liu. 2013. Cost-effective Printing of 3D Objects with Skin-frame Structures. *ACM Trans. Graph.* 32, 6, Article 177 (Nov. 2013), 10 pages. <https://doi.org/10.1145/2508363.2508382>
- Chenming Wu, Chengkai Dai, Guoxin Fang, Yong-Jin Liu, and Charlie C. L. Wang. 2017. RoboFDM: A robotic system for support-free fabrication using FDM. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 1175–1180. <https://doi.org/10.1109/ICRA.2017.7989140>
- Jun Wu, Niels Aage, R diger Westermann, and Ole Sigmund. 2018. Infill Optimization for Additive Manufacturing – Approaching Bone-Like Porous Structures. *IEEE Transactions on Visualization and Computer Graphics* 24, 2 (Feb 2018), 1127–1140. <https://doi.org/10.1109/TVCG.2017.2655523>
- Jun Wu, Charlie C.L. Wang, Xiaoting Zhang, and R diger Westermann. 2016b. Self-supporting rhombic infill structures for additive manufacturing. *Computer-Aided Design* 80 (2016), 32 – 42. <https://doi.org/10.1016/j.cad.2016.07.006>
- Rundong Wu, Huaishu Peng, Fran ois Guimbretiere, and Steve Marschner. 2016a. Printing Arbitrary Meshes with a 5DOF Wireframe Printer. *ACM Trans. Graph.* 35, 4, Article 101 (July 2016), 9 pages. <https://doi.org/10.1145/2897824.2925966>
- Y. Yang, Han Tong Loh, Jerry Fuh Ying Hsi, and Y.G. Wang. 2002. Equidistant path generation for improving scanning efficiency in layered manufacturing. *Rapid Prototyping Journal* 8, 1 (2002), 30–37. <https://doi.org/10.1108/13552540210413284> arXiv:<https://doi.org/10.1108/13552540210413284>
- Jonas Zehnder, Stelian Coros, and Bernhard Thomaszewski. 2016. Designing Structurally-sound Ornamental Curve Networks. 35, 4, Article 99 (2016), 99:1–99:10 pages.
- Jonas Zehnder, Espen Knoop, Moritz B cher, and Bernhard Thomaszewski. 2017. Metasilicone: design and fabrication of composite silicone with desired mechanical properties. *ACM Trans. Graph.* 36, 6 (2017), 240:1–240:13. <https://doi.org/10.1145/3130800.3130881>
- Xiaoting Zhang, Xinyi Le, Athina Panotopoulou, Emily Whiting, and Charlie C. L. Wang. 2015. Perceptual Models of Preference in 3D Printing Direction. *ACM Trans. Graph.* 34, 6, Article 215 (Oct. 2015), 12 pages. <https://doi.org/10.1145/2816795.2818121>
- Haisen Zhao, Fanglin Gu, Qi-Xing Huang, Jorge Garcia, Yong Chen, Changhe Tu, Bedrich Benes, Hao Zhang, Daniel Cohen-Or, and Baoquan Chen. 2016. Connected Fermat Spirals for Layered Fabrication. *ACM Trans. Graph.* 35, 4, Article 100 (July 2016), 10 pages. <https://doi.org/10.1145/2897824.2925958>
- Haisen Zhao, Hao Zhang, Shiqing Xin, Yuanmin Deng, Changhe Tu, Wenping Wang, Daniel Cohen-Or, and Baoquan Chen. 2018. DSCarver: Decompose-and-spiral-carve for Subtractive Manufacturing. *ACM Trans. Graph.* 37, 4, Article 137 (July 2018), 14 pages. <https://doi.org/10.1145/3197517.3201338>
- Qingnan Zhou, Julian Panetta, and Denis Zorin. 2013. Worst-case Structural Analysis. *ACM Trans. Graph.* 32, 4, Article 137 (July 2013), 12 pages. <https://doi.org/10.1145/2461912.2461967>
- Andrea Zocca, Paolo Colombo, Cynthia M. Gomes, and Jens G jnjster. 2015. Additive Manufacturing of Ceramics: Issues, Potentialities, and Opportunities. *Journal of the American Ceramic Society* 98, 7 (2015), 1983–2001. <https://doi.org/10.1111/jace.13700> arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/jace.13700>