Shape from Release: Inverse Design and Fabrication of Controlled Release Structures

JULIAN PANETTA, University of California, Davis, USA
HALEH MOHAMMADIAN, Max Planck Institute for Informatics, Germany
EMILIANO LUCI, Max Planck Institute for Informatics, Germany
VAHID BABAEI, Max Planck Institute for Informatics, Germany

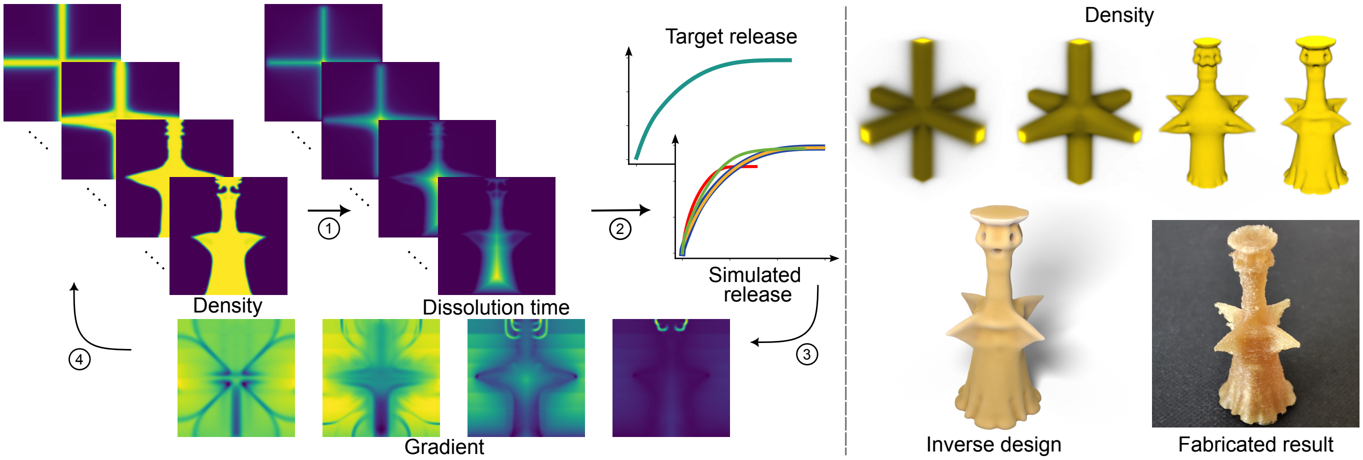


Fig. 1. Our pipeline optimizes for a geometry that achieves a target release curve. Left: starting from the density field at the current iteration, we simulate the release process by ① computing the dissolution time as the solution to a variable-speed Eikonal equation and ② evaluating the release curve by calculating an integral over the domain. The deviation from the target release profile is then measured and differentiated with respect to the density variables ③ to obtain the design objective gradient and update the design ④. Right: visualization of the volumetric density fields describing the evolving 3D design along with the final extracted object and fabricated result.

Objects with different shapes can dissolve in significantly different ways inside a solution. Predicting different shapes' dissolution dynamics is an important problem especially in pharmaceutics. More important and challenging, however, is controlling the dissolution via shape, *i.e.*, designing shapes that lead to a desired release behavior of materials in a solvent over a specific time. Here, we tackle this challenge by introducing a computational inverse design pipeline. We begin by introducing a simple, physically-inspired differentiable forward model of dissolution. We then formulate our inverse design as a PDE-constrained topology optimization that has access to analytical derivatives obtained via sensitivity analysis. Furthermore, we incorporate fabricability terms in the optimization objective that enable physically realizing our designs. We thoroughly analyze our approach on a diverse set of examples via both simulation and fabrication.

Authors' addresses: Julian Panetta, University of California, Davis, USA, jpanetta@ucdavis.edu; Haleh Mohammadian, Max Planck Institute for Informatics, Germany, hmohamma@mpi-inf.mpg.de; Emiliano Luci, Max Planck Institute for Informatics, Germany, eluci@mpi-inf.mpg.de; Vahid Babaei, Max Planck Institute for Informatics, Germany, vbabaei@mpi-inf.mpg.de.

0730-0301/2022/12-ART274
https://doi.org/10.1145/3550454.3555518

CCS Concepts: • **Applied computing** → **Computer-aided manufacturing**; • **Computing methodologies** → **Shape Modeling**; **Simulation**.

Additional Key Words and Phrases: Computational design, controlled release, 3D printing, drug delivery systems, topology optimization, Eikonal equation, sensitivity analysis

ACM Reference Format:
Julian Panetta, Haleh Mohammadian, Emiliano Luci, and Vahid Babaei. 2022. Shape from Release: Inverse Design and Fabrication of Controlled Release Structures. *ACM Trans. Graph.* 41, 6, Article 274 (December 2022), 14 pages. https://doi.org/10.1145/3550454.3555518

1 INTRODUCTION

The dissolution process is a ubiquitous yet fascinating phenomenon. Objects with the same mass but different shapes can dissolve via different dynamics, resulting in vastly different *release profiles*. This property can be exploited, particularly in pharmaceutics, when designing controlled-release systems for drugs, food, and cosmetics. The surge of 3D printing with its unique capabilities for fabricating complex shapes opens new doors for devising novel controlled-release designs [Goole and Amighi 2016].

The first step toward design of controlled-release structures is to understand and model the dissolution process as a function of the shape. Such *forward models* of dissolution have been developed at different levels of abstraction from submolecular simulations [Sagui and Darden 1999] to more approximate but efficient Monte Carlo approaches [Briese et al. 2017] to continuum mechanics models [Zunino et al. 2009]. However, finding a design that matches a desired release profile using only a forward model would require a painstaking manual trial-and-error approach. An outright *inverse design* strategy, where the design is directly optimized to match a target release profile, is crucial to efficiently discover performant designs. Although a handful of existing works address the problem of tuning release profiles [Lu and Anseth 1999], they are limited to predefined setups, such as multi-laminated systems and handicapping assumptions, such as one-dimensional dissolution. To the best of our knowledge, we are the first to consider the problem of *shape from release* for freeform 3D printable geometries.

We propose a gradient-based inverse design framework for the problem of architected release. The backbone of our solution is a simple forward model, based on a geometric abstraction, that is governed by the Eikonal equation. We formulate our problem as a PDE-constrained topology optimization whose objective is to fit the release curve computed from the Eikonal equation solution to the target release profile. We present an exact sensitivity analysis of the discrete fast marching solution and obtain a robust and efficient simulation model with analytical derivatives. Our formulation enables gradient-based optimization for discovering shapes achieving novel release behaviors and exploring trade-offs between release curve closeness and fabricability. We thoroughly analyze our approach on a diverse set of release profiles, both hand crafted and from known shapes. In summary, our contributions are:

- A simple and efficient geometric approach to the release simulation problem that is competitive in accuracy with past methods while being differentiable.
- An adjoint sensitivity analysis for our forward model to support fast gradient calculation.
- A topology optimization algorithm for inverse design of 3D printable geometries that match a prescribed release curve.
- A novel image-based measurement setup for acquiring the physical release curves of 3D printed objects.

2 RELATED WORK

Inverse Design for Functional Fabrication. Functional fabrication refers to generating designs that follow functional goals, or *performances* [Bermano et al. 2017]. The computational design and fabrication community has invented many inverse design frameworks to achieve a diverse set of functional goals related to mechanical properties [Schumacher et al. 2015; Stava et al. 2012; Bickel et al. 2010; Prévost et al. 2013], appearance [Sumin et al. 2019; Babaei et al. 2017; Matusik et al. 2009; Mitra and Pauly 2009; Schüller et al. 2014; Schwartzburg et al. 2014; Piovarči et al. 2020], motion [Bächer et al. 2014; Tang et al. 2020; Coros et al. 2013], photonics [Auzinger et al. 2018; Minkov et al. 2020], buoyancy [Wang and Whiting 2016], acoustics [Li et al. 2016], and more. Most computational approaches solve functional fabrication problems by first simulating the forward process and then inverting the simulation in order to estimate designs meeting a given target performance. In recent years, additive manufacturing has raised new opportunities and challenges for computational design. The most important challenge is navigating the vast design space, leading to the increasing popularity of topology optimization (TO). Once primarily concerned with structural performance [Bendsoe and Sigmund 2013], TO is now the go-to tool for many computationally complex design problems spanning a broad range of physics and application domains [Osanov and Guest 2016]; it enables the discovery of optimal geometries entirely from scratch, requiring no application-specific design parametrization, prior knowledge of the design's topology, or expert-provided initial guess. Our work demonstrates that, thanks to our geometric abstraction of the dissolution process, the functional fabrication problem of inverse release curve design can be tackled by an efficient TO algorithm.

Additive Manufacturing of Release Structures. 3D printing of devices (e.g., tablets) with controlled release has a relatively long history in pharmaceutics (Goole and Amighi [2016] provide a comprehensive review). Multi-material printing, mostly binder-jetting, offers considerable flexibility for designing release profiles as active ingredients can be encapsulated in placebo barrier scaffolds [Katstra et al. 2000; Lam et al. 2002; Thakral et al. 2013]. Filament Deposition Modelling (FDM) has recently gained significant popularity for 3D printing tablets with controlled release [Goyanes et al. 2015; Skowyra et al. 2015; Pietrzak et al. 2015]. So far, most studies of FDM printed structures have focused on empirical evaluation of the effect of the geometry and infill patterns on the release profile. In this work, we employ an FDM 3D printer as our fabrication platform and use a water-soluble filament, typically used as the support material, as the solute.

Forward and Inverse Release. The dissolution process, and other phenomena emerging from intermolecular interactions, are studied in thermodynamics and physical chemistry. These molecular (and submolecular) phenomena can be modeled from first principles via, for example, Density Functional Theory (DFT) [Dreizler and Gross 2012]. Although these first-principles simulations are powerful and general [Sagui and Darden 1999], they are limited at most to a few thousands of atoms. Therefore, higher-level methods like stochastic simulations are popular for capturing the dynamics of larger systems, including the dissolution process [Briese et al. 2017]. Nisser et al. [2019] use a Monte Carlo (MC) approach to predict the overall dissolution time of 3D printed water-soluble support structures. In their approach, each voxel is considered as an atom in a crystalline lattice structure and is eroded with some probability at each simulation time step depending on its connectivity to other voxels. In this work, we compare our proposed distance-based forward model to the MC approach. Siepmann and Siepmann [2008] provide a broader survey of the mathematical models of the release process that have been developed for various device types; most either neglect the effects of geometry or are derived for simple geometries like spheres and cylinders, for which analytical formulas can be obtained. Our method can be seen as a generalization of the semi-empirical formulas derived in [Hopfenberg 1976; Cooney 1972], reformulating the simulation as an Eikonal equation to address arbitrary geometries.

To the best of our knowledge, the problem of computational inverse design of controlled release structures has only been considered for multi-laminates, *i.e.*, matrix devices where strata with different concentrations or thicknesses are juxtaposed [Georgiadis and Kostoglou 2001; Nauman et al. 2010; Zhang 2020; Lu and Anseth 1999]. The common goal of these works is to achieve a desired release profile by tuning the concentrations and/or thicknesses of a handful of layers. For computational simplicity, all but one face of the matrix are insulated using an impermeable shell resulting in a one-dimensional diffusion problem. The forward simulation is then modelled using Fick's law [Fick 1855; Treybal 1980] and the inverse design is formulated as a continuous optimization which is solved via variational calculus [Lu et al. 1998]. Our computational approach works for single-material designs, thus removing the need for a second material, *i.e.*, an impermeable placebo shell. Furthermore, compared to the multi-stage manufacturing of multilaminates, our method enjoys a practical advantage.

Optimal Control of the Eikonal Equation. Using our geometric abstraction, the controlled release design problem can be posed as a PDE-constrained optimization where the state equation (constraint) is the Eikonal equation. This same mathematical formulation has been used to solve travel-time tomography problems in seismology and electromagnetism [Leung and Qian 2006] and, on the design side, to control the propagation of light within a medium by optimizing its composition [Seyyedrezaei and Dadashzadeh 2016]. These works use the iterative fast sweeping method to solve both the forward and adjoint equations, but [Deckelnick et al. 2011] show how both can be solved using the efficient fast marching method; the approach we take is essentially equivalent, but we provide a more accessible derivation (Section 3.5) and address fully general objective functions rather than the specific tomography objective. Finally, Ospald and Herzog [2017] use the Eikonal equation to estimate the fiber orientation produced by the injection molding process for fiber-reinforced plastics, whose anisotropic stiffness strongly influences the elastic behavior of the resulting structure; they propose a compliance minimization TO problem constrained by both linear elasticity and the Eikonal equation to account for this anisotropy during structural optimization.

3 METHOD

Our central goal is to formulate a differentiable forward model of the release process for a given object geometry $\Omega \subset \mathbb{R}^3$ and then employ it in a gradient-based design optimization framework. The following subsections detail the components of our full system for designing 3D printable geometries to achieve an input release curve.

3.1 Simulation of a Fully Solid Object

Motivated by the Noyes-Whitney equation (Appendix A), we propose a simplified geometric model for simulating the release of an object into a stirred solvent that is competitive with other more established simulation approaches and yet is amenable to efficient sensitivity analysis. The fundamental underlying assumption of our model is that the entire exposed boundary of the object dissolves at the same rate. This rate will depend on the current concentration level in the solvent, but to simplify the exposition, we further suppose that the rate is constant. We show in Appendix B that concentration-dependent rates can be handled by a nonlinear remapping of the simulation time, however this was not necessary in the concentration regime of our dissolution experiments.

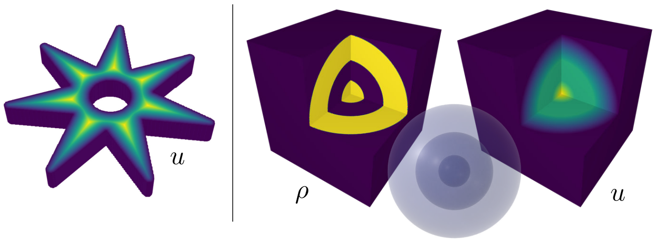


Fig. 2. Left: our geometric abstraction approximates the dissolution time of each point in a solid shape as the distance-to-boundary u, visualized here on a cross-section of a star geometry. Right: geometries with enclosed voids and variable densities are handled by solving a variable-speed Eikonal equation on a rectangular domain, demonstrated here for a sphere nested in a spherical shell. Note that the entire inner void shell region ($\rho = 0$) dissolves at the same instant, releasing zero mass.

By *defining* the simulation time unit to be the length of time required to dissolve through a thickness of 1 mm, we conclude that the time at which a given point $\mathbf{x}$ within a fully solid design Ω dissolves is the point's distance $u(\mathbf{x})$ to the boundary $\partial\Omega$ (Figure 2). Accordingly, the cumulative release at simulation time τ can be computed directly from the distance field $u : \Omega \to \mathbb{R}^+$:

$$C(\tau) = \int_\Omega \rho_0 \{u < \tau\}\,\mathrm{d}\mathbf{x} = \int_\Omega \rho_0\, H(\tau - u)\,\mathrm{d}\mathbf{x}, \quad H(t) := \begin{cases} 1 & t > 0 \\ 0 & t \le 0 \end{cases},$$

where ρ_0 is the mass density of the fabrication material, and H is the Heaviside step function. The distance field u can be computed in a number of ways (*e.g.*, fast marching [Sethian 1999] or the Geodesics in Heat approach [Crane et al. 2017]) and can be characterized as a viscosity solution of the unit-speed Eikonal equation:

$$\begin{aligned} \|\nabla u\| &= 1 \quad \text{in} \quad \Omega \\ u &= 0 \quad \text{on} \quad \partial\Omega. \end{aligned}$$

3.2 Simulation of a Density Field

The formula for $C(\tau)$ just presented has a few limitations. It can simulate only fully solid objects without holes, and it is not clear how to incorporate it in a topology optimization. To address both limitations, we generalize our simulation model by adopting a density-based object representation. Denoting the rectangular design domain (*e.g.*, a 3D printer's build volume) by $\mathcal{D} \subset \mathbb{R}^3$, we represent the object by a *design density function* $\rho : \mathcal{D} \to [0, 1]$ (distinct from the physical material density ρ_0, which we assume in the following to be $\rho_0 = 1$ without loss of generality). For instance, to represent $\Omega \subseteq \mathcal{D}$, now possibly with holes, we take ρ to be the indicator function χ_Ω, which assumes value 1 inside Ω and 0 outside. Our insight for simulating an object of this type is to interpret low density values as material that dissolves at extremely high speed and solve a variable-speed Eikonal equation on $\mathcal{D}$:

$$\begin{aligned} \|\nabla u\| &= n \quad \text{in} \quad \mathcal{D} \\ u &= 0 \quad \text{on} \quad \partial\mathcal{D}, \end{aligned} \tag{1}$$

where $n : \mathcal{D} \to (0, \infty)$ is the *reciprocal* dissolution speed field sometimes called the "slowness field" (speed is $1/n$). This formulation automatically handles structures like interior holes and nested shells, where the inner shell starts dissolving immediately after the outer shell fully dissolves (Figure 2). We use an interpolation law to define n in terms of the design density:

$$n = \rho + (1 - \rho)n_{\min}, \tag{2}$$

where $n_{\min}$ is a small value used to avoid numerical issues in the Eikonal equation solver associated with extreme speeds (we set $n_{\min} = 10^{-6}$ for all experiments). This simplest choice of a linear interpolation law is natural since one would expect, *e.g.*, a half-occupied voxel to dissolve in half the time. Note that since we will penalize our final designs to be nearly binary, the precise choice of interpolation law only influences intermediate designs evaluated during the optimization.

After solving for the "dissolution time" field u, we evaluate the cumulative release curve by integrating over the full design domain:

$$C(\tau) = \int_{\mathcal{D}} \rho\, H(\tau - u)\, \mathrm{d}\mathbf{x}. \tag{3}$$

3.3 Topology Optimization Formulation

Given a target release curve $C^*(\tau)$, we formulate the inverse design problem as the optimization:

$$\begin{aligned} &\min_{0 \le \hat\rho \le 1} J_{\text{fit}}(\rho) + w_b J_{\text{binary}}(\rho) + w_f J_{\text{fab}}(\rho) \\ &J_{\text{fit}} := \frac{1}{T}\int_0^T (C(\tau) - C^*(\tau))^2 \,\mathrm{d}\tau \\ &J_{\text{binary}} := \int_{\mathcal{D}} \rho(1-\rho)\,\mathrm{d}\mathbf{x} \\ &\rho = \texttt{project}(\texttt{smooth}(\hat\rho, r), \beta), \end{aligned} \tag{4}$$

where $\texttt{smooth}(\cdot, r)$ is a standard smoothing filter with a kernel that linearly decays to 0 over the *smoothing radius r* [Bruns and Tortorelli 2001], and `project` is the threshold projection filter from [Wang et al. 2011]:

$$\texttt{project}(\tilde\rho, \beta) := \frac{\tanh(\beta/2) - \tanh(\beta(\tilde\rho - 1/2))}{2\tanh(\beta/2)}.$$

Together, these filters avoid nonphysical mesh-dependency issues like checkerboard patterns while retaining a binary design. J_{binary} is a standard term penalizing non-binary designs [Schumacher et al. 2015], and J_{fab} defined in Section 3.6 is a term promoting robustly printable designs. The total simulation duration T is determined by the maximum possible dissolution time for a fully solid design; for a cube-shaped $\mathcal{D}$, this is half the cube width.

3.4 Discretization and Solvers

We discretize the design domain $\mathcal{D}$ with a regular grid of N cube voxels having edge length Δx. The density ρ is discretized as a piecewise constant field, meaning we define one design variable per voxel. When solving (1), we pad the grid with a layer of zero-density voxels on which the $u = 0$ boundary condition is applied. This is to ensure that parts of the object that intersect the domain boundary are not considered already dissolved at $\tau = 0$.

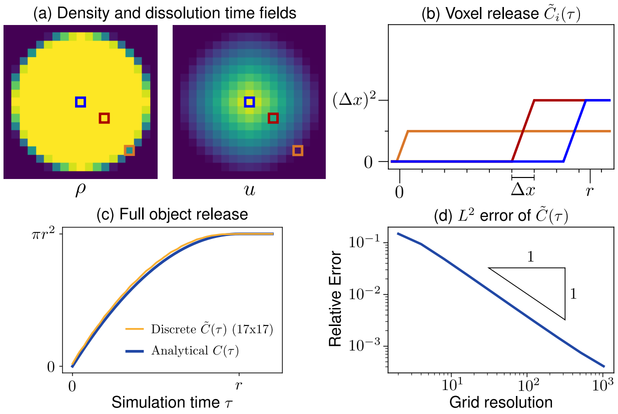


Fig. 3. Simulation process in 2D: using the density and fast marching dissolution time fields, (a) we sum each voxel's linear ramp release curve (visualized in (b) for the three highlighted voxels) to obtain the discrete release curve (c). Under grid refinement, the resulting discrete approximation exhibits first-order convergence to the continuous release curve $C(\tau)$, validated in (d) using an analytical ground-truth formula for the disk's release.

We solve (1) using the implementation of the fast marching method provided by `scikit-fmm` [Furtney 2021], which we modified to track the information necessary for performing the exact adjoint sensitivity analysis discussed in the next subsection. This method is based on a finite difference discretization of the Eikonal equation, and the code supports a second-order scheme for improved accuracy. Unfortunately, the more complicated stencil selection rules involved in the second-order scheme degrade the differentiability of solution u (and thus the computed release curve), and so we used the first-order scheme for all inverse design. Denoting the dissolution time at voxel i by u_i and the neighbor of voxel i in the positive or negative direction along the d axis as $\mathcal{N}(i, \pm d)$, the first-order scheme in 3D solves the discrete equation:

$$\sum_{d=1}^{3} \frac{1}{(\Delta x)^2} \max(u_i - u_{\mathcal{N}(i,d)}, u_i - u_{\mathcal{N}(i,-d)}, 0)^2 = n_i^2 \quad \forall i. \tag{5}$$

Note that the max operation automatically selects the correct upwind stencil for each dimension.

The dissolution time u_i determined by (5) indicates the time at which the voxel i has *fully dissolved*; this interpretation is apparent from studying a 1D version of the discretization. It would be both nonphysical and yield a nondifferentiable model to assume that the entire voxel dissolves instantaneously at time u_i. Instead, we assume that the voxel dissolves at a constant rate over the time interval $(u_i - n_i\Delta x, u_i)$. In other words, voxel i contributes to the release curve a scaled linear ramp function:

$$\tilde{C}_i(\tau) := \rho_i(\Delta x)^3 \operatorname{clamp}\left(\frac{\tau - u_i}{n_i \Delta x} + 1, 0, 1\right). \tag{6}$$

Summing these contributions over the voxels obtains the following discretization of (3):

$$\tilde{C}(\tau) := \sum_{i=1}^{N} \tilde{C}_i(\tau) = \sum_{i=1}^{N} \rho_i \tilde{H}\left(\frac{\tau - u_i}{n_i \Delta x}\right)(\Delta x)^3, \tag{7}$$

where we recognize $\tilde{H}(\tau) := \mathrm{clamp}(\tau + 1, 0, 1)$ to be a smeared version of the Heaviside function.

We discretize the time integral in J_{fit} by picking a time step size $\Delta\tau$ and computing $\frac{1}{T}\sum_i(\tilde{C}(i\Delta\tau) - C^*(i\Delta\tau))^2\Delta\tau$. We use $\Delta\tau = T/1000$ for our results. We note that a literal implementation of (7), summing up the full vector of concentration samples $\tilde{C}(i\Delta\tau)$, has computational complexity $O\left(N\frac{T}{\Delta\tau}\right)$, which becomes a bottleneck when high temporal resolution is requested. This can be accelerated to $O\left(N + \frac{T}{\Delta\tau}\right)$ by exploiting the fact that the second derivative of ramp function (6) is highly sparse, consisting of just two delta functions; therefore, we can accumulate the *second finite difference* of the discrete release curve in $O(N)$ and then integrate it twice (with cumulative sums) to obtain $\tilde{C}(i\Delta\tau)$.

We solve the discrete version of (4) using Knitro's L-BFGS implementation. This algorithm requires the gradients of the design objectives $J_{\mathrm{fit}}, J_{\mathrm{binary}}, J_{\mathrm{fab}}$ with respect to the filtered density variables ρ which are then back-propagated through the density filter chain to obtain the gradients with respect to the design variables $\hat{\rho}$. $J_{\mathrm{binary}} = \sum_{i=1}^{N}\rho_i(1-\rho_i)(\Delta x)^3$ can be directly differentiated, and J_{fab} will be defined and differentiated in Section 3.6. Differentiating J_{fit} is more challenging since it requires derivatives of our forward model; we obtain these efficiently with exact sensitivity analysis using the adjoint method as described in the next section.

3.5 Sensitivity Analysis

While a continuous adjoint equation can be derived for (1), care must be taken to discretize it consistently with the discretization of (1) itself. We prefer to derive the consistent adjoint equation for the discrete equation directly. We begin with an abstract formulation of the discrete Eikonal equation (5):

$$\mathbf{c}(\mathbf{u}, \mathbf{n}) = \mathbf{0}, \tag{8}$$

where $\mathbf{u}$ and $\mathbf{n}$ are vectors collecting the dissolution times and reciprocal speeds of each voxel, and $\mathbf{c}$ is a vector of nonlinear constraint functions, one per voxel. Interpreting this equation as implicitly defining a dissolution time function $\mathbf{u}(\mathbf{n})$, we consider a generic objective function $\tilde{J}(\mathbf{n}) = \hat{J}(\mathbf{u}(\mathbf{n}), \mathbf{n})$ and seek its gradient:

$$\frac{\partial \tilde{J}}{\partial \mathbf{n}} = \frac{\partial \hat{J}}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial \mathbf{n}} + \frac{\partial \hat{J}}{\partial \mathbf{n}}, \tag{9}$$

where $\frac{\partial \mathbf{u}}{\partial \mathbf{n}}$ is the derivative of the solution to the discrete Eikonal equation that we obtain by differentiating both sides of (8):

$$\frac{\partial \mathbf{c}}{\partial \mathbf{u}}\frac{\partial \mathbf{u}}{\partial \mathbf{n}} = -\frac{\partial \mathbf{c}}{\partial \mathbf{n}} \quad \Longleftrightarrow \quad \frac{\partial \mathbf{u}}{\partial \mathbf{n}} = -\left(\frac{\partial \mathbf{c}}{\partial \mathbf{u}}\right)^{-1}\frac{\partial \mathbf{c}}{\partial \mathbf{n}}.$$

Substituting this expression into (9) obtains an explicit gradient:

$$\frac{\partial \tilde{J}}{\partial \mathbf{n}} = -\frac{\partial \hat{J}}{\partial \mathbf{u}}\left(\frac{\partial \mathbf{c}}{\partial \mathbf{u}}\right)^{-1}\frac{\partial \mathbf{c}}{\partial \mathbf{n}} + \frac{\partial \hat{J}}{\partial \mathbf{n}} = -\lambda\frac{\partial \mathbf{c}}{\partial \mathbf{n}} + \frac{\partial \hat{J}}{\partial \mathbf{n}} = \lambda\,\mathtt{diag}(2\mathbf{n}) + \frac{\partial \hat{J}}{\partial \mathbf{n}},$$

which is efficient to calculate after solving the *adjoint equation*:

$$\left(\frac{\partial \mathbf{c}}{\partial \mathbf{u}}\right)^{\top}\lambda = \frac{\partial \hat{J}}{\partial \mathbf{u}}.$$

Structure of the Adjoint Equation. Sparse Jacobian matrix $\frac{\partial \mathbf{c}}{\partial \mathbf{u}}$ can be interpreted as a (weighted) adjacency matrix for the Eikonal equation's data flow graph: the i^{th} equation of (8) determines u_i, and the nonzero entries in this row indicate the variables influencing u_i (the upwind stencil neighbors). This data flow graph has a special acyclic structure that is exploited by the fast marching method to solve the equation in a single pass: sorting the entries of $\mathbf{u}$ in the order the marcher visits them (*i.e.*, ascending) makes matrix $\frac{\partial \mathbf{c}}{\partial \mathbf{u}}$ lower triangular, and the marching algorithm effectively performs a forward substitution process. The adjoint equation features the *transpose* of $\frac{\partial \mathbf{c}}{\partial \mathbf{u}}$; transposition reverses all graph edges and results in an *upper* triangular system that can be solved efficiently in a single *backward* substitution sweep. We implement this sweep efficiently by recording the order in which voxels are visited by the fast marcher and then simply visiting them in *reverse order.*

Formulas for the Adjoint Sweep. To simplify notation, the following assumes that voxels are sorted in ascending order of u_i so that $\frac{\partial \mathbf{c}}{\partial \mathbf{u}}$ is lower triangular. The i^{th} adjoint equation is:

$$\sum_j \frac{\partial c_j}{\partial u_i}\lambda_j = \frac{\partial \hat{J}}{\partial u_i} \quad \Longleftrightarrow \quad \frac{\partial c_i}{\partial u_i}\lambda_i = \frac{\partial \hat{J}}{\partial u_i} - \sum_{j>i}\frac{\partial c_j}{\partial u_i}\lambda_j, \tag{10}$$

where we used the fact that $\frac{\partial \mathbf{c}}{\partial \mathbf{u}}^{\top}$ is upper triangular to ignore entries $j < i$ in the sum. This formula makes it clear that λ_i can be calculated directly once λ_j is found for $j > i$, motivating the reverse sweep. The "boundary conditions" for this reverse sweep appear naturally as well in this equation: any voxel m with no downwind neighbors (i.e., a local maxima of arrival time) will have an empty sum on the right-hand-side, and so its corresponding adjoint equation is simply $\lambda_m = \frac{\partial \hat{J}}{\partial u_m} / \frac{\partial c_m}{\partial u_m}$.

3.6 Fabrication Constraints

To simplify fabrication, we 3D print our geometries without support structures. Therefore they must be free of overhanging features. We do this in one of two ways: (1) generating 2D extruded designs or (2) using the layer-by-layer additive manufacturing approach proposed in [Allaire et al. 2017] that we have adapted to our setting of density-based topology optimization.

For our 2D extrusion design space, we define the density variables on a 2D grid that is replicated in the third dimension up to an extrusion height h, which the user can specify according to, *e.g.*, packaging concerns. We perform the fast marching simulation and sensitivity analysis on the 3D grid and then backpropagate the 3D gradient through the replication step to get the gradient in the 2D design space.

For our fully 3D designs, we evaluate the structural compliance under self weight of a small number of partial designs created at intermediate steps of the incremental, layer-by-layer fabrication process. We refer to the average compliance across all simulations as the *incremental compliance* objective (ICO), which we use as J_{fab}. Specifically, we divide the design along the build direction into L layers, and for the design consisting of layers 1 through l, we perform a linear elasticity finite element simulation under self weight load $\mathbf{f}^{(l)}(\rho)$. This entails solving the linear system

$K^{(l)}(\rho)\mathbf{u}^{(l)} = \mathbf{f}^{(l)}(\rho)$ for the displacements $\mathbf{u}^{(l)}$, and we use a high-performance multigrid-preconditioned conjugate gradient solver. The stiffness matrix K is determined from ρ using the standard SIMP interpolation law using the typical penalization parameter setting $p = 3$. We then evaluate the incremental compliance:

$$J_{\text{fab}} := \frac{1}{2L} \sum_{i=1}^{L} \mathbf{f}^{(l)}(\rho) \cdot \mathbf{u}^{(l)}.$$

Structural compliance is self-adjoint, meaning we can efficiently compute its gradient without solving an adjoint equation:

$$\frac{\partial J_{\text{fab}}}{\partial \rho} = \frac{1}{L} \sum_{l=1}^{L} \left(\frac{\partial \mathbf{f}^{(l)}}{\partial \rho} \cdot \mathbf{u}^{(l)} - \frac{1}{2} \mathbf{u}^{(l)} \cdot \frac{\partial K^{(l)}}{\partial \rho} \mathbf{u}^{(l)} \right).$$

4 EXPERIMENTAL SETUP

We validate our simulation model and inverse design pipeline by 3D printing geometries and measuring their release using a custom experimental setup detailed in the following subsections.

4.1 Fabrication

We fabricate all models using a consumer-level, FDM 3D printer (Prusa i3 MK3s) equipped with a 0.2 mm Kaika nozzle from Tecdia Co. We use AquaSys® 120 from Infinite Materials Solutions™, a water-soluble filament typically used as support structure. We tune the printing profile (CURA 4.13 slicer) to achieve a consistent average print density across different objects (1.32 g/cm^3) and set the nozzle and bed temperatures to the manufacturer recommendations of 245 °C and 85 °C, respectively. The line width is set to 0.2 mm, layer height to 0.1 mm, infill to 100%, and print speed to 200 mm/s. We also found that the concentric infill pattern results in prints that more closely match our simulation model (Figure 5).

4.2 Image-Based Release Measurement

Measuring the dissolution process is important for both the evaluation of our approach and for calibration (Section 4.3). In the pharmaceutical industry, the dissolution process is typically measured using relatively high-cost, specialized dissolution testing instruments. Although devices based on viscosity gauging could be employed, we opt for an image-based setup. This is because our printing material absorbs visible light and allows for a simple, *online* measurement system using an off-the-shelf digital camera. Our setup (Figure 4a) is based on a Nikon D750 DSLR, a Walimex pro LED Sirius 160 Bi Color 65W LED panel, Fisherbrand 600 mL Boro 3.3 beakers (FB33114), and a Heidolph Instruments MR Hei-Tec 145 mm diameter hotplate. We use distilled water from Dr. Starke® as our solvent. The LED panel is oriented parallel to the image plane and positioned such that its center coincides with the image center. An additional diffusing board is placed in front of the LED panel. We turn on the light and the camera around 30 minutes before the capture to let them reach a steady state. We fill the beaker with 500 g of water and cover it with a silicone-edged lid to prevent evaporation. Throughout the experiment, we maintain the water at a constant temperature (70°C) and use a stirring magnet (at 150 rpm) to homogenize the solution. We place the stirring magnet inside a hand-designed holder that reduces turbulence. We capture an image of the water to use as reference and then drop the object inside the beaker. We record images of the dissolution process every 11 seconds using an f/22 aperture, 1/125 s exposure, and native 160 ISO. We stop the capture when we observe the concentration to be steady. The release curve measurement (including heating the water to 70°C) can take up to 3-4 hours depending on the shape of the object.

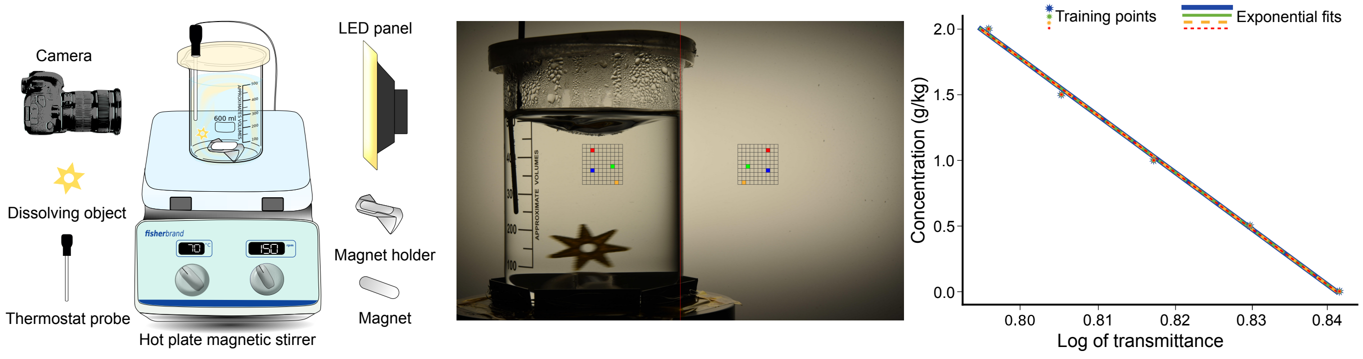
We map the intensity of the captured images to the corresponding concentrations of the solution. We find this mapping through a fitting procedure. We dissolve known quantities of the filament in a fixed amount (500 g) of water and take pictures once the solution is homogeneous. We add pieces of filament to the solution incrementally, waiting for each to completely dissolve before adding the next. As shown in Figure 4b, we select a square crop region of the image inside the beaker and then subdivide this area into a 10x10 grid. To safeguard against possible light variation, we normalize the pixels within the grid by dividing their intensities channel-wise by the intensities of their mirrored counterparts outside the beaker (mirroring across a vertical axis running through the image center). We then average the normalized values over each square, obtaining for each square a single normalized transmittance value per capture. Then, separately for each square, we fit an exponential function to the series of known concentration and normalized transmittance pairs. Our choice of an exponential function is motivated by the negligible scattering of the filament, meaning that Beer-Lambert's law is applicable. The result is a separate map from known concentrations to transmittances for each square that we invert and then average to obtain the final mapping from transmittance to concentration. Images can be noisy due to capture errors or intrusions of the object into the crop region, which manifests as noise in the measured release curves; we employ a Savitzky–Golay filter to smooth the curves [Savitzky and Golay 1964].

4.3 Calibration

Our measurements and simulations of dissolution have different time and concentration units. While measurement is done in physical units (g/kg and seconds), the simulation uses non-physical units for both time and concentration. *Calibrating* the system consists of building a mapping between these units. For this purpose, we print a set of different shapes of equal mass, measure their release profiles, and simulate them using our forward model. We assume that there exist two decoupled linear mappings between the physical concentration and time units and their simulation counterparts. The scalar coefficient defining the linear map for concentration is given simply by the ratio of the final solution concentration in physical units (determined precisely by weighing the object before dissolution) to the simulated final concentration $\tilde{C}(T)$. The time mapping is computed by minimizing the following cost function for the scalar coefficient $k \in \mathbb{R}$:

$$\mathcal{L}(C_s, C_r, k) = \int w_s \left| k C_s'(kt) - C_r'(t) \right| + \left| C_s(kt) - C_r(t) \right| \mathrm{d}t,$$

where C_s and C_r are the cumulative release profiles in simulation time and in real time, respectively, and weight parameter w_s (set to 1 in our implementation) adjusts the relative importance of fitting the curve *slopes*.



(a) (b) (c)

Fig. 4. (a) Our acquisition setup. (b) Image capture of the beaker overlaid with the chosen crop region grid and its mirrored counterpart used for normalization. (c) Measured and fitted exponential relationship between transmittance and concentration shown for the highlighted grid squares.

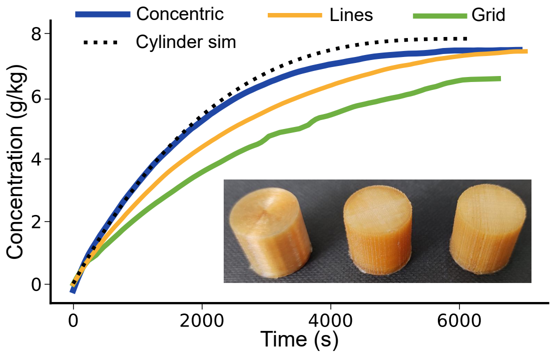


Fig. 5. The same cylinder printed with different infill patterns shows a significantly different release behavior.

The calibration step also compensates for mismatches between our digital designs and their fabricated counterparts. While our forward model assumes a homogeneously filled, isotropic solid, the parts fabricated using FDM exhibit prominent infill-dependent heterogeneities and highly anisotropic properties aligned with the filament direction. In this work, we use the concentric infill pattern for printing all objects as it yielded the highest density. Figure 5 shows a single geometry printed with various infill patterns and measured using our image-based setup. We compare all measurements against the simulated release curve of the digital model, which has been transformed into physical units using our calibration (performed on a set of objects printed with concentric infill). Thanks to the calibration, the measured release profile of the print with concentric infill matches the simulation significantly better than the other infill prints' curves.

5 EVALUATION

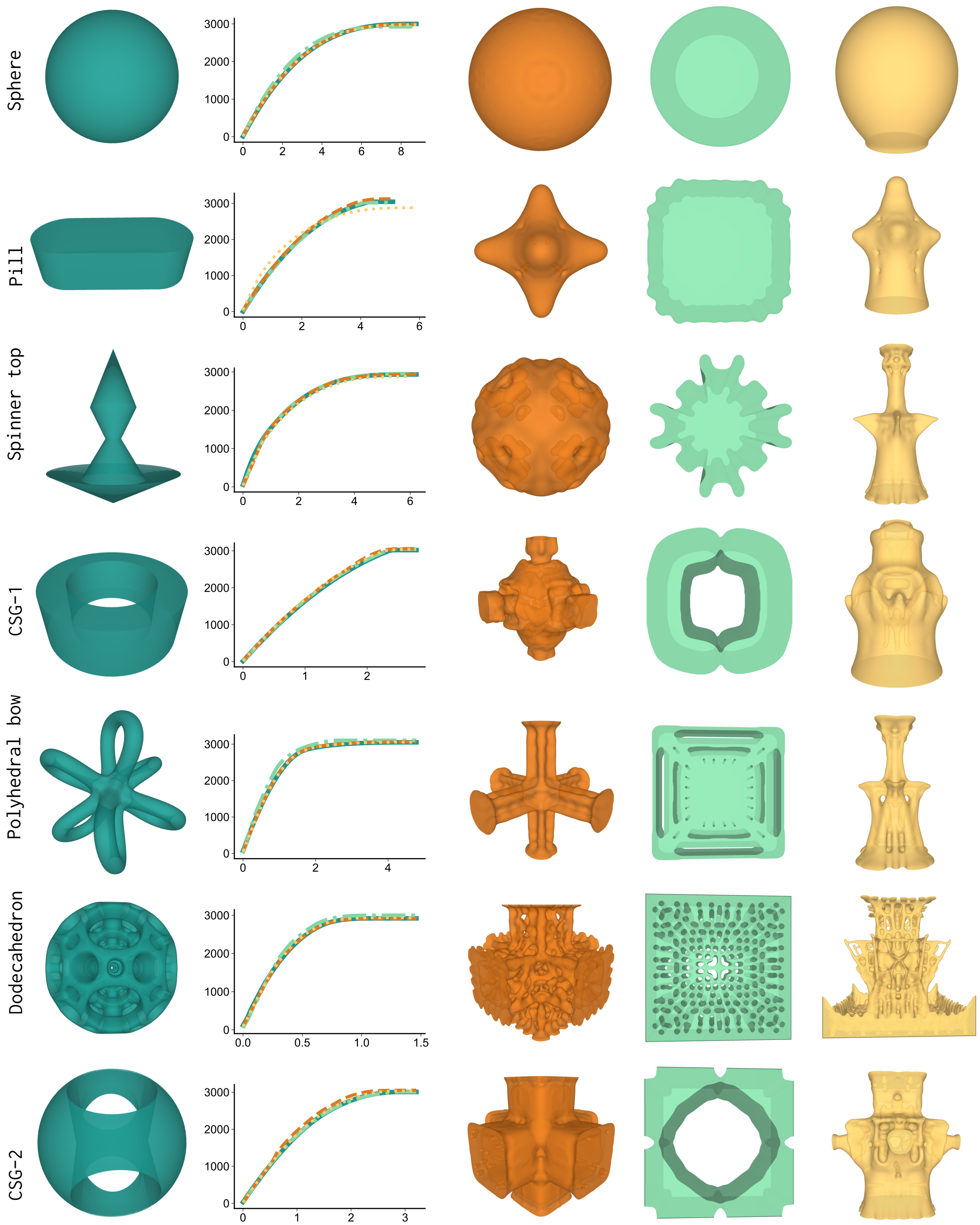
We first evaluate the accuracy of our forward model (in conjunction with the calibration step) and compare it with a standard Monte Carlo approach. Then, we put our inverse design pipeline to work, showing an extensive set of experiments both in simulation and with physical measurements.

5.1 Forward Model Evaluation

We evaluate the accuracy of our forward model by fabricating 5 objects with different shapes but roughly the same mass (~ 1.75 g). For each model, we measure and simulate their release profiles. The simulated release profiles are converted into physical units using our calibration step (Section 4.3) and shown side-by-side with measured release curves in Figure 8. We follow a cross-validation approach where the measurement and simulation data of each individual shape is excluded for computing its calibration. We also include the results of the Monte Carlo simulation, which has been used in the past for forward modeling of the release of 3D printing materials [Nisser et al. 2019]. We tune two important parameters of the MC simulation, namely the pre-exponential factor and the activation energy in the Arrhenius equation, by searching over a grid of 16 combinations [Pauling 2014]. Figure 8 shows the results of the forward model evaluation. The release curves simulated by our method show good agreement with the measured curves—on par with or better than the MC predictions. Apart from its accuracy, our method is distinguished from MC by its differentiability.

5.2 Inverse Design Evaluation

In order to factor out the effects of calibration, our first set of inverse design validation experiments starts from a source shape, simulates its release profile using our forward model, and sets that profile as the target $C^*(\tau)$ for inverse design. For each target, we run three variants of our inverse method with: no fabricability constraint (NFC), the extrusion of a 2D density field (Extruded), and a fully 3D solution with incremental compliance (ICO) penalized by J_{fab}. Figure 6 shows the resulting set of diverse inverse-designed geometries. We observe that our inverse method in all of its variants is able to accurately fit to the target release curve. Note that the plotted release profiles are a result of a full forward simulation of the solid geometries shown in Figure 6 and not simply the release curve evaluated on the density field found by inverse optimization. Specifically, we extracted the $\rho = \frac{1}{2}$ contour of the optimal density field and then voxelized the resulting solid into a purely binary design for simulation.



(a) Source shape (b) Release profiles (c) NFC (d) Extruded (e) ICO

Fig. 6. Inverse designs evaluated in simulation. Starting from a *source shape* (a), we simulate its release curve (solid green line in (b)) and optimize for release-equivalent shapes (c-e) with different fabricability constraints. The resulting release curves are plotted in dashed lines in (b). All axes use simulation units.

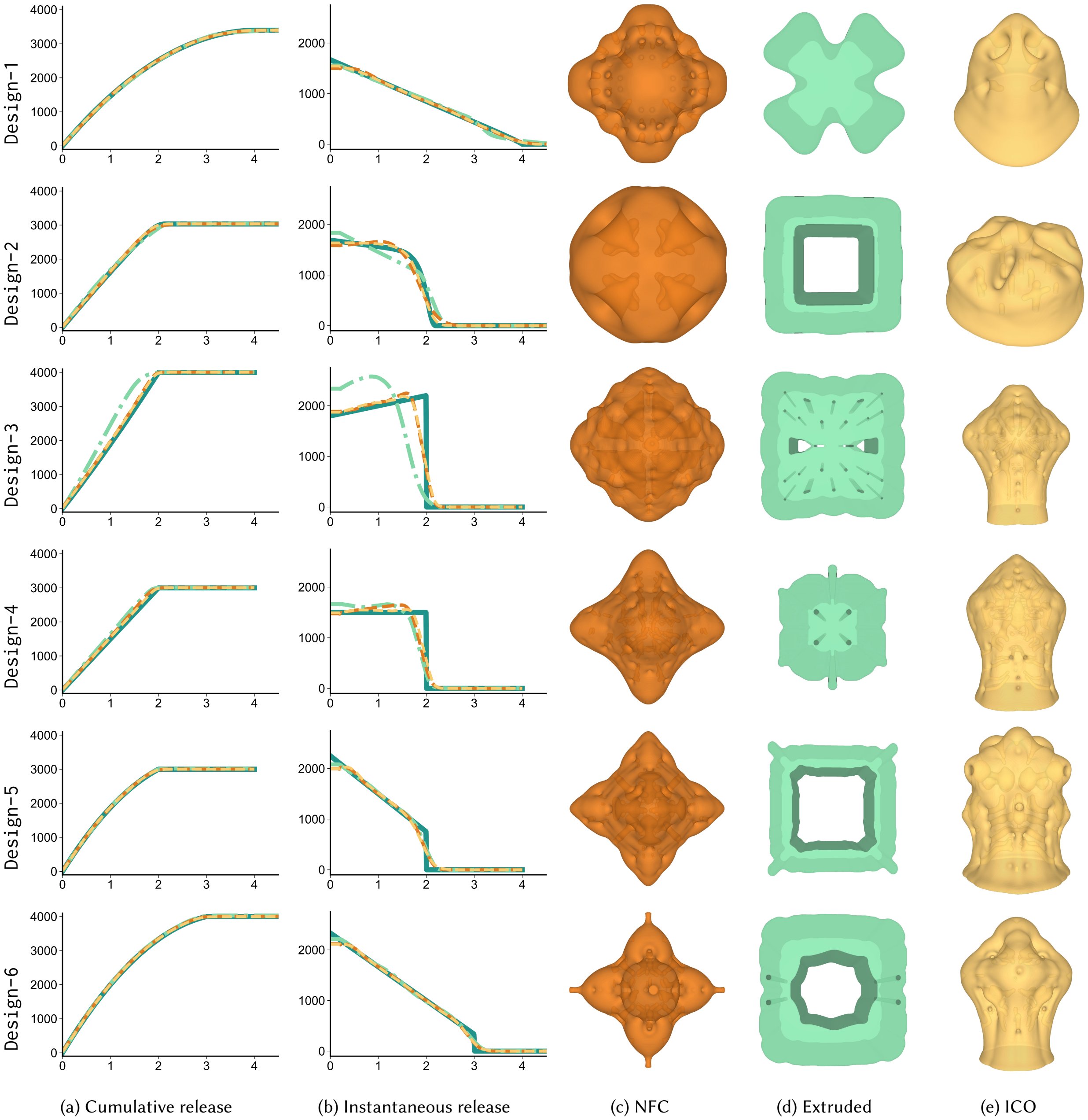


Fig. 7. Geometries fit to *custom-designed* target cumulative release curves (solid green curves in (a)) using different fabrication constraints on the design space (c-e). The resulting simulated cumulative release curves are plotted in dashed lines in (a), and the instantaneous release is validated in (b).

Apart from the high accuracy in matching the release profiles (apparent in Figure 6b), it is interesting to see the *release equivalency* between wildly different shapes. This observation highlights the large degree of multi-modality of the shape-from-release problem. The first row, `sphere`, is a simple validation where, when not constrained, our method recovers the sphere shape from our uniform initialization. Note also how the ICO constraint helps obtain a shape that does not require support by avoiding problematic overhang angles; the NFC solution without this term features unprintable overhangs. The extruded solutions show high accuracy given that they are fast to compute and lend themselves to a wide range of manufacturing methods. We note that the release of the extruded designs match those of sources with extruded geometries (`Pill` and `CSG-1`) perfectly. This is because the extrusion constraint regularizes the problem very well. However, the extruded design space

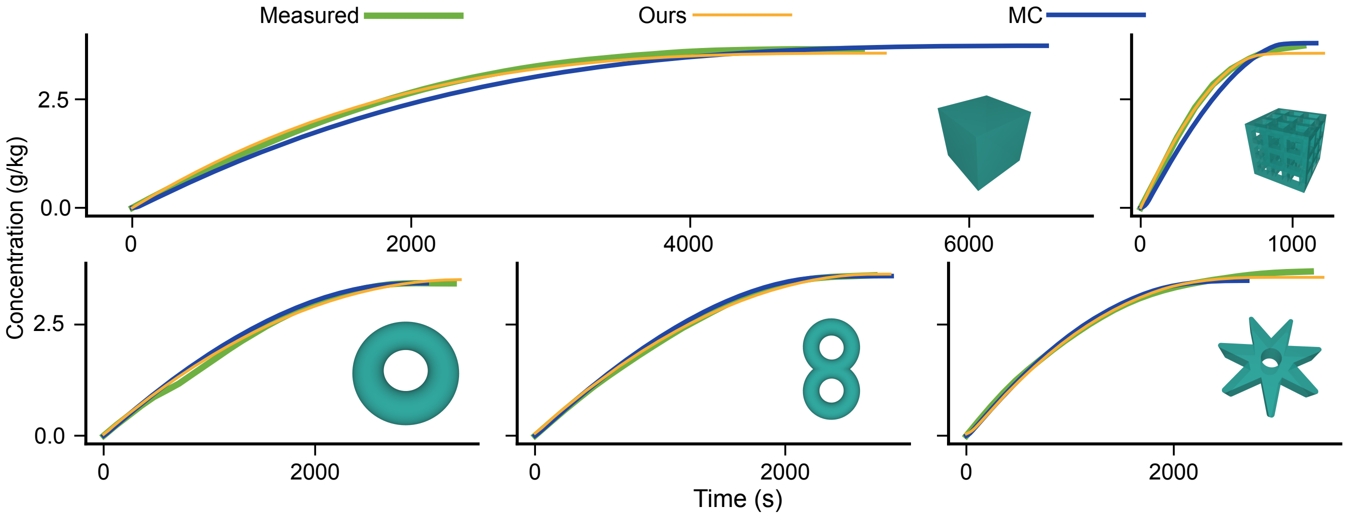


Fig. 8. Evaluation of our forward model in conjunction with our proposed calibration and in comparison with the Monte Carlo simulation (MC).

lacks sufficient expressiveness to emulate more complex source shapes like the `Polyhedral bow` and `Dodecahedron` geometries.

To demonstrate the performance of our designs in a real-world setup, Figure 9a shows a set of printed and measured release-equivalent shapes selected from Figure 6 based on the printability of the source geometry. Note that the computed inverse designs are in general more complex than the geometries used in our forward model evaluation (Figure 8). Except for the ICO design counterpart to the source shape `CSG-1`, we observe reasonable agreement between the target release curve (obtained by simulating the release of the source geometry and converting it into real units using our calibration) and the measured release of the printed geometries. However, the ICO design for `CSG-1` has a significantly slower measured release behavior than predicted by its simulation (which nearly coincides with the target curve shown). We hypothesize that the main source of this inconsistency is the relatively narrow channels within this design. To verify this, we remove the channels from the mesh, print the resulting geometry, and measure its release. As Figure 10 shows, this modification results in a close match between simulation and measurement. Since we verify printing accuracy before each dissolution experiment, the problem occurs during dissolution. Due to the chemistry of our printing material, we observe that the object swells (Figure 11) inside the solution, causing its narrower channels to become blocked. While the simulation assumes that each layer in contact with water will dissolve at the same time, the blocked channels lead to locally different dissolution dynamics as water cannot flow freely even with mixing.

The previous results validated our inverse design on target release curves known to be feasible as they originate from different source shapes. However, the true power of our framework lies in discovering shapes achieving novel release behavior. In Figure 7, we test our pipeline on synthetic input release curves. While the first two curve are hand-drawn, the rest are analytical curves featuring a variety of *constant-acceleration* release behaviors. Of particular interest to pharmaceutical applications is the zero-acceleration, constant release rate shown by `Design-4`. We observe good agreement between the desired and optimized release profiles, though the extruded design space is insufficiently rich to achieve the more challenging behaviors. Imperfections in the fits for the NFC and ICO designs are introduced almost entirely by the density rounding step (contouring and re-voxelizing the continuous densities, which exhibit near-perfect fits). We note that target curves must obey a simple geometric condition to be feasible: the *isoperimetric inequality*. In particular, at every instant in time, the rate of release must exceed a lower bound established by the surface area of a sphere enclosing the volume remaining to be dissolved (spheres minimize the ratio of surface area to enclosed volume). Finally, in Figure 9b, we verify the performance of our inverse designs for hand-crafted input release profiles using a physical dissolution and measurement.

Implementation details. We initialize the optimization with uniform density, picking the spatially constant $\hat{\rho}$ field such that its total mass equals the target final released mass, *i.e.*$\int_{\mathcal{D}} \rho(\hat{\rho})\, \mathrm{d}\mathbf{x} = C^*(T)$. After the optimization terminates, we obtain the solid design's boundary representation by extracting the $\rho = \frac{1}{2}$ contour of the optimal density field. To accelerate simulation and optimization of high-resolution designs, we enforce reflectional symmetry across all three coordinate planes for the designs without fabrication constraints and across the build platform's coordinate axes for the extruded and ICO designs. We use grids with 256 elements along each dimension and divide the design into $L = 8$ layers to evaluate J_{fab}.

Performance. The computational bottlenecks of our method are the dense linear algebra operations within Knitro's L-BFGS implementation; at $N = 256^3$ resolution, and leveraging reflectional symmetry, these take $\sim$ 8 s per iteration; the fast marching, objective evaluation, and gradient computation together take $\sim$ 2.5 s per iteration, and the elasticity solves for evaluating the ICO term take $\sim$ 3.5 s per iteration. Inverse design using the extruded design space is significantly faster due to the smaller number of optimization variables, taking $\sim$ 0.6 s per iteration in total. These reported

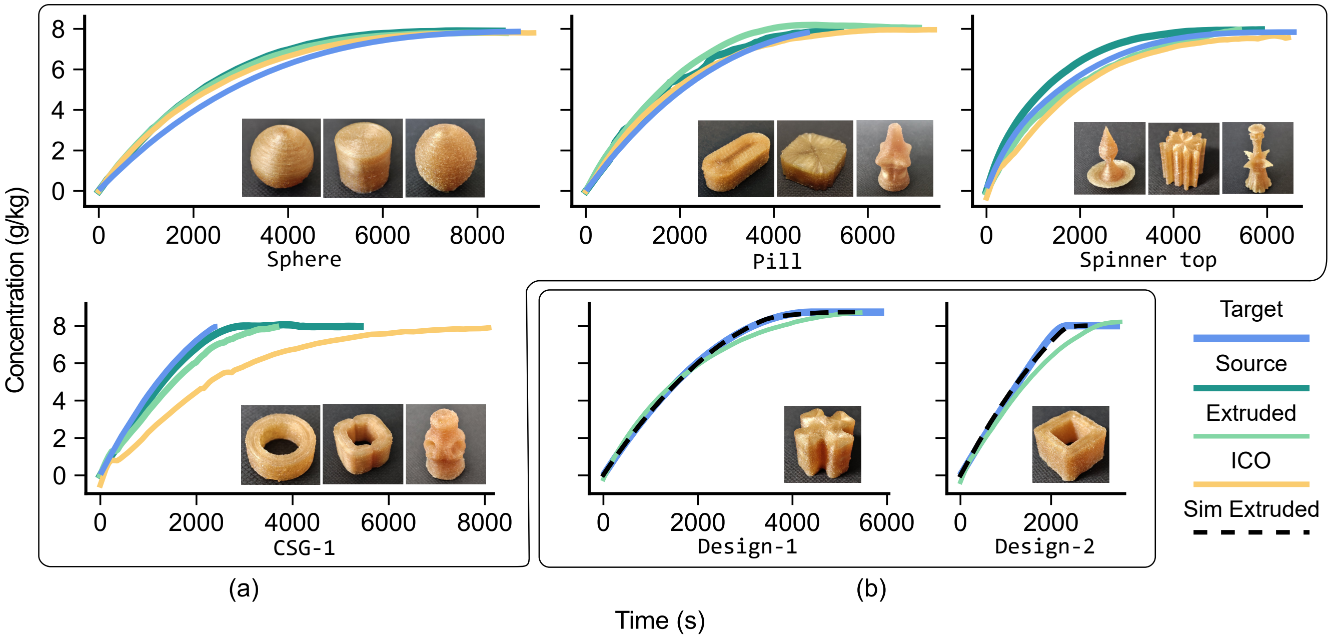


Fig. 9. Physical measurements of fabricated designs. In (a), the measured release curves of the 3D printed source shapes and their corresponding ICO and extruded designs from Figure 6 are plotted against the the target curve used in inverse design (*i.e.*, the *simulated* release of the source shape). The measured release profile of ICO CSG-1 is substantially slower than the target behavior due to the swelling of internal cavities (see Figure 10). In (b), measured releases of inverse designs from Figure 7 are compared against their hand-designed target curves and their simulated release.

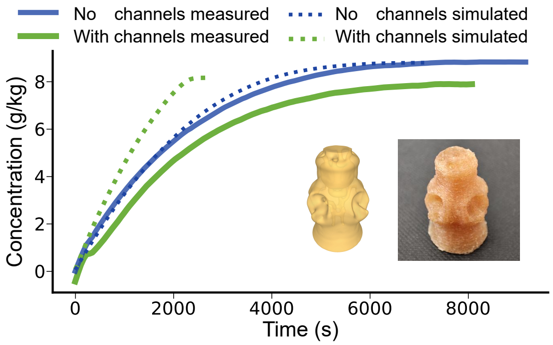


Fig. 10. Comparison of the accuracy of our forward model for a shape with internal channels and the same shape with filled channels. We note that in the physical dissolution process, the inner channels are blocked by swelling material, leading to a mismatch with our prediction.

times are measured on a Ryzen 5750X CPU. Inverse designs were computed with a maximum of 500 iterations.

Hyperparameter tuning. The hyperparameters of our method are: the binary and ICO term weights w_b and w_f, the radius r of the smoothing filter in voxel units, and the steepness β of the projection filter. Typically, for each input release profile, we tune these parameters to obtain a desired shape. The most important performance criterion during tuning is fabricability as the release behavior generally can be obtained without extensive tuning. In case of extrusion, we also manually select the layer height h (in voxels). We seek designs that emerge as a single connected piece and do not exhibit sharp features. In our experience, the most important hyperparameters are the weights of the binary and ICO terms. After finding reasonable settings for these two, we further tune the parameters of our projection and smoothing filters. We have summarized the hyperparameters used for our designs in Tables 1 and 2. Moreover, Figure 12 demonstrates designs that are release-equivalent to the CSG-1 object but optimized with different hyperparameter settings.

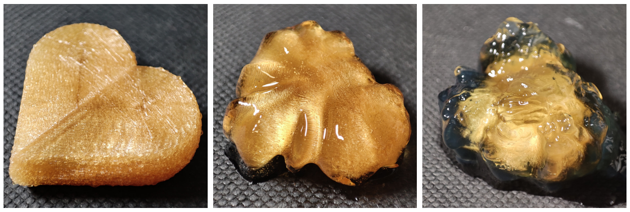
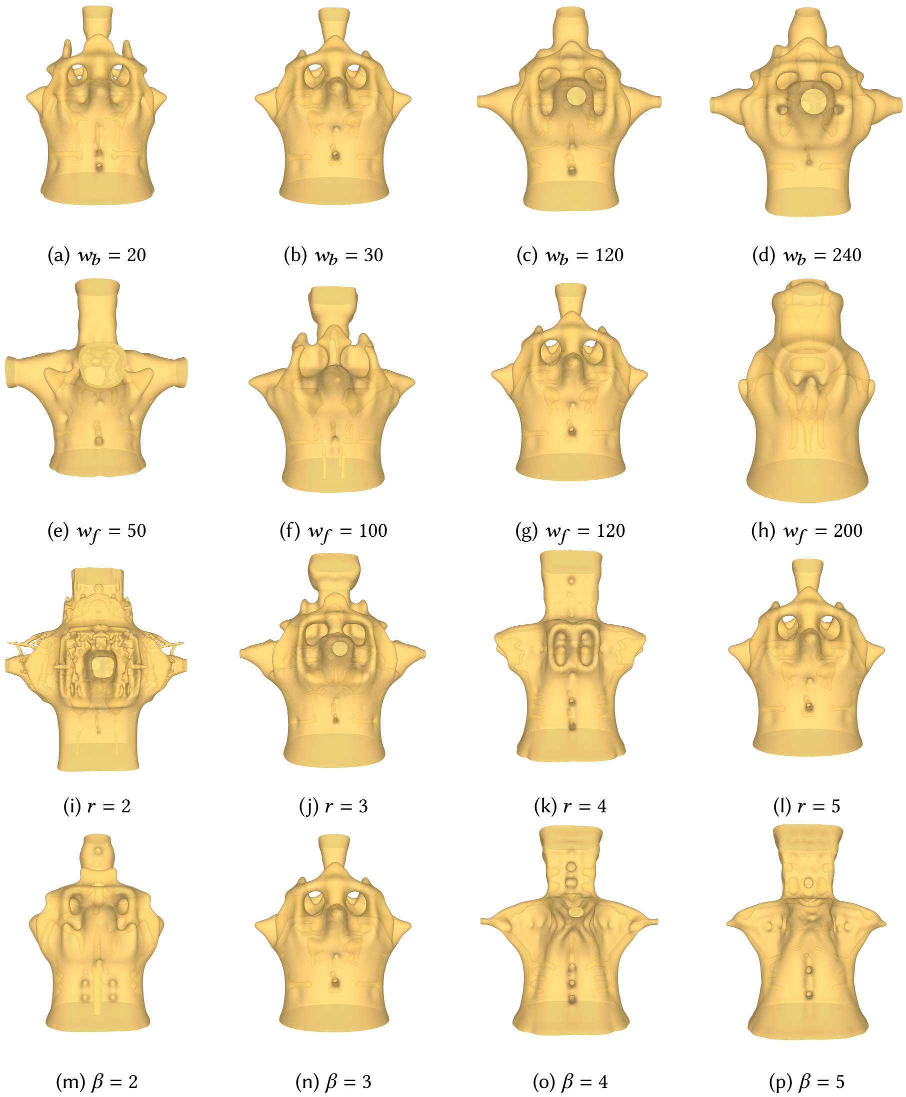


Fig. 11. The evolution of a printed object in undisturbed water. Water seeps into the material, forming a jelly-like substance. This interferes with our release model as the swelling impedes the free flow of water. Though the solution is constantly mixed during our dissolution measurements, here we avoid stirring to simulate local conditions where the flow is minimal.

6 LIMITATIONS AND FUTURE WORK

Some challenging target release profiles are hard to obtain with a single material. Instead of pushing our optimized density field toward binary values, an interesting alternative would be to *dither* mid-range densities. For single-material 3D printing technologies, such as FDM, this can be achieved, for example, using microstructures [Martínez et al. 2018]. On the other hand, we believe our


(a) $w_b = 20$ (b) $w_b = 30$ (c) $w_b = 120$ (d) $w_b = 240$

(e) $w_f = 50$ (f) $w_f = 100$ (g) $w_f = 120$ (h) $w_f = 200$

(i) $r = 2$ (j) $r = 3$ (k) $r = 4$ (l) $r = 5$

(m) $\beta = 2$ (n) $\beta = 3$ (o) $\beta = 4$ (p) $\beta = 5$

Fig. 12. Effect of different hyperparameter settings on optimal designs. From top to bottom: the binary objective term weight w_b, the ICO term weight w_f, the smoothing filter radius r (in voxels), and the projection filter parameter β.

Table 1. Hyperparameters used to compute designs shown in Figure 6.

	ICO				Extruded			
	w_b	w_f	r	β	w_b	h	r	β
Sphere	30	50	4	3	20	160	4	3
Pill	30	120	5	3	20	70	4	3
Spinner top	20	100	4	3	50	120	4	3
CSG-1	30	200	5	3	20	120	4	3
Polyhedral bow	20	120	5	3	20	35	4	3
Dodecahedron	20	50	4	3	50	60	4	3
CSG-2	50	100	4	3	50	120	4	3

Table 2. Hyperparameters used to compute designs shown in Figure 7.

	ICO				Extruded			
	w_b	w_f	r	β	w_b	h	r	β
Design-1	40	200	9	7	50	80	9	7
Design-2	40	200	4	3	50	90	7	5
Design-3	20	10	4	9	40	80	4	9
Design-4	10	10	4	9	10	200	4	9
Design-5	20	10	4	7	20	80	4	3
Design-6	20	10	4	7	20	100	4	7

computational workflow lends itself well to designing, at least, two-material release composites, especially in a scenario where one of the materials acts as a placebo barrier. More exciting will be to design composites in which both release profiles corresponding to both materials are controlled. Our distance-based, forward model proved to be reliable enough for predicting the release behavior for a wide range of shapes under stirred conditions. A potential approach for improving our geometric model's accuracy is to take into account the relationship between release dynamics and other shape descriptors. For example, it is reasonable to assume that regions belonging to the same isosurface but different curvatures can dissolve differently. To implement this modification, we can benefit from research in front-propagation simulation of other physical phenomena [Sethian 1985], such as fire animation. This will, however, require non-trivial changes to our topology optimization.

7 CONCLUSION

The shape of an object can be inferred from a surprisingly wide range of properties, such as shading [Woodham 1980] or interreflections [Nayar et al. 1991]. We introduced a computational framework for computing shapes from a new physical property: its release behavior. Both our physically-inspired, but purely geometric forward modelling, and our PDE-constrained topology optimization proved to be practical solutions for forward prediction and inverse design, respectively. We observed that the problem is highly multi-modal and therefore can be regularized by different fabricability constraints depending on the fabrication technology. With the emergence of pharmaceutical drug 3D printing, we look forward to applying our method in real-world settings and with practically relevant materials.

ACKNOWLEDGMENTS

We are grateful to Serdar Durdyyev, Jigyi Wu, Sebastian Cucerca, Navid Ansari, Parham Zolfaghari, and Sina Ravanbodshirazi for their help with our fabrication and measurement setup.

REFERENCES

Grégoire Allaire, Charles Dapogny, Alexis Faure, and Georgios Michailidis. 2017. Shape optimization of a layer by layer mechanical constraint for additive manufacturing. *Comptes Rendus Mathematique* 355, 6 (2017), 699–717. https://doi.org/10.1016/j.crma.2017.04.008

Thomas Auzinger, Wolfgang Heidrich, and Bernd Bickel. 2018. Computational design of nanostructural color for additive manufacturing. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–16.

Vahid Babaei, Kiril Vidimče, Michael Foshey, Alexandre Kaspar, Piotr Didyk, and Wojciech Matusik. 2017. Color contoning for 3D printing. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–15.

Moritz Bächer, Emily Whiting, Bernd Bickel, and Olga Sorkine-Hornung. 2014. Spin-it: optimizing moment of inertia for spinnable objects. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–10.

Martin Philip Bendsoe and Ole Sigmund. 2013. *Topology optimization: theory, methods, and applications.* Springer Science & Business Media.

Amit H. Bermano, Thomas Funkhouser, and Szymon Rusinkiewicz. 2017. State of the Art in Methods and Representations for Fabrication-Aware Design. *Computer Graphics Forum* 36, 2 (2017), 509–535. https://doi.org/10.1111/cgf.13146 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13146

Bernd Bickel, Moritz Bächer, Miguel A Otaduy, Hyunho Richard Lee, Hanspeter Pfister, Markus Gross, and Wojciech Matusik. 2010. Design and fabrication of materials with desired deformation behavior. *ACM Transactions on Graphics (TOG)* 29, 4 (2010), 1–10.

Laura Briese, Rolf S Arvidson, and Andreas Luttge. 2017. The effect of crystal size variation on the rate of dissolution–A kinetic Monte Carlo study. *Geochimica et Cosmochimica Acta* 212 (2017), 167–175.

Tyler E Bruns and Daniel A Tortorelli. 2001. Topology optimization of non-linear elastic structures and compliant mechanisms. *Computer methods in applied mechanics and engineering* 190, 26-27 (2001), 3443–3459.

David O. Cooney. 1972. Effect of geometry on the dissolution of pharmaceutical tablets and other solids: Surface detachment kinetics controlling. *AIChE Journal* 18, 2 (1972), 446–449. https://doi.org/10.1002/aic.690180234 arXiv:https://aiche.onlinelibrary.wiley.com/doi/pdf/10.1002/aic.690180234

Stelian Coros, Bernhard Thomaszewski, Gioacchino Noris, Shinjiro Sueda, Moira Forberg, Robert W Sumner, Wojciech Matusik, and Bernd Bickel. 2013. Computational design of mechanical characters. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–12.

Keenan Crane, Clarisse Weischedel, and Max Wardetzky. 2017. The Heat Method for Distance Computation. *Commun. ACM* 60, 11 (Oct. 2017), 90–99. https://doi.org/10.1145/3131280

Klaus Deckelnick, Charles M Elliott, and Vanessa Styles. 2011. Numerical analysis of an inverse problem for the eikonal equation. *Numer. Math.* 119, 2 (2011), 245–269.

Reiner M Dreizler and Eberhard KU Gross. 2012. *Density functional theory: an approach to the quantum many-body problem.* Springer Science & Business Media.

Adolf Fick. 1855. Ueber diffusion. *Annalen der Physik* 170, 1 (1855), 59–86.

Jason Furtney. 2021. scikit-fmm: the fast marching method for Python. https://github.com/scikit-fmm/scikit-fmm.

Michael C Georgiadis and Margaritis Kostoglou. 2001. On the optimization of drug release from multi-laminated polymer matrix devices. *Journal of controlled release* 77, 3 (2001), 273–285.

Jonathan Goole and Karim Amighi. 2016. 3D printing in pharmaceutics: A new tool for designing customized drug delivery systems. *International journal of pharmaceutics* 499, 1-2 (2016), 376–394.

Alvaro Goyanes, Pamela Robles Martinez, Asma Buanz, Abdul W Basit, and Simon Gaisford. 2015. Effect of geometry on drug release from 3D printed tablets. *International journal of pharmaceutics* 494, 2 (2015), 657–663.

HB Hopfenberg. 1976. Controlled release from erodible slabs, cylinders, and spheres. ACS Publications.

WE Katstra, RD Palazzolo, CW Rowe, B Giritlioglu, P Teung, and MJ Cima. 2000. Oral dosage forms fabricated by Three Dimensional Printing™. *Journal of controlled release* 66, 1 (2000), 1–9.

Christopher Xu Fu Lam, XM Mo, Swee-Hin Teoh, and DW Hutmacher. 2002. Scaffold development using 3D printing with a starch-based polymer. *Materials Science and Engineering: C* 20, 1-2 (2002), 49–56.

Shingyu Leung and Jianliang Qian. 2006. An adjoint state method for three-dimensional transmission traveltime tomography using first-arrivals. *Communications in Mathematical Sciences* 4, 1 (2006), 249–266.

Dingzeyu Li, David IW Levin, Wojciech Matusik, and Changxi Zheng. 2016. Acoustic voxels: computational optimization of modular acoustic filters. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.

Sanxiu Lu and Kristi S Anseth. 1999. Photopolymerization of multilaminated poly (HEMA) hydrogels for controlled release. *Journal of controlled release* 57, 3 (1999), 291–300.

Sanxiu Lu, W Fred Ramirez, and Kristi S Anseth. 1998. Modeling and optimization of drug release from laminated polymer matrix devices. *AIChE journal* 44, 7 (1998), 1689–1696.

Jonàs Martínez, Samuel Hornus, Haichuan Song, and Sylvain Lefebvre. 2018. Polyhedral Voronoi diagrams for additive manufacturing. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–15.

Wojciech Matusik, Boris Ajdin, Jinwei Gu, Jason Lawrence, Hendrik P. A. Lensch, Fabio Pellacini, and Szymon Rusinkiewicz. 2009. Printing Spatially-varying Reflectance. *ACM Trans. Graph.* 28, 5 (Dec. 2009), 128:1–128:9.

Momchil Minkov, Ian AD Williamson, Lucio C Andreani, Dario Gerace, Beicheng Lou, Alex Y Song, Tyler W Hughes, and Shanhui Fan. 2020. Inverse design of photonic crystals through automatic differentiation. *ACS Photonics* (2020).

Niloy J Mitra and Mark Pauly. 2009. Shadow art. *ACM Transactions on Graphics* 28, CONF (2009), 156–1.

E Bruce Nauman, Kandarp Patel, and Pankaj Karande. 2010. On the design and optimization of diffusion-controlled, planar delivery devices. *Chemical engineering science* 65, 2 (2010), 923–930.

Shree K Nayar, Katsushi Ikeuchi, and Takeo Kanade. 1991. Shape from interreflections. *International Journal of Computer Vision* 6, 3 (1991), 173–195.

Martin Nisser, Junyi Zhu, Tianye Chen, Katarina Bulovic, Parinya Punpongsanon, and Stefanie Mueller. 2019. Sequential support: 3D printing dissolvable support material for time-dependent mechanisms. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction.* 669–676.

Mikhail Osanov and James K Guest. 2016. Topology optimization for architected materials design. *Annual Review of Materials Research* 46 (2016), 211–233.

Felix Ospald and Roland Herzog. 2017. SIMP based topology optimization for injection molding of SFRPs. In *World Congress of Structural and Multidisciplinary Optimisation.* Springer, 850–861.

L. Pauling. 2014. *General Chemistry.* Dover Publications. https://books.google.com/books?id=EpxSzteNvMYC

Katarzyna Pietrzak, Abdullah Isreb, and Mohamed A Alhnan. 2015. A flexible-dose dispenser for immediate and extended release 3D printed tablets. *European journal of pharmaceutics and biopharmaceutics* 96 (2015), 380–387.

Michal Piovarči, Michael Foshey, Vahid Babaei, Szymon Rusinkiewicz, Wojciech Matusik, and Piotr Didyk. 2020. Towards spatially varying gloss reproduction for 3D printing. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–13.

Romain Prévost, Emily Whiting, Sylvain Lefebvre, and Olga Sorkine-Hornung. 2013. Make it stand: balancing shapes for 3D fabrication. *ACM Transactions on Graphics*

(TOG) 32, 4 (2013), 1–10.

Celeste Sagui and Thomas A Darden. 1999. Molecular dynamics simulations of biomolecules: long-range electrostatic effects. *Annual review of biophysics and biomolecular structure* 28, 1 (1999), 155–179.

Abraham. Savitzky and M. J. E. Golay. 1964. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry* 36, 8 (1964), 1627–1639. https://doi.org/10.1021/ac60214a047 arXiv:https://doi.org/10.1021/ac60214a047

Christian Schüller, Daniele Panozzo, and Olga Sorkine-Hornung. 2014. Appearance-mimicking surfaces. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–10.

Christian Schumacher, Bernd Bickel, Jan Rys, Steve Marschner, Chiara Daraio, and Markus Gross. 2015. Microstructures to control elasticity in 3D printing. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 136.

Yuliy Schwartzburg, Romain Testuz, Andrea Tagliasacchi, and Mark Pauly. 2014. High-contrast computational caustic design. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–11.

James A Sethian. 1985. Curvature and the evolution of fronts. *Communications in Mathematical Physics* 101, 4 (1985), 487–499.

J. A. Sethian. 1999. Fast Marching Methods. *SIAM Rev.* 41, 2 (1999), 199–235. https://doi.org/10.1137/S0036144598347059 arXiv:https://doi.org/10.1137/S0036144598347059

Seyyedeh Faezeh Seyyedrezaei and Gholamreza Dadashzadeh. 2016. Optimal Control of Light Propagation Governed by Eikonal Equation within Inhomogeneous Media Using Computational Adjoint Approach. *Journal of Communication Engineering* 5, 1 (2016), 24–37.

L. Shargel, A. Yu, and S. Wu-Pong. 2016. *Applied Biopharmaceutics & Pharmacokinetics, Seventh Edition.* McGraw-Hill Education.

Juergen Siepmann and F Siepmann. 2008. Mathematical modeling of drug delivery. *International journal of pharmaceutics* 364, 2 (2008), 328–343.

Justyna Skowyra, Katarzyna Pietrzak, and Mohamed A Alhnan. 2015. Fabrication of extended-release patient-tailored prednisolone tablets via fused deposition modelling (FDM) 3D printing. *European Journal of Pharmaceutical Sciences* 68 (2015), 11–17.

Ondrej Stava, Juraj Vanek, Bedrich Benes, Nathan Carr, and Radomír Měch. 2012. Stress relief: improving structural strength of 3D printable objects. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–11.

Denis Sumin, Tobias Rittig, Vahid Babaei, Thomas Nindel, Alexander Wilkie, Piotr Didyk, Bernd Bickel, Jaroslav Kurivánek, Karol Myszkowski, and Tim Weyrich. 2019. Geometry-aware scattering compensation for 3D printing. *ACM Trans. Graph.* 38, 4 (2019).

Pengbin Tang, Jonas Zehnder, Stelian Coros, and Bernhard Thomaszewski. 2020. A harmonic balance approach for designing compliant mechanical systems with nonlinear periodic motions. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–14.

Seema Thakral, Naveen K Thakral, and Dipak K Majumdar. 2013. Eudragit®: a technology evaluation. *Expert opinion on drug delivery* 10, 1 (2013), 131–149.

Robert E Treybal. 1980. Mass transfer operations. *New York* 466 (1980).

Fengwen Wang, Boyan Stefanov Lazarov, and Ole Sigmund. 2011. On projection methods, convergence and robust formulations in topology optimization. *Structural and Multidisciplinary Optimization* 43, 6 (2011), 767–784.

Lingfeng Wang and Emily Whiting. 2016. Buoyancy optimization for computational fabrication. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 49–58.

Robert J Woodham. 1980. Photometric method for determining surface orientation from multiple images. *Optical engineering* 19, 1 (1980), 191139.

Xinming Zhang. 2020. An Inverse Problem Solution Scheme for Solving the Optimization Problem of Drug-Controlled Release from Multilaminated Devices. *Computational and Mathematical Methods in Medicine* 2020 (2020).

P. Zunino, C. D'Angelo, L. Petrini, C. Vergara, C. Capelli, and F. Migliavacca. 2009. Numerical simulation of drug eluting coronary stents: Mechanics, fluid dynamics and drug release. *Computer Methods in Applied Mechanics and Engineering* 198, 45 (2009), 3633–3644. https://doi.org/10.1016/j.cma.2008.07.019 Models and Methods in Computational Vascular and Cardiovascular Mechanics.

A PHYSICAL PLAUSIBILITY OF OUR FORWARD MODEL

Our simulation is based on the Noyes-Whitney equation, the standard pharmacokinetic model for predicting dissolution of a solid [Shargel et al. 2016]:

$$\frac{\mathrm{d}m}{\mathrm{d}t} = DA\frac{(C_s - C_b)}{h}, \tag{11}$$

where m is the dissolved mass (kg); D is the diffusion coefficient (m^2/s); A is the exposed surface area (m^2); C_s and C_b are the concentrations of the solid and bulk solvent, respectively (kg/L); and h is the thickness of the concentration gradient between the solid and the bulk solvent (m). This equation is based on Fick's first law of diffusion, which states that the mass flux is proportional to the negative concentration gradient. Implicit in this model is an assumption that, at each instant in time, a layer with uniform infinitesimal thickness is dissolved so that the rate of dissolved mass is proportional to the exposed surface area. This means that, denoting by τ the thickness of the cumulative layer dissolved by time t:

$$\rho A\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{\mathrm{d}m}{\mathrm{d}t},$$

where ρ is the mass density of the solid (kg/m^3).

We make the following simplifying assumptions to obtain a computational efficient and differentiable geometric simulation model:

- D and h do not depend significantly on the exposed geometry (being determined primarily by the choice of solvent and the rate of mixing); and
- the solvent volume into which the geometry is dissolved is sufficiently large that C_b can be regarded as constant ($C_b(t) - C_b(t_0) \ll C_s$). This assumption is not fundamental to our approach, and we analyze the effects of allowing C_b to depend on the dissolved mass in Appendix B.

Under these assumptions, we can regard the entire expression $D\frac{C_s - C_b}{h} := \alpha$ as a *constant* with units kg/(m^2 s):

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \alpha A, \qquad \frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{\alpha}{\rho}.$$

We emphasize that $\frac{\mathrm{d}\tau}{\mathrm{d}t}$ is constant in this model, meaning that the *thickness* of the dissolved shell $S(t)$ grows at a constant rate.

B CONCENTRATION-DEPENDENT RATES

If the solid dissolved is too large or the solvent volume too small, C_b cannot be teated as constant in (11). We can assume without loss of generality that $C_b(t_0) = 0$ (if it were nonzero, we could subtract it from C_s to obtain an equivalent simulation), in which case the concentration in the solvent must come exclusively from the dissolved solid: $C_b(t) = \frac{m(t)}{V}$, where V is the solvent volume (m^3). The assumption of a uniformly thick infinitesimal layer dissolving at each instant remains unchanged (as we still assume the solvent concentration is *spatially homogeneous*), and so:

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \frac{DA}{h}\left(C_s - \frac{m(t)}{V}\right), \qquad A\rho\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{\mathrm{d}m}{\mathrm{d}t}.$$

We first apply a convenient change of variables, expressing the released mass as a function of *depth* rather than time:

$$m(t) := \widehat{m}(\tau(t)),$$

where $\widehat{m} = \int_\Omega \rho H(\tau - u)\,\mathrm{d}\mathbf{x}$ is straightforward to compute with the same distance-based approach as before. To obtain the release curve m, we simply must determine $\tau(t)$, which can be done by numerically integrating the autonomous nonlinear ODE:

$$\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{\mathrm{d}m}{\mathrm{d}t}\Big/ A\rho = \frac{D}{\rho h}\left(C_s - \frac{\widehat{m}(\tau)}{V}\right), \quad \tau(t_0) = 0.$$

The derivative of the release curve objective can then be calculated as before, with an extra chain rule term accounting for the derivative of this "time remapping."