Levitating Rigid Objects with Hidden Support Structures

Sarah Kushner, Risa Ulinski, Karan Singh, David I.W. Levin, Alec Jacobson
University of Toronto

Abstract
We propose a novel algorithm to efficiently generate hidden structures to support arrangements of floating rigid objects. Our optimization finds a small set of rods and wires between objects and each other or a supporting surface (e.g., wall or ceiling) that hold all objects in force and torque equilibrium. Our objective function includes a sparsity inducing total volume term and a linear visibility term based on efficiently pre-computed Monte-Carlo integration, to encourage solutions that are as-hidden-as-possible. The resulting optimization is convex and the global optimum can be efficiently recovered via a linear program. Our representation allows for a user-controllable mixture of tension-, compression-, and bending-resistant rods or tension-only wires. We explore applications to theatre set design, museum exhibit curation, and other artistic endeavours.

1. Introduction
Levitating objects are visually compelling and commonly found in artistic sculptures, film and theatre set design, promotional displays, and museum exhibits (see Figure 2). This effect is especially impressive if the support structure can be hidden from the observer, removing their unsightly distraction and perhaps even giving the impression that the objects in the arrangement are magically floating in space (see Figure 1). Achieving this is a non-trivial task. Physical stability requires a balance of force and torque for each rigid component of the scene. This is readily achieved using many strong, thick struts, but their geometry and scene placement is likely to compete for visual attention with scene objects, or worse, visually obscure objects in the scene (see Figure 3). Hiding these supports by removing or thinning too many struts, on the other hand, will sacrifice physical stability. Thin wires can sometimes be used to hang objects, but wires only resist tension so they alone can not handle situations that are not supported purely from above.

In this paper, we propose modeling the problem of hidden support structure generation for levitating objects as a form of topology optimization. We present a novel convex optimization based on the well-established ground structure method from architecture and engineering. The input to our method is an arrangement of objects in their desired locations and orientations and the distribution of views from which the scene will likely be observed. Our output is a collection of rods and wires, described by their required thicknesses and attachment points on the input rigid objects, and the supporting structural element (e.g., wall or ceiling). Our rods

Figure 1: Our optimization finds hidden supports to hold rigid objects (green) in their locations despite gravity. Rods (orange) resist tension, compression and bending, while wires (black) resist tension. Supports connect between objects or to the input support surface (blue). Rods are hidden behind occlusions in the scene for a possibly disconnected distribution of viewpoints (red) provided by the user. Here, a collection of space-themed objects seemingly hover in the corner of a room. The supporting truss is hidden from the front and through the window.

Figure 2: The skeleton of a blue whale levitates with the support of wires from above and internal rods.
2. Related Work

Our work sits within the larger literature of computational fabrication, construction and assembly. These subfields are rich and vast, so we focus on previous works most similar in methodology or application.

Previous algorithms exist to make objects stand \[\text{PWLS13; VHW2012}\], spin \[\text{BWBS14}\] or hang from wires \[\text{MML16}\]. These works modify the input objects by redistributing mass or changing their shape to achieve the desired goal. In this paper, we explore a complementary contract with the user — how to anchor objects in the environment without changing the objects themselves. We do not assume that objects were fabricated in a particular manner (e.g., 3D printing).

Our approach may be categorized with other structural optimizations for a prescribed static load scenario (i.e., ignoring inertial forces). Recent works increase the stability of fragile objects by adding new new structural elements \[\text{ZPZ23; SVB212; CZT16}\]. For example, Slava et al. add struts to 3D printed objects one-by-one as part of a large optimization loop and use a volumetric simulation as validation. Their strut selection includes an ambient occlusion visibility term, but they do not consider the problem of selecting an optimal set of supports for rigid objects under prescribed viewing conditions. Other methods have considered the interactive design of rod-structures \[\text{PTC215; KSW217; CZS219; Jac219}\] with varying degrees of physical feasibility checking or optimization in the system.

We model the problem of hiding support structures as a form of topology optimization \[\text{LGC218}\]. The general idea of topology optimization is to prune away material from the volume around the input objects or load conditions. The resulting geometries typically have interesting topologies/connectivities that would have been difficult to determine \textit{a priori}. Methods that determine the material occupancy of each voxel in a dense grid are well suited for 3D printing and milling (e.g., \[\text{WDW216}\]), but will in general produce geometries composed curved and varying thickness elements. Our method instead belongs to the class of \textit{ground structure} methods \[\text{Dor64}\], which output a discrete collection of (straight) elements from an initial over-connected graph of candidates (see Figure 4). Methodologically we follow most closely the stress-based formulation of Zegard et al. \[\text{ZP215}\], and utilize the thesis of Freund \[\text{Fre04}\] as a reference. Ground structure-like methods have been applied for designing everything from buildings \[\text{ZHMB20}\] to

![Figure 3: Without our visibility term, optimal rods may be an unsightly distraction.](image)

Expected views of supports without visibility consideration

![Figure 4: Our method constructs a over-connected ground structure of candidate edges (left) then immediately prunes edges that intersect the scene (middle) and finally extracts a small number of hidden rods. Savings from pruning can produce 10× performance improvements.](image)

23600 edges Pruned to 3586 6-rod solution expected view

model tension, compression and bending resistant materials (e.g., wooden dowel rods or steel beams). Our wires model tension only (e.g., fishing line or steel cables).

Unlike Computer Graphics or Virtual Reality where physical laws can be bent or broken, support structures in real scenes are only meaningful if physically valid. Therefore, we enforce physical validity in our optimization as a hard constraint: namely that the rigid objects should achieve force and torque equilibrium and that stresses on rods and wires do not exceed material-dependent yield limits. For ease of assembly, cost of manufacturing, and visibility considerations, we prefer support structures composed of a small number of thin, less visible supports. We model these criteria with a sparsity-inducing cost function defined as a sum over a densely connected graph of edges (i.e., the \textit{ground structure}).

Treating the cross-sectional area of each edge as the primary optimization variable, the traditional ground structure method optimizes the total volume (linear in the areas since lengths are predetermined) and enforces force balance at point loads, by measuring linearized axial tension and compression forces from each rod, subject to yield limits, expressed as linear inequalities in the unknown cross-section areas and axial stresses of the rods. The result is a linear program whose solution — like many \textit{L}_1 or Lasso problems — is sparse (most areas are exactly zero), and often agrees exactly with the NP-hard selection problem (picking the smallest valid subset of edges).

We augment the traditional ground structure method to support embedded rigid objects (via linear static equilibrium equations) and account for bending resistance of rods (via a simple linear bending model derived from proportionality assumptions). We introduce a visibility objective function that is also linear in the unknown edge areas and relies on efficient Monte-Carlo based precomputation. Thus, the optimization remains a (convex) linear program and solutions can be extracted efficiently (in usually less than a minute).

Our experiments satisfyingly confirm that under many conditions structurally valid supports are lurking just out of sight: the space of physically valid supports is vast and finding a completely occluded arrangement, is often possible. We demonstrate the effectiveness of our method across a wide variety of test scenes and prototypical use cases.
struc-tion supports [DPW*14] to 3D printable models [WWY*13; JTSW17; HZH*16] to cable-driven automata [MKS*17]. The stan-dard ground structure method considers only consider axial forces. These methods have been applied for rigid structural elements and adapted to special cases like tensegrities [PTV*17; CW96].

We use a ground structure approach to model the novel problem of creating hidden structural supports from complex viewpoint distributions. Crucially, our method supports structural elements that resist compression, tension and bending forces, as well as wires, without resorting to nonlinear constitutive models or volumetric meshing of prior work [SVB*12; PTC*15; HZH*16]. Our method trivially couples the structure to the rigid objects it supports, correctly accounting for both linear forces and torques, without resorting to displacement-based mechanical formulations (e.g., [PTV*17]). This allows us to formulate our problem as a linear program which can be solved efficiently.

We draw inspiration from algorithms for appearance-driven optimization. For instance, Schuller et al. introduce the problem of generating appearance mimicking surfaces from a specified viewpoint [SPS14]. Several works seek to create 3D shapes that take the form of a set of 2D shapes from corresponding viewpoints or cast the 2D image under certain lighting conditions [MP09; HHC18; STTP14]. Others use viewpoints to create optimal perceptual experiences, for example in 3D printing support structures [ZLP*15] or in skyscraper design [DFL*15].

3. Method
The input to our method is a scene comprised of K rigid objects oriented and positioned in space, a fixed support surface (e.g., wall or ceiling), and a distribution of viewpoints (e.g., discrete set of positions or sample-able probability density function defined on a surface). The output of our method is a supporting structure composed of a small set of rods and wires connecting rigid objects to each other or the supporting surface. Our method ensures that this structure holds the input objects in their prescribed positions and orientations, counter-balancing the force these objects experience due to gravity. Our method optimizes the size and placement of the structure to minimize its overall volume and its visibility with respect to the input viewpoint distribution (see Figure 5). Before describing our optimization, we define our physical model and how we measure visibility.

3.1. Rigid Body Equilibrium
The rigid objects in our scenes experience forces from gravity and the points of attachment to the supporting structure. To hold a rigid body at rest, we must maintain force and torque equilibrium:

\[ \sum_{i \in V_k} f_i = m_k g, \]

\[ \sum_{i \in V_k} \left( x_i - \bar{x}_k \right) \times f_i = 0, \]

where \( m_k, \bar{x}_k, \) and \( V_k \) are the mass, center of mass, and set of attachment points of the \( k \)th object, respectively, and \( x_i, f_i, \tau_i \) are the 3D position of the \( i \)th attachment point and corresponding force and torque vectors, respectively.

3.2. Rods
We assume our support structure undergoes negligible displacement, affording a linearization of the internal forces at play. For stiff rods, we follow the linearised tension and compression model of [ZP15; Fre04], which introduces a signed scalar value per rod \( c_i \in \mathbb{R} \) with units Newtons describing the force in the axial direction parallel to the rod. Assigning an arbitrary direction to the rod \( ij \) between endpoint positions \( x_i \) and \( x_j \), then the axial force contribution at endpoints \( i \) and \( j \) are the product of this scalar \( c_{ij} \) by the rod’s tangent unit direction \( \hat{t}_{ij} = (x_i - x_j)/\|x_i - x_j\|_2 \):

\[ f_i = +c_{ij} \hat{t}_{ij} \quad \text{and} \quad f_j = -c_{ij} \hat{t}_{ij}. \]

Previous methods (e.g., [ZP15; Fre04]) rely solely on tension and compression and ignoring the rods’ resistance to bending.
This is a reasonable assumption in architecture where loads are large relative to the rod’s bending strength. Ignoring bending requires that either the rods are thicker and thus more visible. This is at odds with the intuition that light loads can be held up with a single bending-resistant rod. In reality, a single rod with finite thickness can apply a distribution of forces over its non-zero area contact surface. Since the force is applied at more than one point, torque balance is also possible. Unfortunately, a volumetric rod model couples the unknown rod diameters and forces non-linearly.

To maintain the linearity of our system but also account for bending resistance, we introduce a linearised bending model (see Figure 6). The bending force acts equally (but not oppositely) on either end of the rod (does not account for shearing) and is proportional to the rod’s length. This matches the intuition that it is very difficult to bend a short rod, but easy to bend a long rod of the same thickness. For each rod $ij$, we introduce an arbitrary orthonormal basis $N_{ij} \in \mathbb{R}^{3 \times 2}$ for the 2D space orthogonal to the axial direction. We introduce two dimensional parameter $b_{ij} \in \mathbb{R}^2$ with units Newtons/meters describing the average force over the rod in the basis directions. The bending force contributions at endpoints are:

$$f_i = \epsilon_{ij} N_{ij} b_{ij} \quad \text{and} \quad f_j = \epsilon_{ij} N_{ij} b_{ij}$$

(4)

where $\epsilon_{ij} = \|x_i - x_j\| \in \mathbb{R}^+$ is the rod’s length.

Following previous methods [ZP15; Fre04], we model failure catastrophically. If the stress due to tension, compression or bending exceeds a material-dependent fixed threshold we declare that light loads can be held with a single bending-resistant rod. A wire $ij$ has zero resistance to bending and compression (i.e., $\sigma_{ij}^t = \sigma_{ij}^c = 0$) and very high resistance to tension (i.e., $\sigma_{ij}^t > 0$). Wires made of strong material such as braided steel can be very thin (near invisible) while maintaining high strength. Our method will allow a mixture of tension-compression-bending rods (e.g., wooden dowels) and tension-only wires (steel wires), see Figure 8. As special case, we can limit our optimization to consider only wires, resulting in a hanging optimization (see Figure 7).

3.3. Wires

A special case of our model is a wire, which can be thought of as a tension-only rod. A wire $ij$ has zero resistance to bending and compression (i.e., $\sigma_{ij}^t = \sigma_{ij}^c = 0$) and very high resistance to tension (i.e., $\sigma_{ij}^t > 0$). Wires made of strong material such as braided steel can be very thin (near invisible) while maintaining high strength. Our method will allow a mixture of tension-compression-bending rods (e.g., wooden dowels) and tension-only wires (steel wires), see Figure 8. As special case, we can limit our optimization to consider only wires, resulting in a hanging optimization (see Figure 7).

3.4. Visibility

We define the expected visibility of a rod as function of the input viewpoint distribution, occlusions due to the scene, the rod’s position and orientation and its unknown cross-sectional area. For a rod $ij$, its expected visibility $v_{ij}$ is:

$$v_{ij} = \int_\mathcal{E} p(e) \int_{C_{ij}} s(e, x) d\Omega \, de,$$

(6)

where

$$s(e, x) = \begin{cases} 1 & \text{if the segment } ex \text{ does not intersect the scene}, \\ 0 & \text{otherwise}, \end{cases}$$

where $\mathcal{E}$ defines the set of viewpoints and $p(e)$ is the probability density associated with the point $e \in \mathcal{E}$, and $C_{ij}$ is the surface of the cylindrical rod with cross-sectional area $a_{ij}$ connecting endpoints $x_i$ and $x_j$, and $d\Omega$ is the differential solid angle at the corresponding integration point $x$ subtended at the viewpoint $e$.

Our optimized supports

Rods highlighted

Painted Rods

Figure 7: Wire-only solutions require support from above the arrangement’s center of mass.

Figure 8: We show the rods in orange to demonstrate how hidden they are. But we show that the rods connecting the smaller parts (middle) can be painted to blend in with the ceiling (right).
visibility according to solid angle correctly matches the intuition that the same size rod farther away from an observer is less visible.

The outer integral is immediately as a soft-shadow or area-light source evaluation common in rendering. We can approximate this well by Monte-Carlo importance sampling over the viewpoint distribution. An analytic expression for the inner integral becomes unwieldy, so we instead opt for a simple approximation based on uniform quadrature, accounting for the orientation of the rod resulting in foreshortened projection. Rods are thin relative to the scene and wieldy, so we instead opt for a simple approximation based on uniformly non-negative scalar per-rod where we collect the terms that do not depend on $G_{ij}$ into a single non-negative scalar per-rod $g_{ij} \in \mathbb{R}_{\geq 0}$. In this way, the squared visibility of each rod becomes a linear function of the cross-sectional area: $a_{ij}g_{ij}^2$. Because wires are so thin compared to rods, we happily set $g_{ij} = 0$ for wires and avoid their visibility precomputation. To generate the $n_{ij}$ samples on edge $ij$, we subdivide the edge until all segments are less than a given scene-dependent length threshold (e.g., 0.1 meters for the bedroom scene in Figure 1) and then use the segment barycenters as samples (typically 10-100 samples per edge). Segment queries can be computed in parallel.

### 3.5. Ground Structure

The space of physically feasible supporting structures is high-dimensional and a mixture of discrete variables (e.g., how many rods? connecting between which objects?) and continuous variables (e.g., where rods attach to each object? what are the rod thicknesses?). Navigating this space to find a globally optimal solution is difficult. In response, the ground structure method (e.g., [Dor64; Ped93; ZP15; Fre04] makes the problem tractable by rephrasing the problem into selecting a discrete subset of support elements from an intentionally dense yet finite set of candidate elements. This candidate set is referred to as the “ground structure.”

In our case, we generate a ground structure of candidate rod and wire elements by Poisson disk sampling [Yuk15] all rigid objects and the support surface and then connecting all possible pairs of points from different sources (e.g. for a single rigid object this forms a bipartite graph with the supporting surface, see Figure 4). For each edge in this graph, we label it as a “rod” or “wire” (and possibly create duplicate copies so edges appear as both types). We can discard a bad edge if its attachment angle is self-penetrating or too obtuse (by checking if $\hat{t}_{ij} \cdot \hat{n}_i < \cos \theta_{\text{max}}$), if it intersects objects in the scene (by ray casting), or if its computed visibility coefficient is exceptionally high ($g_{ij} > g_{\text{max}}$). We refer to the result as the pruned ground structure $G$.

### 3.6. Sparse Optimization

The beauty of the ground structure method is that once the candidate set has been chosen, selecting the globally optimal subset can be phrased as an efficient convex optimization, in particular a linear program. In the classic method, the cost function to be minimized is the total volume of material spent on the support structure. Since all edge lengths are known once the candidate set is selected, this cost is a linear function of the yet unknown edge cross-sectional areas. It is important that this cost function is the unsquared volume, which can be thought of as the $L_1$-norm of the vector of edge areas (weighted by edge-lengths), as opposed to the sum of squared per-edge volumes, analogous to the $L_2$-norm. The $L_1$-norm is sparsity inducing and under mild conditions will agree with the optimal solution of the selection problem, analogous to the $L_0$-pseudonorm. As a result, the vast majority of edges in the solution will have exactly zero area.

In our case, we augment the total volume cost function with a least-squares visibility term to penalize choosing highly visible rods. Because our per-edge visibility measurement in Eq. 7 is linear in the square-root of the rod areas, this least-squares energy becomes linear in the areas.

The areas of the rods and wires are the primary unknowns. We introduce auxiliary variables $c_{ij}$ and $b_{ij}$ as described in Sec. 3.2 to facilitate writing our force and torque balance constraints (see Sec. 3.1). These variables are then coupled to the areas via the yield stress inequalities (see Eq. 5).

The resulting optimization is a linear program over the pruned ground structure $G$ containing $m$ candidate edges:

\[
\begin{align*}
\min_{a, c, b} \sum_{ij \in G} a_{ij}(\ell_{ij} + \lambda g_{ij}^2) & \quad \text{(7)} \\
\text{s.t.} \quad \sum_{ij \in E_i} c_{ij}k_{ij} + \ell_{ij}N_{ij}b_{ij} = m_ag, & \quad \forall k = 1, \ldots, K \\
\sum_{ij \in E_i} (c_{ij}k_{ij} + \ell_{ij}N_{ij}b_{ij}) \times (x_j - x_k) = 0, & \quad \forall k = 1, \ldots, K \\
-\sigma^b_{ij}a_{ij} \leq c_{ij} \leq \sigma^b_{ij}a_{ij}, & \quad \forall ij \in G \\
-\frac{\sigma^b_{ij}a_{ij}}{\ell_{ij}} \leq b_{ij} \leq \frac{\sigma^b_{ij}a_{ij}}{\ell_{ij}}, & \quad \forall ij \in G \\
a_{ij} \geq 0, & \quad \forall ij \in G \quad \text{(11)}
\end{align*}
\]

where we stack all $a_{ij}$, $c_{ij}$, and $b_{ij}$ variables into vectors $a \in \mathbb{R}^n$, $c \in \mathbb{R}^m$, and $b \in \mathbb{R}^{2m}$, respectively, and we introduce the user-controllable weighting term $\lambda$ to balance between preference for volume and visibility minimization.

We opt to replace the second-order cone constraint for bending yields in Eq. 5 with the simpler coordinate-wise linear inequality in Eq. 10. This can be thought of as a conservative $L_\infty$ approximation, and albeit coordinate system dependent, does not affect results and admits a faster linear program than a conic program in our experience.

The linear coefficients in the force/torque balance equations and linear inequalities (Eqs. 8-10) can be collected in large sparse matrices (see App. 5). Many efficient solvers exist for such large sparse linear programs; we use MOSEK [AA00].
A solution is a guarantee to exist as long as force and torque balance can be achieved. This could fail to happen for very sparse ground structures (e.g., less than six edges per object) or degenerate situations (e.g., all edges are parallel). Our very dense ground structure (hundreds of thousands of edges) enjoys the general position of its random position. We never fail to find a feasible solution.

4. Experiments & Results

We implemented our algorithm in MATLAB using GPToolbox [Jac*18] for geometry processing and MOSEK [AA00] to solve the linear program formulated in Section 3.6. Pre-computation of the integrated visibility, in our input scene is accelerated using the恩施 [WWB*14] ray-tracer as interfaced by LIBIGL [JP*20]. We report statistics and timings for the results in our paper in Table 1. All times are reported on a MacBook Pro with 3.5 GHz Intel Core i7 and 16GB of RAM. Visibility pre-computation is computed in parallel, but is still typically the bottleneck (≈ 80%).

The number of degrees of freedom in the system is the size of the ground structure which generally scales quadratically in the number of objects \( m = O(K^2) \), typically generated by taking all inter-object pairs over 10-100 Poisson disk sample points on each object. Starting with a dense ground structure leads to better qualitative results, but the exact positions of the samples do not drastically affect the hidden-ness of the result. Figure 9 shows how little the solution changes as a function of the ground structure sampling.

Pruning often significantly reduces the ground structure size and consequently, the number of degrees of freedom (see, e.g., Figure 4). Perhaps unsurprisingly, we typically experience a speedup the same ratio of original ground structure edges to pruned ground structure edges. The number of constraints in our optimization is six times the number of objects \( K \). After pruning, MOSEK finds a solution for the above problem configuration within a few minutes.

In our accompanying video, we show animations of results in this paper including traversals of the viewing distributions to demonstrate the robustness of our methods ability to hide supports.

Levitating 3D objects has a wide range of applications including scientific visualization, film and theater set design, home decor, anamorphic 3D art installations, as well as objects for zoetropes and 3D stop-motion animation. Each application has specific design requirements, and our algorithm is designed to enable the exploration of a number of aesthetic and structural parameters and design choices, which significantly impact the resulting solution. We elaborate on some of these design use cases.

Both scientific exhibits (see, e.g., Figures 1, 2) and illusory art installations require an unobscured view of the levitating objects. While it is feasible for designers to hand-craft support structures from a single fixed viewpoint, the interplay between visibility and structural stability is quite complex for multi-view distributions. In Figure 1, we show the ability of our algorithm to adapt its optimal solution to multiple viewpoint distributions.

Our algorithm is able to holistically optimize the support structure using a mix of rods and wires. We color our rods bright orange for evaluation in this paper, but in practice they can be further camouflaged by matching their appearance to the background or scene objects (see Figure 8).

The choice of using a rod or wire is both aesthetic (as determined by a user) and functional. For example, supporting a levitating object with a wire would require a potential attachment points on the fixed surface or other levitating objects, to be vertically higher than the given object (see, e.g., Figure 7). The inset figure shows a parade float suspended by optimized wires (the net force pointing upward due to buoyancy). Previous methods have considered hanging objects [PWLS13] or more generally mobiles [MML16] by placing a single support “perfectly” placed...
in alignment above the center of mass. While this strategy requires the fewest supports, it is an unstable solution (see Figure 10). Our method relies on random sampling of points in general position, typically producing multiple wires per hanging object, but resulting in a more stable configuration.

Mounting objects off the side of a support such as a wall is best achieved with a mixture of wires and rods. Figure 11 shows a giant promotional display suspended in front of a contemporary art museum. We provide a symmetric dense ground structure and our optimization naturally finds a symmetric sparse solution.

The pterosaur in Figure 8 has 57 separate bones and requires a complex support structure, acting as a stress test on our optimization. In practice, skeleton displays often pre-plaster-fuse bones to reduce the number of pieces (see spine of whale in Figure 2).

The idea of stop-motion animation and 3D zoetropes is over a century old [Mar90], with modern examples including “Feral Fount” by Gregory Barsamian at the Museum of the Moving Image in Queens and the Toy Story zoetrope featured in Pixar’s Museum Exhibit. The portrayal of levitating objects in this medium is particularly challenging. We demonstrate a prototypical result of a backflipping boy in Figure 12 by hiding supporting rods out of sight. For this example to be structurally stable both at rest and while spinning, we first find the optimal set of rods for each frame under gravity and then re-run the linear program on just these rods subject to centrifugal forces. The final rod thickness is the maximum over the two solves. Incorporating more elaborate multi-load handling (cf. [Fre04]) is left as future work.

5. Limitations & Future Work

Our rod model includes linearised tension, compression, and bending forces. Like many past methods, we do not handle the self-weight of the rods by assuming that the force of gravity is much larger than the force of the rods on themselves. This is a trivial addition of gravity forces on each rod proportional to their length. Ground structure methods may produce solutions where thickened rods intersect each other; ours is no exception. Previous methods have considered penalty terms or post-pressing to deal with intersecting (e.g., [JTSW17]). Wire-wire intersections are extremely unlikely due to the very thin nature of wires. Our visibility model considers direct line of sight, but not other cues such as reflections or shadows. Transparency of objects is not accounted for. Depending on the setup of the scene, there may not be a solution invisible to every viewpoint (e.g., Figure 13). Since we model physical validity as a hard constraint, we are still able to find a solution, albeit a visible one.

The precise solution depends on the initial ground structure. In general, denser ground structures produce higher quality solutions — both in terms of total structure volume and hidden-ness — with diminishing returns. Rod areas are directly proportional to stress limits, so accurate fabrication relies on accurate (or at least conservative) material measurement. In fact, fabrication of our digital results would also be an exciting next step.

Fabrication may be within reach utilizing recent wire-frame methods [KSW*17; CZS*19; Jac19] or via 3D printing. A voxel-based topology optimization formulation of our problem may be better suited for 3D printing or milling pipelines. It may be possible to forgo fabricating a literal support structure entirely by exploring magnetic levitation.

Our algorithm assumes that the input is a well-crafted scene to begin with and leaves it perfectly as inputted. The creative design process for these scenes is itself non-trivial. In the future, we are interested in pursuing an interactive design tool which would provide hints to increase occlusion by applying simple transformations (translations, rotations and scales) to the objects in the scene or even provide automatic layout optimizations given the objects and the viewpoints.

We model the problem of hidden supports as an efficient linear program that leverages fast ray-casting from computer graphics.
We see an exciting future in combining techniques from rendering and geometry processing with structural optimization in architecture and engineering. We hope this combination of appearance-driven design will be beneficial to scientific and artistic endeavours.

**References**

[AA00] Andersen, E. D. and Andersen, K. D. “The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm”. *High Performance Optimization*. 2000 5, 6, 9.

[BWBS14] Bacher, Moritz, Whiting, Emily, Bickel, Bernd, and Sorkine-Hornung, Olga. “Spin-it: optimizing moment of inertia for spinnable objects”. *ACM Trans. Graph.* 33.4 (2014), 96:1–96:10 2.

[CW06] Connelly, Robert and Whiteley, Walter. “Second-Order Rigidity and Prestress Stability for Tensegrity Frameworks”. *SIAM J. Discrete Math.* 9.3 (1996), 453–491 3.

[CFZ19] Chidambaram, Subramanian, Zhang, Yunbo, Sundararajan, Venkatraghavan, et al. “Shape Structuralizer: Design, Fabrication, and User-driven Iterative Refinement of 3D Mesh Models”. *Proc. CHI*. 2019, 663 2, 7.

[CTZ16] Coros, Stelian, Zehnder, Jonas, and Thomaszewski, Bernhard. “Designing structurally-sound ornamental curve networks”. *ACM Trans. Graph.* 35.4 (2016), 99:1–99:10 2.

[DFL15] Doraiswamy, Harish, Ferreira, Nivan, Lage, Marcos, et al. “Topology-based catalogue exploration framework for identifying view-enhanced tower designs”. *ACM Trans. Graph.* 34.6 (2015), 230:1–230:13 3.

[Dor64] Dorn, W. “Automatic design of optimal structures”. *J. de Meccanique 3* (1964), 525–522 3.

[DPW*14] Deuss, Mario, Panuzzo, Daniele, Whiting, Emily, et al. “Assembling self-supporting structures”. *ACM Trans. Graph.* 33.6 (2014), 214:1–214:10 3.

[Fre04] Freund, Robert M. “Truss Design and Convex Optimization”. PhD thesis. MIT, 2004 2–5, 7.

[HHC18] Hsiao, Kai-Wen, Huang, Jia-Bin, and Chu, Hung-Kuo. “Multi-view wire Art”. *ACM Trans. Graph.* 37.6 (2018) 3.

[HZJH16] Huang, Yi-Jiang, Zhang, Juyong, Hu, Xin, et al. “FrameFab: robotic fabrication of frame shapes”. *ACM Trans. Graph.* 35.6 (2016), 224:1–224:11 3.

[Jac*18] Jacobson, Alec et al. *gptoolbox*: Geometry Processing Tool box. http://github.com/alecJacoby/gptoolbox. 2018 6.

[Jac19] Jacobson, Alec. “RodStewArt: A Design-to-Assembly System for Fabrication using 3D-Printed Joints and Precision-Cut Rods”. *Comput. Graph. Forum* 38.7 (2019), 765–774 2, 7.

[JP*20] Jacobson, Alec, Panuzzo, Daniele, et al. *libigl*: A simple C++ geometry processing library. http://libigl.github.io/libigl/. 2020 6.

[JTWW17] Jiang, Caigui, Tang, Chengcheng, Seidel, Hans-Peter, and Wonka, Peter. “Design and volume optimization of space structures”. *ACM Trans. Graph.* 36.4 (2017), 159:1–159:14 3, 7.

[KSW*17] Kovacs, Robert, Seufert, Anna, Wall, Ludwig, et al. “TrussFab: Fabricating Sturdy Large-Scale Structures on Desktop 3D Printers”. *Proc. CHI*. 2017, 2606–2616 2, 7.

[LGC*18] Liu, Jikai, Gaynor, Andrew T., Chen, Shikui, et al. “Current and Future Trends in Topology Optimization for Additive Manufacturing”. *Struct. Multidiscip. Optim.* 57.6 (2018) 2, 6.

[Mar90] Marey, E.J. *Physiologie du mouvement: vol des oiseaux*. 1890 7.

[MKS*17] Megaro, Vittorio, Knoop, Espen, Spielberg, Andrew, et al. “Designing cable-driven actuation networks for kinematic chains and trees”. *Proc. SCA*. 2017, 151–1510 3.

[MML16] Makatura, Liande, Most, Catherine, and Li, Gloria. *Balancing Act*: An Interactive Tool for Fabricating Calder-Style Hanging Mobiles. Tech. rep. 2016 2, 6.

[MP09] Mitra, Niloy J. and Pauly, Mark. “Shadow Art”. *ACM Trans. Graph.* 28.5 (Dec. 2009), 156:1–156:7 3.

[Ped93] Pedersen, Paul. “Topology optimization of three-dimensional trusses”. *Topology design of structures*. Springer, 1993, 19–30 5.

[PTC*15] Perez, Jesus, Thomaszewski, Bernhard, Coros, Stelian, et al. “Design and fabrication of flexible rod meshes”. *ACM Trans. Graph.* 34.4 (2015), 138:1–138:12 3, 7.

[PTV*17] Pietroni, Nico, Tanari, Marco, Vaxman, Amir, et al. “Position-based tensegrity design”. *ACM Trans. Graph.* (2017) 3.

[PWS13] Prevost, Roman, Whiting, Emily, Lefebvre, Sylvain, and Sorkine-Hornung, Olga. “Make it stand: balancing shapes for 3D fabrication”. *ACM Trans. Graph.* (2013) 2, 6.

[SPS14] Schuller, Christian, Panuzzo, Daniele, and Sorkine-Hornung, Olga. “Appearance-mimicking Surfaces”. *ACM Trans. Graph.* 33.6 (Nov. 2014), 216:1–216:10 3.

[STFP14] Schwartzburg, Yuli, Testuz, Romain, Tagliasacchi, Andrea, and Pauly, Mark. “High-contrast Computational Caustic Design”. *ACM Trans. Graph.* (2014) 3.

[SVE*12] Stava, Ondrej, Vanek, Juri, Benes, Bedrich, et al. “Stress relief: improving structural strength of 3D printable objects”. *ACM Trans. Graph.* 31.4 (2012), 48:1–48:11 2, 3.

[VHPW12] Vouga, Etienne, Höbinger, Mathias, Wallner, Johannes, and Pottmann, Helmut. “Design of self-supporting surfaces”. *ACM Trans. Graph.* 31.4 (2012), 87:1–87:11 2.

[WDW16] Wu, Jun, Dick, Christian, and Westermann, Rüdiger. “A System for High-Resolution Topology Optimization”. *IEEE Trans. Vis. Comput. Graph.* 22.3 (2016), 1195–1208 2.

[WWB*14] Wald, Ingo, Woop, Sven, Benthin, Carsten, et al. “Embree: a kernel framework for efficient CPU ray tracing”. *ACM Trans. Graph.* 33.4 (2014), 143:1–143:8 6.

[WYY*13] Wang, Weiming, Wang, Tuanfeng, Yang, Zhioung, et al. “Cost-effective printing of 3D objects with skin-frame structures”. *ACM Trans. Graph.* 32.6 (2013), 177:1–177:10 3.

[Yuk15] Yuksel, Cem. “Sample Elimination for Generating Poisson Disk Sample Sets”. *Proc. Eurographics* (2015) 5.

[ZHM20] Zegard, Tomáš, Hertz, Christian, Mazurek, Arek, and Baker, William F. “Advancing building engineering through structural and topology optimization”. *Structural and Multidisciplinary Optimization* (2020) 2.

[ZLP*15] Zang, Xiaoting, Le, Xin, Panotopoulou, Athina, et al. “Perceptual models of preference in 3D printing direction”. *ACM Trans. Graph.* 34.6 (2015), 215:1–215:12 3.

[ZP15] Zegard, Tomás and Paulino, Glauco H. “GRAND3: Ground structure based topology optimization for arbitrary 3D domains using MATLAB”. *Structural and Multidisciplinary Optimization* (2015) 2–5.
Appendix: Matrix Form

Solvers like MOSEK [AA00] expect the problem to be provided in matrix form. We spell out the coefficients of the relevant sparse matrices implementing the linear program in Eq. 8.

For our pruned ground structure with \( m \) candidate edges connecting \( N \) vertices, introduce a unit-less sparse matrix \( C \in \mathbb{R}^{3N \times m} \) where:

\[
C_{ji} = \begin{cases} 
\hat{t}_i & \text{if rod } i \text{ points toward } j \\
-\hat{t}_i & \text{if rod } i \text{ points away from } j \\
0 & \text{otherwise.}
\end{cases}
\]

(12)

Introduce a sparse matrix \( B \in \mathbb{R}^{3N \times 2m} \) with units meters where:

\[
B_{ji} = \begin{cases} 
\hat{N}_i \ell_i & \text{if rod } i \text{ points toward } j \\
-\hat{N}_i \ell_i & \text{if rod } i \text{ points away from } j \\
0 & \text{otherwise.}
\end{cases}
\]

(13)

Introduce a sparse unit-less selection matrix \( S \in \mathbb{R}^{3K \times 3N} \), where

\[
S_{kj} = \begin{cases} 
I_3 & \text{if vertex } i \text{ lies on object } k \\
0 & \text{otherwise}
\end{cases}
\]

(14)

Introduce a sparse cross-product matrix \( S \in \mathbb{R}^{3K \times 3N} \) with units meters, where

\[
G_{kj} = \begin{cases} 
[x_j - x_k] \times & \text{if vertex } i \text{ lies on object } k \\
0 & \text{otherwise}
\end{cases}
\]

(15)

where

\[
[g] \times = \begin{bmatrix} 0 & -g_3 & g_2 \\
g_3 & 0 & -g_1 \\
-g_2 & g_1 & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3}
\]

(16)

Finally, the full linear program in matrix form may be written:

\[
\begin{align*}
\min_a & \quad \ell^T a \\
\text{subject to} & \quad \begin{bmatrix} 0 & SC & SB \\ 0 & GC & GB \end{bmatrix} \begin{bmatrix} a \\ c \\ b \end{bmatrix} = \begin{bmatrix} m \otimes g \\ 0 \end{bmatrix} \\
\text{and} & \quad -\sigma_b a_i \leq c_i \leq a_i \sigma_c, \forall i \\
\text{and} & \quad -\sigma_b a_i \ell_i / \ell_i \leq b_i \leq a_i \sigma_b / \ell_i, \forall i.
\end{align*}
\]

(17)