Wire Mesh Design

Akash Garg[1*] Andrew O Sageman-Furnas[2] Bailin Deng[3] Yonghao Yue[1]
Eitan Grinspun[1] Mark Pauly[3] Max Wardetzky[2]

[1]Columbia University [2]University of Göttingen [3]EPFL

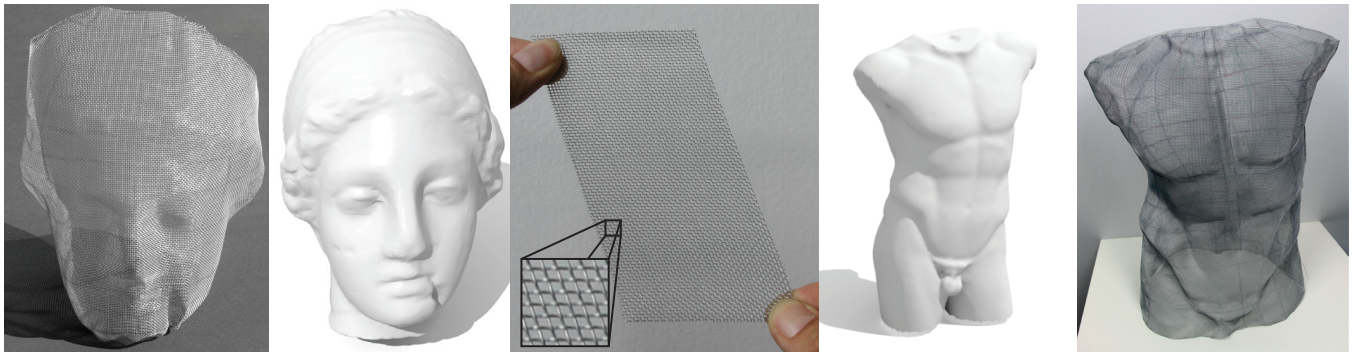


Figure 1: *Wire mesh design allows creating physical realizations (1^{st} and 5^{th} images) of a given design surface (2^{nd} and 4^{th} images) composed of interwoven material (middle image) in an interactive, optimization-supported design process. Both the torso and the Igea face are constructed from a* single *sheet of regular wire mesh.*

Abstract

We present a computational approach for designing *wire meshes*, i.e., freeform surfaces composed of woven wires arranged in a regular grid. To facilitate shape exploration, we map material properties of wire meshes to the geometric model of *Chebyshev nets*. This abstraction is exploited to build an efficient optimization scheme. While the theory of Chebyshev nets suggests a highly constrained design space, we show that allowing controlled deviations from the underlying surface provides a rich shape space for design exploration. Our algorithm balances globally coupled material constraints with aesthetic and geometric design objectives that can be specified by the user in an interactive design session. In addition to sculptural art, wire meshes represent an innovative medium for industrial applications including composite materials and architectural façades. We demonstrate the effectiveness of our approach using a variety of digital and physical prototypes with a level of shape complexity unobtainable using previous methods.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling; J.2 [Physical sciences and engineering]: Engineering—Computer-aided design

Keywords: Wire mesh, interactive shape modeling, Chebyshev nets, discrete differential geometry, design, global optimization

Links: DL PDF

*e-mail: akg2110@columbia.edu

ACM Reference Format
Garg, A., Sageman-Furnas, A., Deng, B., Yue, Y., Grinspun, E., Pauly, M., Wardetzky, M. 2014. Wire Mesh Design. ACM Trans. Graph. 33, 4, Article 66 (July 2014), 12 pages. DOI = 10.1145/2601097.2601106 http://doi.acm.org/10.1145/2601097.2601106.

DOI: http://doi.acm.org/10.1145/2601097.2601106

1 Introduction

Wire meshes enjoy broad application in art, architecture, and engineering, including handmade sculptures, filters, support structures in composite materials, and architectural façades (see Fig. 3). Despite their widespread use, a systematic design methodology for freeform wire meshes is lacking. While physical exploration helps build intuition in early concept design, rationalizing a surface entails numerous constraints that are often globally coupled. Artists currently use an incremental, trial-and-error approach, where an initially flat piece of wire mesh is gradually bent by hand to conform to a desired surface. Likewise, in architecture wire meshes are restricted to very simple shapes, such as planes, cylinders, cones, or half-spheres, despite great interest in freeform façades. We show that a much richer space of wire meshes can be more effectively explored using digital tools, which automatically account for the strong global coupling of physical and geometric constraints.

While in our fabrication examples (but not for our design tool), we have focused on wire mesh made of steel, wire mesh encompasses a much broader range of materials, such as fishnet stockings, woven reinforcements in composite materials, or even onion nets. Indeed, even something as prosaic as a simple onion net reveals some of the core structural properties of wire mesh: inextensible fibers that are woven in a criss-cross pattern such that the warp and weft directions cannot stretch but may significantly shear towards (or away from) one another (see Fig. 5). In order to gain intuition for designing with wire mesh, one may try to "dress" a given target shape, such as a vase, a bust, or a ball with an onion net. Soon one then discovers that due to shearing some features cannot be captured, that more material may be required in certain areas, or that it is difficult to preserve the fine details of the given target shape.

Such difficulties are ubiquitous when working and designing with wire mesh: If a wire mesh is required to lie exactly on a given target design surface, incremental layout often fails to adequately represent the desired shape. We substantiate this observation by modeling wire meshes as discrete Chebyshev nets (§3), revealing fundamental limitations in the kind of shapes that can be equipped with a single wire mesh. Further insights from the theory of Chebyshev nets allow us to formulate an optimization scheme where the mesh can deviate

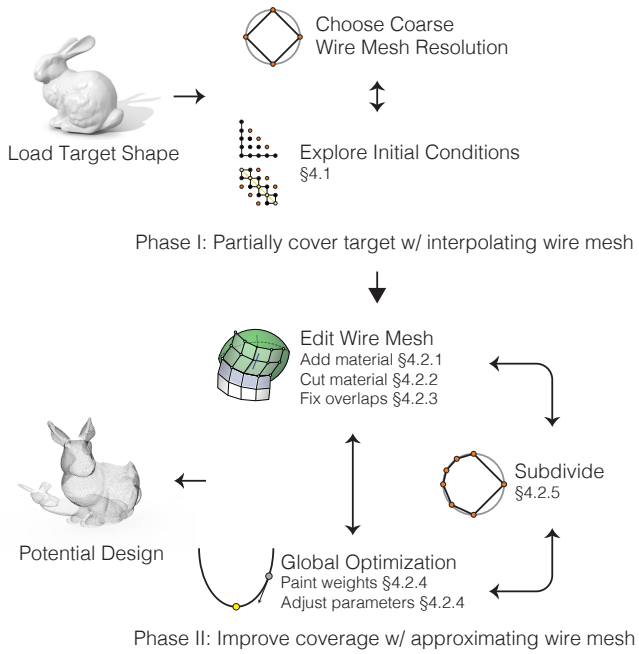


Figure 2: *The wire mesh design process.*

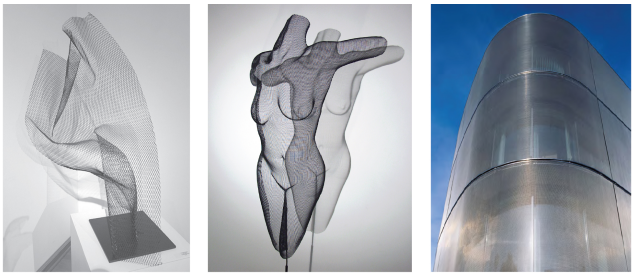
from the target surface in a controlled way. This scheme balances inextensibility of wires and limits on shearing angles with design objectives imposed by the user. Optimization is then incorporated into an interactive design tool that leverages the user's high-level understanding of shape (§4). The tool interleaves user input on artistic decisions with global optimization to explore the design space. This facilitates an effective pipeline from surface modeling to physical realization.

In contrast to previous approaches, our methodology (i) lifts the restriction of a priori curvature bounds for a given target shape (which is omnipresent in the mathematical literature and previous computer-aided tools) and (ii) works without insertion of darts (i.e., folds stitched into the material). Indeed, unlike garments, where the inclusion of darts are considered a premium and a signature of thoughtful design, darts generally introduce a point of failure, over-engineering, or manufacturing complexity in wire mesh structures. Instead, our design tool allows the user to interactively and iteratively grow a wire mesh on a target shape, thus allowing fabrication with a *single piece of material*.

Our contribution is an example of how geometric modeling and optimization-based shape exploration can lead to new material-aware design solutions that enable creative works not feasible before. The design process (Fig. 2) has two phases: a setup phase that directly extends previous work on *interpolating* Chebyshev nets which partially cover the target and a novel design loop where the designer interactively explores *approximating* Chebyshev nets to increase coverage.

2 Related Work

Computational tools for material- and fabrication-aware design have recently become a prominent topic in computer graphics research. A typical example are algorithms for the design and optimization of discrete freeform surfaces with planar polygons [Liu et al. 2006; Wang et al. 2008; Bouaziz et al. 2012; Poranne et al. 2013]. Ensuring planarity of mesh elements facilitates the use of cost-effective materials and construction technologies, for example in stone or



© Donald Kolberg © Randy Cooper wiremesh facade

Figure 3: *Wire meshes are a popular medium in abstract (left) and figurative (middle) sculptural art. These freeform shapes are created by manually bending a single, flat piece in an incremental, trial-and-error process. Lacking an effective digital design process, the use of wire meshes in architectural façades (right) is currently limited to simple geometries.*

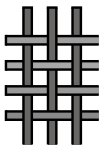
glass façades. Additional constraints such as torsion-free nodes can be incorporated to improve structural performance and further simplify the fabrication process [Liu et al. 2006]. Related examples include rationalization and shape exploration for developable surfaces [Pottmann et al. 2008], geodesic patterns [Pottmann et al. 2010], curved panelings [Eigensatz et al. 2010], or circular arc structures [Bo et al. 2011]. A series of papers exploits geometric abstractions of compression-only surfaces to facilitate the design of self-supporting structures [Vouga et al. 2012; Panozzo et al. 2013; de Goes et al. 2013; Liu et al. 2013]. The construction of planar intersecting pieces has been investigated by Schwartzburg and Pauly [2013] who map assembly and fabrication restrictions to geometric constraints of a mixed discrete/continuous optimization. The common theme of these and other related methods is that material behavior or physical restrictions are mapped to *geometric properties* or constraints of the design surface—a methodology that we also follow here. This strategy avoids the complexities of a full physical simulation and enables efficient computations for interactive form-finding and design exploration.

As explained in §3, wire meshes are best modeled by Chebyshev nets—a geometric model of woven materials using a two dimensional net composed of inextensible yarns, first proposed by Chebyshev in 1878 [Tschebyscheff 1878]. The same model is used in the mechanical theory of pure networks, i.e., grids of inextensible yarns with no shear resistance [Rivlin 1958; Rivlin 1964; Rivlin 1997]. When shearing is incorporated, the model is known as a reinforced network [Adkins 1956]; investigations into bending resistant inextensible networks have been considered in [Wang and Pipkin 1986]. Pipkin analyzed stress in reinforced networks on arbitrary curved surfaces [Pipkin 1984] and the distribution of wrinkles of the solution [Pipkin 1986], but assumes—different from our approach—that the reinforced network already lies on the surface. A purely discrete version of Chebyshev nets has been used in computer aided design to model the forming of woven reinforced composite materials to surfaces. However, many studies focus only on simple geometries such as a hemisphere [Robertson et al. 1981; Ye and Daghyani 1997], polyhedral and rounded cones [Robertson et al. 2000; Baillargeon and Vu-Khanh 2001] or translated sine curves [Wang et al. 1999].

In terms of application domain, virtual design with Chebyshev nets was considered by Aono et al. [1994; 1996; 2001]. Building upon initial computational investigations [van West et al. 1990], Aono et al. presented a method for finding Chebyshev nets *interpolating* a given surface via automatic dart insertions. Their approach mimics the process by which a tailor drapes garments over the human form, pinning initial lines of material onto a dress form, then working outwards from these constraints, making cuts or inserting darts as

required to fit the underlying form. Our work is inspired by and builds upon the work of Aono and coworkers, while (i) lifting the restriction that total Gaussian curvature be bounded by 2π (thereby expanding the scope of possible designs), (ii) working with a *single* piece of material without the need of inserting darts, (iii) expanding the range of admissible target shapes (including cylindrical topologies), and (iv) extending modality by deeply integrating the user into the design loop with a varied tool set.

3 Chebyshev Nets



As prototypical wire meshes we consider metal (steel) wires woven in a *plain weave*. In these most ubiquitous wire meshes, longitudinal *warp* and transversal *weft* wires are interwoven (but not welded) in a criss-cross pattern (see inset figure). For the typical forces and deformations applied to wire meshes, the stretching of each metal wire may be reasonably neglected, thus each wire is adequately modeled as an inextensible elastic curve. The elastic response of the ensemble is then governed by the bending of each curve and the interactions induced by the weave pattern.

The weave induces a "soft" interlocking of wire: while each wire may slightly slide over the crossing wires, significant sliding is uncharacteristic because it occurs only under exceedingly large forces. Consequently, adjacent contact points maintain their intrinsic distance, even for large extrinsic deformations, resulting in a structure where warp and weft directions cannot stretch but significantly shear towards or away from one another. The corresponding mathematical model is the theory of *Chebyshev nets*.

Chebyshev nets are akin to the conformal parameterizations commonly used for texture mapping. While conformal maps exactly preserve *angles* but allow for uniform stretching, Chebyshev nets preserve *lengths* along two parameter (warp and weft) directions but allow for shearing of angles between warp and weft. Let $r : \mathbb{R}^2 \to \mathbb{R}^3$ be a parameterization of a smooth surface describing a patch of a surface in space. Then $r(u, v)$ is *Chebyshev* if $|r_u| = |r_v| = 1$, where (u, v) is an orthonormal coordinate system for $\mathbb{R}^2$. A collection of such patches describing a whole surface, such that coordinate transitions are given by *translations* only, is called a *smooth Chebyshev net*.

3.1 Smooth Chebyshev Net Theory

One of the intricate mathematical difficulties for constructing smooth Chebyshev nets results from the fact that while one can *locally* equip every smooth surface in 3-space with a Chebyshev net [Bieberbach 1926], this is no longer the case *globally* without producing singularities. This mathematical difficulty translates into very concrete challenges in the design process. While it is locally possible to fit a wire mesh to a given shape, there are strong global obstructions; moreover, small local changes might have drastic global effects—making manual design cumbersome, time consuming, or intractable.

Curvature and shear The coupling between shearing of the wire mesh and curvature makes *global* existence of Chebyshev nets on an arbitrary smooth surface a delicate matter. Let $\gamma(u, v)$ be the *shear angle* of a Chebyshev net $r(u, v)$, i.e., $\sin\gamma(u, v) = r_u(u, v) \cdot r_v(u, v)$. Simply, γ measures the signed angle deviation of the originally orthogonal warp and weft directions under the deformation of the surface. The so-called Gauß equation for the local parameterization reads [Pipkin 1984]

$$\mathcal{K}(u, v) \cos\gamma(u, v) = \gamma_{uv}(u, v) \,, \tag{1}$$

where $\mathcal{K}(u, v)$ denotes the Gaussian curvature at the point $r(u, v)$. This equation reveals that changes in the shearing angle directly correspond to the encoding of curvature. Indeed, regions where the magnitude of Gaussian curvature is high correspond to large magnitude (close to $\pi/2$) shearing angles or a high rate of change of shearing angles (which then leads to large magnitude shearing angles nearby). For wire mesh design, where large magnitude shear angles are prohibitive (see Fig. 5), this results in difficulties for covering regions of high curvature.

Global obstructions to existence An important obstruction that results from the coupling between shear angle and curvature is provided by the *formula of Hazzidakis [1879]*: Consider an axis-aligned (with respect to the u and v parameter directions) rectangular domain $D \subset \mathbb{R}^2$, then it follows from Equation (1) that

$$\int_D \mathcal{K}(u, v) \mathrm{d}A = 2\pi - \sum_{i=0}^{3} \alpha_i \,, \tag{2}$$

where $\mathrm{d}A$ is the area element on the surface and the α_i are the interior angles of the quadrangle given by the image of the axis-aligned rectangle D under the Chebyshev net r.

As a consequence of Hazzidakis' formula, *if* one requires the boundary of a rectangular patch D to coincide with parameter lines, then it is impossible to cover the image of D with a Chebyshev net if this image has total Gauß curvature greater than 2π. Perhaps surprisingly, despite the Hazzidakis obstruction, Voss [1882] showed that there exists a global Chebyshev net on any surface of revolution which does not meet the rotation axis—even if total Gauß curvature exceeds 2π. Recently, Ghys [2011] proposed a Chebyshev net on the sphere (with singularities along two spherical arc segments at the south pole) that is different from the solution of Voss (with singularities at the poles where the profile curve meets the rotation axis). These results show that Hazzidakis' obstruction ceases to be valid *if one gives up on insisting on axis-aligned parameter domains*.

Inspired by these observations, our design tool introduced in §4, allows the user to both: construct *non axis-aligned domains* and to add or remove material, thus *changing the shape of the domain*.

Sufficient conditions for existence Hazzidakis' formula provides *necessary* conditions for the existence of Chebyshev nets. The search for optimal results about *sufficient* conditions for the existence of global Chebyshev nets on surfaces is still ongoing [Bakelman 1965; Samelson 1991; Samelson and Dayawansa 1995; Burago et al. 2007]. Although some of these works offer constructive proofs, they do not immediately lead to a computationally feasible algorithm. More importantly, these works assume that total negative and positive Gauß curvature does not exceed 2π in magnitude—a bound too restrictive for real-world design—and that the resulting Chebyshev net lies *exactly* on the given surface—a requirement that is neither practical nor strictly necessary in design.

Summary The above mathematical properties translate into practical difficulties when designing shapes with wire mesh. We tackle these challenges in our design tool by (i) allowing the parameter domain to be changed by the user by adding or removing material, (ii) working with Chebyshev nets *nearby* (but not exactly on) a given target shape, and (iii) accommodating for the global nature of the problem with the help of an optimizer.

3.2 Discrete Chebyshev Nets and the Role of Shear

We model wire mesh by *discrete Chebyshev nets*, rhombic nets comprised of inextensible equilateral edges such that each interior

mesh vertex has valence four. Notice that we do *not* require the rhombi to be planar. Akin to the smooth case, discrete Chebyshev nets have a long history in mathematics. Originally introduced for the special case of constant Gauß curvature surfaces [Wunderlich 1951; Sauer 1970; Bobenko and Pinkall 1996; Hoffmann 1999; Pinkall 2008], their more general theory is still an active area of research.

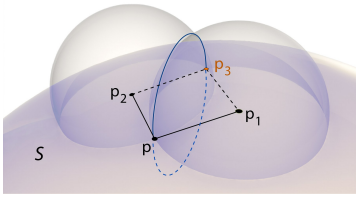


Figure 4: *Local construction of a discrete Chebyshev net.*

As in the smooth case, there exists a discrete Chebyshev net locally around any point p on a target surface $S \subset \mathbb{R}^3$. Indeed, given a small neighborhood U of p on S, choose an edge length ℓ and two unit vectors $v, w \in \mathbb{R}^3$ such that $p_1 = p + \ell v$ and $p_2 = p + \ell w$ are in U. Generically the three points p, p_1, p_2 determine a unique fourth point $p_3 \in S$ such that the quadrilateral (p, p_1, p_2, p_3) is a (non-planar) rhombus (see Fig. 4). To obtain p_3, consider the two spheres of radius l around p_1 and p_2, respectively, which intersect in a circle (shown in blue). Both p and p_3 lie on the intersection of this circle with S. Generically, p_3 is unique and distinct from p.

The role of shear While the wires of a wire mesh do not stretch and offer some resistance to bending of warp or weft directions, the quadrilateral structure of the weave allows for a considerable amount of *shear*, which is ultimately responsible for the rich set of possible wire mesh deformations. We measure the discrete signed shear angle, γ, as the deviation from 90° of the interior angles of the quadrilaterals. Wire meshes exhibit little resistance to in-plane shearing as long as the magnitude of the shear between warp and weft lines does not exceed a certain threshold. Beyond this point the required shear force increases drastically.

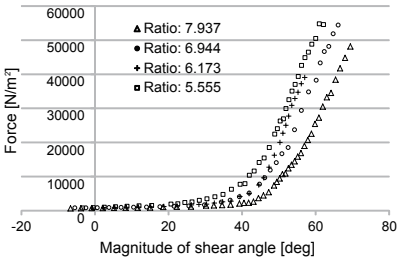
This claim is validated by several experiments that we conducted on real materials using an *Instron* machine, a device that measures tensile (or compressive) forces under a prescribed deformation. As shown in Fig. 5, the energetic cost of shearing wire mesh samples of varying gauge is negligible for shears with magnitude below a (consistent) threshold of approximately 45°. To deform a wire mesh beyond this threshold requires excessive force.

These experiments validate a *bounded-shear* model. When optimizing for a discrete Chebyshev net, we thus restrict the magnitude of the allowable shearing to a user-defined bound; our examples set the *shear limit* or *shear bound* to $\gamma_{max} = \pi/4$. Each rhombus is thus individually restricted to interior angles between $[\pi/2 - \gamma_{max}, \pi/2 + \gamma_{max}]$. We point out that the motivation for our specific choice of γ_{max} arises from experiments—other choices are indeed possible. We use the adjective *realizable* to refer to *discrete, bounded-shear* Chebyshev nets.

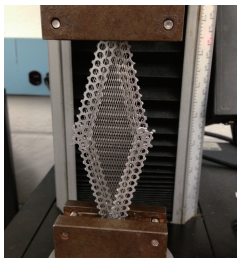
3.3 Building Discrete Chebyshev Nets

A prevalent way for constructing discrete Chebyshev nets is through a process called *integration* from appropriate *initial data* or *initial conditions*. This approach plays a central role in our implementation for *initializing* a discrete Chebyshev net on a target surface.

Interpolating integration The observation that three points of a rhombus in a discrete Chebyshev net on a (smooth or triangulated) surface S determine the fourth point leads to a construction of



Force vs. magnitude of shear angle



Instron measurement setup

Figure 5: *Measurements of shear resistance for four different wire meshes. While very little force is required to shear the mesh initially, an exponential increase in force can be observed starting at a shear angle magnitude of about 45°. In the measurements shown above, we have tested four wire meshes with different mesh opening / wire thickness ratios. The diameter of the wires in all meshes is 0.009 mm and the meshes have per inch 14 cells (ratio: 7.937), 16 cells (ratio: 6.944), 18 (ratio: 6.173) cells and 20 cells (ratio: 5.555), respectively.*

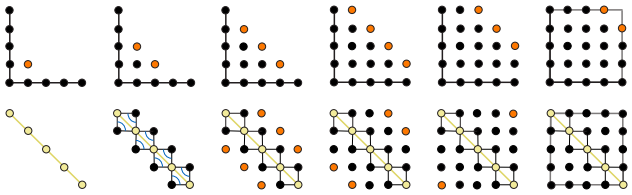


Figure 6: Top: *integration using Cauchy initial data (black polygonal curve).* Bottom: *diagonal initial data given by a curve (yellow dots) and desired angles (blue, at black dots in 2nd picture from left); diagonal data determine two zig-zag curves along the diagonal.* Left-to-right: *orange dots denote new data that are computed from previously known ones.* Rightmost figure: *bounding box shows region in the parameter domain that can be covered.*

Chebyshev nets from certain *initial condition* curves. To this end, consider a curve on S that is equidistantly sampled (with respect to the extrinsic metric of $\mathbb{R}^3$). We refer to such a curve as *Cauchy* initial data. Any vertex of this curve whose adjacent edges (with respect to its adjacent sample points) form an angle that obeys the shear limit constraint can be used as a seed for integration, see Fig. 6 (top) for a schematic illustration of this process. We additionally allow for specifying what we call *diagonal* initial data, which, different from Cauchy data, are given by a (discrete) curve on S together with angles (obeying the shear limit) for each curve segment. In this case, curve segments serve as diagonals of rhombi and angles specify the two (necessarily equal) angles opposite to the diagonal in each rhombus, see Fig. 6 (bottom). Notice that on a topological cylinder, diagonal initial data are *required* since they allow for covering the entire cylinder, while Cauchy data do not—see Fig. 7 for a schematic illustration.

Limits to interpolating integration While there exists a discrete Chebyshev net of some resolution ℓ around every point p on S given by initial data, it is impossible to know in advance how far these data will propagate the net. This makes it difficult to know if particular initial data are sufficient to construct a global discrete Chebyshev net. There are three ways in which initial data can fail to propagate a realizable Chebyshev net to globally cover the entire target surface: (i) the generic construction of Fig. 4 fails to find a new point p_3 because the circle of intersection of the constraint spheres only meets the surface at the original point p; (ii) the three points p, p_1, p_2 already determine an angle that violates the shear limit; (iii) the

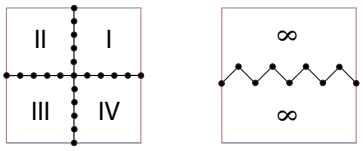


Figure 7: *Influence of initial data for a topological cylinder—in both figures, left and right side of the square are identified (glued) to form a cylinder.* Left: *axis-aligned Cauchy data allow for covering a finite region only, i.e., the depicted parts of quadrants I to IV; additionally, in general there is no guarantee that the results of integration match on left and right—possibly leading to discontinuities.* Right: *zig-zag (from diagonal data) allows for covering an infinite cylinder—without further restrictions except for the shear limit.*

prescribed initial data were not sufficient to cover the target surface because shearing in one place pulls material from another part, so more material is actually required. These three failure modes are dependent on the initial data and on the initial resolution, ℓ, at which the Chebyshev net was formed. Choosing ℓ too large in comparison to the target surface might produce an extreme approximation such as a single large rhombus for the entire target surface, while choosing ℓ too small might introduce unnecessary curvature detail that is either not of artistic interest or stems from artifacts of a triangulation.

Translation surface integration We employ an alternative method to construct discrete Chebyshev nets when interpolating integration fails. This translation surface integration method relies on the fact that three non collinear points p, p_1, p_2 in 3-space such that $|p - p_1| = |p - p_2|$ uniquely determine a fourth point p_3 such that (p, p_1, p_2, p_3) is a *planar* rhombus. That is, instead of propagating a Chebyshev net such that the fourth vertex resides on the target shape, we offer the possibility to propagate, from an initial curve on S, such that the resulting rhombi are *planar*, while maintaining the general integration paradigm depicted in Fig. 6 (top). Notice that in general the resulting net will deviate from the target shape—a property that is desirable for initializing an approximate (instead of interpolating) Chebyshev net in scenarios where integration fails. We refer to §4 for details of when we use translation surfaces instead of interpolating integration. We remark that discrete Chebyshev nets that are entirely comprised of planar rhombi are *discrete translation surfaces*—in analogy to smooth translation surfaces that are defined by $r(u, v) := a(u) + b(v)$, where $a, b : \mathbb{R} \to \mathbb{R}^3$ are smooth curves [Voss 1882; Pipkin 1984].

Next, we describe how these insights guide our design of wire mesh in practice.

4 Design Tool

We facilitate the creation of a *single, contiguous piece* of wire mesh that can be *cut out of the plane*, and bent *without inserting darts* and with *bounded shear* to approximate a desired surface, or *guide form*.

Theory informs us that the Chebyshev net constraints are globally coupled; when we additionally constrain the net to interpolate a given guide form, the design space is intractably small. Fortunately, considerable additional freedom can be gained by allowing for slight deviations from the guide form. We therefore seek designs that *approximate* rather than interpolate a given shape.

With a guide form given, the first phase is to create initial wire mesh material that *interpolates* a part of the guide form. Due to the limitations of interpolating integration laid out in the previous section, to extend coverage a second *approximating* phase is required. This second phase interweaves adding or removing wire mesh material, weight-painting, and globally optimization.

Coarse-to-fine design We find that a coarse-to-fine design process is effective. The designer first situates the wire mesh and resolves the coarsest features, before refining to focus on details. The design session begins by establishing a coarse (large cell size) wire mesh, and proceeds by iteratively subdividing the wire mesh, revising the shape, and repeating, until the finest details are resolved. The revisions employ several types of tools, categorized as either *local* or *global* in effect.

Local vs. global tools Tools with local effect alter only the selected region of the wire mesh, e.g., adding or removing mesh material; these tools do not allow the *deforming* of the wire mesh: the Chebyshev constraints are inherently global in nature, prohibiting such a local deformation. To deform the mesh, we employ a *global* optimizer; it necessarily alters almost the entire wire mesh shape. This optimizer is incorporated into the interactive workflow, and the user controls the optimization by painting weights to prioritize approximation of some target regions over others.

We now survey these tools and refer the reader to the accompanying demonstrations in the supplementary video.

4.1 Phase I - Interpolating the Guide Form

We present two interpolating integration tools to quickly and easily lay out the initial patch of material. The first is a novel zig-zag tool for both cylindrical and disk topologies, while the second is the Cauchy integration tool for disk topology similar to the work of Aono et al. [1994; 1996; 2001].

Zig-zag tool The designer draws a single "diagonal" curve segment on the surface, and specifies the (possibly varying) shear along the diagonal; using these data, the computer generates two curves that zig-zag on and off the diagonal, one on each side of the diagonal (see Fig. 6-bottom); the computer then integrates an interpolating wire mesh patch outward, using the diagonal initial data. To specify *cylindrical topology* a closed loop on the surface is specified as the initial diagonal curve instead of a curve segment.

Cauchy tool The designer draws two curve segments that *intersect* at a point; the computer then integrates an interpolating wire mesh patch from each quadrant of Cauchy initial data (see Fig. 6-top). Using the Cauchy tool with a geodesic curve, for instance, ensures a constant shear along the geodesic. Indeed, the geodesic curvature of the u-parameter lines and v-parameter lines are given by γ_u and $-\gamma_v$, respectively [Pipkin 1984]. Therefore, a Chebyshev net's parameter line lies along a geodesic if and only if the shearing γ is constant along that line. Using geodesics as initial conditions makes the Cauchy approach appealing if a geodesic is made to pass over multiple hills and valleys. We found this useful when designing the Igea head and for capturing the face and neck of the bunny (see Fig. 10).

Discussion A cylindrical Chebyshev net *requires* diagonal initial conditions and thus the zig-zag tool. In the case of a disk topology, either type of initial conditions may be used, and the choice is one of aesthetics, versatility, and convenience.

The zig-zag approach is otherwise preferred because of its versatility. One example of a recurring strategy with this tool is to pass a diagonal through a high curvature region: Knowing that curvature will decrease in magnitude away from the diagonal, the designer

specifies a high shear along the diagonal, thus allowing the mesh to reduce in shear when integrated outward.

With either of the tools, the user selects the resolution of integration, and specifies whether the shear bound should be respected. Integration then continues as far as possible subject to the initial conditions, boundary of the guide form, and (optionally) the shear bound.

Both integration methods have been optimized using spatial hashing for accurate and fast surface intersection tests. This allows interactive exploration of various integration options in order to find a good wire mesh patch that can be used in Phase II of the design process.

Drawing curves These tools require the user to draw curves on the guide surface. In our implementation we allow for three simple approaches: (i) the designer positions a plane, and the algorithm computes the plane-surface intersection; (ii) the designer picks a surface point, a direction, and a distance, and the algorithm integrates out a geodesic; (iii) the designer picks a sequence of points forming a polyline, and the algorithm projects the polyline to the guide form.

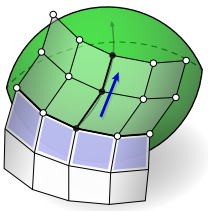
4.2 Phase II - Approximating the Guide Form

The second part of the design process allows for modeling tools that *approximate* instead of *interpolate* the guide form. There are multiple reasons why interpolating integration alone falls short: First, interpolating integration typically does not create as much material as desired, e.g., due to exhaustion of initial data. Second, *interpolation* may be too strong a request (§3.3). After other editing tools are used, the wire mesh will *approximate* rather than interpolate the guide form.

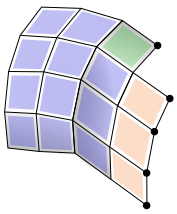
4.2.1 Adding Material

New material must be added to a given wire mesh patch when interpolating integration can no longer proceed. The tools described here allow for the creation of new material that respect the Chebyshev conditions at the expense of interpolating the guide form.

Translational surface tool This tool extends an existing wire mesh by propagating parallel to a profile curve drawn on the guide form, but does not additionally seek proximity to the guide form. The designer selects a wire mesh boundary and draws a profile curve on the guide form. The computer then creates additional wire mesh material following the translation surface integration described in §3 using the profile guide form and the wire mesh boundary as the two translation curves. This tool is particularly useful for adding wire mesh material along regions of high or oscillatory curvature.



Reflection tool This tool extends an existing wire mesh (blue quads in inset figure) by a small, local addition. After selecting a wire mesh region, the user taps the tool's hot key, and the computer extends the wire mesh by one cell. At corner inclusions, the remaining fourth vertex is uniquely determined by the Chebyshev and planarity conditions (green quad), i.e., just as for the translation surface tool. At boundary edges, the two remaining vertices are uniquely determined by *reflecting* the face across the boundary edge, trivially ensuring compatibility with adjacent extensions (orange quads). If reflection is repeated without an optimization pass (discussed below), material can be constructed that protrudes far from the guide form.



There are two hotkeys for each of these tools. Both create new Chebyshev material using the specified tool, but one guarantees the shear limit is not violated while the other allows a user specified amount of violation, usually ten percent. Therefore adding new approximate material using these tools either strictly satisfies or nearly satisfies the wire mesh realizability constraints, thereby drastically facilitating the optimizer's task (described below). By allowing the user to create new material which neither satisfies the shear constraint nor interpolates the surface, new material may always be added.

4.2.2 Removing Material

The designer uses the *cutting* tool to eliminate excess material that may otherwise buckle, to trim boundaries, or to punch holes. We do not allow cut edges to be stitched together into a dart as this would introduce valence-three vertices which cannot be fabricated.

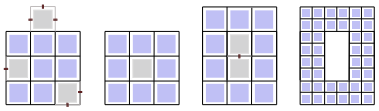
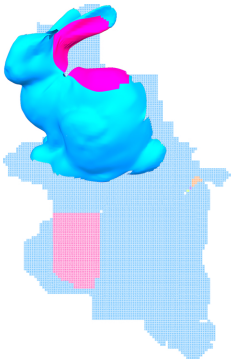


Figure 8: *From left to right: (1) cutting a boundary cell removes one, two, or three wire edges; (2) cutting an interior cell marks the cell as deleted, but does not remove wires; (3) when two adjacent cells are missing, the separating wire is removed; (4) when the mesh is subdivided, wires are not inserted in deleted cells.*

The cutting tool is simple to use: the designer selects wire mesh cells, and presses the cutting hot key. The selected cells are then marked as deleted. Wires that bound a live (not deleted) cell are retained, and the remaining wires are discarded.

4.2.3 Visualizing and Correcting Parametric Overlap

Repeated cutting and material addition can unintentionally create a design that cannot be cut out of a planar wire mesh, by adding excess material that overlaps in the parametric domain. Throughout the design process, overlapping regions of the parametric domain are brought to the designer's attention via highlighted wire mesh cells, as depicted in the inset figure of the bunny.



Detecting cells that overlap in the parametric domain is a straightforward exercise in reference counting. Because the wire mesh has strictly regular grid connectivity, each cell is easily indexed by integer Cartesian coordinates; when two or more cells have identical coordinates, they overlap.

Our interface automatically detects and highlights, but does not prevent parameteric overlap. We have found the ability to temporarily create overlaps to be an indispensable discretion during the design process. We benefited from freely extending material, temporarily ignoring the highlighted overlaps, and later choosing whether to correct them by cutting from the new material, or from the older overlapping region. The inset shows that the wire mesh on the left ear of the bunny and on its back refer to the same region in the parameter domain, so the artist must make a choice. Similar choices were made in many of our examples (see Fig. 10), where the designer traded: the entire left arm of the *armadillo man* for the shell on its back; the bump on the *Stanford bunny's* back for the exterior of its left ear; and the back of the *Igea* head for the front.

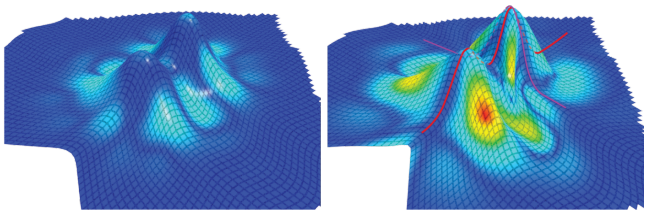


Figure 9: *Unconstrained global optimization finds a wire mesh close to the target (left), while fixing the wire mesh along two curves (in red) as hard constraints during global optimization produces large deviations from the target (right). Deviation from the target is colored from blue to red.*

4.2.4 Optimization and Form Shaping

Due to the global nature of Chebyshev nets, our design workflow uses a global optimization approach to improve the shape quality while ensuring the realizability of a design. The optimizer is intended to make a quick calculation that does not impede the interactive, multi-faceted system of tools afforded for design. To use the optimizer, the designer paints weights onto the guide surface to indicate where a close fit should be prioritized. The optimizer seeks to balance the quality of the fit against the fairness of the wire mesh. Global optimization with no hard constraints is essential to find a satisfactory result as illustrated in Figure 9.

Solver The constraints are enforced using a generic geometric framework [Bouaziz et al. 2012]. The objective function, together with these constraints, are solved using the augmented Lagrangian technique of [Deng et al. 2013] with one modification: the closeness term is based on distance measured to the guide form, as opposed to displacement from original vertex positions. Briefly, this solver works by introducing auxiliary variables to transform the original problem into separable subproblems with closed-form solutions. This separable structure allows subproblems to be solved in parallel leading to significant speedups on multi-core systems [Bouaziz et al. 2012] and rapid convergence to approximate solutions [Deng et al. 2014].

Problem statement Given a design represented as a Chebyshev net, we write the optimization problem in terms of vertex positions $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$:

$$\begin{aligned} \min_{\mathbf{x}} \;& w_{\text{fair}} F_{\text{fair}}(\mathbf{x}) + F_{\text{close}}(\mathbf{x}) \\ \text{s.t.} \;& \|\mathbf{x}_i - \mathbf{x}_j\| = \ell \;\; \forall (i,j) \in \mathcal{E}, \text{(Chebyshev net)} \\ & |\frac{\pi}{2} - \angle \mathbf{x}_i \mathbf{x}_j \mathbf{x}_k| \leq \gamma_{max} \;\; \forall (i,j,k) \in \mathcal{A}, \text{(Shear limit)} \end{aligned}$$

where the function F_{close} penalizes the deviation between the mesh and guide surface, F_{fair} measures the fairness of the mesh, $\mathcal{E}$ is the index set of vertices that lie on a common edge, $\mathcal{A}$ is the index set of vertices that form a corner of a quad face; ℓ is the constant edge length, γ_{max} is the maximum amount of shear in radians, and w_{fair} is a user-specified positive weight to control the tradeoff between closeness and fairness. F_{fair} is a quadratic energy defined using the second order difference of vertices

$$F_{\text{fair}}(\mathbf{x}) = \sum_{(i,j,k) \in \mathcal{F}} \|\mathbf{x}_i - 2\mathbf{x}_j + \mathbf{x}_k\|^2,$$

where $\mathcal{F}$ is the index set of three consecutive vertices that lie on a common wire. Such a fairing term inhibits the wire mesh from folding onto itself. F_{close} is the weighted sum of squared distance from the mesh vertex to the guide form

$$F_{\text{close}}(\mathbf{x}) = \sum_{i=1}^{n} W(P(\mathbf{x}_i)) \|\mathbf{x}_i - P(\mathbf{x}_i)\|^2,$$

where $P(\mathbf{x}_i)$ is the closest projection of $\mathbf{x}_i$ onto the guide form, and W is a local weight function painted onto the guide form by the user, or a global constant when no local weights are specified.

Closeness term The closest projection $P(\mathbf{x}_i)$ of a vertex of the wire mesh is approximated using a signed distance field. This provides significant speed improvements to the optimization. The field is precomputed on a very high resolution grid (with 20% padding) to capture all the details of the guide form. To emphasize important features of the guide form during global optimization, the local weights $W(P(\mathbf{x}_i))$, are painted onto the guide form vertices and linearly interpolated across the faces.

Discussion The optimization has three parameters which may be changed at any point during the design loop: (i) the global fairness weight w_{fair}, (ii) the surface closeness weights $W(P(\mathbf{x}_i))$, and (iii) the number of iterations to perform. Recall that the design loop tools guarantee that the mesh is Chebyshev and in almost every instance that the shear limit constraint is also satisfied, even though material is being added and removed. Additionally, any new material roughly approximates the guide form. Therefore the modified wire mesh is a good initialization for the global optimization.

Timings We chose a solver which finds approximate solutions quickly. The compute time for a single iteration is 27ms on a modern laptop computer for a mesh with 15,000 vertices. This allows us to achieve interactive performance even for 500-1000 iterations which satisfies the Chebyshev net constraint usually within 1% and the shear limit constraint within 3-5%. The designer therefore has good intuition of the final result at interactive speeds. At the highest subdivided resolution (the edge length is that of the physical wire mesh) we run a final optimization. As for many non-convex optimization problems, there is no guarantee that the solver converges. In practice, however, we observed that 10,000 iterations were sufficient to fabricate the results.

4.2.5 Enriching detail via subdivision

To resolve finer details, the designer invokes the subdivision tool. The subdivision scheme globally quadrisects each cell keeping all the original ("even") vertices fixed, while introducing new ("odd") vertices at the cell centroids and edge midpoints. This subdivision automatically preserves the Chebyshev constraints.

5 Results

Our design tool is implemented as a plugin of OpenFlipper [Möbius and Kobbelt 2012]. The accompanying video provides a didactic, visual explanation of the design and interaction process. Figures 10 & 13 showcase designs created with this process, and Table 1 summarizes the associated statistics. These examples demonstrate coverage of large parts of intricate geometries with a single sheet of wire mesh. In all designs, the Chebyshev nets are restricted to a shear limit of $\pi/4$. The optimization automatically distributes shear non-uniformly (see shear distributions, Fig. 10, fourth column) so as to simultaneouly satisfy the wire mesh constraints and adhere closely to the guide surface (see deviations plotted in Fig. 10, third column).

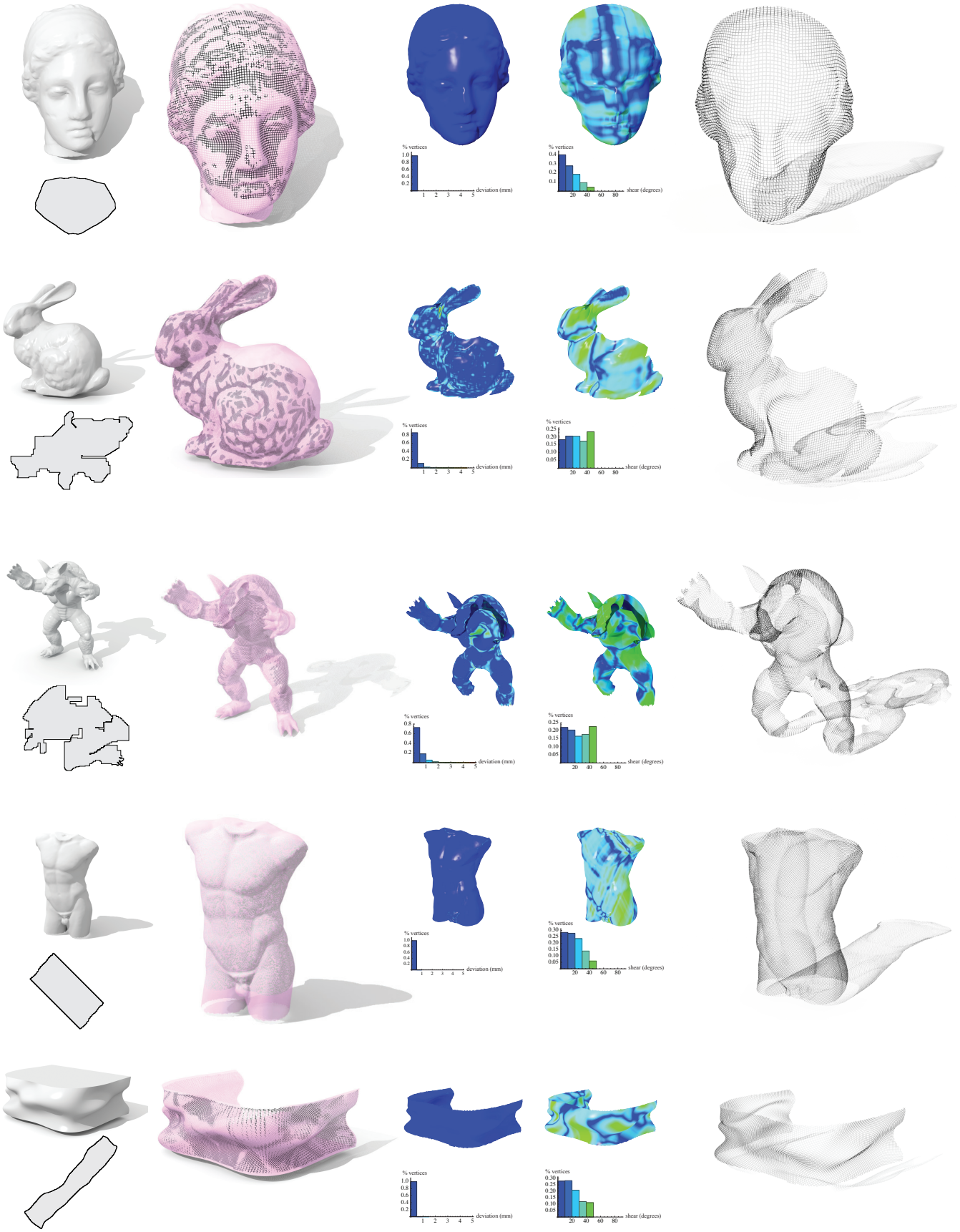


Figure 10: *Three computer graphics classics, a male torso, and a freeform facade modeled as wire meshes. From left to right: guide surface and final flattened wire mesh, overlay of wire mesh and guide surface, deviation from guide surface, shear distribution, final wire mesh.*

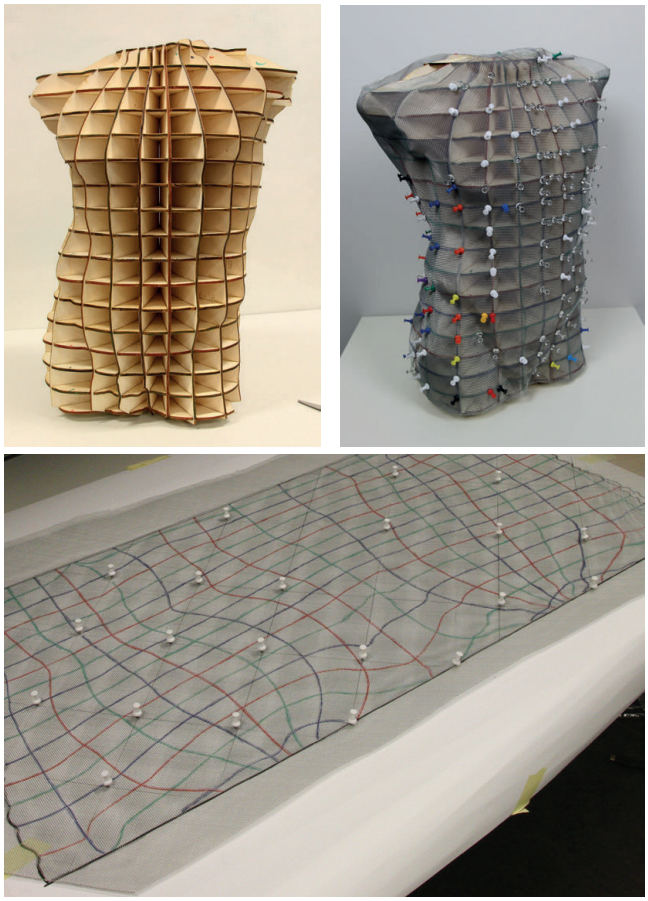


Figure 11: *Physical fabrication workflow: A 3D scaffold is created by laser cutting intersecting planar pieces (top-left); the planar wire mesh material is labeled and cut according to the flattened Chebyshev design net (bottom); the mesh is bend into place according to the labelled curves and pinned to the support (top-right); after removing the support, a free standing sculpture of the torso is obtained. The slight tilt of the model results from the non-horizontal lower boundary curve.*

For the Igea, bunny, and Armadillo models, the domain of each Chebyshev net has been designed interactively using the tools described in §4. Through the combination of interaction and optimization, we can capture not only geometrically delicate features such as the bunny ears, but also the global surface structure of the guide surfaces. The Armadillo model is particularly challenging due to the high total curvature resulting from the geometric complexity of its salient features. In the design of this wire mesh, clear tradeoffs have to be made between surface coverage and guide surface adherence. For example, as seen on the inset figure to the left, when trying to retain the bump on the back of the Armadillo the left arm cannot be covered without overlap in the parametric domain.

The male torso, female torso, Moai statue, and facade models have been designed starting from a cylindrical topology. After finding an initial layout of a coarse cylindrical mesh, the design is refined by interleaving mesh edits, subdivision and optimization, to gradually capture prominent features of increasing geometric frequency. While the facade may seem simpler than the torso, there is more curvature variation on the facade model, making the design process more time consuming as the exact amount of material required for a full covering has to be found via interaction. As one bump on the facade is captured better through optimization, material is pulled away from other regions, necessitating addition of material and further optimization. After three levels of subdivision, the optimized meshes fit well to the target surface (see Fig. 10, third column). The flat back of the facade model was trimmed away to reveal the final disk topology.

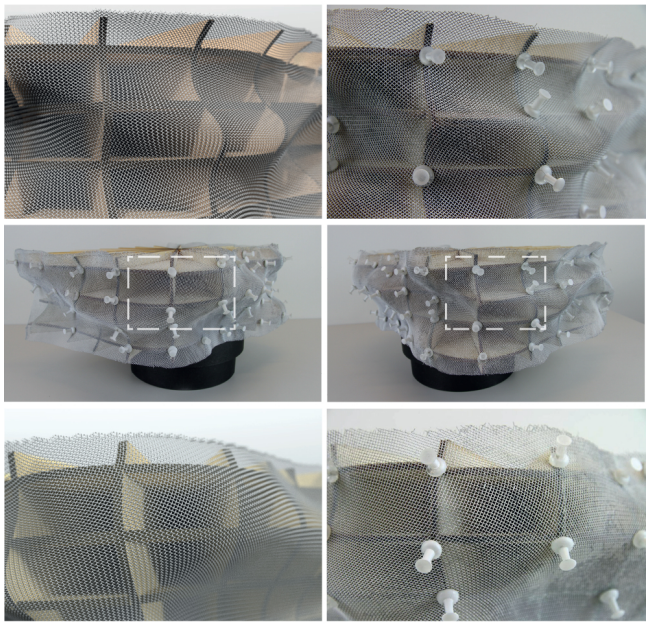


Figure 12: *The fabricated facade (center), with a comparison between renderings of the designed Chebyshev net (top and bottom left) and photographs of the physical model (top and bottom right).*

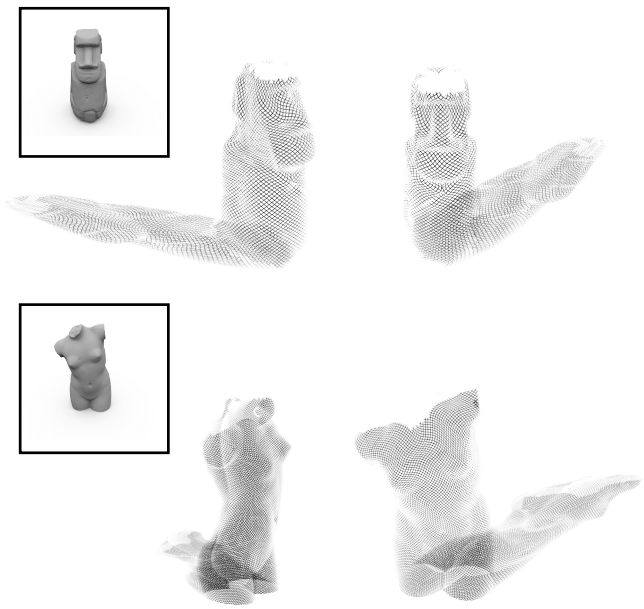


Figure 13: *Wire meshes of a Moai statue (top) and a female torso sculpture (bottom), designed using zig-zag initial conditions to account for their cylindrical topologies.*

Model	N	K	Time
IGEA	41,700	66.21	10 min
ARMADILLO	66,019	175.96	2 hr
BUNNY	98,239	65.72	2.5 hr
FACADE	66,351	107.04	45 min
TORSO	230,880	124.40	10 min

Table 1: *Statistics for our design studies. N denotes the number of vertices, K measures the total discrete Gaussian curvature as the sum of all vertex angle defects. Time is the total design time including exploration. All numbers refer to the final wire mesh.*

Observe the choice of a diagonal orientation of the wires in many cases and the non axis-aligned domains, thus circumnavigating the restrictions enshrined by the Hazzidakis formula. Indeed, the total discrete Gaussian curvature in all examples significantly exceeds the fundamental 2π limit that is imposed on axis aligned rectangular domains by the Hazzidakis formula. This illustrates that the choice of the right domain is essential when aiming for *single-sheet* coverage of curved surfaces.

While numerous artists have manually created compelling wire mesh sculptures of the human body, to our knowledge, no example exists that has cylindrical topology like our torso model, i.e., represents a *complete* section of a body. Previous examples like the one shown in Fig. 3 only show the front part with a wire mesh of disk topology.

The facade model illustrates the potential for architectural applications. We cover a complex facade with a single sheet of wire mesh, avoiding patch boundaries with their attendant inconsistencies and visible seems; such contiguous designs improve the visual quality of wire mesh claddings and freeform facades.

Fabrication To validate the agreement of physical wire meshes with digitally designed Chebyshev nets, we fabricated four of the designs. We use 0.34 mm gauge stainless steel wires, woven with a plain weave into a wire mesh with 1 mm square openings; correspondingly, our digital designs have at their finest resolution a 1.34 mm centerline spacing. We fabricate in three stages (see Fig. 11):

First, we fabricate a scaffold: The wire mesh design is triangulated; to avoid the bias introduced by cutting a quadrilateral by a diagonal, we add a vertex at the center of each quadrilateral. We employ *Autodesk*'s *123D Make* to transform the triangulated mesh into two orthogonal families of planar cross sections, which we laser-cut from 4mm softwood, glue together, and sand at each contour plane intersection (see Fig. 11-top-left).

Figure 14: *Physical realizations (middle & right) of the female torso (left).*

Second, we color the scaffold, referring to an intersection map: We intersect the digital wire mesh against the digital scaffold model; the intersecting faces typically form a network of curves, to which we assign three colors. We then color the contours of the fabricated scaffold according to this color convention. Correspondingly, we digitally map the colored curve network to the parametric plane using the parametric plane construction of §4.2.3. The flattened network of colored curves forms our *intersection map*, which we print on paper and transfer onto a large piece of planar shear-free wire mesh (see Fig. 11-bottom).

Third, the planar piece of wire mesh is manually bent: We bend the mesh so as to bring the guide curves into alignment with the contours of the scaffold, using push pins to fix the mesh in place. We chose pins with heads large enough to prevent too much slippage from occurring, but small enough to allow the wire mesh to shear. We leave the mesh affixed to the scaffold for 48–72h, allowing time for plastic flow under the applied strain (see Fig. 11-top-right), at which time we remove the pins, and the wire mesh, from the scaffold, yielding a freestanding wire mesh (see Figs. 1 & 12).

6 Conclusion & Future Work

Computational wire mesh design is a new approach for creating compelling 3D models composed of woven materials. We employ results from the theory of Chebyshev nets to shed light on the intrinsic difficulties of designing with wire meshes. Our analysis calls for a *global approach* with *local control*. We leverage and coordinate the human ability to *understand shape*, and the computer's ability to *optimize shape* subject to thousands of constraints.

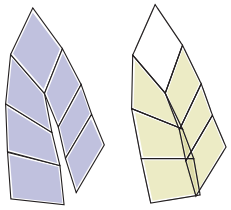


Figure 15: *Left: before optimization. Right: after optimization, causing self-intersections.*

Limitations A fundamental challenge, both theoretically and practically, of Chebyshev nets is whether a given guide form can be covered in its entirety. Consequently, in our digital design process it is difficult to anticipate the amount of material required to cover the target surface. Our user therefore iteratively adds material, guided by intuition. However, if too much material is supplied at a coarse resolution, subdivision introduces buckles and folds, which are most easily corrected by backtracking the design process. Conversely, if too little material is available near a cut, the optimizer may attempt to close the cut, producing non-local self-intersections; while the fairing energy helps reduce local self-intersections and buckling, it does not prevent more general self-intersections (see Fig. 15). Providing more powerful tools to address buckles and self-intersections would accelerate the design process.

Our fabrication process also poses challenges when dealing with the "spring-back" of the physical wire mesh material. A real wire mesh must be "over bent" in certain regions so that it springs back into the desired form. It would be interesting to investigate how to account for this over bending in future work. Even without over bending, producing scaffolds for arbitrary shapes is not easy; *123D Make* constructs strong scaffolds for closed, nearly convex surfaces such as the torso or facade. Generating such scaffolds for highly non-convex shapes, such as the bunny or armadillo, is more difficult. While *123D Make* generates two families of planar contours orthogonal to each other, the works of Cignoni et al. [2014] and Schwartzburg & Pauly [2013] construct scaffolds from planar pieces cut at arbitrary angles to one another; these works can hopefully be extended to provide sufficiently strong scaffolds for our fabrication process.

Future work While we make no attempt to directly advance the theory of Chebyshev nets, we hope that by exposing empirical evidence for the rich space of Chebyshev nets that abound under relaxed constraints on surface adherence, our work might inspire new theoretical investigations on approximative nets.

We look forward to extensions of the optimization step that incorporate additional design objectives, such as accounting for the influence of gravity, or optimizing shadows and shade. Indeed, wire meshes are popular in facade applications to reduce solar radiation, where wire thickness, spacing, and shearing all affect shadowing capacity. Extending our software to the design of wire meshes with desirable spatially varying shading capacity could therefore be a powerful tool in architectural form finding.

Acknowledgements

We thank Cosmo Wenman for providing the digital models of the torsos from their physical counterparts in The Louvre; Zohreh Sasanian for helping with the fabrication process; Keenan Crane, Henrique Maia and Nora Wixom for helping prepare the final version of this paper; Keith Yeager for allowing us to use the Instron machine; and Mario Botsch for initial discussions on wire mesh design.

This work was supported in part by the JSPS Postdoctoral Fellowships for Research Abroad, NSF (Grants IIS-1319483, CMMI-1331499, IIS-1217904, IIS-1117257, CMMI-1129917, IIS-0916129), Israel-US BSF, Niedersachsen-Israel (grant "Spectral Methods in Geometry Processing: Theory and Applications"), SNSF (grant 200021_137626), Intel, The Walt Disney Company, Autodesk, Side Effects, and NVIDIA. This research has received funding from the European Research Council under the European Unions Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement 257453, ERC Starting Grant COSYM.

References

ADKINS, J. E. 1956. Finite Plane Deformation of Thin Elastic Sheets Reinforced with Inextensible Cords. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 249*, 961, 125–150.

AONO, M., DENTI, P., BREEN, D. E., AND WOZNY, M. J. 1996. Fitting a woven cloth model to a curved surface: dart insertion. *IEEE Computer Graphics and Applications 16*, 5, 60–70.

AONO, M., BREEN, D. E., AND WOZNY, M. J. 2001. Modeling methods for the design of 3D broadcloth composite parts. *Computer-Aided Design 33*, 13, 989–1007.

AONO, M. 1994. *Computer-Aided Geometric Design for Forming Woven Cloth Composites*. PhD thesis, Rensselaer Polytechnic Institute.

BAILLARGEON, Y., AND VU-KHANH, T. 2001. Prediction of fiber orientation and microstructure of woven fabric composites after forming. *Composite Structures 52*, 3-4, 475–481.

BAKELMAN, I. Y. 1965. Chebyshev networks in manifolds of bounded curvature. *Trudy Matematicheskogo Instituta im. VA Steklova 76*, 124–129.

BIEBERBACH, L. 1926. über Tchebychefsche Netze auf Flächen negativer krümmung, sowie auf einigen weiteren flächenarten. *Preuss. Akad. Wiss., Phys. Math. Kl 23*, 294–321.

BO, P., POTTMANN, H., KILIAN, M., WANG, W., AND WALLNER, J. 2011. Circular arc structures. *ACM Trans. Graph. (SIGGRAPH '11) 30*, 4, 101:1–101:11.

BOBENKO, A. I., AND PINKALL, U. 1996. Discrete surfaces with constant negative Gaussian curvature and the Hirota equation. *Journal of Differential Geometry 43*, 3, 527–611.

BOUAZIZ, S., DEUSS, M., SCHWARTZBURG, Y., WEISE, T., AND PAULY, M. 2012. Shape-up: Shaping discrete geometry with projections. *Comp. Graph. Forum (SGP '12) 31*, 5, 1657–1667.

BURAGO, Y. D., IVANOV, S. V., AND MALEV, S. G. 2007. Remarks on Chebyshev Coordinates. *Journal of Mathematical Sciences 140*, 4, 497–501.

CIGNONI, P., PIETRONI, N., MALOMO, L., AND SCOPIGNO, R. 2014. Field-aligned mesh joinery. *ACM Trans. Graph. 33*, 1, 11:1–11:12.

DE GOES, F., ALLIEZ, P., OWHADI, H., AND DESBRUN, M. 2013. On the equilibrium of simplicial masonry structures. *ACM Trans. Graph. (SIGGRAPH '13) 32*, 4, 93:1–93:10.

DENG, B., BOUAZIZ, S., DEUSS, M., ZHANG, J., SCHWARTZBURG, Y., AND PAULY, M. 2013. Exploring local modifications for constrained meshes. *Computer Graphics Forum (EUROGRAPHICS '13) 32*, 2, 11–20.

DENG, B., BOUAZIZ, S., DEUSS, M., KASPAR, A., SCHWARTZBURG, Y., AND PAULY, M. 2014. Interactive design exploration for constrained meshes. *Computer-Aided Design*. To appear.

EIGENSATZ, M., KILIAN, M., SCHIFTNER, A., MITRA, N., POTTMANN, H., AND PAULY, M. 2010. Paneling architectural freeform surfaces. *ACM Trans. Graph. (SIGGRAPH '10) 29*, 4, 45:1–45:10.

GHYS, É. 2011. Sur la coupe des vêtements: variation autour d'un thème de Tchebychev. *L'Enseignement Mathématique Revue Internationale 2e Sèrie 57*, 1-2, 165–208.

HAZZIDAKIS, J. N. 1879. Über einige Eigenschaften der Flächen mit constantem Krümmungsmass. *Journal für die reine und angewandte Mathematik 88*, 68–73.

HOFFMANN, T. 1999. Discrete Amsler surfaces and a discrete Painlevé III equation. *Oxford Lecture Series in Mathematics and its Applications 16*, 83–96.

LIU, Y., POTTMANN, H., WALLNER, J., YANG, Y.-L., AND WANG, W. 2006. Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph. (SIGGRAPH '06) 25*, 3, 681–689.

LIU, Y., PAN, H., SNYDER, J., WANG, W., AND GUO, B. 2013. Computing self-supporting surfaces by regular triangulation. *ACM Trans. Graph. (SIGGRAPH '13) 32*, 4, 92:1–92:10.

MÖBIUS, J., AND KOBBELT, L. 2012. Openflipper: An open source geometry processing and rendering framework. In *Curves and Surfaces*, J.-D. Boissonnat, P. Chenin, A. Cohen, C. Gout, T. Lyche, M.-L. Mazure, and L. Schumaker, Eds., vol. 6920 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 488–500.

PANOZZO, D., BLOCK, P., AND SORKINE-HORNUNG, O. 2013. Designing unreinforced masonry models. *ACM Trans. Graph. (SIGGRAPH '13) 32*, 4, 91:1–91:12.

PINKALL, U. 2008. Designing cylinders with constant negative curvature. In *Discrete Differential Geometry*. Springer, 57–66.

PIPKIN, A. C. 1984. Equilibrium of Tchebychev nets. *Archive for Rational Mechanics and Analysis 85*, 1, 81–97.

PIPKIN, A. C. 1986. Continuously distributed wrinkles in fabrics. *Archive for Rational Mechanics and Analysis 95*, 2, 93–115.

PORANNE, R., OVREIU, E., AND GOTSMAN, C. 2013. Interactive planarization and optimization of 3D meshes. *Computer Graphics Forum 32*, 1, 152–163.

POTTMANN, H., SCHIFTNER, A., BO, P., SCHMIEDHOFER, H., WANG, W., BALDASSINI, N., AND WALLNER, J. 2008. Freeform surfaces from single curved panels. *ACM Trans. Graph. (SIGGRAPH '08) 27*, 3, 76:1–76:10.

POTTMANN, H., HUANG, Q., DENG, B., SCHIFTNER, A., KILIAN, M., GUIBAS, L., AND WALLNER, J. 2010. Geodesic patterns. *ACM Trans. Graph. (SIGGRAPH '10) 29*, 4, 43:1–43:10.

RIVLIN, R. S. 1958. The deformation of a membrane formed by inextensible cords. *Archive for Rational Mechanics and Analysis 2*, 1, 447–476.

RIVLIN, R. S. 1964. Networks of inextensible cords. In *Nonlinear Problems of Engineering*. Academic Press Professional, Inc, New York, 51–64.

RIVLIN, R. S. 1997. Plane Strain of a Net Formed by Inextensible Cords. In *Collected Papers of R.S. Rivlin*, G. I. Barenblatt and D. D. Joseph, Eds. Springer New York, 511–534.

ROBERTSON, R. E., HSIUE, E. S., SICKAFUS, E. N., AND YEH, G. S. Y. 1981. Fiber rearrangements during the molding of continuous fiber composites. I. Flat cloth to a hemisphere. *Polymer composites 2*, 3, 126–131.

ROBERTSON, R. E., CHU, T. J., GERARD, R. J., KIM, J. H., PARK, M., KIM, H. G., AND PETERSON, R. C. 2000. Three-dimensional fiber reinforcement shapes obtainable from flat, bidirectional fabrics without wrinkling or cutting. Part 2: a single n-sided pyramid, cone, or round box. *Composites Part A: Applied Science and Manufacturing 31*, 11, 1149–1165.

SAMELSON, S. L., AND DAYAWANSA, W. P. 1995. On the Existence of Global Tchebychev Nets. *Transactions of the American Mathematical Society 347*, 2, 651–660.

SAMELSON, S. L. 1991. Global Tchebychev Nets on Complete Two-Dimensional Riemannian Surfaces. *Archive for Rational Mechanics and Analysis 114*, 3, 237–254.

SAUER, R. 1970. *Differenzengeometrie*. Springer Verlag, Berlin.

SCHWARTZBURG, Y., AND PAULY, M. 2013. Fabrication-aware design with intersecting planar pieces. *Computer Graphics Forum (EUROGRAPHICS '13) 32*, 2, 317–326.

TSCHEBYSCHEFF, P. L. 1878. Sur la coupe des vêtements, "On the cutting of garments". *Association francaise pour l'avancement des sciences*, 154–155.

VAN WEST, B. P., PIPES, R. B., AND KEEFE, M. 1990. A Simulation of the Draping of Bidirectional Fabrics over Arbitrary Surfaces. *Journal of the Textile Institute 81*, 4, 448–460.

VOSS, A. 1882. Über ein neues Princip der Abbildung krummer Oberflächen. *Mathematische Annalen 19*, 1–26.

VOUGA, E., HÖBINGER, M., WALLNER, J., AND POTTMANN, H. 2012. Design of self-supporting surfaces. *ACM Trans. Graph. (SIGGRAPH '12) 31*, 4, 87:1–87:11.

WANG, W. B., AND PIPKIN, A. C. 1986. Inextensible networks with bending stiffness. *Quarterly Journal of Mechanics & Applied Mathematics 39*, 3, 343–359.

WANG, J., PATON, R., AND PAGE, J. R. 1999. The draping of woven fabric preforms and prepregs for production of polymer composite components. *Composites Part A: Applied Science and Manufacturing 30*, 6, 757–765.

WANG, W., LIU, Y., YAN, D., CHAN, B., LING, R., AND SUN, F. 2008. Hexagonal meshes with planar faces. *HKU CS Tech Report TR-2008-13*.

WUNDERLICH, W. 1951. *Zur Differenzengeometrie der Flächen konstanter negativer Krümmung*. Springer Verlag.

YE, L., AND DAGHYANI, H. R. 1997. Characteristics of woven fibre fabric reinforced composites in forming process. *Composites Part A: Applied Science and Manufacturing 28*, 9, 869–874.