

# Computational Design of Weingarten Surfaces

DAVIDE PELLIS, EPFL, Switzerland  
MARTIN KILIAN, TU Wien, Austria  
HELMUT POTTMANN, KAUST, Saudi Arabia  
MARK PAULY, EPFL, Switzerland

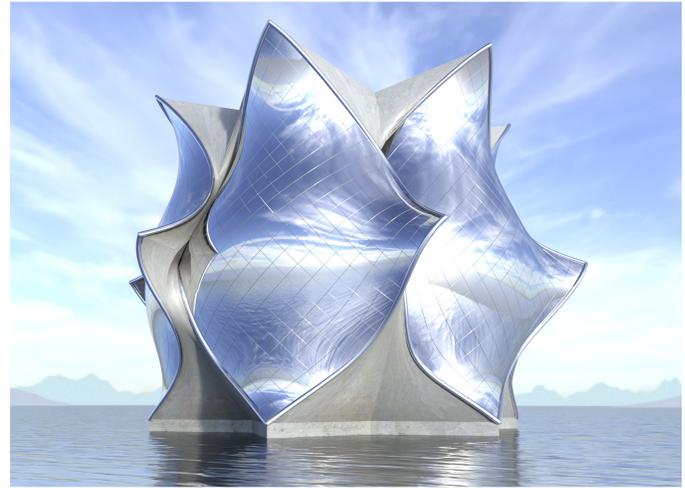
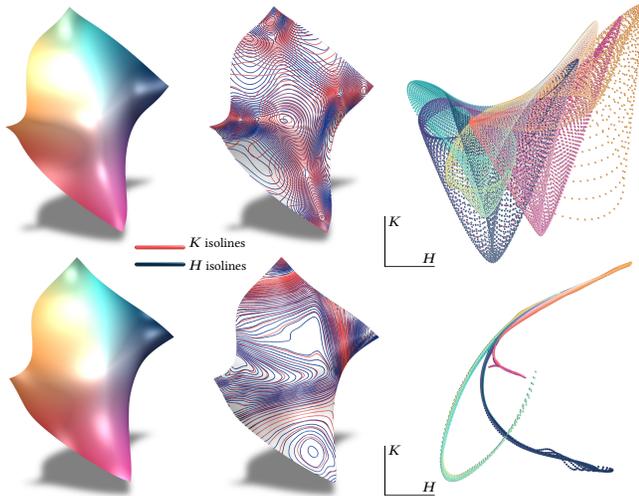


Fig. 1. Design of Weingarten surfaces. *Top row:* B-spline surface  $S$ , corresponding isolines of Gaussian curvature  $K$  (red) and mean curvature  $H$  (blue), and the associated curvature diagram in the  $(H, K)$ -plane. *Bottom row:* Result of optimization of  $S$  towards a Weingarten surface. Curvature isolines become aligned and the curvature diagram assumes a curve-like shape. Colors show corresponding points on the surface and in the  $(H, K)$ -plane. Potential areas of application of Weingarten surfaces include the computation of cost efficient paneling solutions on architectural freeform skins.

In this paper we study Weingarten surfaces and explore their potential for fabrication-aware design in freeform architecture. Weingarten surfaces are characterized by a functional relation between their principal curvatures that implicitly defines approximate local congruences on the surface. These symmetries can be exploited to simplify surface paneling of double-curved architectural skins through mold re-use.

We present an optimization approach to find a Weingarten surface that is close to a given input design. Leveraging insights from differential geometry, our method aligns curvature isolines of the surface in order to contract the curvature diagram from a 2D region into a 1D curve. The unknown functional curvature relation then emerges as the result of the optimization. We show how a robust and efficient numerical shape approximation method can be implemented using a guided projection approach on a high-order B-spline representation. This algorithm is applied in several design studies to illustrate how Weingarten surfaces define a versatile shape space for fabrication-aware exploration in freeform architecture. Our optimization

algorithm provides the first practical tool to compute general Weingarten surfaces with arbitrary curvature relation, thus enabling new investigations into a rich, but as of yet largely unexplored class of surfaces.

CCS Concepts: • **Computing methodologies** → **Shape modeling**.

Additional Key Words and Phrases: fabrication-aware design, computational differential geometry, geometric optimization, Weingarten surface, architectural geometry, paneling, mold reduction

## ACM Reference Format:

Davide Pellis, Martin Kilian, Helmut Pottmann, and Mark Pauly. 2021. Computational Design of Weingarten Surfaces. *ACM Trans. Graph.* 40, 4, Article 114 (August 2021), 11 pages. <https://doi.org/10.1145/3450626.3459939>

## 1 INTRODUCTION

Freeform surfaces play a prominent role in contemporary architecture. While digital modeling tools have simplified their design, construction remains highly challenging. Architectural designs are typically only built once and at large scale, which impedes cost reduction measures common for mass-produced freeform surfaces, such as cars or consumer products.

Constructing doubly curved architectural skins within a reasonable budget requires the freeform design surface to be segmented into panels that are pre-fabricated and mounted on site. A smooth global appearance can often only be achieved with doubly curved panels, which usually requires the fabrication of molds. Paneling

Authors' addresses: Davide Pellis, EPFL, Lausanne, Switzerland, [davide.pellis@epfl.ch](mailto:davide.pellis@epfl.ch); Martin Kilian, TU Wien, Vienna, Austria, [kilian@tuwien.at](mailto:kilian@tuwien.at); Helmut Pottmann, KAUST, Thuwal, Saudi Arabia, [helmut.pottmann@kaust.edu.sa](mailto:helmut.pottmann@kaust.edu.sa); Mark Pauly, EPFL, Lausanne, Switzerland, [mark.pauly@epfl.ch](mailto:mark.pauly@epfl.ch).

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2021 Copyright held by the owner/author(s).  
0730-0301/2021/8-ART114

<https://doi.org/10.1145/3450626.3459939>

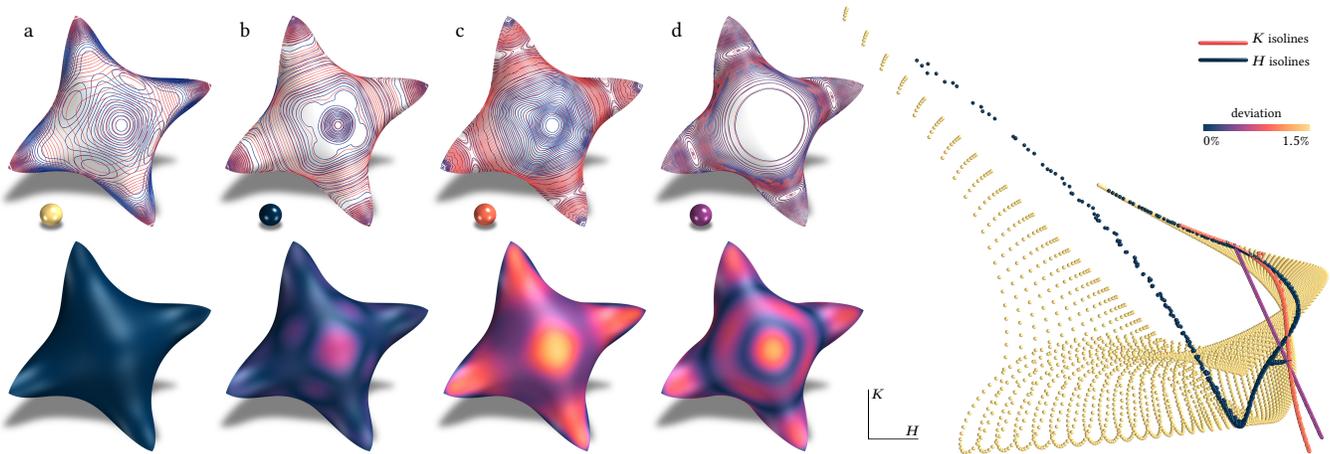


Fig. 2. Surface approximation with W-surfaces. The reference surface shown in (a) is optimized towards a Weingarten surface through curvature isolines alignment (b). The same shape is approximated also with Weingarten surfaces constrained to a B-spline (c), and a linear (d) curvature relation in the  $(H, K)$ -plane. The corresponding curvature diagrams are shown on the right. Below, surface colors indicate the deviation from the reference shape, expressed as percentage of its bounding box diagonal. We observe that without constraining the curvature relation  $g(H, K)$ , we achieve a better shape approximation.

algorithms therefore aim at computing panel layouts with high surface quality, while using as few and as simple molds as possible to reduce cost. The state of the art in this field is the post-rationalization method of [Eigensatz et al. 2010], which exploits tolerances at the panel seams (small gaps, kink angles) and controlled deviations from the target geometry to keep the production cost as low as possible within the chosen tolerances. This approach has been applied in various architectural projects (see [Pottmann et al. 2015]).

Clearly, symmetries can significantly reduce the number of required molds. For example, rotational and helical surfaces, which can continuously move in themselves, can be covered with congruent panels along the circular or helical paths of the generating motion. Panel congruence facilitates mold re-use and thus simplifies fabrication, yet such restricted classes of surfaces are in general too limited for freeform design.

Recently, a different class of "repetitive" surfaces, so-called *Weingarten surfaces* (*W-surfaces*), have received interest in the present context [Pellis et al. 2020a,b]. These surfaces are characterized by a relation between their principal curvatures  $\kappa_1, \kappa_2$  or between their Gaussian curvature  $K$  and mean curvature  $H$ :

$$f(\kappa_1, \kappa_2) = 0 \quad \text{or} \quad g(H, K) = 0. \quad (1)$$

They are named in appreciation of the work of Julius Weingarten [1861], who studied them first<sup>1</sup>. He obtained the following remarkable result: The focal surface (set of principal curvature centers) of a W-surface is isometric to a rotational surface, which is determined only by the curvature relation  $f(\kappa_1, \kappa_2) = 0$ . Hence, the focal surface of a W-surface is intrinsically repetitive.

A smooth surface can be approximated locally up to second order by the osculating paraboloid, which can be considered a locally well fitting panel. In an adapted local frame, its equation is  $2z = \kappa_1 x^2 + \kappa_2 y^2$ . Since a W-surface has only a one-parameter family of different

<sup>1</sup>In the very same paper, the shape operator (also called Weingarten map) appears for the first time.

osculating paraboloids, we can expect a significant reduction in the number of needed molds for paneling compared to a general freeform surface. Roughly speaking, a design with  $N \times N$  panels should only require  $N$  different molds, assuming the panels are sufficiently small compared to the variation of curvature in the design surface. [Pellis et al. 2020a,b] verify this expectation for some known Weingarten surfaces with simple prescribed curvature relations.

However, little is known about the possible shapes of Weingarten surfaces for arbitrary curvature relations. This raises the hope that they define a sufficiently rich shape space for freeform design, while offering crucial benefits for architectural surface paneling.

Essentially, we are faced with the problem of *approximating a given freeform surface by a Weingarten surface*. Here, a general functional relation  $f(\kappa_1, \kappa_2) = 0$  is not given a priori, but discovered as the result of an optimization. In the present paper we show how to solve this challenging problem.

For a general surface, the curvature pairs  $(\kappa_1, \kappa_2)$  form a region in the plane that contracts to a curve  $\gamma$  for a W-surface (see Fig. 1). This contraction restricts the shape much less than the contraction of the Gauss image (formed by the unit normal vectors of a surface) to a curve. There, one obtains developable surfaces ( $K = 0$ ), which are of course also W-surfaces, but strongly shape restricted and immediately recognizable as such.

Making a surface more "Weingarten" through optimization raises numerous questions and opens up possibilities which go far beyond the motivating architectural application. Which shape effects can be seen in a surface when  $(\kappa_1, \kappa_2)$  form a curve  $\gamma$  instead of a 2D-region? Which connections between a surface  $S$  and its curvature diagram  $\gamma$  can one observe? How many W-surfaces, if any, are close to a given shape? If we perturb a W-surface, does optimization return to it or lead to another nearby W-surface? These are just a few of many questions, some of which are touched in the present

paper. They include deep mathematical problems on the shape space of  $W$ -surfaces which hopefully will receive interest in the future.

## 1.1 Contributions and Overview

In this paper, we study general Weingarten surfaces with a focus on applications in freeform paneling. Our main technical contribution is a guided projection algorithm to approximate a given freeform surface with a  $W$ -surface by aligning the isolines of  $H$ ,  $K$  or  $\kappa_1$ ,  $\kappa_2$ . In this way, a general functional relationship in Eq. 1 emerges automatically during the optimization. We found that fourth-order B-splines are well suited to formulate this optimization, since they simplify the evaluation of higher-order derivatives required to characterize  $W$ -surfaces to an unknown curvature relation. We show how to incorporate additional regularization to ensure that curvature isolines are close to geodesic parallels. This improves the performance of a subsequent curvature clustering algorithm that computes the mold shapes for paneling. Our experiments confirm that general Weingarten surfaces offer a rich space for freeform design, while significantly reducing paneling complexity.

After a brief discussion of related work, we first introduce the key geometric concepts required for our work (Sec. 2). We then present our main algorithm for approximating a given freeform surface by a Weingarten surface (Sec. 3). In Sec. 4 we present and discuss results. We conclude with a summary and an outlook of future research.

## 1.2 Related work

The literature on general Weingarten surfaces is not particularly rich, and explicit examples are known only for very simple curvature relations. An overview of results and references which are relevant in the present context can be found in [van Brunt and Grant 1996]: We mention that all ruled  $W$ -surfaces have been classified. It is also known that a channel  $W$ -surface (envelope of a one-parameter family of spheres with non-constant radius) is a rotational surface. S. Lie proved that the principal curvature lines on a  $W$ -surface can be computed by quadratures. Although [van Brunt and Grant 1996] has applications in the title, these concern just simplifications in curvature-based shape interrogation.

Umbilics (points with  $\kappa_1 = \kappa_2$ ) on  $W$ -surfaces have been studied by several authors, e.g. by H. Hopf [1951]. Analyzing umbilics, K. Voss [1959] proved that a closed analytic  $W$ -surface of sphere topology is a rotational surface. Since the result is no longer true if analyticity is dropped, it is not an essential shape restriction in the present setting.

Umbilics are the singularities in the network of principal curvature lines. [van Brunt and Grant 1994] showed that on those  $W$ -surfaces that are locally solutions of 2nd order hyperbolic PDEs, characterized by  $(\partial f / \partial \kappa_1)(\partial f / \partial \kappa_2) < 0$ , the umbilics are also singularities in the net of characteristics. If, in addition, there is a linear relation  $aH + bK = c$  between curvatures, the characteristics form a Chebyshev net.

These linear  $W$ -surfaces to  $aH + bK = c$  are still an active topic of research (see e.g. [Pámpano 2020]). A variational approach to their computation has recently been proposed by [Tellier 2020; Tellier et al. 2019], motivated by structural advantages in membrane, shell and gridshell design.

Linear  $W$ -surfaces contain the familiar special cases of constant mean curvature (CMC) surfaces ( $H = c$ ), in particular minimal surfaces ( $H = 0$ ), and surfaces of constant Gaussian curvature ( $K = c$ ) including developable surfaces ( $K = 0$ ). It would lead too far to discuss the rich literature on these surfaces. From a computational viewpoint, contributions from discrete differential geometry are probably the most interesting ones. For the discrete theory based on quad meshes we refer to [Bobenko and Suris 2008]. An efficient computation of CMC surfaces using triangle meshes has been proposed by [Pan et al. 2012]. Let us also mention the pioneering paper [Pinkall and Polthier 1993] on discrete minimal surfaces and the software package "Surface Evolver" by K. Brakke [1992]. Typically, triangle mesh approaches are well suited when boundaries are prescribed, while the quad mesh approaches are preferred when the combinatorics of special parameterizations is given. For example, [Bobenko et al. 2006] determine discrete minimal surfaces from the combinatorics of the principal curvature net. Since we use B-spline surfaces to approximate  $W$ -surfaces, we also point to a few contributions on spline representations of minimal or CMC surfaces. [Xu and Wang 2010; Xu et al. 2015] investigate polynomial minimal surfaces, which constitute a small class of surfaces within all minimal surfaces. In [Monterde 2004; Pan and Xu 2011], B-spline surfaces or subdivision surfaces are used for approximation of minimal or CMC surfaces.

There is another type of "linear"  $W$ -surfaces, namely to a relation  $a\kappa_1 + b\kappa_2 = c$ . They include CMC surfaces ( $a = b$ ), and thus the main interest is on the case  $a \neq b$ . On surfaces with a constant ratio of principal curvatures, i.e.,  $\kappa_1 - b\kappa_2 = 0$  all curvature elements (osculating paraboloids) are scaled versions of each other. For  $b < 0$ , the surfaces are negatively curved and their asymptotic directions form a constant angle. This property is useful for realizations as asymptotic gridshells [Schling 2018], which possess a curved support structure consisting of bent rectangular strips that are orthogonal to the surface. The structure follows the network of asymptotic curves (see [Jimenez et al. 2020], [Schling et al. 2018]). Surfaces of type  $a\kappa_1 + b\kappa_2 = c$  have recently been derived via special principal symmetric meshes [Pellis et al. 2020b], demonstrating also the advantages in paneling.

Paneling is an important topic where Geometry and Geometry Processing can contribute to architecture. Without any further assumptions on shape and fabrication technology, the algorithm of choice is [Eigensatz et al. 2010]. This post-rationalization method is computationally very involved, however, making it less suitable for design exploration. Our approach aims at fabrication-aware design, where a suitable shape space ( $W$ -surfaces) is explored to simplify down-stream rationalization tasks and avoid costly design iterations.

Very recently, [Gavriil et al. 2020] developed a design system for glass facades where cold bent glass panels are used wherever possible. The judgement whether a panel can be produced via cold bending comes from a neural network that has been trained on a large number of simulation results. Some results from discrete differential geometry offer possibilities for paneling surfaces smoothly with simple patches: Based on a quad patch layout along principal curvature lines, one can use Dupin cyclide patches [Bo et al. 2011; Bobenko and Huhnen-Venedey 2012]. If the panel boundary

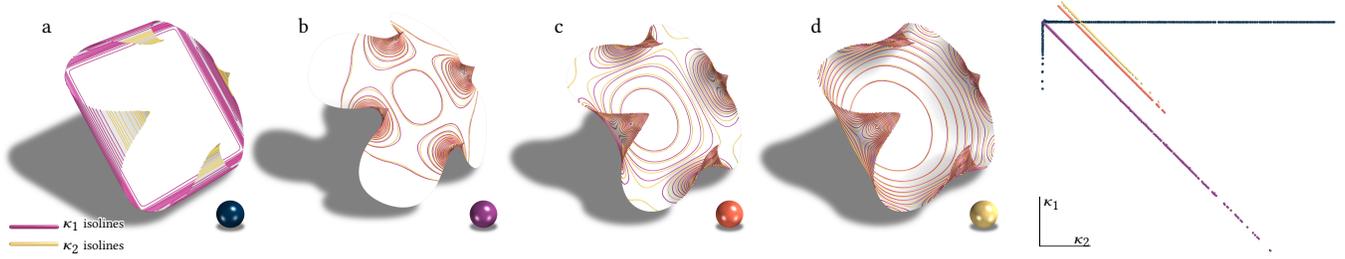


Fig. 3. Linear Weingarten surfaces designed with our method. From left to right: A developable surface, a minimal surface, two CMC surfaces of increasing mean curvature. The surfaces are obtained from the same B-spline surface where we use a high penalty term to prevent the boundary curve from moving. The corresponding curvature diagrams in the  $(\kappa_1, \kappa_2)$ -plane are shown on the right.

layout follows asymptotic curves on a negatively curved surface, ruled quadric panels are sufficient to smoothly cover the surface [Huhnen-Venedey and Rörig 2014]. This does not reduce the number of molds, but depending on the material to be used, may yield a cost reduction in the fabrication of panels.

[Eigensatz et al. 2008] presented a mesh-based algorithm for curvature-domain shape processing. Their method allows explicitly prescribing principal curvature values on a surface and solves for the corresponding vertex positions using a non-linear least-squares optimization. While effective for certain shape filtering tasks such as smoothing or feature enhancement, this approach is not sufficiently robust to optimize for general Weingarten surfaces. In our approach, we do not prescribe curvature directly, but implicitly optimize for an unknown functional curvature relation. In general we found that a mesh-based approach is less suited to our problem, since we are dealing with surface derivatives of up to fourth order. This motivates our preference for higher order B-splines over polygonal surface representations.

## 2 WEINGARTEN SURFACES

In this section we collect the main concepts needed in our implementation. As we use B-spline surfaces in our algorithm, we can base everything on the standard parametric representation  $\mathbf{s}(u, v)$  of a surface.

Writing partial derivatives as lower indices preceded by a comma, ( $\mathbf{s}_{,u} := \partial\mathbf{s}/\partial u, \dots$ ), the coefficients  $E, F, G$  of the first fundamental form  $Edu^2 + 2Fdu dv + Gdv^2$  are defined as

$$E = \langle \mathbf{s}_{,u}, \mathbf{s}_{,u} \rangle, \quad F = \langle \mathbf{s}_{,u}, \mathbf{s}_{,v} \rangle, \quad G = \langle \mathbf{s}_{,v}, \mathbf{s}_{,v} \rangle. \quad (2)$$

Using the determinant  $\Delta = EG - F^2$ , and the (non normalized) normal vector field  $\mathbf{n}(u, v) = \mathbf{s}_{,u} \times \mathbf{s}_{,v}$ , the coefficients  $L, M, N$  of the second fundamental form are

$$L = \frac{\langle \mathbf{s}_{,uu}, \mathbf{n} \rangle}{\sqrt{\Delta}}, \quad M = \frac{\langle \mathbf{s}_{,uv}, \mathbf{n} \rangle}{\sqrt{\Delta}}, \quad N = \frac{\langle \mathbf{s}_{,vv}, \mathbf{n} \rangle}{\sqrt{\Delta}}. \quad (3)$$

This yields Gaussian curvature  $K$  and mean curvature  $H$ ,

$$K = \frac{LN - M^2}{\Delta}, \quad H = \frac{LG - 2MF + NE}{2\Delta}. \quad (4)$$

For a  $W$ -surface, since principal curvatures are in functional relation, the constancy of one curvature entails the constancy of the

other one<sup>2</sup>. This implies that an isoline (level set) of the function  $\kappa_1(u, v)$  is also an isoline of  $\kappa_2(u, v)$ ,  $H(u, v)$  and  $K(u, v)$ . Hence, a  $W$ -surface has only one family of curvature isolines. This can be used to characterize a  $W$ -surface as one where the  $K$ -isolines and  $H$ -isolines are aligned. The preimages of these curves in the parameter domain are orthogonal to the gradients  $\nabla K = (K_{,u}, K_{,v})^T$  and  $\nabla H = (H_{,u}, H_{,v})^T$ , respectively. Thus, isoline alignment is expressed by linear dependence of these vectors, leading to the well-known characterization of  $W$ -surfaces via

$$H_{,u}K_{,v} - H_{,v}K_{,u} = 0. \quad (5)$$

It is important to note that this characterization also holds for the most widely known  $W$ -surfaces, namely those with constant  $H$  or  $K$ . There, the  $H$ - or  $K$ -isolines are not well defined, but (5) is satisfied due to vanishing  $\nabla H$  or  $\nabla K$ .

### 2.1 Geodesic parallel isolines

In order to control the spacing between curvature isolines, we need the intrinsic gradients  $\nabla_s H, \nabla_s K$ . Recall that the intrinsic gradient  $\nabla_s \psi$  of a function  $\psi(u, v)$  defined on a surface  $\mathbf{s}(u, v)$  is a tangent vector  $\lambda_u \mathbf{s}_{,u} + \lambda_v \mathbf{s}_{,v}$  that is orthogonal to the isoline and points in the direction of the steepest ascent, with its norm equal to that steepest ascent. Using the inverse  $\mathbb{I}^{-1}$  of the matrix  $\mathbb{I}$  of the first fundamental form,

$$\mathbb{I}^{-1} = \frac{1}{\Delta} \begin{pmatrix} G & -F \\ -F & E \end{pmatrix},$$

the intrinsic gradient  $\nabla_s \psi$  follows from  $\nabla \psi = (\psi_{,u}, \psi_{,v})^T$  via

$$\nabla_s \psi = \lambda_u \mathbf{s}_{,u} + \lambda_v \mathbf{s}_{,v} \quad \text{with} \quad (\lambda_u, \lambda_v)^T = \mathbb{I}^{-1} \nabla \psi, \quad (6)$$

and its squared norm is given by

$$\|\nabla_s \psi\|^2 = \nabla \psi^T \cdot \mathbb{I}^{-1} \cdot \nabla \psi. \quad (7)$$

For our application, it is advantageous if the curvature isolines do not diverge too much. Ideally, they should be geodesically parallel. In that case, the curves orthogonal to the isolines are geodesics. We base our work on the following lemma, whose proof is given in the Appendix.

<sup>2</sup>In regions where one of the two principal curvatures is constant and the other is not, the condition is entailed by the constancy of the latter only.

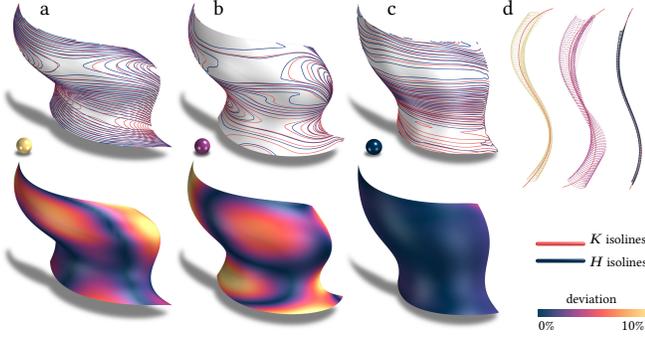


Fig. 4. Geodesic parallel curvature isolines. (a) A reference surface close to a helical surface. (b) The reference surface optimized for curvature isolines alignment. (c) The reference surface optimized also for geodesic parallel curvature isolines. On the *bottom*, surface colors display the deviation of the surface from a helical one. Deviation is expressed as percentage of the corresponding helical surface radius. (d) Surface points projected on a half-plane trough helical motion along the surface.

**LEMMA 2.1.** *The isolines of a function  $\psi(u, v)$  on a surface  $s$  are geodesic parallels (geodesic offsets of each other), if and only if  $\|\nabla_s \psi\|^2$  is constant along each isoline.*

A well-known special case is that of a distance function  $\psi$ , characterized by the eikonal equation  $\|\nabla_s \psi\|^2 = 1$ . However, being a distance function is not necessary for geodesic parallel isolines.

We will later regularize our optimization with a term expressing geodesic parallel curvature isolines. For that we have to align the isolines of the functions  $\|\nabla_s H\|^2$  or  $\|\nabla_s K\|^2$  with the curvature isolines, which can be expressed as

$$(\|\nabla_s H\|^2)_{,u} H_{,v} - (\|\nabla_s H\|^2)_{,v} H_{,u} = 0, \quad (8)$$

or analogously using  $K$ . Note that these conditions involve 4th order derivatives of  $s$ , which is a strong argument for avoiding meshes and working with splines of sufficiently high degree.

Kinematic surfaces, i.e. helical surfaces, rotational surfaces and general cylinder surfaces, are examples of W-surfaces with geodesic parallel curvature isolines (helices, circles, rulings, respectively). Our numerical experiments (see Figure 4 and Sec. 4) lead to the conjecture that these are the only examples. As this does not play a major role in the present paper, we leave a complete classification of W-surfaces with parallel curvature isolines for future research.

### 3 OPTIMIZATION ALGORITHM

To model W-surfaces, we start from a given generic input surface. We then optimize this surface by aligning the isolines of mean curvature  $H$  and Gaussian curvature  $K$  according to Eq. 5, while staying as close as possible to the input surface. We implement this optimization using the *guided projection* algorithm of [Tang et al. 2014]. This method is a Gauss-Newton variant that works best for constraints formulated as polynomials of degree two in the unknowns. Higher order constraints are reformulated as quadratic polynomials through the introduction of auxiliary variables and additional constraints. In the following, we describe this approach for the computation of W-surfaces.

### 3.1 Variables and constraints

**3.1.1 Surface representation.** The surface  $s(u, v)$  is represented as a tensor product B-spline

$$s(u, v) = \sum_i \sum_j N_i^{d_u}(u) N_j^{d_v}(v) c_{ij}, \quad (9)$$

where  $c_{ij}$  are the control points that shall form a regular quadrilateral mesh,  $N_i^d$  are the B-spline basis functions of degree  $d$ , and  $d_u, d_v$  are the degrees in  $u$  and  $v$  direction respectively. The main variables of the problem are then the positions  $c_{ij}$  of the control points. The target function and the auxiliary variables are estimated at sample points  $s(\bar{u}, \bar{v})$ , where  $\bar{u}$  and  $\bar{v}$  are obtained by homogeneous subdivision of the  $u$  and  $v$  domains respectively (see Figure 5). The bi-degree  $(d_u, d_v)$  of the B-spline surface and the density of the samples can be set by the user.

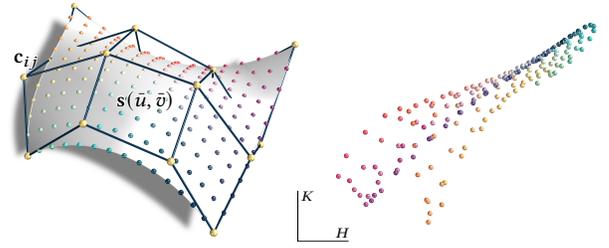


Fig. 5. Surface representation. On the *left*: a B-spline surface and its control mesh. In color, sampling points  $s(\bar{u}, \bar{v})$  are shown. On the *right*: Curvature diagram computed at sampling points, shown with the same colors.

**3.1.2 Auxiliary variables and constraints.** To express Eq. 5 as a quadratic constraint, we add the following auxiliary variables and constraints at each sample point  $s(\bar{u}, \bar{v})$ :

- The non unitized normal vector  $\mathbf{n} = \mathbf{s}_{,u} \times \mathbf{s}_{,v}$
- The coefficients of the first fundamental form  $E, F$  and  $G$ , defined by Eq. 2.
- The determinant of the first fundamental form, given by  $\Delta = EG - F^2$ .
- The coefficients of the second fundamental form  $L, M$  and  $N$ , defined by Eq. 3, where we use the value of  $\Delta$  taken from the previous iteration.
- The Gaussian curvature  $K$  and the mean curvature  $H$ , given by Eq. 4.

Note that the surface derivatives  $\mathbf{s}_{,u}, \mathbf{s}_{,v}, \mathbf{s}_{,uu}, \mathbf{s}_{,uv},$  and  $\mathbf{s}_{,vv}$  can be computed analytically from Eq. 9 and depend linearly on the control points  $c_{ij}$ . For details, see [Prautzsch et al. 2002].

**3.1.3 Curvature isolines alignment.** Eq. 5 can now be expressed as a quadratic function of the unknowns  $K$  and  $H$ . Partial derivatives of scalar functions  $\psi(\bar{u}, \bar{v})$  are computed through central differences:

$$\begin{aligned} \psi_{,u}(\bar{u}, \bar{v}) &= \frac{\psi(\bar{u}_+, \bar{v}) - \psi(\bar{u}_-, \bar{v})}{\bar{u}_+ - \bar{u}_-}, \\ \psi_{,v}(\bar{u}, \bar{v}) &= \frac{\psi(\bar{u}, \bar{v}_+) - \psi(\bar{u}, \bar{v}_-)}{\bar{v}_+ - \bar{v}_-}, \end{aligned} \quad (10)$$

where  $\bar{u}_-$  and  $\bar{u}_+$  denote the  $\bar{u}$  value of the previous and next sample point of  $\bar{u}$ , respectively.  $\bar{v}_-$  and  $\bar{v}_+$  are defined analogously.

**3.1.4 Shape preservation and closeness to the reference surface.** Let  $\mathbf{c}_{ij}^0$  be the control points of the initial reference geometry. The closeness to the reference surface is implemented by adding the soft constraint  $\mathbf{c}_{ij} - \mathbf{c}_{ij}^0 = \mathbf{0}$ . We also add a shape preservation term through the soft constraint  $l_i - l_i^0 = 0$ , where  $l_i$  are the lengths of edges and face diagonals of the control mesh, and  $l_i^0$  the corresponding lengths in the reference shape. To constrain the boundary, we let the boundary control points glide along a curve  $\beta$  that interpolates the boundary control points of the reference geometry. To this end, boundary control points are projected at each iteration onto the curve  $\beta$  and constrained to its tangent line.

## 3.2 Additional variables and constraints

**3.2.1 Geodesic parallel curvature isolines.** To optimize for geodesic parallel curvature isolines, we impose Eq. 8 for  $H$  and  $K$  in addition to Eq. 5. For that, we introduce the following additional auxiliary variables and constraints:

- The products of the derivatives of  $K$  and  $H$ ,  $K_{,u}^2$ ,  $K_{,u}K_{,v}$ ,  $K_{,v}^2$ , and  $H_{,u}^2$ ,  $H_{,u}H_{,v}$ ,  $H_{,v}^2$ .
- The norm of intrinsic gradients from Eq. 7:

$$\|\nabla_s K\|^2 = \frac{EK_{,u}^2 - 2FK_{,u}K_{,v} + GK_{,v}^2}{\sqrt{\Delta^k}},$$

$$\|\nabla_s H\|^2 = \frac{EH_{,u}^2 - 2FH_{,u}H_{,v} + GH_{,v}^2}{\sqrt{\Delta^k}}.$$

Here, derivatives of  $H$  and  $K$  are computed through central differences with Eq. 10, and  $\Delta^k$  is the first fundamental form determinant  $\Delta$  taken from the previous iteration  $k$ . With the new variables introduced, Eq. 8 can be formulated as a quadratic constraint by expressing again derivatives with Eq. 10.

**3.2.2 Linear Weingarten surfaces.** A relation of the form  $g(H, K) = 0$  as in Eq. 1 is discovered as a side effect during optimization. Since  $H$  and  $K$  are directly available as variables we can influence this relationship directly. In the case of linear Weingarten surfaces we add a constraint of the form  $aH + bK + c = 0$ . The coefficients  $a$ ,  $b$ , and  $c$  are computed at each iteration by linear regression to the values of  $H$  and  $K$  taken from the previous iteration. Alternatively, the coefficients can be set by the user. See Figures 3 and 2d for results.

**3.2.3 B-spline curvature relation.** We can also optimize a surface to have a  $(H, K)$ -curvature diagram that follows a B-spline curve  $\gamma$ , see Figure 2c. At each iteration, each  $(H, K)$ -point is constrained on the tangent line through the closest point of  $\gamma$ . The B-spline curve can be initialized and eventually updated at each iteration through minimization of the sum of squared distances to each  $(H, K)$ -point.

## 3.3 Numerical solver

Each constraint of Sec. 3.1 and 3.2 can be written as a quadratic function

$$\phi_i(x) = \frac{1}{2}x^T A_i x + b_i x + c_i = 0,$$

where  $x \in \mathbb{R}^{N \times 1}$  collects the unknowns, and  $A_i \in \mathbb{R}^{N \times N}$ ,  $b_i \in \mathbb{R}^{1 \times N}$  and  $c_i \in \mathbb{R}$  collect, respectively, the quadratic, linear and constant coefficients. At each iteration  $k+1$ , with current unknowns vector  $x^k$ , the constraints are approximated with a first order Taylor expansion:

$$\phi_i(x) \approx \phi_i(x^k) + \nabla \phi_i(x^k)(x - x^k),$$

where  $\nabla \phi_i(x^k) = A_i x^k + b_i$ . The linearized constraints are multiplied by weights  $\omega_i$  and then rearranged as a system of linear equations  $Cx = r$ . To stabilize the solution, we add a fairness energy  $E_f$  to the control mesh polylines that at each control point  $\mathbf{c}_{ij}$  is given by

$$E_f(\mathbf{c}_{ij}) = \left\| \mathbf{c}_{ij} - \frac{\mathbf{c}_{ij-1} + \mathbf{c}_{ij+1}}{2} \right\|^2 + \left\| \mathbf{c}_{ij} - \frac{\mathbf{c}_{i-1j} + \mathbf{c}_{i+1j}}{2} \right\|^2.$$

The total energy can be written in matrix form as  $\|Px\|^2$ . The linear system is then solved in least squares, where the fairness energy, together with a closeness term to the current variables  $x^k$ , act as a regularizer:

$$\|Cx - r\|^2 + \omega_f^2 \|Px\|^2 + \varepsilon^2 \|x - x^k\|^2 \rightarrow \min. \quad (11)$$

Here  $\omega_f$  and  $\varepsilon$  are weights set by the user. The resulting linear system is solved by Cholesky factorization.

## 4 RESULTS AND APPLICATIONS

### 4.1 Design of W-surfaces

Our main goal is to provide an algorithm that makes Weingarten surfaces accessible for shape modeling and shape processing. In the following, we evaluate our algorithm and its performance on some common design tasks. Run times and computation settings for the presented examples are listed in Tables 1 and 2.

**4.1.1 Workflow.** Our design workflow of W-surfaces starts from a given B-spline surface. The starting B-spline can act as a reference geometry or serve just as a starting point for shape design. In the former case, during the optimization the surface is kept close to the reference shape by enabling the constraints of Sec. 3.1.4. In the latter case, the user can edit the B-spline surface by moving its control points; the edited surface is then used as a target geometry

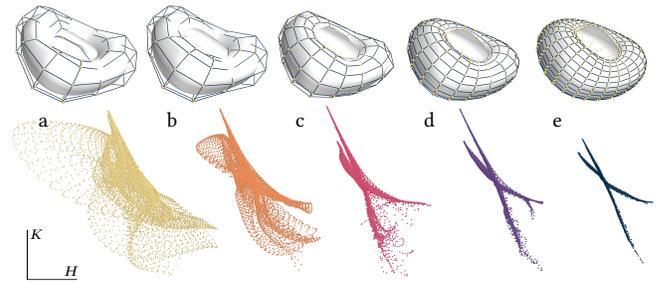


Fig. 6. Multiresolution modeling. A given B-spline surface shown in (a) is optimized towards a Weingarten surface through an iterative procedure. At each step, the control mesh is refined and the surface is re-optimized for alignment of curvature isolines. From (b) to (e), four steps of this procedure with corresponding curvature diagrams are shown. We observe that, as we increase the number of control points, the optimized B-spline surface converges to a Weingarten surface.

for some iterations of our optimization. This process can be applied repeatedly as an interactive design tool. For a broad preservation of the input shape during the modeling session, the shape preservation constraints of Sec. 3.1.4 can be enabled with a low weight.

To achieve a  $W$ -surface that is close to some reference geometry, it is necessary to work with B-splines that have a sufficiently dense control mesh, since the number of control points corresponds to the degrees of freedom available for optimization. For this purpose, we found it advantageous to use a multiresolution approach where, at each step, the control mesh is refined and the surface is optimized again until convergence. This procedure is illustrated in Figure 6. In our implementation, the refinement of the control mesh is performed by computing a new B-spline surface with the desired number of control points through least squares fitting. For shape design, it is preferable to work with a B-spline with a coarse control mesh until a desired shape is approximately reached. One can then proceed with a multiresolution refinement. Sample shapes designed with this procedure are shown in Figures 10 and 11.

**4.1.2 Shape approximation.** For effective shape approximation with Weingarten surfaces we expect the space of Weingarten surfaces to be dense, i.e., for a given target shape there should be a Weingarten surface sufficiently close to it. We do not have a mathematical proof of this, but our examples provide strong numerical evidence that the shape space of Weingarten surfaces is dense enough as we generally observe rather small shape changes during the alignment of curvature isolines, as shown in Figures 1, 8, 9, and 13. Figure 2 shows the approximation of the target shape also with  $W$ -surfaces to a B-spline and a linear curvature relation, through constraints described in Sec. 3.2.3 and 3.2.2. We observe that we can achieve a better approximation of the target shape by optimizing for the alignment of curvature isolines alone, that is without constraining the curvature function  $g(H, K)$ . In Figure 7, we explore the neighborhood of a  $W$ -surface through perturbation and re-optimization. Results suggest the existence of a dense neighborhood of  $W$ -surfaces, but we leave further investigation as future research.

**4.1.3 Fairing curvature isolines.** In our procedure, we can also intervene on the layout of curvature isolines. In fact, a smoother and fair layout can be beneficial in applications where curvature isolines are used to subdivide the surface into regions, as for instance in architectural paneling.

Since curvature functions involve second order derivatives on the surface, smooth curvature isolines can be achieved only for surfaces with a sufficiently high degree of continuity. This property can be easily fulfilled with B-spline surfaces. Figure 8 shows a comparison of curvature isolines alignment using a B-spline surface of bi-degree (4, 4) against a standard cubic B-spline.

Furthermore, to obtain more evenly spaced isolines, we can ask for geodesic parallel curvature isolines as described in Sec. 3.2.1. Obtaining truly geodesic parallel curvature isolines is not possible for a general Weingarten surface, however we can use this constraint as a fairness term during the optimization. The effect of this fairness on the isolines layout is shown in Figure 9.

Figure 4 shows the effect of geodesic parallel isolines as a hard constraint. In this test, we start from a B-spline close to a helical surface. After optimization for aligned and geodesic parallel isolines,

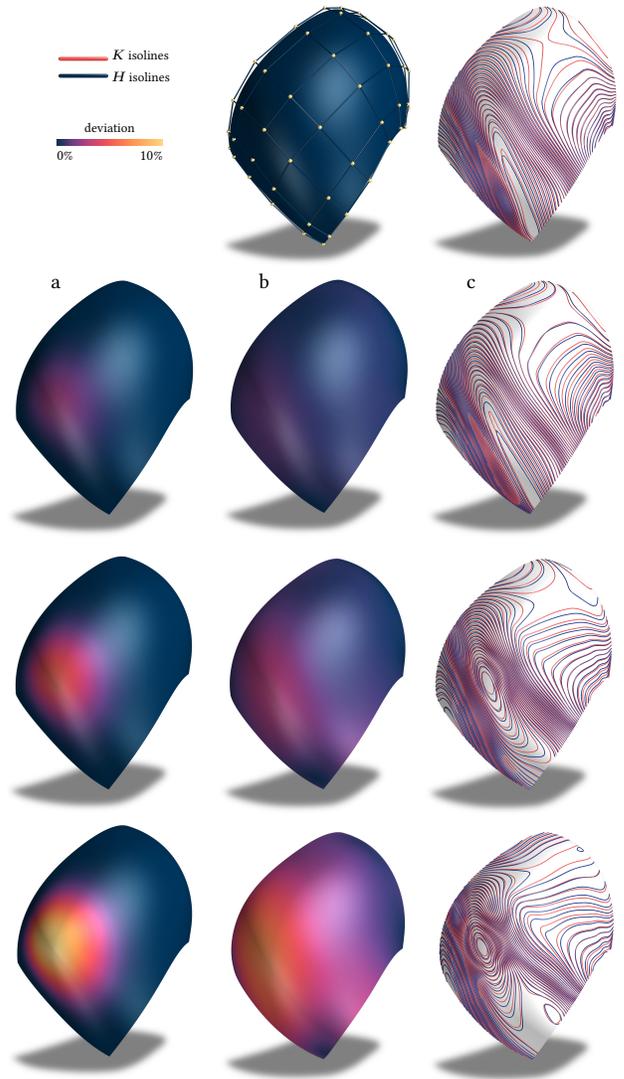


Fig. 7. Shape exploration. On top: A reference  $W$ -surface designed with our method with corresponding curvature isolines shown on the right. (a) The reference surface is perturbed by moving a control point. From top to bottom, increasing perturbations are shown. (b, c) The perturbed shape on the left is optimized for alignment of curvature isolines. Surface colors indicate the deviation from the reference shape, expressed as percentage of its bounding box diagonal. Results suggest the existence of a rich space of  $W$ -surfaces.

we observe the shape converging to a helical surface. If we optimize the same shape only for curvature isolines alignment, this effect does not occur. These results suggest that  $W$ -surfaces with geodesic parallel curvature isolines are kinematic surfaces.

## 4.2 Paneling architectural surfaces

Weingarten surfaces maximize repetitivity of curvature elements, i.e., the osculating paraboloid at one location is congruent to the osculating paraboloids at many other points on the design surface.

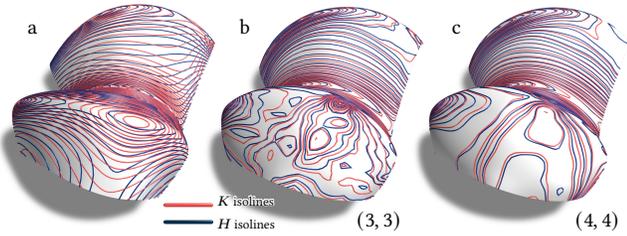


Fig. 8. Influence of B-spline degree. The reference shape shown in (a) is optimized for curvature isolines alignment with a B-spline surface of bi-degree (3, 3), and with one of bi-degree (4, 4). On surfaces of higher degree, we achieve smoother curvature isolines.

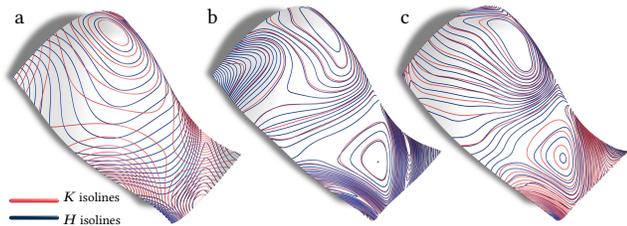


Fig. 9. Geodesic parallel curvature isolines as fairness energy. (a) Reference surface. (b) The reference surface optimized for alignment of curvature isolines. (c) The reference surface optimized for alignment of curvature isolines with geodesic parallel curvature isolines constraints used as a fairness energy. We can observe that curvature isolines are more regularly spaced. This property can be advantageous for architectural panelization.

This indicates that congruent panels can be used in those regions, all of which can be manufactured with the same mold or tool configuration. Such regions form clusters in the curvature diagram.

The widely adopted post-rationalization design loop that tries to maximize repetitivity on a generic surface involves costly and time consuming iterations between designers and engineers [Eigensatz et al. 2010]. Applying our method in an architectural context can be seen as pre-rationalization, i.e., the surfaces designed with our algorithm have repetitivity already ‘baked in’ as an essential property.

The main ingredients for paneling are a subdivision of the design surface into segments by a network of seam curves and an assignment of molds to faces of the seam network. Each face of the seam network corresponds to a panel. We generate the layout of seams by overlaying the parameter domain of the designed B-spline surface with a quad mesh (or any other desired pattern) and then map this mesh to the surface.

Theoretically, one should be able to approximate a Weingarten surface that has been subdivided into  $N$  panels reasonably well by  $\sqrt{N}$  unique molds if the panel dimensions are not too large. In practice, panel dimension are often fixed and cannot be freely chosen. To account for this we allow more clusters; specifically we used  $k\sqrt{N}$  clusters with  $k = 2, 3$ .

Clustering is performed in the  $(H, \sqrt{K})$ -plane instead of  $(H, K)$  to obtain the same scale on each axis. To this end, we overlay the plane with a uniform grid obtained by homogeneous subdivision of

each axis. We increase iteratively the density of the grid until we reach a subdivision such that all  $(H, \sqrt{K})$  pairs fall into a prescribed numbers of cells. The pairs inside such cells define a cluster whose panels are assigned the same mold.

Figures 1, 11, and 13 show results obtained in this way. Figure 13 shows an evaluation of reflection lines, gaps and kink angles, as well as a comparison with [Eigensatz et al. 2010]. Their method aims to minimize manufacturing cost, while we aim to minimize the number of molds. To enable a comparison, we restrict the output of [Eigensatz et al. 2010] to cubic panels which effectively turns the cost-based formulation into a formulation based on the number of molds, since panels of the same type amount to the same cost. In all examples we used cubics instead of paraboloids for practical reasons; paraboloids often require rather small panel sizes to achieve the desired thresholds on gaps and kink angles.

### 4.3 Limitations and future work

As Figure 12 illustrates, our algorithm is not always successful in aligning the curvature isolines. Here the optimization is stuck in a local minimum before sufficiently contracting the curvature diagram. However, we observe that making a surface “more Weingarten”, even if not achieving a clear curvature relation, improves the performance of subsequent paneling algorithms. The computed local minima depend on the weights chosen in Eq. 11, which we currently choose through experimentation. Further investigations are needed to better understand the dependency of the result on these parameters.

The curvature relation of a Weingarten surface has a global influence on the shape. While we found that reasonably complex freeform shapes can often be well approximated with a W-surface,

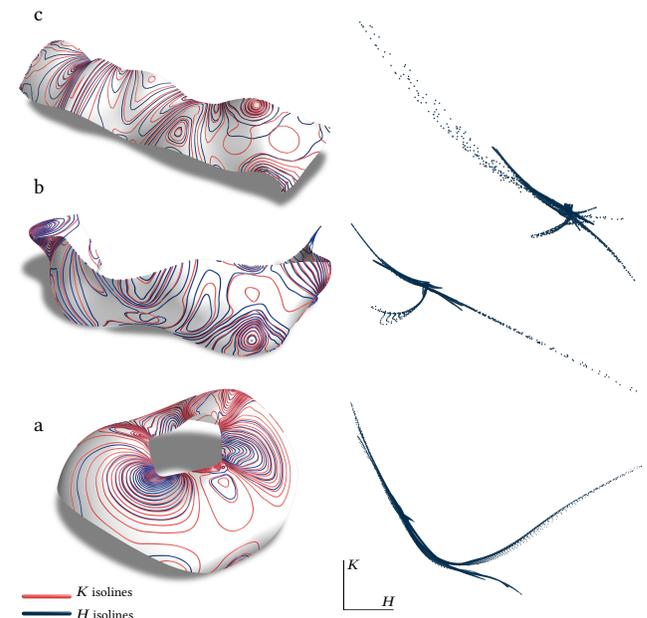


Fig. 10. W-surfaces designed with our method (left). On the right, the corresponding curvature diagram is shown.

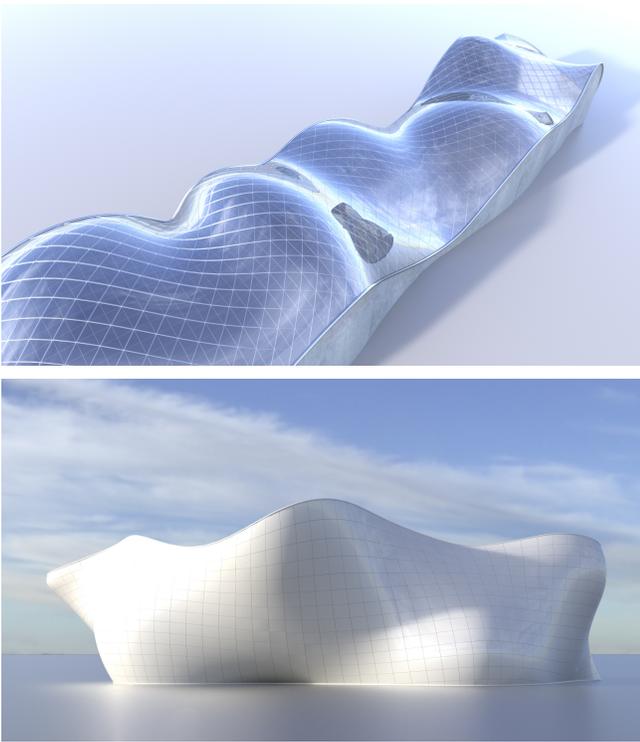


Fig. 11. Architectural panelizations of the shapes shown in Figure 10a and b (cf. Sec. 4.2). We used  $3\sqrt{N}$  molds to generate all  $N$  panels. As we can observe, the cladded surfaces show good visual smoothness.

other shapes like a model of a person or the Stanford bunny are not suitable for approximation with a single Weingarten surface. To broaden applicability, future work could investigate how to decompose more complex surfaces into smaller components for which an approximation with a W-surface is adequate.

The curve-like curvature diagrams that we obtain through optimization come in many different forms. Beyond linear W-surfaces which are well understood, it is unclear what other functional relations can be linked to which geometric properties of the surface.

More broadly, we do not yet have a good understanding of the shape space of W-surfaces. For example, it is unclear if there exists a continuous path between two W-surfaces in the space of W-surfaces. This poses interesting questions for future research, both in terms of new mathematical theory and for the design of practical geometry processing tools.

A related topic is direct manipulation of the functional curvature relation. For example, if we represent the curvature relation with a B-spline curve as in Figure 2, we could edit the corresponding control polygon and continuously update the underlying W-surface. It will be interesting to study whether this form of indirect modeling facilitates intuitive shape exploration.

## 5 CONCLUSIONS

General Weingarten surfaces offer a new means for freeform shape design. We show that for a large variety of input designs, we can find

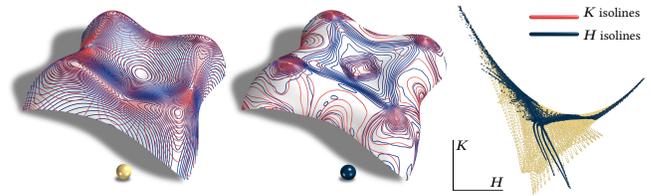


Fig. 12. Limitations. The surface on the *left* is optimized for curvature isolines alignment (*center*). In this case, further optimization steps do not improve the result. The curvature diagrams before and after optimization are shown on the *right*.

a close-by Weingarten surface through optimization. These surfaces lead to significant improvements for subsequent panelization.

Optimizing in curvature space is generally challenging, as the required higher-order derivatives are numerically very sensitive. Consequently, our initial attempts failed to formulate such an optimization using polygonal meshes. We found, however, that higher-order B-splines in combination with a guided projection approach are ideally suited to address this challenging problem. The main advantages of B-splines over meshes are intrinsic smoothness, low number of parameters, and analytical higher-order derivatives. Guided projection allows re-formulating the optimization objective as a quadratic function using a suitable defined cascading set of auxiliary variables. The resulting algorithm makes the general class of Weingarten surfaces accessible for shape exploration, which will hopefully trigger further investigations and lead to new theoretical and practical advances.

## ACKNOWLEDGMENTS

This work was supported by the Swiss National Science Foundation through NCCR "Digital Fabrication" (agreement No. 51NF40-141853), and by the Austrian Science Fund FWF through SFB "Advanced Computational Design" (project No. F77).

Fig.	$\omega_{al}$	$\omega_{bs}$	$\omega_{ln}$	$\omega_{gp}$	$\omega_{cl}^*$	$\omega_{sh}^*$
1,7,8,12,13	0.1	0	0	0	0.1	0
2b/c/d	0.1/0/0	0/10/0	0/0/1	0	0.1	0
3	0	0	1	0	0.001	0.001
4b/c	0.1	0	0	0/0.2	0.001	0.01
9b/c	0.1	0	0	0/0.05	0.1	0
10	0.1	0	0	0.01	0.05	0.001

Table 1. Weights  $\omega$  of target functions used for the optimization of the presented examples. From the *left*:  $\omega_{al}$  = isolines alignment of Eq. 5,  $\omega_{ln}$  = linear curvature relation of Sec. 3.2.2,  $\omega_{bs}$  = B-spline curvature relation of Sec. 3.2.3,  $\omega_{gp}$  = geodesic parallel isolines of Eq. 8,  $\omega_{cl}$  = closeness to reference control points of Sec. 3.1.4,  $\omega_{sh}$  = shape preservation of Sec. 3.1.4. Weights marked with an asterisk can slightly change during an interactive procedure and can be used as design parameters. As for the regularization terms of Eq. 11, we set the weight of the fairness energy  $\omega_f$  to 0.2 and the weight of closeness to the previous iteration  $\epsilon$  to 0.001 for all examples. Before optimization, the input B-splines are scaled so that the average of the absolute principal curvatures at sample points is equal to 5.

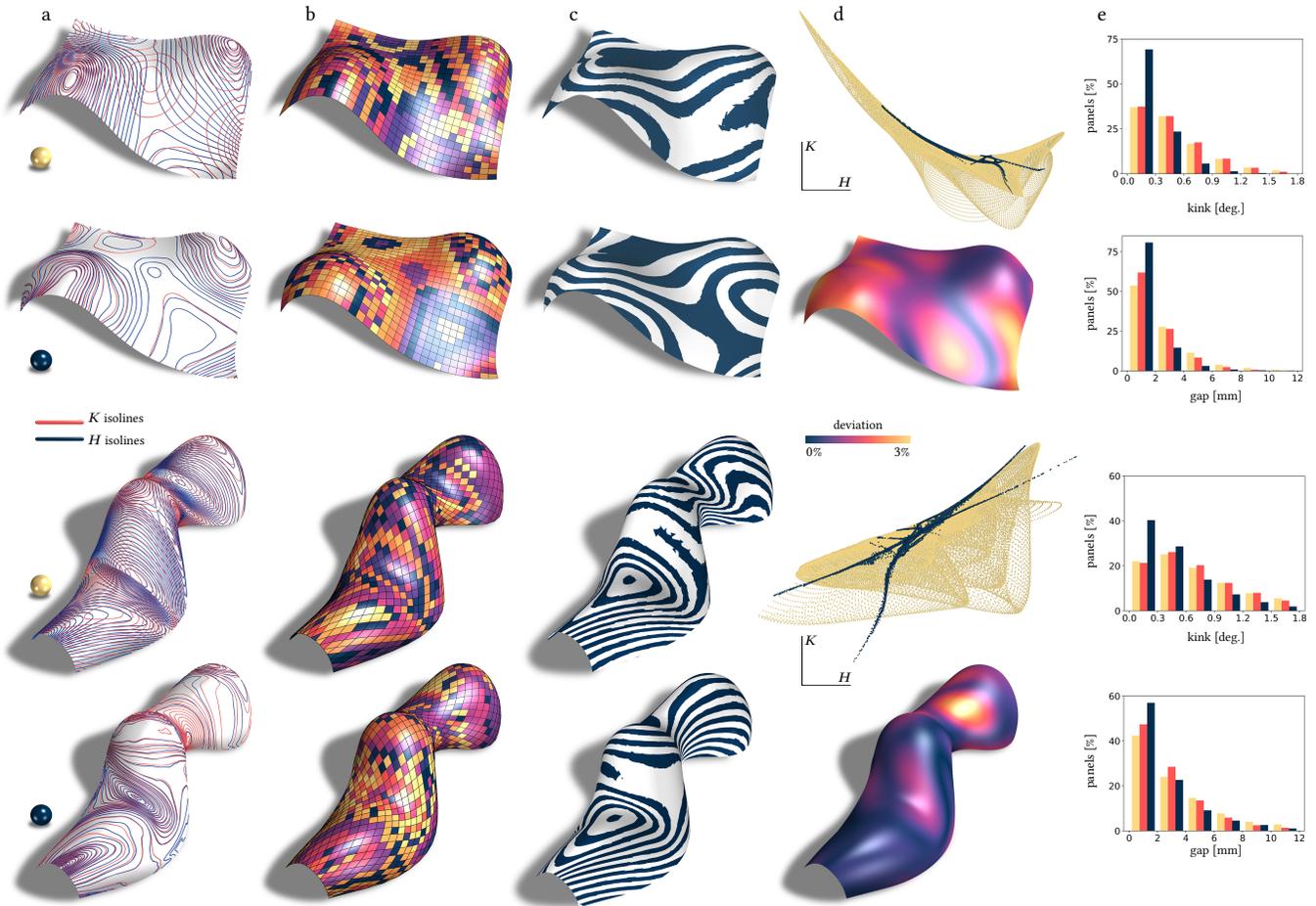


Fig. 13. Architectural paneling with Weingarten surfaces. (a) The surface on *top* is optimized towards a Weingarten surface. The resulting shape, with its curvature isolines, is shown *below*. (b) We subdivide both surfaces into panels and group them into the same number of clusters, as described in Sec 4.2. Panels belonging to the same cluster are shown with the same color. (c) To each cluster, we fit panels belonging to the same cubic surface, while minimizing kinks and gaps along the seams between panels. A zebra striping of the panels is shown where smoother strips indicate a better surface continuity. (d) On *top*, the curvature diagrams of the starting shape and the optimized one. On the *bottom*, colors show the deviation of the optimized surface from the starting one, expressed as percentage of the diagonal of the bounding box. (e) The histograms show the distribution of kinks and gaps between panels. In red, we also show the results obtained with [Eigensatz et al. 2010] on the non-optimized surface, using the same number of clusters. Histograms shifted to the left indicate better continuity of the cladding. These results show that, with our optimization, with a rather small shape change, we can achieve smoother panelizations with the same number of molds.

## REFERENCES

- Pengbo Bo, Helmut Pottmann, Martin Kilian, Wenping Wang, and Johannes Wallner. 2011. Circular Arc Structures. *ACM Trans. Graph.* 30, 4, Article 101 (July 2011), 12 pages. <https://doi.org/10.1145/2010324.1964996>
- Alexander Bobenko, Tim Hoffmann, and Boris Springborn. 2006. Minimal surfaces from circle patterns: Geometry from combinatorics. *Ann. Math.* 164 (2006), 231–264.
- Alexander I. Bobenko and Emanuel Huhnen-Venedey. 2012. Curvature line parametrized surfaces and orthogonal coordinate systems: discretization with Dupin cyclides. *Geom. Dedicata* 159 (2012), 207–237. <https://doi.org/10.1007/s10711-011-9653-5>
- Alexander I. Bobenko and Yuri B. Suris. 2008. *Discrete differential geometry. Integrable structure*. Graduate Studies in Mathematics, Vol. 98. American Mathematical Society, Providence, RI.
- Kenneth A. Brakke. 1992. The Surface Evolver. *Experimental Mathematics* 1, 2 (1992), 141–165. <https://doi.org/10.1080/10586458.1992.10504253>
- Michael Eigensatz, Martin Kilian, Alexander Schiftner, Niloy J. Mitra, Helmut Pottmann, and Mark Pauly. 2010. Paneling Architectural Freeform Surfaces. *ACM Trans. Graph.* 29, 4 (2010), 45:1–45:10.
- Michael Eigensatz, Robert W. Sumner, and Mark Pauly. 2008. Curvature-Domain Shape Processing. *Computer Graphics Forum* 27, 2 (2008), 241–250. <https://doi.org/10.1111/j.1467-8659.2008.01121.x>
- Konstantinos Gavriil, Ruslan Guseinov, Jesus Perez, Davide Pellis, Paul Henderson, Florian Rist, Helmut Pottmann, and Bernd Bickel. 2020. Computational design of cold bent glass facades. *ACM Trans. Graphics* 39, 6 (2020), 208:1–208:16. Proc. SIGGRAPH Asia.
- Heinz Hopf. 1951. Über Flächen mit einer Relation zwischen den Hauptkrümmungen. *Math. Nachr.* 4 (1951), 232–249.
- Emanuel Huhnen-Venedey and Thilo Rörig. 2014. Discretization of asymptotic line parametrizations using hyperboloid surface patches. *Geom. Dedicata* 168, 1 (2014), 265–289. <https://doi.org/10.1007/s10711-013-9830-9>

- Michael R. Jimenez, Christian Müller, and Helmut Pottmann. 2020. Discretizations of Surfaces with Constant Ratio of Principal Curvatures. *Discrete Comput. Geom.* 63, 3 (2020), 670–704.
- Wolfgang Kühnel. 2003. *Differentialgeometrie* (second ed.). Friedr. Vieweg & Sohn, Braunschweig, viii+256 pages. Kurven—Flächen—Mannigfaltigkeiten.
- Juan Monterde. 2004. Bézier surfaces of minimal area: The Dirichlet approach. *Computer Aided Geometric Design* 21, 2 (2004), 117 – 136.
- Alvaro Pámpano. 2020. A variational characterization of profile curves of invariant linear Weingarten surfaces. *Differential Geom. Appl.* 68 (2020), 101564, 27.
- Hao Pan, Yi-King Choi, Yang Liu, Wenchao Hu, Qiang Du, Konrad Polthier, Caiming Zhang, and Wenping Wang. 2012. Robust Modeling of Constant Mean Curvature Surfaces. *ACM Trans. Graph.* 31, 4, Article 85 (2012), 11 pages. <https://doi.org/10.1145/2185520.2185581>
- Qing Pan and Guoliang Xu. 2011. Construction of minimal subdivision surface with a given boundary. *Computer-Aided Design* 43, 4 (2011), 374 – 380.
- Davide Pellis, Martin Kilian, Hui Wang, Caigui Jiang, Christian Müller, and Helmut Pottmann. 2020a. Architectural freeform surfaces designed for cost-effective paneling through mold re-use. In *Advances in Architectural Geometry*.
- Davide Pellis, Hui Wang, Florian Rist, Martin Kilian, Helmut Pottmann, and Christian Müller. 2020b. Principal Symmetric Meshes. *ACM Trans. Graphics* 39, 4 (2020), 127:1–127:17.
- Ulrich Pinkall and Konrad Polthier. 1993. Computing discrete minimal surfaces and their conjugates. *Experiment. Math.* 2, 1 (1993), 15–36. <https://projecteuclid.org/443/euclid.em/1062620735>
- Helmut Pottmann, Michael Eigensatz, Amir Vaxman, and Johannes Wallner. 2015. Architectural Geometry. *Computers and Graphics* 47 (2015), 145–164.
- Hartmut Prautzsch, Wolfgang Boehm, and Marco Paluszny. 2002. *Bézier and B-Spline Techniques*. Springer-Verlag.
- Eike Schling. 2018. *Repetitive Structures*. Ph.D. Dissertation. TU Munich.
- Eike Schling, Martin Kilian, Hui Wang, Denis Schikore, and Helmut Pottmann. 2018. Design and construction of curved support structures with repetitive parameters. In

- Adv. in Architectural Geometry*, Lars Hesselgren et al. (Ed.), Klein Publ. Ltd, 140–165.
- Chengcheng Tang, Xiang Sun, Alexandra Gomes, Johannes Wallner, and Helmut Pottmann. 2014. Form-finding with Polyhedral Meshes Made Simple. *ACM Trans. Graph.* 33, 4 (2014), 70:1–70:9.
- Xavier Tellier. 2020. *Morphogenesis of curved structural envelopes under fabrication constraints*. Ph.D. Dissertation. Univ. Paris-Est.
- Xavier Tellier, Cyril Douthe, Laurent Hauswirth, and Olivier Baverel. 2019. Linear Weingarten surfaces for conceptual design of double-curvature envelopes. In *Proceedings Int. Symposium on Conceptual Design of Structures, Madrid*.
- Bruce van Brunt and Katina Grant. 1994. Hyperbolic Weingarten Surfaces. *Math. Proc. Camb. Phil. Soc.* 116 (1994), 489–504.
- Bruce van Brunt and Katina Grant. 1996. Potential applications of Weingarten surfaces in CAGD, Part I: Weingarten surfaces and surface shape investigation. *Computer Aided Geometric Design* 13, 6 (1996), 569 – 582.
- Konrad Voss. 1959. Über geschlossene Weingartensche Flächen. *Math. Annalen* 138 (1959), 42–54.
- Julius Weingarten. 1861. Über eine Klasse aufeinander abwickelbarer Flächen. *J. reine u. angewandte Mathematik* 59 (1861), 382–393.
- Gang Xu and Guozhao Wang. 2010. Quintic parametric polynomial minimal surfaces and their properties. *Differential Geometry and its Applications* 28, 6 (2010), 697 – 704.
- Gang Xu, Yaguang Zhu, Guozhao Wang, Andre Galligo, Li Zhang, and Kin chuen Hui. 2015. Explicit form of parametric polynomial minimal surfaces with arbitrary degree. *Appl. Math. Comput.* 259 (2015), 124 – 131.

## A APPENDIX

*Proof of Lemma 2.1:* Since intrinsic gradients are independent of the parameterization, we can adapt the parameterization  $\mathbf{s}(u, v)$  to the isolines of the given function. We assume that the isolines of  $\psi$  are the  $v$ -lines on  $\mathbf{s}$  and that the  $u$ -lines are orthogonal to them. Hence, we have  $F = 0$  and by equations (6) and (7),

$$\nabla_s \psi = \frac{\psi_{,u}}{E} \mathbf{s}_u + \frac{\psi_{,v}}{G} \mathbf{s}_v, \quad \|\nabla_s \psi\|^2 = \frac{1}{E} \psi_{,u}^2 + \frac{1}{G} \psi_{,v}^2.$$

Alignment of the  $\psi$ -isolines with  $v$ -lines is possible exactly if  $\nabla_s \psi$  is tangent to the  $u$ -lines. This requires  $\psi_{,v} = 0$  and thus  $\psi$  is just a function of  $u$ , and

$$\|\nabla_s \psi\|^2 = \frac{1}{E} \psi_{,u}^2.$$

Therefore,  $\|\nabla_s \psi\|^2$  is constant along the isolines, i.e.  $\|\nabla_s \psi\|^2$  depends on  $u$  only if and only if  $E$  depends just on  $u$ . This is precisely the condition for the parameterization  $\mathbf{s}(u, v)$  to represent geodesic parallel coordinates [Kühnel 2003, p. 110]. There, the  $v$ -lines form a family of geodesic parallel curves and the  $u$ -lines are geodesics.

□

Fig.	ctrl. points	bi-degree	sampling	# var.	t./iter. [s]
1	(14, 14)	(4, 4)	(28, 28)	9996	3.32
2	(18, 18)	(4, 4)	(36, 36)	16525	5.71
3	(25, 25)	(4, 4)	(50, 50)	31875	52.04
4b	(20, 18)	(4, 4)	(40, 36)	18360	10.89
4c	(16, 10)	(5, 5)	(32, 20)	13280	1.71
6e	(25, 18)	(4, 4)	(50, 36)	26406	14.15
6b	(10, 8)	(4, 4)	(20, 16)	5616	0.32
6c	(14, 10)	(4, 4)	(28, 20)	9060	0.83
6d	(18, 14)	(4, 4)	(36, 28)	15540	3.26
7	(6, 8)	(4, 4)	(18, 24)	2448	0.39
8b	(16, 20)	(3, 3)	(48, 60)	35520	15.71
8c	(16, 20)	(4, 4)	(48, 60)	35520	15.82
9b	(8, 10)	(4, 4)	(16, 20)	4080	0.44
9c	(8, 10)	(4, 4)	(16, 20)	6640	0.47
10a	(35, 12)	(4, 4)	(70, 24)	21420	17.36
10b	(40, 12)	(4, 4)	(80, 24)	24480	25.56
10c	(40, 15)	(4, 4)	(80, 30)	33480	57.27
13top	(16, 20)	(4, 4)	(32, 40)	16320	8.40
13bot	(20, 30)	(4, 4)	(40, 60)	30600	48.25
12	(10, 10)	(4, 4)	(20, 20)	5100	0.59

Table 2. For the presented examples we list, from the left, the number of control points in  $u$  and  $v$  direction, the corresponding bi-degree, the number of sample points, the number of variables subject to optimization, and the computation time per iteration. For convergence, from five to ten iterations are generally needed. Values refer to the last step of a multiresolution optimization (see Sec. 4.1.1). Computation times refer to our implementation in Python and are tested on a Intel® Core™ i7-10750H CPU with 32 GB RAM.