Deployable strip structures

DAOMING LIU*, KAUST, Saudi Arabia
DAVIDE PELLIS*, ISTI-CNR, Italy
YU-CHOU CHIANG, National Chung Hsing University, Taiwan
FLORIAN RIST, KAUST, Saudi Arabia
JOHANNES WALLNER, TU Graz, Austria
HELMUT POTTMANN, KAUST, Saudi Arabia

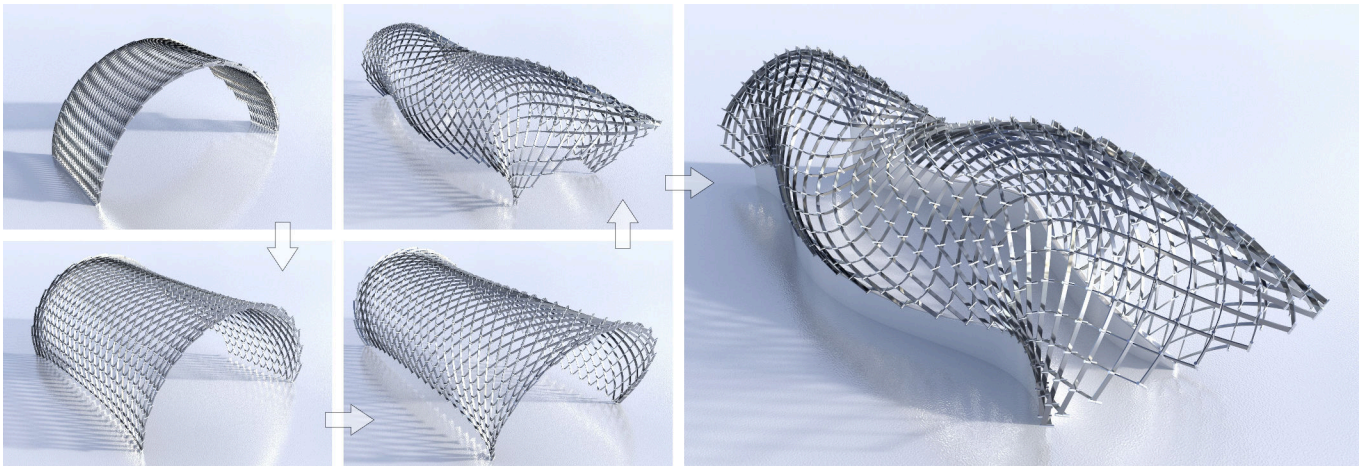


Fig. 1. We design and edit C-meshes which guide structures that can be deployed from a flat collapsed state. Geometrically, they are quad meshes or hex meshes with constant edge length and the additional property of spherical vertex stars.

We introduce the new concept of C-mesh to capture kinetic structures that can be deployed from a collapsed state. Quadrilateral C-meshes enjoy rich geometry and surprising relations with differential geometry: A structure that collapses onto a flat and straight strip corresponds to a Chebyshev net of curves on a surface of constant Gaussian curvature, while structures collapsing onto a circular strip follow surfaces which enjoy the linear-Weingarten property. Interestingly, allowing more general collapses actually leads to a smaller class of shapes. Hexagonal C-meshes have more degrees of freedom, but a local analysis suggests that there is no such direct relation to smooth surfaces. Besides theory, this paper provides tools for exploring the shape space of C-meshes and for their design. We also present an application for freeform architectural skins, namely paneling with spherical panels of constant radius, which is an important fabrication-related constraint.

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

Additional Key Words and Phrases: discrete differential geometry, digital fabrication, architectural geometry, computational design, inverse design, deployable structure, spherical paneling

ACM Reference Format:
Daoming Liu, Davide Pellis, Yu-Chou Chiang, Florian Rist, Johannes Wallner, and Helmut Pottmann. 2023. Deployable strip structures. *ACM Trans. Graph.* 42, 4 (August 2023), 16 pages. https://doi.org/10.1145/3592393

1 INTRODUCTION

Deployable structures arise in a variety of applications based on the fundamental property that such a structure is deployed from a small and compact rest state. Even if deployment is performed only once at the moment of installation, we already have the benefit of easy storage and transport. The potential range of applications is large, from small devices to flexible architectural designs, see Fig. 2. It is not surprising that deployable structures have been the focus of computational design already, see e.g. [Ren et al. 2022]. However, the design of complex double-curved shapes which are able to collapse onto a simple flat initial rest state is a challenging computational problem. Its solution in general heavily depends on the material behavior of the individual components and the way in which they are tied together to form a flexible and deployable structure. There

*equally contributing authors

Authors' addresses: Daoming Liu, KAUST, Saudi Arabia, liudaoming3@gmail.com; Davide Pellis, ISTI-CNR, Pisa, Italy, davidepellis@gmail.com; Yu-Chou Chiang, National Chung Hsing University, Taichung, Taiwan, chiang.yuchou@gmail.com; Florian Rist, KAUST, Saudi Arabia, florian.rist@kaust.edu.sa; Johannes Wallner, TU Graz, Austria, j.wallner@tugraz.at; Helmut Pottmann, KAUST, Saudi Arabia, helmut.pottmann@gmail.com.

0730-0301/2023/8-ART $15.00
https://doi.org/10.1145/3592393

Fig. 2. The 2021 "kinetic umbrella" by Jonas Schikore and Eike Schling is an elastically transformable lattice structure manufactured from straight lamellas. (© J. Schikore).

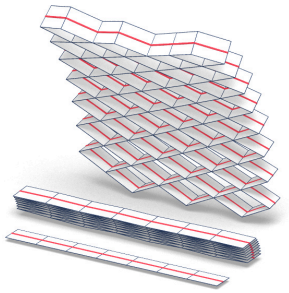


Fig. 3. A simple deployable structure based on a quad mesh. We show the ideal collapsed state, a more realistic rendering of an actual physical collapsed state, and the deployed state. The central mesh is highlighted in red.

is no general theory which would solve these computational design problems, but almost all solutions are formed by an interesting blend of tools from geometry, geometry processing, optimization and simulation. Our contribution to this area identifies a class of deployable structures which minimizes the influence of materials and imperfections and allows us to bypass simulation, making the problem a geometric one.

Inspiration for our research comes from deployable systems that are formed by elastic rods [Baek et al. 2018; D'Amico et al. 2015; Panetta et al. 2019; Pillwein et al. 2020; Pillwein and Musialski 2021; Tellier 2022], and in particular from the kinetic elastic gridshells of Schikore and Schling [2021; 2022]. They consist of straight flat lamellas that are bent into shape and arranged in a quadrilateral grid. After deployment, lamellas are in a position orthogonal to a smooth design surface (see Fig. 2).

The structures treated by this paper enjoy a full collapse onto a rest state formed by a stack of flat straight or flat circular strips that lie on top of each other. They open up to form either a quadrilateral or a hexagonal strip structure, see Figures 1 and 3. Deployment usually involves twisting individual strips, but does not have to: In case of a straight collapsed state, we can always deploy through an intermediate state which is still flat and simply a regular grid lying on the ground (Fig. 4, left).

We propose the novel concept of C-meshes as a discrete approximation to structures that deploy in the way described in the previous paragraph. In order to validate this simplification we also perform simulation based on discrete elastic rods [Bergou et al. 2008]. Our simulation model can handle all types of flat initial states, even if they do not collapse to a straight stack. Such flat initial states arise e.g. in the asymptotic gridshells of [Schling 2018]. On a smaller scale, flat initial states may also be considered as a type of *geometric material*, fabricated while flat and subsequently deformed into the target shape. Such geometric materials received a lot of attention in recent years, see e.g. [Chen et al. 2021; Duncan et al. 2018; Malomo et al. 2018; Özdemir et al. 2022]. Our deployable structures constitute a further contribution to this area.

1.1 Overview and contributions

We organize our paper as follows. § 2 defines C-meshes which serve as a discrete approximation for the central mesh that guides a deployable strip structure. We discuss how to compute C-meshes and deployable strip structures via optimization. The section concludes with a brief description of physical simulation of deployable strip structures, based on a fast implementation of discrete elastic rods.

The geometry of quad C-meshes is investigated in detail in § 3. They enjoy either a straight or a circular collapsed state, and are related to K-surfaces (straight case) and to linear-Weingarten surfaces (circular case). We also deduce the surprising fact that it does not make sense to allow more general collapsed states, as this dramatically reduces the degrees of freedom available for design. This phenomenon is related to a property of mechanisms that is actually well known in kinematic geometry: The ability of a mechanism to flex sometimes does not only depend on the combinatorics of its setup, but on very special geometric properties. We also discuss how to explore the design space of linear-Weingarten surfaces from its boundary, which is the pipe surfaces and is thus easily accessible.

§ 4 treats C-meshes with hexagonal combinatorics. While they enjoy more degrees of freedom in their design, a local analysis reveals that hex meshes can approximate smooth surfaces only in a slightly zigzagging way. We propose several methods to generate hex C-meshes, one of them being an initialization from a brick wall pattern.

In § 5 we provide a computational solution for the inverse design problem. For quad meshes, this works by optimizing surfaces to become hyperbolic linear Weingarten, using previous work on optimizing for the (simple) Weingarten property. Design methods for hex meshes are based on the quad case – here we do not fully explore the design space.

§ 6 contains results and discussion, in particular the paneling with spheres of constant radius associated with each quadrilateral C-mesh. We also provide statistics on our computations, and we present a validation of the theory by means of experiments with physical models. The paper concludes with a discussion of limitations, and proposals for future work.

1.2 Related Work

We start with an overview of related work in *Geometry* and *Geometric Computing*. Our deployable quad structures are related to *K-surfaces* which possess constant negative Gaussian curvature, and also to *linear Weingarten surfaces of hyperbolic type*. These are defined by the property that their Gaussian curvature K and mean curvature H obey a linear relation $aK + bH + c = 0$ with $b^2 - 4ac < 0$; they are equivalently characterized as offsets of K-surfaces. In particular, we show that deployable structures are naturally approximated by discrete surfaces studied first by [Wunderlich 1951]; see also the

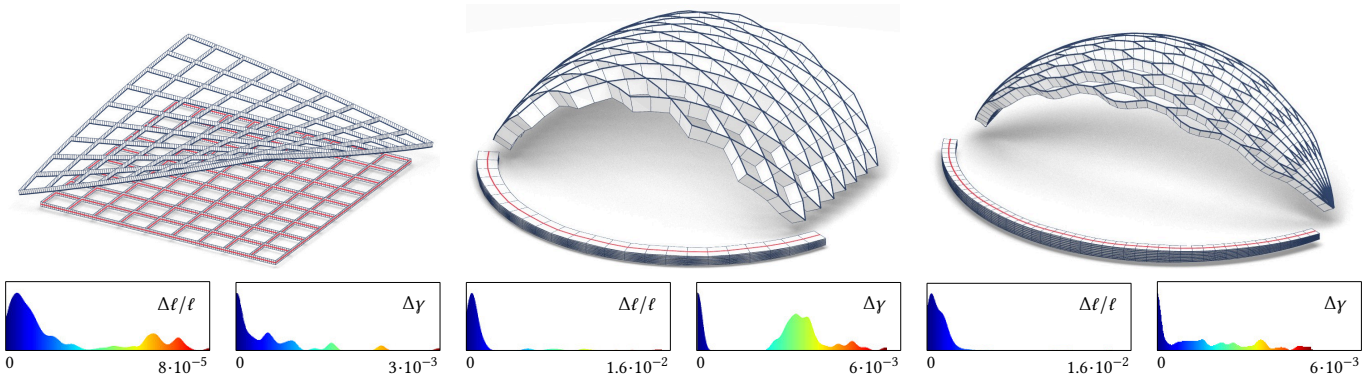


Fig. 4. Physical simulation of deployable structures. Here the deployment from the collapsed state has been computed by a generalized discrete elastic rod simulation according to [Jawed et al. 2018]. To verify that our simplified approach via C-meshes is valid, we extract the central mesh and verify that it is almost a C-mesh. According to Lemma 2.4 being a C-mesh means edge lenghts are constant, and so are the angles between edges and normals in the vertices . We therefore display the relative deviation of edge length "$\frac{\Delta\ell}{\ell}$" as well as the deviation "$\Delta\gamma$" of the angle from its average (except for angles at the boundary). The statistics confirm the C-mesh property to a satisfactory extent.

modern treatment by [Bobenko and Suris 2008]. Our discrete models constitute a subset of principal symmetric meshes in the sense of [Pellis et al. 2020]. Pellis et al. [2021] have shown how to approximate a given shape by a *Weingarten surface* (where H, K are related), and their procedures can be adapted to the special case of linear Weingarten surfaces of hyperbolic type. We use their work for inverse design.

The quadrilateral *C-meshes* occurring in our work are special discrete *Chebyshev nets*, meaning they have constant edge length. Patchworks of general Chebyshev nets can approximate arbitrary shapes [Garg et al. 2014; Sageman-Furnas et al. 2019]. There is much less related work on hexagonal meshes. Wunderlich [1973] studied rotationally symmetric hex meshes with constant edge length in static equilibrium and derived them from the quadrilateral counterparts by edge insertion. Such a strategy will turn out to be useful also in our case. Wunderlich's meshes belong to a special class of hex C-meshes which can be reconnected into hexagonal meshes with planar faces. The latter have turned out to be interesting for applications in architecture; we only mention the most recent contribution by Pluta et al. [2021] and point to the references given there. The Caravel meshes of [Tellier et al. 2020] include certain hex meshes with constraints on nodes to achieve repetitive elements for architectural applications.

A further topic related to ours concerns the computational design of various types of structures from developable or nearly developable strips, such as arrangements of strips following plane elastic curves [Hafner and Bickel 2021], rotational symmetric arrangements of strips arising from a flat initial state [Mhatre et al. 2021] or strips that form weaving patterns [Ren et al. 2021; Vekhter et al. 2019]. Asymptotic gridshells [Schling 2018] and asymptotic geodesic hybrid gridshells [Schling et al. 2022] also belong to this type of structures. [Hong et al. 2022] take a geometric approach to turn 2D strip patterns into 3D shapes through appropriate boundary actuation.

Architecture and Engineering. We already pointed to research on deployable grid shells from elastic rods [Baek et al. 2018; Panetta et al. 2019; Pillwein et al. 2020; Pillwein and Musialski 2021; Tellier 2022]. The shapes of most of these types are strongly dominated by geometry, which is a property supporting inverse design. Geometry does not play such a major role for the X-shells proposed by [Panetta et al. 2019], which may be a reason for missing inverse design results. A prominent early example in architecture is provided by the 1975 Mannheim *Multihalle* designed by Frei Otto. [D'Amico et al. 2015] used the same construction method but aided by modern numerical simulation.

Even if they are not strictly relevant for our paper, we also briefly address deployable structures with *rigid* elements. The flexible "spheres" of Hoberman [1990] are a contribution to this area of transformable design. A combination of rigid and elastic elements is used by the *umbrella meshes* recently proposed by Ren et al. [2022]. Another kind of mechanism are programmable auxetic metamaterials [Konaković-Luković et al. 2018] which are actuated by inflating a balloon. Inflatables in general are deployable in the wider sense; they have a long history in Architecture. For a recent approach to inverse design we refer to [Panetta et al. 2021].

Physical Simulation. There is, of course, a huge body of literature on the elastic deformation of shapes. Relevant to this paper, early contributions to thin elastic strips are due to Sadowsky and Wunderlich, cf. translation by [Fosdick and Fried 2016], which are still relevant in recent contributions to ribbon simulation [Charrondière et al. 2020]. That part of physical simulation which is closest to our work is the simulation of elastic rods. Here popular reduced models are based on Kirchhoff-Love rod theory or Cosserat rod theory [Jawed et al. 2018]. An early contribution from the graphics community is [Pai 2002] where the Cosserat rod model was used for interactive thin solid simulation. More recently, [Deul et al. 2018; Umetani et al. 2015] implemented the Cosserat rod model within the position based dynamics framework and [Soler et al. 2018] implemented it within the projective dynamics framework. In this paper, we adopt the discrete elastic rod model proposed by Bergou et al. [2010; 2008]. The method has been extended by a variety of contributions, e.g., [Pérez et al. 2015] adopted rigid transformation for the rod joints. [Panetta et al. 2019] proposed a nine-variables

rod joint model especially for connections of rod ends at the joints. More relevant to this paper, [Ren et al. 2021; Vekhter et al. 2019] adapted this approach to strip-like elements and weaving. Recently, [Pillwein et al. 2020] used it to model geodesic grids with rotational and slightly-sliding strip joints.

2 A SIMPLIFIED GEOMETRIC MODEL

We are going to present a simplified geometric model for deployable structures composed of individual strips joined together with hinges. The simplified model, which is the topic of § 2.1, is used for design and interactive modelling. For verification of the results we perform a more realistic physical simulation, see § 2.2.

We are modelling flexible arrangements of elastic strips which are connected at nodes in the manner of a regular quad mesh or alternatively as a regular hex mesh. The edges of the mesh represent the central lines of strips which extend in a direction orthogonal to the mesh surface. The strips are connected along hinges which pass through the vertices of the *central mesh*, and which are approximately orthogonal to the mesh (see Fig. 3). The strips deform in an elastic manner, in particular they experience twisting.

We wish to model structures exhibiting a collapsed state where all involved strips are arranged along a curve. Fig. 3 shows an example where the collapsed state is described by a straight line. The hinges are orthogonal to this central line. For simplicity we assume that all edge lengths are equal. We also consider the case where the collapsed structure is aligned with a circle, with the hinges lying orthogonal to this central circle.

This restriction to lines and circles may seem arbitrary, but this is not actually the case. Firstly, the geometry of central meshes corresponding to these cases is very interesting, and their geometric properties allow us to derive information on degrees of freedom for deployment. Secondly, and more importantly, one can show that a collapsed state which is neither straight nor circular considerably restricts the shapes that can be achieved. The structure then is automatically generated by rotation or by translation (see Th. 3.8).

2.1 C-meshes

Definition 2.1 (Strip arrangement). *Assume the vertices of a mesh $M = (V, E, F)$ are equipped with unit "normal" vectors $n(v)$. The strip arrangement defined by these data consists of one quadrilateral strip per edge $e = (v, w)$, which has vertices $v \pm \delta n(v)$ and $w \pm \delta n(w)$.*

We imagine the strips to be twisted versions of flat quadrilaterals, and we consider only the case where the vectors $n(v)$ are not far from consistently oriented normal vectors of a smooth surface.

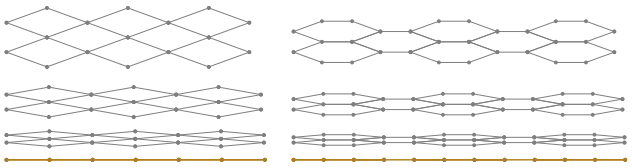


Fig. 5. A C-mesh from the combinatorial viewpoint is part of a regular 2D lattice. In this sequence of images showing a collapse we see how this regularity in the collapsed state is also expressed geometrically.

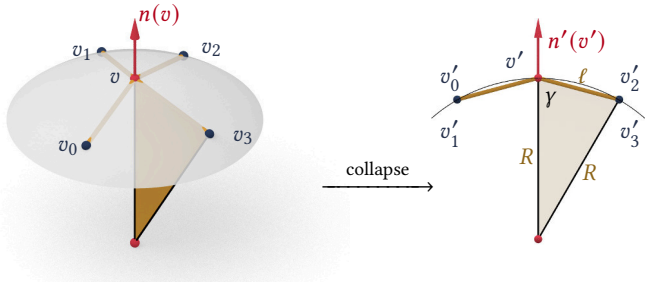


Fig. 6. *Left:* A vertex v with neighbours v_0, v_1, v_2, v_3 in a C-mesh. The constant edge length ℓ of the mesh and the sphere radius R determine the angle γ between the normal vector $n(v)$ and the edges incident with v: An isosceles triangle with edge lengths ℓ, R, R yields $\cos\gamma = \frac{1}{2}\ell/R$. *Right:* This vertex star collapses onto a circle.

Definition 2.2 (deployment, collapsed state). *A deployment of a mesh M according to Def. 2.1 is a combinatorially equivalent mesh M' where corresponding edges have the same length, and corresponding angles between normal vectors and edges are equal. If in addition M' is contained in a plane, with normal vectors being parallel to this plane, we say M' is a collapsed state of M.*

We think of this deployment as imitating an elastic deformation of the strip arrangement. If a mesh is in a collapsed state, the strips associated with it do not experience any twisting, and they lie on top of each other. We are interested in meshes which collapse onto a curve:

Definition 2.3 (C-mesh). *A C-mesh M is either a quad mesh or a hex mesh of regular combinatorics, with all edge lengths being equal, which can be deployed onto a collapsed state M' whose vertices are arranged in a regular manner along a straight line, or along a circle, as illustrated by Figure 5.*

The letter C in "C-mesh" refers to the property that the mesh is the central mesh of the strip arrangement. It may also refer to the arrangement being collapsible. It turns out that the C-mesh property has a nice equivalent characterization:

Lemma 2.4. *A regular quad mesh resp. hex mesh with constant edge length ℓ is a C-mesh if and only if the angles between edges and normal vectors $n(v)$ are all equal*

Equivalently, the mesh is a C-mesh, if for each vertex v there is a sphere $S(v)$ which contains v together with its immediate neighbours, and all spheres $S(v)$ have the same radius R. $S(v)$ degenerates into a plane if $R = \infty$.

Proof. $n(v)$ has the same angle γ with all edges emanating from v if and only if the vertex star of v can be collapsed to lie on a circle with radius R. Fig. 6 shows that

$$\cos\gamma(\ell) = \ell/2R, \tag{1}$$

with the limit case $\gamma(\ell) = \frac{\pi}{2}$ for $R = \infty$. We write the angle as a function of the edge lengths to emphasize the dependence on ℓ in formulas where both ℓ and γ occur. If the angle is the same also for the other vertices, the collapse onto the same circle propagates through the mesh.

The equivalence of the angle property with the sphere property is elementary. □

In § 3 we are going to show further geometric properties of C-meshes. For computational purposes we model C-meshes and their deployment by the following constraints and energies.

Energies for modeling C-meshes. Constant edge lengths and angles are expressed by the constraints

$$c_{\text{len}}(e) = \|e\|^2 - \ell^2, \quad c_{\text{angle}}(v,w) = \langle w - v, n(v)\rangle - \ell\cos\gamma(\ell).$$

Together with the normalization constraint $c_{\text{norm}}(v) = \|n(v)\|^2 - 1$ we define an energy which vanishes if and only if the mesh under consideration has the C property:

$$E(M) = \alpha_1 \sum\nolimits_{e\in E} c_{\text{len}}(e)^2 + \alpha_2 \sum\nolimits_{v\in V} c_{\text{norm}}(v)^2. \\ + \alpha_3 \sum\nolimits_{(v,w)\in E} \left(c_{\text{angle}}(v,w)^2 + c_{\text{angle}}(w,v)^2\right)$$

For the actual computation we add a fairness energy which in the quad mesh case expresses small 2nd order differences of all triples (u,v,w) of successive vertices in the mesh:

$$E_{\text{fair}}(M) = \alpha_{\text{fair}} \sum_{\text{triples } u,v,w} \|u - 2v + w\|^2$$

In the hex mesh case we employ the fairness energy proposed by [Jiang et al. 2015] which penalizes a configuration of two neighbouring faces if it is not symmetric about the common edge midpoint.

Energies for Deployment. As to deployment, a mesh M deploys to M' if M, M' are combinatorially equivalent, and both are C-meshes with the same parameters ℓ, γ. To regularize the deformation experienced by the individual strips, for each edge $e = (v,w)$ the amount of twisting about an edge v, w, measured by

$$\text{twist}(v,w) = \|(n(v) - n(w)) \times (v - w)\|, \tag{2}$$

should be small. Since $\text{twist}(v,w)^2 = \|n(v) - n(w)\|^2\|v - w\|^2 - (\langle n(v), v - w\rangle - \langle n(w), v - w\rangle)^2$ and all terms except the first are constant, we define the energy $E_{\text{twist}}(M, M')$ by letting

$$E_{\text{twist}} = \alpha_{\text{twist}} \sum_{(v,w)\in E} (\|n(v) - n(w)\|^2 - \|n'(v') - n'(w')\|^2)^2.$$

Here v', w' are the vertices corresponding to v, w in the mesh M'. $E_{\text{twist}}(M, M')$ is small if the strips of M experience the same torsion as the strips of M'. If M' is in a collapsed state, there is no twist at all, and the energy simply penalizes large twists in M. We will see that this energy is actually not necessary in the quad mesh case, since Prop. 3.2 ensures constant twist.

Finally, we give an energy which expresses that vertex stars in M look similar to vertex stars in M' (which is definitely not the case if M' is in a collapsed state and M is not). We define

$$E_{\text{stiff}}(M, M') = \alpha_{\text{stiff}} \sum_{v\in V} \sum_{v_1,\ldots,v_k \in \text{link}(v)} (\|v_i - v_{i+1}\|^2 - \|v_i' - v_{i+1}'\|^2)^2.$$

Here again the vertex v_i' of M' is the vertex corresponding to v_i of M. In the summation over the link of v, indices are modulo k. We used this energy as a regularizer with low weight in certain examples.

Optimization. The examples in this paper have been produced by minimizing a combination of the energies defined above. Depending on the application, additional constraints are added, e.g. soft or hard positional constraints, or soft proximity constraints when a C-mesh is to follow a reference surface Φ. The corresponding energies are quadratic and read

$$E_{\text{pos}} = \alpha_{\text{pos}} \sum \text{dist}(v_i, v_i^0)^2, \quad E_{\text{prox}} = \alpha_{\text{prox}} \sum \text{dist}(v_j, T_{v_j^*})^2.$$

Summation in each case is over those vertices for which constraints are to be enforced. Following [Tang et al. 2014], proximity to Φ is achieved by penalizing deviation of vertices v_i from the tangent plane $T_{v_i^*}$ in $v_i^* \in \Phi$, which is the closest point to v_i. In our iterative optimization procedure, we recompute points v_i^* after each round, using the approximate nearest neighbour algorithm on a triangle mesh Φ [Mount and Arya 2010].

For optimization we used a Levenberg-Marquardt method according to [Nocedal and Wright 2006, Ch. 10]. The linear systems occurring there have been solved by a preconditioned conjugate gradient method. We emphasize that intermediate positions in the deployment are not present in the optmization.

2.2 Verification of the Geometric Model

Our choice to use C-meshes for the description of deployable structures has been validated by physical simulation. For verification we perform a more realistic physical simulation, where strips are modelled as discrete elastic rods with elliptic cross-section, which is a generalization of the discrete elastic rods approach proposed by [Bergou et al. 2008]. This choice is mainly motivated by the fact that this model has been recently applied and validated successfully in several computational design and fabrication applications. We could just as well have used a rectangular cross-section.

To be precise, we use the setup of equations proposed by [Jawed et al. 2018, p. 89, Eq. (8.18)]. The system of constraints expressing the elastic behaviour of strips is augmented by constraints which fix the relative position of strips at nodes. Figure 4 shows the deployment of some examples computed in this way. For the actual computation, a dynamic relaxation method according to [Otter et al. 1966] was adopted (we used linear damping proportional to velocity). This physical simulation is not considered a contribution of the paper. Therefore we do not go into details and refer only to literature references and to Figure 4. The results of simulation are also borne out by measurements taken from physical models, see § 6.2 and Fig. 22.

3 GEOMETRY OF QUADRILATERAL C-MESHES

This section discusses in detail the geometry of quadrilateral C-meshes. It turns out that they are either discrete surfaces of constant Gaussian curvature, or offsets of such surfaces. This surface class is known as the *linear-Weingarten surfaces of hyperbolic type*, which immediately leads us to a discussion of surfaces with the linear Weingarten property in general. This discussion serves two purposes: Firstly it is important to appreciate that being a quadrilateral C-mesh comes with shape restrictions. Secondly, the *boundary* of the admissible variety of surfaces turns out to be easily accessible

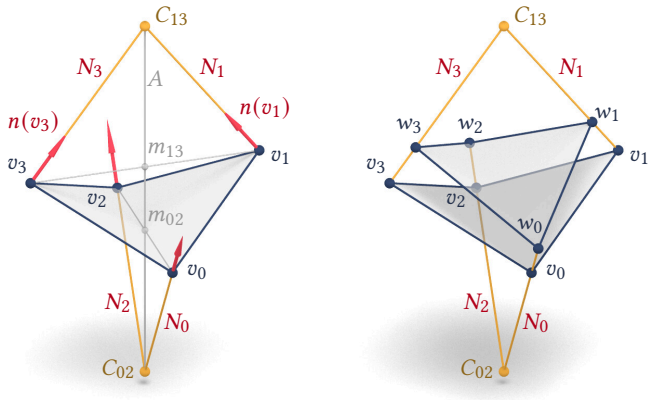


Fig. 7. A quadrilateral face $v_0v_1v_2v_3$ of a C-mesh. *Left:* The normals N_i, N_j in opposite vertices v_i, v_j intersect on the axis of rotational symmetry. The quad also enjoys a reflection symmetry w.r.t. the plane $\sigma_{ij} = \mathrm{span}(v_i, v_j, A)$, where $i, j = 0, 2$ or $i, j = 1, 3$. *Right:* Offsetting vertices v_i along normals changes the radius of the collapsed state of the mesh.

for design and can thus serve as initialization (that boundary consists of the pipe surfaces which are defined by a central curve and a radius). In § 3.5 we investigate what happens if we depart form the requirement that C-meshes must collapse onto straight lines or circles. We show that allowing this unexpectedly diminishes the variety of available shapes.

3.1 Meshes with Straight Collapsed State

A quad structure that collapses onto a straight line is described by a C-mesh exhibiting constant edge length ℓ and planar nodes. Such meshes are well known in discrete differential geometry. The constant edge length property makes them discrete *Chebyshev nets*. The planar vertex stars reveal them as *asymptotic nets*, meaning that the parameter lines of the mesh are discrete asymptotic curves. In the smooth case, the combination of these properties implies constant negative Gaussian curvature. Indeed, C-meshes with planar nodes are precisely the discrete surfaces of constant Gaussian curvature which were studied in detail by [Bobenko and Suris 2008; Sauer 1970; Wunderlich 1951]. They are commonly referred to as *K-nets*.

As a preparation for the discussions below, we study a single quad with vertices $v_0, \ldots, v_3$ where all edge lengths equal ℓ (see Fig. 7). The midpoints m_{02} and m_{13} of diagonals v_0v_2 and v_1v_3, respectively, span a line A. The quad is symmetric w.r.t. rotation about A by 180°. This rotation permutes the pairs v_0, v_2 and also v_1, v_3. Further it is clear from symmetry that the quad has a mirror symmetry w.r.t. two planes, namely

$$\sigma_{02} = \mathrm{span}(A, v_0, v_2), \quad \sigma_{13} = \mathrm{span}(A, v_1, v_3).$$

In each vertex v_i a discrete surface normal N_i is spanned by the normal vectors $n(v_i)$ (red arrows in Fig. 7). Since normals are orthogonal to the edges, the following is also clear from symmetry: The configuration of normals is symmetric w.r.t. the reflections mentioned above. Consequently, normals at N_1, N_3 intersect in a point C_{13}. Likewise the normals N_0, N_2 intersect in a point C_{02}. Both these centers lie on the axis A.

3.2 C-Meshes with Circular Collapsed State

A C-mesh collapsing onto a circle with radius R has a constant edge length ℓ, and the star of each vertex is contained in a sphere with radius R.

Wunderlich [1951] pointed out that the K-nets of §3.1 possess constant-distance offsets which are C-meshes in our sense: offsetting turns a C-mesh with straight collapse into a C-mesh with circular collapse. Remarkably, also the converse is true. We are able to show a new result stating that all C-meshes can be created in this way:

Theorem 3.1. *For a C-mesh collapsing onto a circle, equipped with unit normal vectors $n(v_i)$ in its vertices, there is a value d such that the constant-distance offset mesh at distance $|d|$, defined by vertices*

$$w_i = v_i + dn(v_i)$$

is a K-net, i.e., a C-mesh collapsing onto a straight line.

Proof. By Equ. (1) all edges emanating from v_i have the same length ℓ, and they intersect the normal N_i under the angle γ, where $\ell = 2R\cos\gamma$. Since $R \neq \infty$, γ is not 90° (that would be the K-net case). Fig. 7, right, illustrates a face $f = (v_0v_1v_2v_3)$. Since we require a consistent orientation of unit normal vectors $n(v_i)$, it is not possible that, say $n(v_1)$ points upwards and $n(v_3)$ points downwards. Rather, they look exactly like shown by Fig. 7. By symmetry, normals N_0, N_2 intersect in a point $C_{02} \in A$, normals N_1, N_3 intersect in a point $C_{03} \in A$.

We now construct an offset mesh with vertices w_i which arise from v_i by movement along the normal N_i. It is well known that the shortest distance between straight lines N_i, N_j occurs between points $w^*_{ij} = v_i + d_{ij}n(v_i)$ on N_i and $w^*_{ji} = v_j + d_{ji}n(v_j)$ on N_j, and that the segment $w^*_{ij}w^*_{ji}$ is orthogonal to both normals N_i, N_j. Lemma A.1 in the appendix shows that offsetting distances d_{ij}, d_{ji} are equal. By mirror symmetry, this offsetting distance is the same for all edges of a face f; we call this value $d(f)$ and we let $w_i = v_i + d(f)n(v_i)$. If faces f, f' share an edge v_iv_j then obviously $d(f) = d_{ij} = d_{ji} = d(f')$. Thus the constant offset distance propagates through the mesh. The orthogonality mentioned above means that the edges of the offset mesh are orthogonal to the normals. □

3.3 Strip Geometry in C-Meshes

The following is an interesting fact concerning the individual strips making up the strip arrangement derived from a C-mesh:

Proposition 3.2. *The twist of strips as defined by Equ. (2) is constant throughout the mesh; all quadrilateral strips associated with edges according to Defn. 2.1 are congruent to each other.*

Proof. Consider first the case of a K-net, i.e., a C-mesh collapsing onto a straight line. For any edge vw, normal vectors $n(v)$ and $n(w)$ are orthogonal to $v - w$, and the angle enclosed by $n(v), n(w)$ is known to be constant [Wunderlich 1951]. Thus $\|n(v) - n(w)\|$ is constant. Since edge lengths are likewise constant, so is the twist. This remains true if offsetting is applied to create general C-meshes, since vectors $n(v), n(w)$ do not change.

A quadrilateral strip associated with the edge vw has vertices $v \pm dn(v), w \pm dn(w)$, cf. Defn. 2.1. The equality of angles mentioned above ensures they are all congruent, which also follows from the mirror symmetries visualized by Fig. 7. □

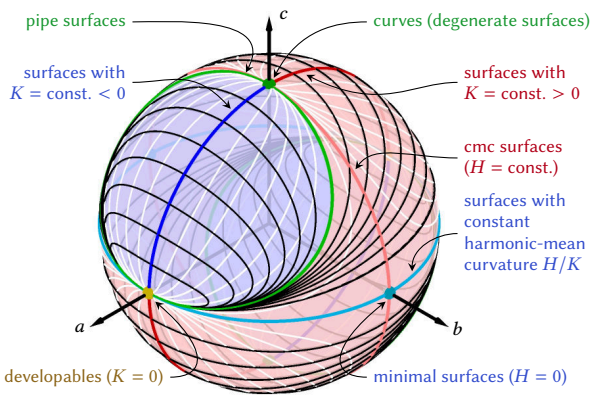


Fig. 8. Classification of linear Weingarten surfaces. A surface enjoying the relation $aK + bH + c = 0$ in this image is represented by the point (a, b, c) which is normalized such that $a^2 + b^2 + c^2 = 1$. Thin white and black curves correspond to special transformations of surfaces: If a surface undergoes offsetting resp. scaling, it moves along a black resp. white curve. The blue part of the sphere corresponds to hyperbolic surfaces which represent the shapes of C-quad meshes.

3.4 Smooth Surfaces Related to C-Meshes

3.4.1 Linear Weingarten Surfaces of Hyperbolic Type. Theorem 3.1 establishes C-meshes as offset meshes of K-nets. Smooth counterparts of C-meshes therefore are found by offsetting surfaces with constant Gaussian curvature $K < 0$. It is well known that such offsets enjoy a linear relation between mean curvature H and Gauss curvature K:

$$aK + bH + c = 0, \tag{3}$$

where a, b, c are constant throughout the surface. Such surfaces are called *Linear Weingarten*. For us, the following result is convenient:

Lemma 3.3. *If a smooth linear-Weingarten surface W with unit normal vector field n enjoys the property $aK + bH + c = 0$, then its offset surface $W' = W + dn$ at distance $|d|$ is also linear-Weingarten. Its curvatures obey $a'K' + b'H' + c' = 0$, with*

$$a' = a - bd + cd^2,\ b' = b - 2cd,\ c' = c. \tag{4}$$

Proof. Consider the principal curvatures κ_1, κ_2, where $H = (\kappa_1 + \kappa_2)/2$ and $K = \kappa_1\kappa_2$, and similar κ_1', κ_2' for the offset surface. Offsetting causes the inverses of principal curvatures to shift by a constant amount (the principal centers of curvature do not change under offsetting): $1/\kappa_i' = 1/\kappa_i + d$.

A straightforward computation now allows us to relate H, K with the offset's mean curvature H' and Gaussian curvature K':

$$K = \frac{K'}{1 - 2dH' + d^2K'}, \quad H = \frac{H' - dK'}{1 - 2dH' + d^2K'} \tag{5}$$

Inserting this into (3) shows the result. □

Linear Weingarten surfaces are traditionally classified as having elliptic, parabolic, or hyperbolic type, if $b^2 - 4ac$ is positive, zero, or negative, respectively. It is very interesting to know that the shapes of C-meshes are represented exactly by one of these categories:

Lemma 3.4. *The linear-Weingarten surfaces of hyperbolic type are exactly the offsets of surfaces of constant negative Gaussian curvature.*

Proof. (1) Surfaces with $K = \text{const}$ obey a linear-Weingarten relation $aK + bH + c = 0$ where $b = 0$. Hyperbolic type means $4ac > 0$, i.e., $K < 0$. (2) Equ. (4) shows that for any linear Weingarten surface, an offset with $d = -b/2c$ has curvatures H', K' obeying $a'K' + c' = 0$, i.e., $K' = -c'/a'$ is constant. This value expands to

$$K' = \frac{4c^2}{b^2 - 4ac}, \tag{6}$$

which is negative for hyperbolic type. (3) The class of hyperbolic linear Weingarten surfaces is invariant under offsetting, since $(b')^2 - 4a'c' = b^2 - 4ac$. □

Figure 8 gives an overview of the various types of linear Weingarten surfaces according to the defining relation $aK + bH + c = 0$. Offsetting changes the coefficients a, b, c according to (4), and Fig. 8 shows the progress of selected surfaces through such an offset family. Likewise, scaling a surface by a factor λ changes Gauss curvature K and mean curvature H to K/λ^2 and H/λ, so coefficients a, b, c are replaced by $a\lambda^2, b\lambda, c$. Also the progress of a surface through such a scaling family is illustrated.

In §3.1 we related C-meshes with a straight collapse to K-surfaces as their smooth counterpart. We established that the edges of the mesh correspond to a Chebyshev net of curves exhibiting zero normal curvature. The previous paragraphs established how other C-meshes are generated by an offsetting operation. Analogously, general linear Weingarten surfaces of hyperbolic type are generated from K-surfaces by an offsetting operation. The edges of the C-meshes again correspond to a special curve network on the surface. If the analogy between the mesh case and the surface case is true, this must again be a Chebyshev net (since edge lengths in the mesh are constant). Further, the curves in the network should exhibit constant normal curvature, since in the C-mesh any triple of successive vertices of a mesh polyline lies in a sphere of constant radius R which is tangent to the surface (this discrete version of normal curvature has already been used by [Pellis et al. 2020]).

The following result establishes that the analogy is indeed true: A linear-Weingarten surface of hyperbolic type is covered by a Chebyshev net of curves with constant normal curvature κ_n, and consequently is discretized by a C-mesh with radius $R = 1/\kappa_n$.

Proposition 3.5. *Any linear Weingarten surface W of hyperbolic type which obeys the relation $aK + bH + c = 0$ contains a Chebyshev net of curves exhibiting constant normal curvature*

$$\kappa_{nc} = -b/2a. \tag{7}$$

The angle α between these curves and the first principal direction obeys

$$\tan^2\alpha = -\frac{\kappa_1(2c + b\kappa_1)}{\kappa_2(2c + b\kappa_2)}. \tag{8}$$

Proof. We know from (6) that an appropriate offset W' at distance $d = -b/2c$ has constant Gaussian curvature $K' < 0$. W' has a parameterization $x'(u, v)$ whose parameter lines are a Chebyshev net (i.e., $\|x_u'\| = \|x_v'\| = \alpha = \text{const.}$) of asymptotic curves, and there is a normal vector field $n(u, v)$ with $\|n\| = \nu = \text{const.}$ obeying the so-called Lelieuvre conditions $x_u' = n \times n_u$, $x_v' = -n \times n_v$ [Bobenko

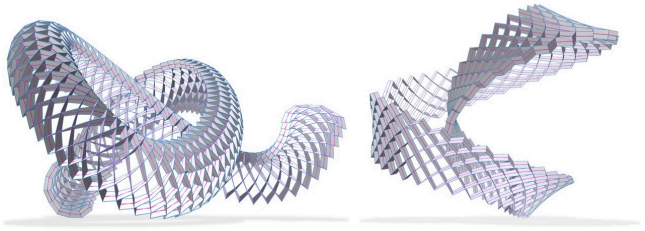


Fig. 9. Deployable structures whose central C-meshes have been initialized from a pipe surface (left) and from a developable (right). Optimization towards the C-mesh property changes the shape slightly.

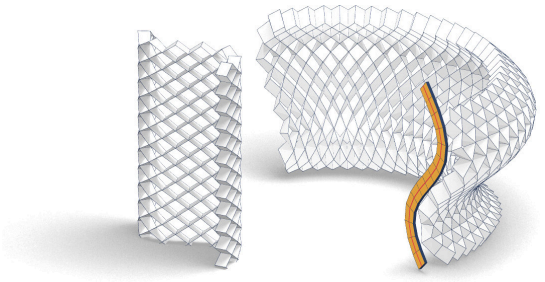


Fig. 10. Meshes deploying from a general profile (neither circle nor straight line) have rotational symmetry or are cylinders. This theoretical result is confirmed by these two examples which have been generated by optimization from C-mesh constraints, without prior knowledge of the global shape.

and Suris 2008]. From $\|n\| = \text{const.}$ we see that n, n_u are orthogonal. From $x'_u = n \times n_u$ we derive that x'_u is orthogonal to both n, n_u and further, $\|n_u\| = \frac{\alpha}{v}$. A similar statement is true for x'_v, n, n_v. Undoing the offsetting operation yields a parameterization $x(u,v) = x'(u,v) \pm \frac{d}{v} n(u,v)$ of the original surface. From $\|x_u\|^2 = \|x'_u \pm \frac{d}{v} n_u\|^2 = \alpha^2 + d^2\alpha^2/v^4 = \text{const.}$ and the analogous relation for x_v we see that $x(u,v)$ is a Chebyshev net. Equations (7) and (8) are proved in the Appendix. □

3.4.2 Parabolic linear Weingarten surfaces. General C-meshes correspond to *hyperbolic* linear-Weingarten surfaces, i.e., to the blue region in Fig. 8. Their shapes are not so easily understood, so we suggest to approach the design space from its boundary (the green curve in Fig. 8). These turn out to be a well known class of surfaces:

Proposition 3.6. *A linear-Weingarten surface W of parabolic type is either developable or a pipe surface, which is defined as the points of constant distance r from a central spine curve W'.*

This fact should be well known, but we could not find an explicit literature reference. To see it is true, we argue as follows.

Proof. Recall that *parabolic* type means that the coefficients in the linear-Weingarten relation $aK + bH + c = 0$ obey $b^2 - 4ac = 0$. By Equation (4), this condition is invariant under offsetting. The developables which exhibit $K = 0$ fall into this class; they obey the relation $aK + bH + c = 0$ with $(a, b, c) = (a, 0, 0)$. In the other parabolic cases we try to find an offset W' of constant Gauss curvature according to Prop. 3.5. This does not work, because Equ. (6) shows that W' has infinite Gauss curvature. W' therefore is not a surface, it has degenerated into a curve. The offset family consists of pipe surfaces with W' as its spine.

On the other hand, the differential geometry of a pipe surface (tubular surface) W is well known. It is the envelope of spheres of radius r centered at the spine W'. These spheres touch W along circles of radius r which constitute one family of principal curvature lines. Thus one principal curvature equals $1/r$, and we have $H = (1/r + \kappa)/2$, $K = \kappa/r$, with κ being the other principal curvature. It follows that W obeys a linear-Weingarten relation $aK + bH + c = 0$ with $a = 1$, $b = -2/r$, $c = 1/r^2$. Obviously, $b^2 - 4ac = 0$. □

When a hyperbolic linear-Weingarten surface (contained in the blue region in Fig. 8) evolves towards the parabolic boundary (green in Fig. 8), the special Chebyshev nets relevant for discretization degenerate: (8) yields $\alpha = 0$, which says that the net degenerates to the family of principal circles of constant normal curvature $1/r$.

Remark 3.7. Developables are envelopes of planes, they can be seen as pipe surfaces with $r = \infty$. Principal circles appear as rulings.

The discussion above inspires the following design method for C-meshes: We cover a pipe surface with a quad mesh whose edges lie symmetric w.r.t. the principal circles and have a small angle with those circles. Our experiments show that subsequent optimization towards the C-mesh property causes only a small shape change. The same method works by starting from a developable surface and its rulings. Results are shown by Fig. 9.

3.5 Quad Meshes Collapsing to General Curves

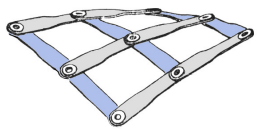
So far we considered only structures collapsing onto straight lines or circles. It is natural to get rid of this restriction. Remarkably, this does not increase the variety of available shapes, but quite the opposite is the case.

We define a *generalized C-mesh* as a regular quad mesh M with constant edge length ℓ that collapses onto a 2D polyline. To be more precise, the polyline's vertices are labelled w_k, and the vertices of M are labelled $v_{i,j}$, with i, j, k as integers. The immediate neighbours of $v_{i,j}$ are $v_{i\pm1,j}$ and $v_{i,j\pm1}$. We now require that M deploys to a mesh M' with vertices $v'_{i,j} = w_{i-j}$.

There is the following result concerning the global shape of generalized C-meshes. It is illustrated in Fig. 10.

Theorem 3.8. *Generalized C-meshes which are not C-meshes represent either surfaces of revolution or cylindrical surfaces.*

The proof is given in the Appendix. It shows that the shape restriction also applies in case the collapsed state is a union of circular arcs. The proof reveals the reason for the loss of degrees of freedom: It is the loss of symmetries in the configuration of faces plus normals shown by Fig. 7.



Remark 3.9. The configuration of vertices plus normals of Fig. 7 can be physically realized by rigid strips along the edges which are connected by revolute joints in the vertices. A twist in the strip is used to model the constant torsion discussed by Prop.

3.2. In general, a quad formed by four such strips is rigid, but if its dimensions are derived from a C-mesh it is flexible. One can even join an arbitrary number of such quads to form a flexible quad mesh, see [Wunderlich 1951] and inset figure. Quads derived from generalized C-meshes do not enjoy flexibility. This corresponds to their restricted shapes as described by Theorem 3.8.

4 HEXAGONAL C-MESHES

The detailed study of quadrilateral C-meshes in § 3 shows that they have rich and interesting geometry but their shape space is restricted. We therefore turn to hexagonal C-meshes in order to increase the degrees of freedom in designing deployable structures that collapse onto curves. Interestingly, the correspondence between discrete C-meshes and their smooth counterparts is not of the same nature as it was for quad meshes. For a quad C-mesh with a straight collapsed state, we relate the edges to asymptotic curves on a smooth surface. This cannot work in the hex mesh case (we would have 3 asymptotic curves passing through every point). In fact, a hexagonal C-mesh approximating a smooth surface must do so in a zigzagging way. This non-smoothness is not visually apparent, however. Below we use a local analysis to investigate the possible shapes of vertex stars in hexagonal C-meshes.

4.1 Shapes of Vertex Stars in Hexagonal C-Meshes

Consider a hexagonal C-mesh M according to Def. 2.3 which in particular has constant edge length ℓ. We are interested in meshes whose vertices lie in a smooth surface S and start our discussion with the case that M has a straight collapsed state, i.e., each vertex v has a plane $T(v)$ containing the neighbours of v. Their distance from v equals the constant value ℓ. Possible locations can therefore be found by intersecting a circle with radius ℓ with S. Replacing S by its osculating paraboloid, we see that in any case $T(v) \cap S$ can be approximated by a quadratic curve. Figures 11c,e,f show the intersection of several such planes with S.

Tangential approximation of a smooth surface. The plane $T(v)$ may be tangent to S (subfigure e), or may be inclined (subfigures c,f), in which case it is parallel to a tangent plane of S not in v, but in another point. If $T(v)$ is the tangent plane in v, an intersection only exists if S has negative curvature. The intersection configuration is

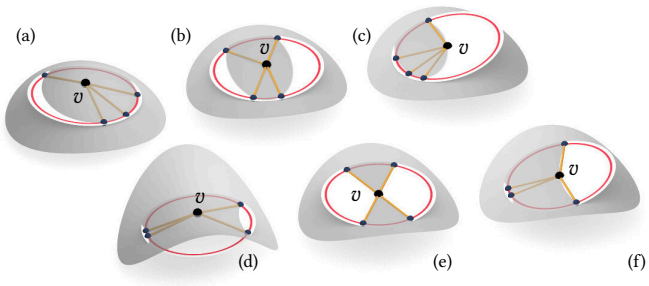


Fig. 11. Vertex configuration of a C-mesh inscribed in a smooth surface S, which can have positive Gaussian curvature (top row) or negative Gaussian curvature (bottom row). Potential vertices can be found by intersecting a certain circle with S. We explore possibilities by positioning that circle in different ways.

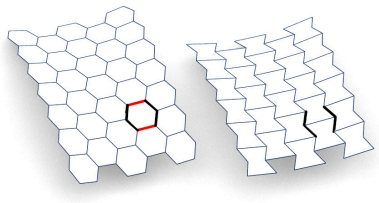


Fig. 12. Left: A hex mesh with planar vertex stars and one family of edges (red) with zero torsion. *Right*: Conversion to a hex mesh with planar nonconvex faces.

symmetric, and thus better corresponds to a quad mesh than a hex mesh. A hex mesh where $T(v)$ is positioned tangentially uses only 3 of the 4 potential edges emanating from v, and will look like the brick wall mesh shown by Fig. 13.

Non-convex faces in case of positive curvature. An interesting fact is shown by Fig. 11c. Here S has positive curvature. Obviously in such a case all intersection points with S lie to one side of a straight line inside $T(v)$. Thus the faces of the C-mesh cannot be convex, the mesh looks like Fig. 12, right. In the positively-curved case, an intersection exists only if the plane $T(v)$ is inclined, so a C-mesh cannot approximate S tangentially.

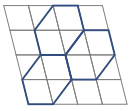
C-meshes collapsing onto circles. In the more general case of C-meshes which collapse onto a circle, the plane $T(v)$ is replaced by a sphere, but the points in the sphere at distance ℓ from the central vertex still lie on a circle. All the previous arguments apply. This case is illustrated by Fig. 11a,b,d. Also here a tangential approximation of S by a C-mesh leads to a symmetric intersection and is not suitable for hex meshes. We conclude that hex meshes (other than the brick wall case) do not approximate smooth surfaces smoothly, they can do so only in a zigzagging way. The brick wall case is still useful for initializing the computation of a hex mesh, see § 4.2.1.

4.2 Design of Hex C-Meshes From Quad C-Meshes

4.2.1 Initializing Hex Meshes from Brick Wall Patterns. By deleting edges in a quadrilateral C-mesh as shown by Fig. 13, we formally convert it to a hexagonal C-mesh. It still enjoys the defining properties of a C-mesh (constant edge length, planar/spherical vertex stars).

Clearly, these special cases may not be those naturally occurring under deployment of a strip structure. We may apply regularization to achieve the expected shape of faces. Furthermore, the quad mesh we started from might not follow our original design surface (quad C-meshes are restricted in their shape). We can therefore use a round of optimization to modify the brick wall mesh to approximate the intended design surface. This automatically cause the shape of faces to change, and in fact the examples in this paper have been produced in this way. There is a catch, however: Hex C-meshes, apart from brick wall meshes, are not capable of approximating smooth surfaces tangentially at all. This phenomenon may or may not be relevant for particular practical applications. It is in detail discussed in § 4.1 above.

4.2.2 Initializing Hex meshes from Diagonals of Quad Meshes. The inset figure shows another method of initializing a hex mesh: By introducing diagonals and deleting all interior edges in a 2×2 block of



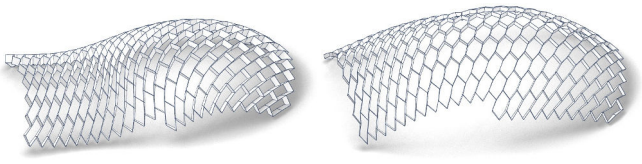


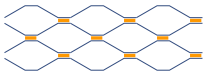
Fig. 13. Hex C-meshes whose vertices lie on a smooth surface and whose vertex planes or spheres are tangent to it, appear as brick wall patterns. We initialize such a brick wall hex mesh from a quad mesh (left), and optimize for more regular faces, or for fitting a different shape (result on the right).

faces, we convert a quad mesh into a hex mesh. It no longer has the C-property, which needs to be restored by optimization. However we take this opportunity to generalize the notion of C-hex mesh somewhat. We allow two different edge lengths according to Fig. 16 and we say that such meshes are C-meshes in the wider sense. It is easy to generalize the conditions expressing the C-property to this case:

Lemma 4.1. *The characterization of C-meshes by Lemma 2.4 remains valid for hexagonal C-meshes in the wider sense (with two different edge lengths according to Fig. 16) if the angle condition is rephrased: For each edge vw, the angle $\gamma(\ell)$ between a normal vector $n(v)$ and the edge vw obeys Equ.* (1)*, where $\ell = \|v - w\|$.*

Accordingly, the optimization has to be changed slightly; the constraint $c_{\text{angle}}(u, v)$ now involves a value $\gamma(\ell)$ that depends on the intended length of the edge uv. Fig. 14 shows a result.

4.2.3 Hex Meshes with Edges of Zero Torsion. A hexagonal strip structure can be fabricated by gluing. Individual strips are folded in a zigzag fashion as shown by the inset figure and glued where indicated by the yellow color. This yields a hexagonal strip structure with one family of thicker strips along which it tends to have less torsion. We account for this in our simplified geometric model by requiring zero torsion for such edges. For such an edge vw we require that the vectors $n(v)$, $n(w)$, and $w - v$ are co-planar. In the optimization, this is achieved by 3 constraints expressing orthogonality of these vectors to an auxiliary unity vector $m(v, w)$, which is added as a variable to the optimization. Fig. 14 shows such a result.



Coplanarity of the two normals at both ends of an edge is a well known discrete version of the property that this edge follows a principal direction. Usually, this is difficult to achieve. However, if we convert a C-quad mesh into hex mesh by inserting diagonals

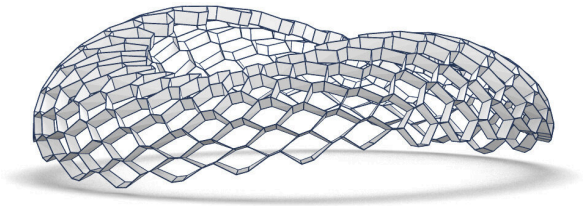


Fig. 14. A hexagonal C-mesh in the wider sense, exhibiting two different edge lengths. After cutting it becomes simply connected and collapses onto a circular strip.

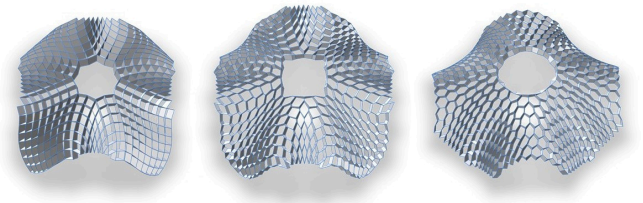


Fig. 15. Hex meshes via edge insertion. Splitting vertices in the quad-C-mesh on the left results in a hex mesh (center). Optimization yields a hex C-mesh. The design surface approximated here is the one of Fig. 17a.

Fig. 16. A hexagonal C-mesh in the wider sense collapsing onto a polyline. It has both short and long edges in an alternating way.

according to § 4.2.2, the newly inserted edges will already approximately follow principal directions, according to Prop. 3.5. Thus we can expect optimization to achieve zero torsion for these edges. Fig. 14 shows an example of this kind.

Remark 4.2. Consider the special case of C-hex meshes with planar vertex stars, where one family of edges has zero torsion (Fig. 12, left). We delete those edges and reconnect the mesh by inserting edges where the planes of vertex stars intersect (Fig. 12, right). This creates a hex mesh with nonconvex planar faces. Such meshes are a well investigated topic (see e.g.[Pluta et al. 2021]), however not under our additional edge length constraints.

4.2.4 Hex Meshes Via Edge Insertion. Splitting all vertices converts a quad mesh to a hex mesh. In the specific case of a C-quad mesh approximating a smooth design surface, Prop. 3.5 states that we can expect the principal directions to bisect the edges. We can therefore easily effect a vertex split where the newly introduced edges are aligned with principal directions. In our examples we made these edges very short, and subsequently optimized towards a C-mesh in the wider sense with two different edge lengths. The two edge lengths which occur in the mesh are variables ℓ_1, ℓ_2 in the optimization. Previously existing edges are required to have length ℓ_1, newly inserted edges have length ℓ_2 (c_{len} has to be modified in the obvious way, with either ℓ_1 or ℓ_2 instead of ℓ). We add the constraint $\ell_1 - \alpha\ell_2 = 0$ ($\alpha > 0$) to our optimization, where α is the intended ratio between the edge lengths.

Figure 15 shows an example. Since newly inserted edges follow principal directions, they can also be optimized towards zero torsion, as explained in § 4.2.3 above. We should mention that already Wunderlich [1973] mentions simple meshes of this kind (which enjoy rotational symmetry).

5 INVERSE DESIGN

5.1 Inverse Design of Quadrilateral C-Meshes

In this section we illustrate a method for the inverse design of quadrilateral C-meshes. It is based on the relation of such meshes with their continuous counterparts (see § 3.4), namely, linear-Weingarten surfaces of hyperbolic type. Briefly, the design pipeline is as follows:

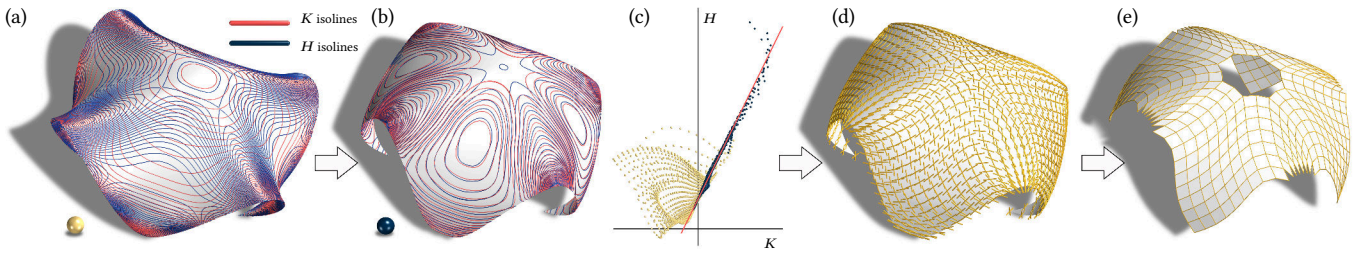


Fig. 17. Inverse design of quadrilateral C-meshes. (a) Given shape with isolines of Gaussian curvature K and mean curvature H. (b) The result of optimization towards a Weingarten surface with a linear relation between H and K. Every H isoline is a K isoline and vice versa. (c) The image of both the original surface (yellow) and the optimized surface (blue) in the H-K-plane. By fitting a straight line to the diagram corresponding to the optimized surface we derive a linear relation between H and K which applies in an approximate way. (d) C-directions are computed at sample points. (e) A quad mesh which follows the C-directions is optimized for equal edge length and for spherical vertex stars of constant radius.

(1) Optimize a design surface to become linear-Weingarten;
(2) Remesh, guided by the cross field of directions representing a Chebyshev net of constant normal curvature;
(3) Optimize for the C-mesh property.

5.1.1 Design of a Hyperbolic Linear Weingarten Surface. The optimization of surfaces towards the Weingarten property is treated by Pellis et al. [2021]. Their method allows to impose further properties like *Linear-Weingarten*. It works with B-spline surfaces and optimizes the control mesh of that B-spline such that it approximately fulfills a generic or a specific relation between the Gaussian curvature K and the mean curvature H at sample points. It is based on a standard Levenberg–Marquardt algorithm with quadratic constraints and quadratic target functions – for details we refer to [Pellis et al. 2021]. For our application, we enforce a linear relation $aK + bH + c = 0$, with a normalization constraint $a^2 + b^2 + c^2 = 1$. To achieve *hyperbolic type*, we use the constraint $b^2 - 4ac = -\delta^2 + \delta_{\max}$, where δ is an auxiliary variable, and $\delta_{\max}$ is a negative threshold. Figures 17a–c illustrate this procedure. Subfigures 17a,b show the original surface S and the optimized surface M' and how the isolines of H and K on M' align. Fig. 17c shows the image of both M and M' in the H-K-plane, by plotting pairs (H, K) for a dense sampling of the surface. The image of M' concentrates along a straight line. By performing PCA on this 2D point cloud we compute an ideal relation of the form $aK + bH + c = 0$ which is approximately fulfilled by the optimized surface M'.

5.1.2 Quadrilateral Remeshing. In any point of the surface produced by § 5.1.1 we can compute the principal directions as well as Gaussian curvature K and mean curvature H. Equation (8) computes the direction of edges of a quadrilateral C-mesh which follows this surface (we call them C-directions). We found that we achieve better results if we use these formulae only after we modify the values H, K by closest point projection onto the ideal linear-Weingarten line $aK + bH + c = 0$. Having thus achieved a cross field of C-directions in a dense set of points (Fig. 17d), we convert to a triangle mesh and apply the integer quadrangulation method of [Bommes et al. 2009] to derive a quad mesh.

We cannot apply integer quadrangulation directly, since that method requires an *orthogonal cross-field* as input. In order to make the C-directions orthogonal, we apply a deformation that makes these directions orthogonal. We represent the given surface as a triangle mesh. For each face $v_0v_1v_2$, we express the C-directions as a linear combination $\alpha_i(v_1 - v_0) + \beta_i(v_2 - v_0)$, $i = 1, 2$. We then move vertices from original positions v_i^0 to new positions v_i, such that C-directions become orthogonal, i.e.,

$$c_{\text{ortho}}(f) = \langle \alpha_1(v_1 - v_0) + \beta_1(v_2 - v_0), \alpha_2(v_1 - v_0) + \beta_2(v_2 - v_0) \rangle$$

vanishes. We thus minimize

$$\sum_{f \in F} c_{\text{ortho}}(f)^2 + \epsilon_1 \sum_{v \in V} \|v_i - v_i^0\|^2 + \epsilon_2 \sum_{\substack{\text{interior} \\ \text{vertices}}} \|\Delta(v)\|^2,$$

using a Levenberg-Marquardt algorithm. The terms multiplied with small factors ϵ_i are for regularization; $\Delta(v)$ means the umbrella vector of [Kobbelt et al. 1998]. We then apply integer quadrangulation to the deformed mesh, and map the result back to the original surface (using barycentric coordinates). This quad mesh finally is optimized for the C-mesh property (Fig. 17e).

Remark 5.1. In this paper we impose constant edge length on quadrilateral C-meshes. It is possible to generalize and loosen this requirement, even if Fig. 5 shows that edge lengths must obey many conditions. We expect that such a generalized C-mesh, assuming fairness, still discretizes a linear-Weingarten surface, but parametrized in a more general way. This would mean that the design space essentially is not restricted by the constant edge length requirement.

5.2 Inverse Design of Hexagonal C-Meshes

We do not claim to solve the problem of inverse design of hexagonal C-meshes in general. Basically we compute quadrilateral C-meshes first, which involves a deviation from the original shape. Then we apply one of the methods described by § 4.2 to derive a hex mesh from it. Subsequent optimization for the C-mesh property involves proximity to the original surface. The additional degrees of freedom which are available for hex meshes mean that this procedure usually succeeds. Figure 15 shows one example created by adding diagonals. Another example is shown by Fig. 23: A quad mesh, the result of inverse design, is projected back towards the original design surface which it was unable to approximate. A vertex split and optimization according to § 4.2.4 yield a hexagonal C-mesh.

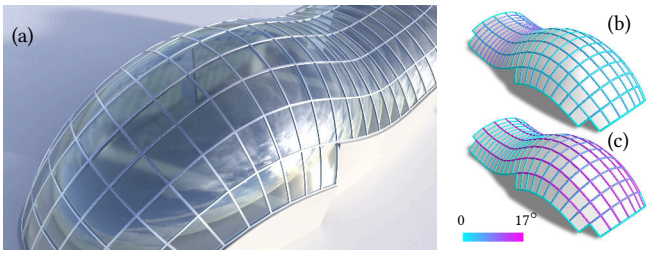


Fig. 18. Paneling with spherical panels. A diagonal net of a C-mesh has planar faces and allows paneling not only with planar panels (c), but even with spherical panels of constant radius (a,b). The overall smoothness of such a paneling is shown by visualizing angles between panels in subfigures b, c for spherical resp. planar panels. For each panel the maximum angle occurring at the boundary seems to be smaller in the spherical case. Individual angles do not have to be smaller, especially in negatively curved areas. In positively curved areas, spherical panels lead to a much smoother surface than planar ones.

6 RESULTS

This section presents a result which is not directly connected with the deployment of our strip structures, but nevertheless we feel is important for freeform architecture. It concerns the possibility of paneling quadrilateral C-meshes with spherical panels. The rest of this section discusses generalizations (combinatorial singularities), and the verification of our theory by means of physical models. It concludes with limitations and proposals for future research.

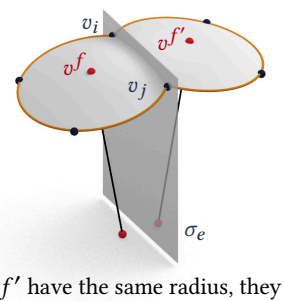
6.1 Paneling with spheres

Quadrilateral C-meshes have an interesting application for architectural skins, namely one with spherical panels of constant radius, and a torsion-free support structure associated with the edges. The construction is as follows:

Proposition 6.1. *Consider a quadrilateral C-mesh deploying from a circular collapsed state, and a "diagonal mesh" D of M, i.e., a quad mesh formed by face diagonals of M. A face f of D is formed by the immediate neighbours of a central vertex $v^f \in M$. The sphere associated with v^f is labelled $S(f)$.*

Then D is equipped with a torsion-free support structure compatible with its sphere $S(f)$. I.e., each vertex v_i of D has a "normal" N_i, such that each edge $e = v_iv_j$ of D together with the normals N_i, N_j lie in a common plane σ_e. Further, if an edge $e = f \cap f'$ is the intersection of faces f, f', spheres $S(f), S(f')$ intersect precisely in σ_e.

Proof. The vertices of a face f of D are contained in the sphere $S(f)$. They have constant distance from $v^f \in M$, so f has a circumcircle (which is contained in $S(f)$). Consider two faces f, f' sharing an edge $e = v_iv_j$. The configuration of spheres $S(f), S(f')$ is symmetric w.r.t. the symmetry plane of sphere centers, which we take as the plane σ_e. Since circumcircles of f, f' have the same radius, they too enjoy symmetry w.r.t. σ_e. All spheres have the same size, so $S(f) \cap S(f') \subset \sigma_e$.



Now consider a vertex $v_0 \in D$, and the edges and faces of D incident with it. Each of the four edge planes σ_{e_i} are symmetry planes of two sphere centers. We wish to show that they contain a common line N_0; this would conclude the proof.

Consider M', the mesh of sphere centers of M. It is an offset of M since spheres have constant radius. By § 3.2, M' is a C-mesh. Those sphere centers corresponding to faces of D form a diagonal mesh $\tilde{D}$ of M'. Consider specifically the sphere centers associated with faces incident with v_0. They form a face $\tilde{f}$ of $\tilde{D}$, which has a circumcircle ($\tilde{D}$ being a diagonal mesh of a C-mesh). A plane σ_{e_i} bisects two vertices of $\tilde{D}$, so σ_{e_i} contains the axis "N_0" of the aforementioned circumcircle. □

It follows that the planes of the torsion-free support structure serve as natural boundaries of spherical panels associated with faces. For an actual application in architecture, it is necessary to be able to produce spherical panels. Fortunately, this is possible, and in fact they are already in use, see Fig. 19. Figure 18 shows spherical panelings of the diagonal mesh D of a quadrilateral C-mesh illustrating Prop. 6.1.

Fig. 19. Spherical glass panels are used by one of the largest spherical buildings in the world, the *Nur Alem* sphere in Astana, Kazakhstan (photo Saliha Kanaeva).

Remark 6.2. A torsion-free support structure with bisecting planes as described by Prop. 6.1 implies that D is not only a circular mesh, but at the same time a conical mesh. This fact might contribute to an eventual classification of meshes which are both circular and conical. We should also emphasize that all circular meshes have panelings with spherical panels – one can easily fit spheres on the circles associated with each face. However the intersections of neighbouring spheres will in general not be related to any torsion-free support structure.

6.2 Further Results and Discussion

C-meshes With Combinatorial Singularities. We established that the edges of a quad-C-mesh correspond to a uniquely defined curve network on a curvature-continuous smooth surface, more precisely an offset of the asymptotic curves on a K-surface. The combinatorial singularities of this curve network are well known: There are none, since negative Gauss curvature implies a local saddle shape, and absence of umbilics. However, if we relax our smoothness assumptions to tangent continuity, we may paste individual curvature-continuous K-surface pieces together. This topic has been studied in depth in differential geometry, e.g. [Shearman and Venkataramani 2021], and in fact such surfaces seem to occur even in nature, e.g. on curly leaves. On the discrete side, it is no problem to stitch individual regular quad C-meshes together and obtain results as shown by Fig. 21. Edge polylines have a distinct change in direction

Fig. 20. Deployment of a quadrilateral C-mesh from a circular polyline, and an actual paper model of this mesh made from lasercut strips assembled manually. For an animation of the continuous deployment we refer to the accompanying video.

across seams (as would the aforementioned curve network in the continuous case).

The inverse design pipeline for quadrilateral C-meshes is based on curvature information, and in fact a C-mesh that is to approximate a given shape is guided by the network of principal curves. So in general there will be combinatorial singularities (see Figures 17 and 25). The previous paragraph shows that the approximation by C-meshes cannot involve approximation of curvatures also. Meshes which are not simply connected or which have combinatorial singularities usually cannot deploy from a single flat state; this is possibly only after cutting has removed topological obstacles.

Experimental Verification of the C-mesh Property. We experimentally tested the approximation capabilities of C-meshes by machining a design surface and checking if a strip structure made from paper can be made to deploy onto that surface, see Fig. 20.

For verifying the C-mesh property itself, we built several deployable structures and performed measurements. Figure 22 shows both a quad mesh and a hex mesh we used for this purpose. The white boundary, which is an offset of the central mesh, has been laser scanned and measured. In the quad case we checked that boundary, since offsets of quadrilateral C-meshes are C-meshes themselves. With the notation ℓ for the average edge lengths, the standard deviation of edge lengths in the quad case is 0.009ℓ. Sphericity of a vertex star is measured by its diameter in direction of the normal; we get standard deviation 0.006ℓ. These numbers confirm the C-mesh property to a satisfactory extent.

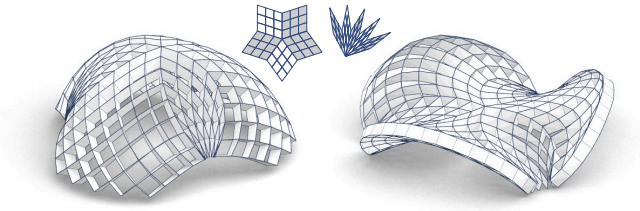


Fig. 21. These two examples of C-meshes with a combinatorial singularity correspond to surfaces which are tangent-continuous and piece-wise a K-surface (or offset of a K-surface). A combinatorial singularity never corresponds to a single piece of curvature-continuous surface. The small symbolic figure shows how such a mesh can deploy from a flat state after cutting it open.

The hex mesh of Fig. 22, left, has been fabricated by gluing strips as shown in § 4.2.3, so edges are unequal. Consequently the standard deviation of edge lengths of the reconstructed central mesh has the larger value 0.035ℓ. Sphericity of a vertex star is measured by its diameter in direction of the normal; we get a standard deviation of 0.005ℓ. These numbers confirm the C-mesh property to a lesser extent than in the quad case.

Relations to Geometric Metamaterials. Meshes deploying from a collapsed flat state have potential applications in situations where transport and mounting is an issue, even if the mesh under consideration no longer undergoes flexion in its final state. The principle is similar to the deployment of geometric metamaterials, e.g. the programmable auxetics of Konakovic et al. [2018]. As an example, Fig. 23 shows a sandwich structure.

Statistics. Table 1 illustrates run times for examples of inverse design according to the pipeline of Fig. 17 which involves the optimization of a surface towards the hyperbolic linear Weingarten property first. Computation times refer to a Python implementation which has been tested on a Intel® Core™i7-10750H CPU with 32 GB RAM. Computation times for further examples are given by Table 2. These were treated by a C++ implementation running on an Intel® Xeon® CPU E5-2699 v3 (2.30GHz, 256 GB RAM). Levenberg-Marquardt optimization uses adaptive regularization weights and stepsize control. Depending on the quality of the initialization and the complexity of the example, we generally used 20–50 iterations.

Limitations. An obvious limitation of quadrilateral C-meshes is that they cannot represent arbitrary shapes. A limitation of our

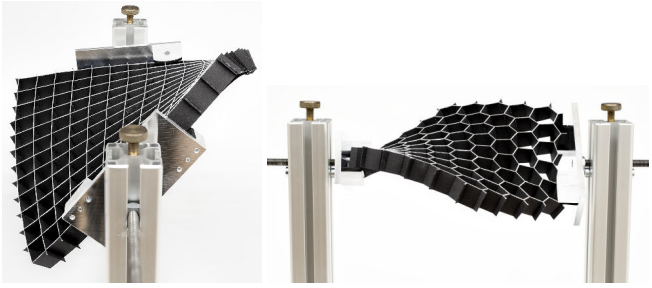


Fig. 22. Verification of the C-mesh property by means of scanning physical models. Edge lengths are 20 mm (quad case) resp. 14 mm (hex case).

Table 1. Optimization of C-meshes, Part 1. T_{prep} refers to the time spent for approximating the given design surface by a linear-Weingarten surface of hyperbolic type by the method of [Pellis et al. 2021]. The table further gives the number of vertices of the meshes involved, the number of variables in the optimization, the number of constraints, and the weights employed by the energies defined in § 2.1.

Fig.	$\lvert V\rvert$	# var.	# const.	α_1	α_2	α_3	α_{fair}	α_{pos}	α_{prox}	T_{prep}	T [s]	# it.
1	725	4352	5575	3	20	20	.8	.1	.2	17.5	1.03	14
17	464	2786	3496	5	50	50	.8	.1	.2	14.1	0.68	14
20	431	2588	3268	3	20	20	.8	.2	.4	25.5	0.59	12
23l	431	2396	3072	5	50	50	.5	.1	.1	17.9	0.85	18
25	629	3776	4792	3	80	80	.5	.05	1	36.5	1.42	20

Table 2. Optimization of C-meshes, Part 2. The table gives the number of vertices, the number of variables in the optimization, and the weights of energies defined in § 2.1. For inverse design examples, fairness and proximity weights are adjusted to become zero as optimization progresses. The adjustment rate proved difficult to automatize, so it was performed interactively by the user.

Fig.	$\lvert V\rvert$	# var	α_1	α_2	α_3	α_{fair}	α_{prox}	α_{twist}	α_{stiff}	α_{pos}	t [s]
9r	474	2844	1	1	1	0	0	0	0	0	.06
9l	794	4764	1	1	1	0	0	0	0	0	.10
10r	230	1380	1	1	1	0	0	0	0	0	.02
10l	230	1380	1	1	1	0	0	0	0	0	.02
13r	405	2430	1	1	1	.01→0	.01→0	0	0	0	.04
14	500	3000	1	1	1	0	0	1	0	0	.07
15r	828	4968	1	1	1	.01→0	.01→0	0	0	0	.12
20	431	2586	100	100	100	0	0	.1	.01	0	.05
26	804	4824	1	1	1	0	0	0	0	0	.11

paper is our focus on the theory and design of quadrilateral C-meshes; §4.1 suggests that a theory on hexagonal C-meshes will be more difficult to establish.

Future Research. There are plenty of potential directions for future research. Properties of materials have been neglected so far. More work is needed to fully understand the shape space of hex structures, and to answer geometric questions (e.g. if more general collapsed states provide additional degrees of freedom). A further rewarding topic are meshes with spherical faces and circular edges in general, and their applications in architecture. Here additional fabrication-related constraints are important, like a small number of sphere radii, or torsion-free support structures.

ACKNOWLEDGMENTS

The authors wish to thank the anonymous reviewers for their helpful comments. D. Liu, Y.-C. Chiang, and F. Rist have been supported by KAUST baseline funding (grant BAS/1/1679-01-01).

REFERENCES

Changyeob Baek, Andrew O. Sageman-Furnas, Mohammad K. Jawed, and Pedro M. Reis. 2018. Form finding in elastic gridshells. *PNAS* 115, 1 (2018), 75–80.
Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky, and Eitan Grinspun. 2010. Discrete viscous threads. *ACM Trans. Graphics* 29, 4 (2010), 116:1–10.
Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly, and Eitan Grinspun. 2008. Discrete elastic rods. *ACM Trans. Graphics)* 27, 3 (2008), 63:1–12.
Alexander Bobenko and Yuri Suris. 2008. *Discrete differential geometry: Integrable Structure.* American Math. Soc.

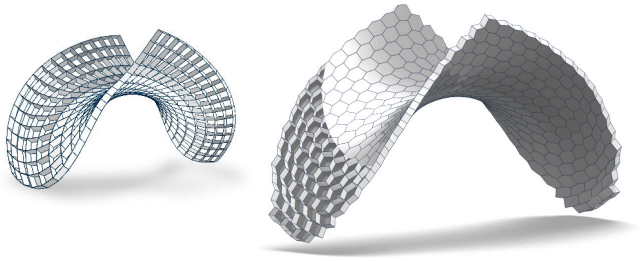


Fig. 23. Inverse Design. The C-quad mesh on the left is the result of the inverse design pipeline described by § 5.1. The hex mesh on the right has been created from it by edge insertion and re-optimization. Here it serves as core of a sandwich structure.

David Bommes, Henrik Zimmer, and Leif Kobbelt. 2009. Mixed-integer quadrangulation. *ACM Trans. Graphics* 28, 3 (2009), 77:1–10.
Raphaël Charrondière, Florence Bertails-Descoubes, Sébastien Neukirch, and Victor Romero. 2020. Numerical modeling of inextensible elastic ribbons with curvature-based elements. *Computer Methods in Applied Mech. & Engrg.* 364, Article 112922 (2020), 24 pages.
Tian Chen, Julian Panetta, Max Schnaubelt, and Mark Pauly. 2021. Bistable auxetic surface structures. *ACM Trans. Graph* 40, 4 (2021), 39:1–9.
Bernardino D'Amico, Abdy Kermani, Hexin Zhang, Alberto Pugnale, Sofia Colabella, and Sergio Pone. 2015. Timber gridshells: Numerical simulation, design and construction of a full scale structure. *Structures* 3 (2015), 227–235.
Crispin Deul, Tassilo Kugelstadt, Marcel Weiler, and Jan Bender. 2018. Direct position-based solver for stiff rods. *Computer Graphics Forum* 37, 6 (2018), 313–324.
Manfredo do Carmo. 1976. *Differential Geometry of Curves and Surfaces.* Prentice-Hall.
Olly Duncan, Todd Shepherd, Charlotte Moroney, Leon Foster, Praburaj Venkatraman, Keith Winwood, Tom Allen, and Andrew Alderson. 2018. Review of Auxetic Materials for Sports Applications: Expanding Options in Comfort and Protection. *Applied Sciences* 8, 6, Article 941 (2018), 33 pages.
Roger Fosdick and Eliot Fried (Eds.). 2016. *The mechanics of ribbons and Möbius bands.* Springer.
Akash Garg, Andrew O. Sageman-Furnas, Bailin Deng, Yonghao Yue, Eitan Grinspun, Mark Pauly, and Max Wardetzky. 2014. Wire mesh design. *ACM Trans. Graph.* 33, 4 (2014), 66:1–12.
Christian Hafner and Bernd Bickel. 2021. The design space of plane elastic curves. *ACM Trans. Graph.* 40, 4 (2021), 126:1–20.
Charles Hoberman. 1990. Reversibly expandable doubly-curved truss structure. U.S. Patent 4942400.
Yaoye Hong, Yinding Chi, Shuang Wu, Yanbin Li, Yong Zhu, and Jie Yin. 2022. Boundary curvature guided programmable shape-morphing kirigami sheets. *Nature communications* 13, 1 (2022), 530.
M. Khalid Jawed, Alyssa Novelia, and Oliver M O'Reilly. 2018. *A Primer on the Kinematics of Discrete Elastic Rods.* Springer.
Caigui Jiang, Chengcheng Tang, Amir Vaxman, Peter Wonka, and Helmut Pottmann. 2015. Polyhedral patterns. *ACM Trans. Graph.* 34, 6 (2015), 172:1–12.
Leif Kobbelt, Swen Campagna, Jens Vorsatz, and Hans-Peter Seidel. 1998. Interactive multi-resolution modeling on arbitrary meshes. In *Proc. SIGGRAPH.* 105–114.
Mina Konaković-Luković, Julian Panetta, Keenan Crane, and Mark Pauly. 2018. Rapid deployment of curved surfaces via programmable auxetics. *ACM Trans. Graph.* 37, 4 (2018), 106:1–13.
Luigi Malomo, Jesús Pérez, Emmanuel Iarussi, Nico Pietroni, Eder Miguel, Paolo Cignoni, and Bernd Bickel. 2018. FlexMaps: Computational design of flat flexible shells for shaping 3D objects. *ACM Tran. Graph.* 37, 6 (2018), 214:1–14.
Saurabh Mhatre, Elisa Boatti, David Melancon, Ahmad Zareei, Maxime Dupont, Martin Bechthold, and Katia Bertoldi. 2021. Deployable structures based on buckling of curved beams upon a rotational input. *Adv. Funct. Mater.* 31, Article 2170261 (2021), 7 pages.
David Mount and Sunil Arya. 2010. ANN Library (Version 1.1.2). http://www.cs.umd.edu/~mount/ANN/
Jorge Nocedal and Stephen Wright. 2006. *Numerical Optimization* (2nd ed.). Springer.
Joseph Reuben Harry Otter, Alfred Carlo Cassell, and Roger Edwin Hobbs. 1966. Dynamic relaxation. *Proc. Institution of Civil Engineers* 35, 4 (1966), 633–656.
Eda Özdemir, Laura Kiesewetter, Karen Antorveza, Tiffany Cheng, Samuel Leder, Dylan Wood, and Achim Menges. 2022. Towards Self-shaping Metamaterial Shells:. In *Proceedings of the 2021 DigitalFUTURES.* Springer, 275–285.
Dinesh K Pai. 2002. Strands: interactive simulation of thin solids using Cosserat models. *Computer Graphics Forum* 21, 3 (2002), 347–352.

Julian Panetta, Florin Isvoranu, Tian Chen, Emmanuel Siéfert, Benoît Roman, and Mark Pauly. 2021. Computational inverse design of surface-based inflatables. *ACM Trans. Graph.* 40, 4 (2021), 40:1–14.
Julian Panetta, Mina Konaković-Luković, Florin Isvoranu, Etienne Bouleau, and Mark Pauly. 2019. X-Shells: A New Class of Deployable Beam Structures. *ACM Trans. Graph.* 38, 4 (2019), 83:1–15.
Davide Pellis, Martin Kilian, Helmut Pottmann, and Mark Pauly. 2021. Computational design of Weingarten surfaces. *ACM Trans. Graph.* 40, 4 (2021), 114:1–11.
Davide Pellis, Hui Wang, Martin Kilian, Florian Rist, Helmut Pottmann, and Christian Müller. 2020. Principal symmetric meshes. *ACM Trans. Graph.* 39, 4 (2020), 127:1–17.
Jesús Pérez, Bernhard Thomaszewski, Stelian Coros, Bernd Bickel, José A. Canabal, Robert Sumner, and Miguel A. Otaduy. 2015. Design and fabrication of flexible rod meshes. *ACM Trans. Graph.* 34, 4 (2015), 138:1–12.
Stefan Pillwein, Kurt Leimer, Michael Birsak, and Przemyslaw Musialski. 2020. On Elastic Geodesic Grids and Their Planar to Spatial Deployment. *ACM Trans. Graph.* 39, 4 (2020), 125:1–12.
Stefan Pillwein and Przemyslaw Musialski. 2021. Generalized Deployable Elastic Geodesic Grids. *ACM Trans. Graph.* 40, 6 (2021), 271:1–15.
Kacper Pluta, Michal Edelstein, Amir Vaxman, and Mirela Ben-Chen. 2021. PH-CPF: Planar Hexagonal Meshing Using Coordinate Power Fields. *ACM Trans. Graph.* 40, 4 (2021), 156:1–19.
Yingying Ren, Uday Kusupati, Julian Panetta, Florin Isvoranu, Davide Pellis, Tian Chen, and Mark Pauly. 2022. Umbrella meshes: elastic mechanisms for freeform shape deployment. *ACM Trans. Graph.* 41, 4 (2022), 152:1–15.
Yingying Ren, Julian Panetta, Tian Chen, Florin Isvoranu, Samuel Poincloux, Christopher Brandt, Alison Martin, and Mark Pauly. 2021. 3D weaving with curved ribbons. *ACM Trans. Graph.* 40, 4 (2021), 127:1–15.
Andrew O. Sageman-Furnas, Albert Chern, Mirela Ben-Chen, and Amir Vaxman. 2019. Chebyshev nets from commuting PolyVector fields. *ACM Trans. Graph.* 38, 6 (2019), 172:1–16.
Robert Sauer. 1970. *Differenzengeometrie.* Springer.
Jonas Schikore, Eike Schling, Thomas Oberbichler, and Anna Bauer. 2021. Kinetics and design of semi-compliant grid mechanisms. In *Adv. in Architectural Geometry 2020.* Presses des Ponts, 108–129.
Eike Schling. 2018. *Repetitive Structures – Design and construction of curved support structures with repetitive parameters.* Ph. D. Dissertation. TU Munich.
Eike Schling and Jonas Schikore. 2022. Morphology of kinetic asymptotic grids. In *Towards Radical Regeneration.* Springer, 374–393. Proc. Design Modelling Symposium.
Eike Schling, Hui Wang, Sebastian Hoyer, and Helmut Pottmann. 2022. Designing asymptotic geodesic hybrid gridshells. *Computer-Aided Design* 152, Article 103378 (2022), 17 pages.
Toby L Shearman and Shankar C Venkataramani. 2021. Distributed branch points and the shape of elastic surfaces with constant negative curvature. *J. Nonlinear Science* 31, 1, Article 13 (2021), 60 pages.
Carlota Soler, Tobias Martin, and Olga Sorkine-Hornung. 2018. Cosserat rods with projective dynamics. *Computer Graphics Forum* 37, 8 (2018), 137–147.
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), 70:1–9.
Xavier Tellier. 2022. Bundling elastic gridshells with alignable nets. Part I: Analytical approach. Part II: Form-finding. *Automation in Construction* 141 (2022). Articles 104291, 104292, 19+13pp..
Xavier Tellier, Cyril Douthe, Laurent Hauswirth, and Olivier Baverel. 2020. Caravel meshes: A new geometrical strategy to rationalize curved envelopes. *Structures* 28 (2020), 1210–1228.
Nobuyuki Umetani, Ryan Schmidt, and Jos Stam. 2015. Position-based elastic rods. In *Proc. SCA'14.* Eurographics Association, 21–30.
Josh Vekhter, Jiacheng Zhuo, Luisa F. Gil Fandino, Qixing Huang, and Etienne Vouga. 2019. Weaving geodesic foliations. *ACM Trans. Graph.* 38, 4 (2019), 34:1–12.
Walter Wunderlich. 1951. Zur Differenzengeometrie der Flächen konstanter negativer Krümmung. *Sitzungsber. Österr. Ak. Wiss. II* 160 (1951), 39–77.
Walter Wunderlich. 1973. Drehsymmetrische Gleichgewichtsformen von Rhomben- und Sechsecknetzen. *Zeitschrift Angew. Math. Mechanik* 53 (1973), 593–600.

APPENDIX

Lemma A.1. *Consider nonparallel straight lines N_0, N_1 and the points $w_i \in N_i$ which realize the shortest distance between N_0 and N_1. If the straight line L intersects N_0, N_1 under the same angle γ at points v_0, v_1, then distances $\|v_0 - w_0\|$ and $\|v_1 - w_1\|$ are equal.*

Proof. For a geometric proof, we parallel translate lines L, N_0, N_1 through the origin, which yields lines $\widetilde{L}$, $\widetilde{N}_0$, $\widetilde{N}_1$. The line $\widetilde{L}$ is contained in the cone of all lines enclosing the angle γ with the axis $\widetilde{N}_0$, and in a similar cone with axis $\widetilde{N}_1$. We visualize this situation in Fig. 24. The two subfigures on the left view the situation in direction of the segment w_0w_1, which is orthogonal to both N_0, N_1. The two cones being congruent, $\widetilde{L}$ is contained in a plane bisecting $\widetilde{N}_0, \widetilde{N}_1$. The bisector property is visible in the projection and directly implies that the two distances of L from w_i, firstly measured along N_0, secondly measured along N_1, are the same. □

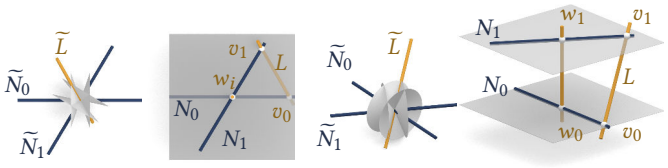
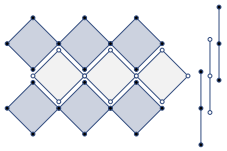
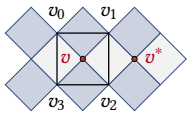


Fig. 24. Proof of Lemma A.1.

Proof of Theorem 3.8. The idea of the proof is to show that the ability to collapse onto a curve imposes so many conditions that only the special meshes described in the theorem can do it. The proof is divided into 6 parts. In Parts (1–3) we construct a certain arrangement of circles, in Step (4) we show it must be symmetric, and in Step (6) we draw conclusions about the shape of the mesh. However, this chain of arguments does not work if the curves we collapse onto are special (namely, circles and straight lines). This case is treated in Step (5).



(1) *Decomposition Into C-Meshes, Definition of Normals.* Recall that the generalized C-mesh M collapses onto a polyline. Since a short polyline with only 3 vertices always lies on a circle, the given mesh is the union of diagonal C-mesh strips which collapse onto just 3 vertices and therefore are only 1 face wide, see inset figure. Any vertex star is contained in one such C-mesh, and so every vertex has a well defined sphere and normal.



(2) *Diagonal Quads are Circular.* Consider a vertex v and its star $v_0v_1v_2v_3$. Recall that the normal in the central vertex is contained in the symmetry plane of neighbours v_i, v_{i+1} (indices modulo 4). This symmetry is shown by Fig. 7 – please note however that labels $v_0, \dots, v_3$ in Fig. 7 denote the vertices of a face, while here, $v_0, \dots, v_3$ denote the 1-ring of a central vertex v.

Neighbours v_i have constant distance from v and lie on the sphere of v, so the quad $v_0v_1v_2v_3$ is circular, i.e., has a circumcircle C.

(3) *Construction of a Parallel Quad From Normals.* The stars of two successive vertices, say v_1, v_2, have two vertices in common (v and v^* in inset figure), so both normals N_1, N_2 lie in the symmetry plane of v, v^*; they are *co-planar*. This allows us to perform the following construction: Parallel translate normals through the origin and intersect them with a plane parallel to $\mathrm{span}(v_0, \dots, v_3)$. This yields a quad $\bar{v}_0\bar{v}_1\bar{v}_2\bar{v}_3$ whose edges are parallel to the corresponding edges of the original quad. The circular property depends on angles only, so also $\bar{v}_0\bar{v}_1\bar{v}_2\bar{v}_3$ has a circumcircle $\bar{C}$.

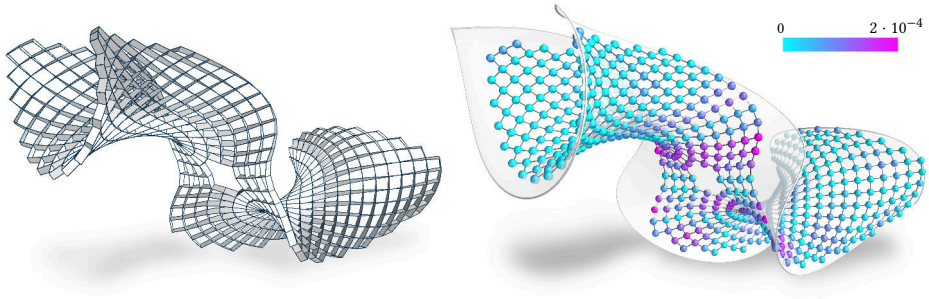
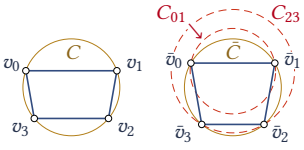


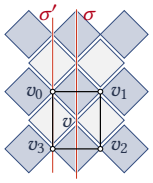
Fig. 25. A result of inverse design of quad C-meshes according to the method of § 5 and Fig. 17. *Left:* Final strip structure. *Right:* the C-mesh which serves as its central mesh, and the linear-Weingarten surface it has been derived from. The deviation of the mesh from the surface is shown as a multiple of the bounding box diagonal.



Normals N_0, N_1 belong to opposite vertices in a face of a C-mesh, so they are in the symmetric position illustrated by Fig. 7. In particular, v_0, v_1 have the same distance from the intersection point $N_0 \cap N_1$. After parallel translation this symmetry is expressed as $\|\bar{v}_0\| = \|\bar{v}_1\|$. Similarly, $\|\bar{v}_2\| = \|\bar{v}_3\|$. The inset figure shows the corresponding level sets of the function $\|x\|$ in the plane of the quad. They are concentric circles $C_{01} \ni \bar{v}_0, \bar{v}_1$ and $C_{23} \ni \bar{v}_2, \bar{w}_3$.

(4) *Diagonal Quads are Trapezoids.* Consider the reflection in the symmetry plane σ of $\bar{v}_0$, $\bar{v}_1$. It leaves circles $\bar{C}$ and C_{01} invariant, and also C_{23} because it has the same center as C_{01}. If $C_{01} \neq C_{23}$ we can reconstruct vertices from circles. Thus not only the circle arrangement is symmetric, but also the quad $\bar{v}_0\bar{v}_1\bar{v}_2\bar{v}_3$. A regularity assumption similar to Th. 3.1 excludes the nonconvex case. We conclude that $\bar{v}_0\bar{v}_1\bar{v}_2\bar{v}_3$ is a trapezoid. By parallelity, so is $v_0v_1v_2v_3$.

(5) *The Special Case of C-Meshes.* If the mesh under consideration is actually a C-mesh, then also the normals N_1, N_2 correspond to opposite vertices of a face in a C-mesh, and so are in a symmetric position. This implies $\|\bar{v}_1\| = \|\bar{v}_2\|$, and circles C_{01}, C_{23} coincide. We can no longer reconstruct vertices from the circles present, and we cannot conclude that $v_0v_1v_2v_3$ is a trapezoid.



(6) *Global Shape of Generalized C-Meshes.* Above we showed that the star $v_0v_1v_2v_3$ of any vertex v is symmetric w.r.t. the reflection in a plane σ which maps $v_0 \longleftrightarrow v_1$. That reflection is already defined by two vertices, so σ is the symmetry plane of v_2, v_3 as well as the symmetry plane of vertices v_0, v_1. Thus, symmetry w.r.t. σ extends to neighbours, and in turn extends to an entire column of trapezoids. By symmetry, $v \in \sigma$, and in fact an entire column of vertices lies in σ. Repeating the argument for a neighbour of v we get a similar reflective symmetry w.r.t. a plane σ' which again contains a column of vertices. Since that column is sill symmetric w.r.t. σ, the symmetry plane of the column positioned on the other side of σ is found by reflecting σ' in σ. The columns involved in this argument are now symmetric w.r.t. the composition of reflections in σ, σ': If σ, σ' intersect in a line, this composition is a rotation, otherwise a parallel translation. This local rotational/translational symmetry propagates through the mesh, which concludes the proof. □

Remark A.2. The proof shows that already a profile composed of only two circular segments leads to these special surfaces. The argument in the proof is applied to a vertex where the two segments join. The rotational symmetry established locally propagates through the entire mesh.

Proof of Equations (8) *and* (7). We follow the terminology of [do Carmo 1976] and switch to a *principal* parametrization $w(u, v)$ of W, so that principal curvatures $\kappa_i(u, v)$ obey $n_u = -\kappa_1 w_u$, $n_v = -\kappa_2 w_v$ and both the first and second fundamental forms I, II are diagonal matrices. Without loss of generality, in the point of interest $\mathrm{I} = \mathrm{diag}(1, 1)$, so $\mathrm{II} = \mathrm{diag}(\kappa_1, \kappa_2)$. Offsetting yields the parametrization $w'(u, v) = w(u, v) + dn(u, v)$, with

$$w'_u = (1 - d\kappa_1)w_u, \quad w'_v = (1 - d\kappa_2)w_v.$$

It follows that in the point of interest, the 1st and 2nd fundamental forms of the offset surface are given by $\mathrm{I}' = \mathrm{diag}((1 - d\kappa_1)^2, (1 - d\kappa_2)^2)$, $\mathrm{II}' = \mathrm{diag}(\kappa_1(1 - d\kappa_1), \kappa_2(1 - d\kappa_2))$.

A direction $(\dot{u}, \dot{v})$ defines a vector $\dot{u}w_u + \dot{v}w_v$ tangent to the surface W and a corresponding vector $\dot{u}w'_u + \dot{v}w'_v$ tangent to W'. The normal curvature associated with this direction is

$$\kappa_n(\dot{u}, \dot{v}) = \frac{(\dot{u}, \dot{v})\,\mathrm{II}\,(\dot{u}, \dot{v})^T}{(\dot{u}, \dot{v})\,\mathrm{I}\,(\dot{u}, \dot{v})^T}, \tag{9}$$

in the surface W. An analogous expression involving $\mathrm{I}', \mathrm{II}'$ expresses the normal curvature $\kappa'_n(\dot{u}, \dot{v})$ in the offset surface W'.

To show (8), we recall that the curves in question are asymptotic curves in W' which have zero normal curvature:

$$\kappa'_n(\dot{u}, \dot{v}) = 0 \iff \kappa_1(1 - d\kappa_1)\dot{u}^2 = -\kappa_2(1 - d\kappa_2)\dot{v}^2. \tag{10}$$

This yields the direction $(\dot{u}, \dot{v})$ up to sign. Since we chose $\mathrm{I} = \mathrm{diag}(1, 1)$, we can compute angles between tangent vectors directly in the $(\dot{u}, \dot{v})$-plane. The angle α enclosed by $(\dot{u}, \dot{v})$ and the 1st principal direction $(1, 0)$ obeys $\tan\alpha = \dot{v}/\dot{u}$, so (8) follows directly from (10). As to Equation (7), the value κ_n is computed by inserting $(\dot{u}, \dot{v})$ in (9), and some elementary manipulations. □

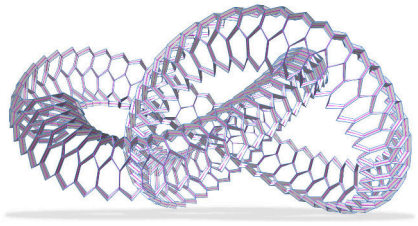


Fig. 26. A hexagonal C-mesh in the shape of a trefoil knot constructed from a pipe surface according to § 3.4.2.