

Circular Arc Structures

Pengbo Bo
Univ. Hong Kong, TU Wien

Helmut Pottmann
KAUST, TU Wien

Martin Kilian
Evolute, TU Wien

Wenping Wang
Univ. Hong Kong

Johannes Wallner
TU Graz, TU Wien

Abstract

The most important guiding principle in computational methods for freeform architecture is the balance between cost efficiency on the one hand, and adherence to the design intent on the other. Key issues are the simplicity of supporting and connecting elements as well as repetition of costly parts. This paper proposes so-called circular arc structures as a means to faithfully realize freeform designs without giving up smooth appearance. In contrast to non-smooth meshes with straight edges where geometric complexity is concentrated in the nodes, we stay with smooth surfaces and rather distribute complexity in a uniform way by allowing edges in the shape of circular arcs. We are able to achieve the simplest possible shape of nodes without interfering with known panel optimization algorithms. We study remarkable special cases of circular arc structures which possess simple supporting elements or repetitive edges, we present the first global approximation method for principal patches, and we show an extension to volumetric structures for truly three-dimensional designs.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

Keywords: architectural geometry, circular arc, repetitivity, congruent nodes, cyclides, volumetric meshes, double-curved and single-curved panels, discrete differential geometry

Links: [DL](#) [PDF](#)

1 Introduction

Our work is motivated by the geometric challenges posed by freeform architecture, and, in particular, by the problem of *rationalization* of a freeform design. This means its decomposition into smaller parts, thereby meeting two competing objectives: feasibility, and consistency with the designer’s intentions. Depending on what constitutes the design, there have been different approaches to this problem which have led to different kinds of specific geometric and computational questions. Mostly these questions involve replacing smooth surfaces (possibly with an additional curve network on them) by other structures like meshes with special properties. The guiding thought in all considerations is the efficient manufacturing of the surface parts and their respective necessary supporting/connecting elements. Both simple geometry and *repetition* of elements contribute to this goal of efficiency.

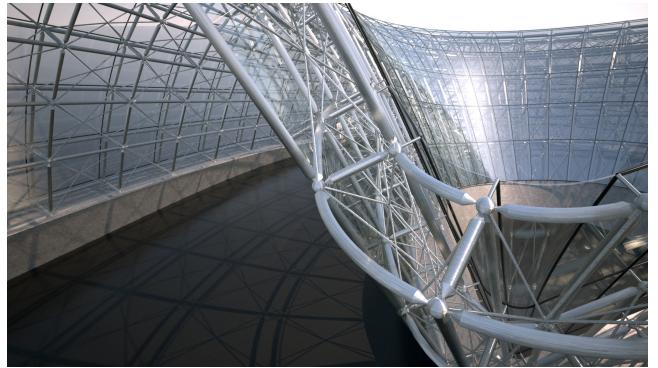


Figure 1: Architectural freeform designs based on circular arc structures exhibit smooth skin, congruent node elements, and simple shapes of beams. In special cases like for the cyclidic CAS shown here, they also admit offsets at constant distance.

Much work deals with decomposing a freeform surface design into flat panels with straight beams between them. However, this process of approximating a smooth surface by a polyhedral surface inevitably shifts complexity to the nodes (vertices): In general no two nodes are congruent and, which is worse, a typical node exhibits *torsion*, i.e., is a truly spatial object whose manufacturing is challenging (see Figure 2). It is possible to optimize nodes to make them torsion-free, which simplifies production and enhances the aesthetic appearance (cf. [Liu et al. 2006; Pottmann et al. 2007] for quad meshes and [Schiftner et al. 2009] for hexagonal meshes).

Often the faceted appearance of planar panels is not intended, and as a natural next step, rationalization with single-curved panels has been proposed by [Pottmann et al. 2008]. This method leads to a surface which is smooth in one direction, but non-smooth in the other. Setting aside the cladding of surfaces by bendable panels (e.g. made of wood and useful for interior design, cf. [Pottmann et al. 2010]), the faithful reproduction of a smooth outer skin necessitates very costly manufacturing of double curved panels.



Figure 2: Node complexity. Manufacturing the connecting element (yellow) via plasma cutting requires much effort if the node has ‘torsion’, because of its truly spatial shape.

This task can be rendered feasible by employing *repetitive* elements which recently have become a focus of study: Eigensatz et al. [2010] show how a given smooth surface with given panel boundaries may be decomposed into panels whose production requires as few costly molds as possible, such that all changes to the original design are within prescribed tolerances. Thus not the panels themselves are repeated, but the auxiliary elements needed for their manufacturing. During this panel optimization the given curve network remains unchanged. The design of curve networks is not addressed by [Eigensatz et al. 2010].

Both [Singh and Schaefer 2010] and [Fu et al. 2010] derive structures which aim at repetitive (i.e., congruent) panels. These panels

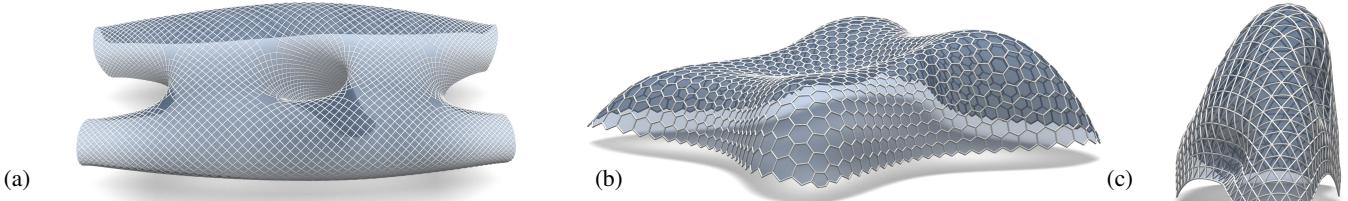


Figure 3: Circular arc structures. (a) Quadrilateral CAS with an edge angle of 90° covering an architectural design. (b) Hexagonal CAS (edge angle 120°) covering the top of the Lilium Tower, Warszaw, by Zaha Hadid Architects. (c) Triangular CAS (edge angle 60°) on the Eindhoven ‘Blob’ by M. Fuksas.

are triangular or quadrilateral, with the quadrilaterals not necessarily planar. These papers show that repetitive panels impose limitations: in the case of triangles, the aesthetic quality is reduced if the number of repetitions increases. Fu et al. [2010] take fairness into account, but in general significantly change the original geometry to reach their goal.

Circular arc structures. This paper takes up the idea of repetitiveness and applies it to panel boundaries and nodes. Our guiding principle is: if we are to have a smooth surface and therefore cannot avoid double-curved panels anyway, we are still very interested in other significant contributions to manufacturing efficiency which do not interfere with the manufacturing of panels.

We propose the class of *circular arc structures* which are built from circular arcs meeting smoothly at *congruent nodes*. In particular we think of two-dimensional structures of quad mesh or hex mesh combinatorics and circular arcs as edges, but we also consider triangular and three-dimensional combinatorics. This way we enhance the state of the art concerning the realization of architectural freeform designs:

- (i) We provide structures with smooth appearance, congruent nodes, and the simplest possible elements for the curved edges.
- (ii) Our cost-effective construction of smooth curve networks and supporting elements does not interfere with an optimized skin panelization according to [Eigensatz et al. 2010].

Previous work. Work concerned with rationalization of architectural freeform designs has been reviewed in the introduction above. Our work specifically deals with structures composed of circular arcs, which in the univariate case are well-studied in Computer-Aided Geometric Design. There are many contributions to curves made of circular arcs (*arc splines*). See e.g. [Leopoldsdeder 2001] for space curves and numerous papers by D. Meek and D. Walton for planar curves (cf. e.g. [Walton and Meek 1995]). Approximation with arc splines has been addressed by [Song et al. 2009].

Special arc structures can have panels which are part of a *cyclide*. Introduced to Geometric Modeling by [Martin et al. 1986], they play a role e.g. in the construction of blending surfaces. The most general surfaces carrying circles which seem to have been considered for geometric modeling are the Dupin cyclides (whose principal curves are circles), and to a lesser extent some of their generalizations. We do not give detailed references but refer to the survey paper [Degen 2002] instead. We are not aware of any previous work on general discrete structures with circular arcs as edges.

Our work constitutes a new type of discretization of surfaces, with obvious relations to discrete differential geometry (see e.g. the monograph [Bobenko and Suris 2008]). We employ a connection between circular and conical meshes on the one hand, and smooth surfaces consisting of cyclidic panels on the other hand, which has

been observed already by E. Huhnen-Venedey in his thesis [2007], later published as [Bobenko and Huhnen-Venedey 2011].

The geometric structures considered in this paper belong to the conformal geometry of surfaces [Gu and Yau 2008]. In their construction we also make use of conformal mappings (e.g. the method of [Zeng et al. 2009]).

Contributions of the present paper. Our contributions in detail are the following:

- We introduce circular arc structures (CAS) which have circular arcs as basic elements and which possess congruent nodes.
- We show how freeform surfaces may be rationalized (for the purposes of architectural design), using circular arc structures. This rationalization preserves surface smoothness, and exhibits properties which are key for efficient manufacturing: planar and simple edge elements; repetition in the nodes.
- We consider repetitions not only in the nodes, but also in the radii of circular edges. This in particular has applications for freeform designs realized in concrete, and is furthermore relevant to 5-axis CNC machining.
- We consider the subclass of *cyclidic structures*, thus providing the first real solution of freeform modeling with Dupin cyclides. Cyclidic structures lead to derived structures which can be very useful for freeform architecture, e.g. for supporting elements.
- We extend our constructions to fully three-dimensional structures (cyclidic cells, variable distance offsets, and others).
- These constructions have nice relations to discrete differential geometry and to the sphere geometries: e.g. they provide a discretization of orthogonal curve networks on surfaces.

2 Circular arc structures

2.1 Definition and computation of CAS

DEFINITION 1. A *circular arc structure* consists of 2D mesh combinatorics (V, E) , where edges are realized as circular arcs, such that in each vertex the adjacent arcs touch a common tangent plane. We require congruence of interior vertices, and we consider the following three cases:

- Hexagonal CAS have valence 3 vertices. Angles between edges equal 120° degrees;
- Quadrilateral CAS have valence 4 vertices. Angles between edges have values $\alpha, \pi - \alpha, \alpha, \pi - \alpha$, if one walks around a vertex;
- Triangular CAS have valence 6 vertices. Angles between edges equal 60° degrees.

For boundary vertices appropriately modified conditions are imposed. The three cases are illustrated by Figure 3. A quadrilateral

CAS with an edge angle of 60 degrees is shown in Figure 5. For quadrilateral CAS we only really consider the case that at each vertex the two pairs of opposite edges are joined smoothly. Thus we have smooth *flow lines* of edges (the definition would allow rather irregular structures). Also in the triangular and hexagonal cases we could allow different angles, but for aesthetic reasons we stick to 60° and 120°.

The notion of ‘arc’ employed here includes the limit case of infinite radius, i.e., straight line segments. Most images show arcs as pipes with circular cross-section, which incidentally is also a way of realization in practice.

Data structure. A circular arc structure is uniquely determined by the combinatorics (V, E) , the position of each vertex $v \in V$, and by the collection of *edge vectors* a_{vw} (see Figure 4): a_{vw} is the unit vector attached to the vertex v which is tangent to the arc \widehat{vw} . These variables are not independent, but for purposes of optimization are treated as if they were – the constraints which apply are built into the target functional. This target functional is a rather straightforward collection of nonlinear least squares terms, which turns out to be sufficient for our purposes. In the following we describe the setup, numerical methods employed in optimization, and how to initialize optimization. The computation of circular arc structures with special properties which are introduced later is discussed in the respective subsections.

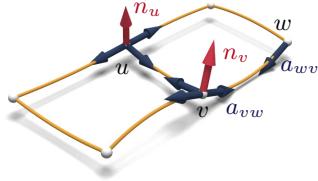


Figure 4: Quadrilateral CAS with vertices u, v, w , normal vectors n_u, \dots (red), and edge vectors a_{uv}, \dots (blue).

Setting up a target functional. For optimization we employ a composite target functional which has the general form

$$F = \lambda_1 f_{prox} + \lambda_2 f_{pl} + \lambda_3 f_{cons} + \lambda_4 f_\Delta + \lambda_5 f_{ang}, \quad (1)$$

where $\lambda_1, \lambda_2 \dots$ are nonnegative weights. The individual contributions to this target functional are as follows.

The term f_{prox} penalizes deviation of vertices from a reference surface Φ , and likewise deviation of the boundary vertices from the reference surface’s boundary. We use the symbol $\pi(v)$ for the point in Φ which is closest to v , and similar $\pi_\partial(v)$ for the closest point to v in the reference surface’s boundary $\partial\Phi$. The symbols $T_{\pi(v)}$, $T_{\pi_\partial(v)}$ denote the tangent plane or tangent, respectively. We define

$$f_{prox} = \sum_{v \in V} \text{dist}(v, T_{\pi(v)})^2 + \sum_{v \in V_\partial} \text{dist}(v, T_{\pi_\partial(v)})^2.$$

As for smoothness, the term f_{pl} ensures that edges sharing a vertex have a common tangent plane. This is expressed by linear dependence of the initial tangent vectors of such edges:

$$f_{pl} = \sum_{v \in V} \sum_{\widehat{vw}_i \in E} \det(a_{vw_{i-1}}, a_{vw_i}, a_{vw_{i+1}})^2.$$

For geometric consistency we must deal with the fact that the vectors a_{vw}, a_{wv} , serving as end tangent vectors of an arc, are not independent. There exists a circular arc connecting vertices v, w which fits these vectors if and only if the entire configuration of vertices and edge vectors is mirror-symmetric:

$$\rho_{vw}(a_{vw}) = a_{vw}, \quad \text{where}$$

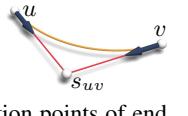
$$\rho_{vw}(x) = x - 2 \left\langle \frac{v-w}{\|v-w\|}, x \right\rangle \frac{v-w}{\|v-w\|}.$$

Accordingly we define a contribution to the target functional, which expresses this relation between vectors a_{vw}, a_{wv} , by

$$f_{cons} = \sum_{\widehat{vw} \in E} \|\rho_{vw}(a_{vw}) - a_{wv}\|^2.$$

For regularization it is not enough to employ second order differences of vertices; in order to punish a wiggly sequence of arcs we use auxiliary points s_{uv} which are defined as the intersection points of end tangents of the arc \widehat{uv} . We let

$$f_\Delta = \sum_{\substack{\text{smooth union} \\ \widehat{uv}, \widehat{vw}}} \|u - 2v + w\|^2 + \sum_{\substack{\text{smooth union} \\ \widehat{uv}, \widehat{vw}, \widehat{wz}}} \|s_{uv} - 2s_{vw} + s_{wz}\|^2. \quad (2)$$



In a hexagonal CAS there are no smoothly joining edges, so the contribution of f_Δ is zero. We use the graph Laplacian

$$\sum_{v \in V} \|v - \frac{1}{\deg(v)} \sum_{u: \widehat{uv} \in E} u\|^2$$

for regularization instead. Finally we deal with the various constraints on angles between edges. With the temporary notation $\phi_{uv, uw}$ for the desired angle between edges \widehat{uv} and \widehat{uw} we let

$$f_{ang} = \sum_{\text{angles } \phi_{uv, uw}} (\langle a_{uv}, a_{uw} \rangle - \cos \phi_{uv, uw})^2.$$

The sum is taken over all edge pairs where an angle is prescribed.

Numerical optimization. To compute a minimizer of (1) we employ a classical Gauss-Newton method. All required first order derivatives are calculated exactly (i.e., are not approximated). The linear systems to be solved in each round of iteration are sparse, since the single contributions to (1) involve only local operations defined in terms of small vertex neighborhoods. We employ Levenberg-Marquardt regularization [Kelley 1999], so the occurring matrices are positive definite, and we can employ sparse Cholesky factorization, using the TAUCS library [Toledo 2003]. In order to enforce the geometric constraints, the weights of terms f_{pl} , f_{cons} , f_{ang} , are increased in each round of iteration.

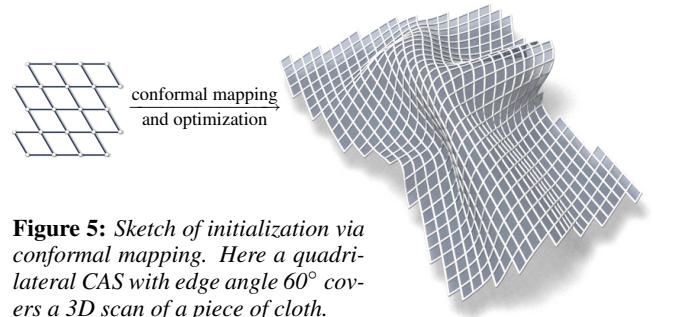


Figure 5: Sketch of initialization via conformal mapping. Here a quadrilateral CAS with edge angle 60° covers a 3D scan of a piece of cloth.

Initialization. Most components of our target functional (1) are highly nonlinear and there is no guarantee that standard numerical methods like Gauss-Newton converge. Fortunately we are able to start our optimization from a configuration which is already very close to the solution. This is because any CAS on a reference surface Φ can be seen as a discrete version of a curve network with prescribed angles of intersection. We therefore construct a mesh in a planar domain with the desired intersection angles between edges and use a conformal mapping to transfer it to Φ . Optimization is initialized from the resulting curve network in Φ (see Figure 5). We tried various types of conformal mapping from a planar domain to

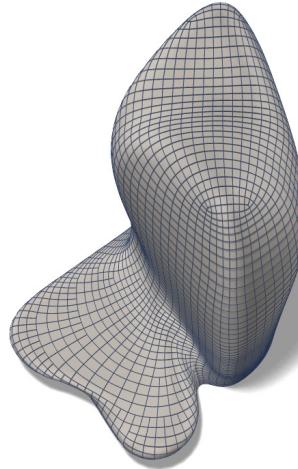
surfaces. One is the method of [Zeng et al. 2009], which has been used for Figures 3c and 8. Another method is the as rigid-as-possible mapping of [Alexa et al. 2000], which is used for Figure 5, because in this example the reference surface is isometric to a planar domain.

Another method of initialization, which from the abstract viewpoint contains a discrete-conformal mapping, is to approximate Φ by a mesh which enjoys the circle-packing property according to [Shiftner et al. 2009]. Such a mesh can be directly used to initialize optimization of a triangular CAS; a hex mesh dual to it can be used to initialize optimization of a hexagonal CAS (this has been done for Figure 3b). Just to mention yet another way: Figure 3a has been initialized from the diagonals of a conical mesh whose faces happened to be roughly square.

There are as many ways to initialize a CAS as there are ways to find curve networks on surfaces which have the prescribed intersection angles. For a quadrilateral CAS, which is initialized by two families of curves, it is possible to choose the first family arbitrarily and find the second one by the condition of constant intersection angle (this method has been used for Figure 6, supported by the conformal mapping of [Zeng et al. 2009]). Later we discuss CAS with special properties whose initialization is different.

In any case the decision on how many CAS mesh elements to use in each direction has to be made in the initialization phase. A typical constraint which arises in applications is maximum panel size.

Figure 6: Quadrilateral CAS with edge angle 90° which covers the design by Zaha Hadid Architects for the National Holding Headquarters, Abu Dhabi. This CAS exhibits one family of ‘horizontal’ flow lines, which implies either T-junctions or a very uneven distribution of the second family of flow lines (we decided for the former). Note that in the ‘steep’ part, where no T-junctions occur, load-bearing elements can follow the CAS.



2.2 Generalizations

Definition 1 is sometimes too restrictive. It is important in applications to extend the notion of CAS to slightly more general situations such as combinatorial singularities (i.e., vertices which do not have the same valence as the majority of other vertices, cf. Figure 6), or other vertices where the edge angle condition is either modified or is absent. They are otherwise treated in the same way as other vertices and are easily incorporated into our optimization framework. Another important generalization is quad meshes with *T-junctions*. Those are usually avoided for load-bearing elements because of the high bending stresses they cause. For pure panelization, however, T-junctions make perfect sense, see Figure 6.

As regards computation, Figure 7 shows such a mesh together with its geometric data. We still require that each face is bounded by four circular arcs. This leads to the situation that we have circular arcs (like \widehat{vw} in Figure 7) which contain certain smaller edges (like \widehat{uu}' in Figure 7). In our optimization we must take care of the fact

that the edge vectors of the edge \widehat{uu}' are actually taken from the bigger arc \widehat{vw} , which means

$$a_{uu'} = \rho_{wu}(a_{vw}), \quad a_{u'u} = \rho_{vu'}(a_{vw}).$$

An appropriate sum of squares, which expresses this condition for every sub-edge adjacent to a T vertex is added to the objective function (1).

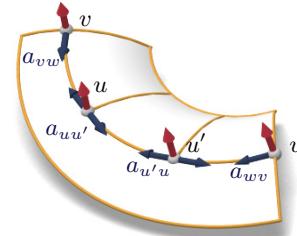


Figure 7: CAS can have T-junctions. Here T-vertices u, u' sit on a circular edge \widehat{vw} . Also shown are normal vectors (red) and edge vectors (blue). The edge \widehat{uu}' is part of the arc \widehat{vw} .

2.3 Geometry and curvatures of CAS

Recall the geometric data defining a circular arc structure (Figure 4): Each vertex v of a CAS is equipped with a unit normal vector n_v . For each edge \widehat{vw} , the unit vectors a_{vw}, a_{wv} are respectively attached to vertices v, w and serve as initial tangent vectors of that curved edge.

It is not difficult to obtain a smooth surface by filling each quadrilateral face by an appropriate surface patch (for methods to do this, see e.g. [Farin et al. 2002]). Each such surface has the same normal vectors as the CAS. However, there is in general no curvature-continuous surface which contains a given CAS if vertex valences are 4 or higher. This is because the curvatures of the arcs adjacent to a vertex do not match. For a hexagonal CAS, where only three arcs meet in a vertex, this is different and we can actually find, for each vertex, not only a normal vector but a second fundamental form (i.e., a curvature element) which matches the three arcs emanating from this vertex. This is because in 2D any symmetric quadratic form is uniquely determined by its values at three linearly independent vectors.

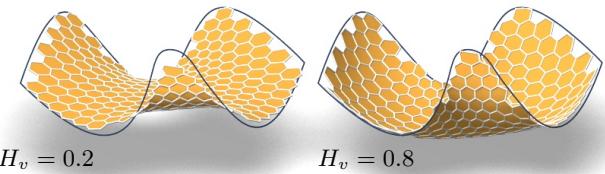


Figure 8: Soap films under pressure. A sequence of CAS was produced by first covering a minimal surface by a hexagonal CAS which is optimized for vanishing mean curvature ($H_v = \text{const.} = 0$). Subsequently we recursively initialize the computation of a hexagonal CAS with mean curvature $H_v = \frac{i}{10}$, $i = 1, 2, \dots$ by a hexagonal CAS with mean curvature $\frac{i-1}{10}$. We require proximity to the given boundary, but not to any reference surface.

Mean curvature. It turns out that we can find a nice formula for the mean curvature which has an interesting interpretation in Möbius geometry. This is the content of the following paragraphs which have no direct relation to applications, but which are interesting from the viewpoint of discrete differential geometry. We start with a well known lemma.

LEMMA 2. Consider a regularly arranged collection of m directions tangent to a C^2 surface in a common point. Using coordinates w.r.t. the principal frame, these directions are indicated by vectors

$$\begin{pmatrix} \cos \psi_j \\ \sin \psi_j \end{pmatrix}, \quad \psi_j = \phi + j \frac{2\pi}{m} \quad (j = 0, \dots, m-1).$$

Each direction having normal curvature $\kappa_{j,n}$, the mean curvature of the surface equals $\frac{1}{m}(\kappa_{1,n} + \dots + \kappa_{m,n})$, provided $m \geq 3$.

Proof. Elementary surface theory implies that normal curvatures can be expressed in terms of the principal curvatures κ_1, κ_2 as $\kappa_{j,n} = \kappa_1 \cos^2 \psi_j + \kappa_2 \sin^2 \psi_j$ [do Carmo 1976]. Observing $\cos^2 \alpha = \frac{1}{2}(1 + \cos 2\alpha)$, we get $\frac{1}{m} \sum_{j=1}^m \kappa_{j,n} = \frac{\kappa_1 + \kappa_2}{2}$. \square

LEMMA 3. For any curvature-continuous surface containing a hexagonal CAS, the value of mean curvature in the vertex v equals

$$H_v = \frac{1}{3} \sum_{e=\widehat{vw} \in E} \kappa_n(e), \text{ where } \kappa_n(\widehat{vw}) = 2 \left\langle \frac{w-v}{\|w-v\|^2}, n_v \right\rangle.$$

Here n_v is the unit normal vector in the vertex v . The symbol $\kappa_n(e)$ means normal curvature of the edge e .

Proof. We employ coordinates w.r.t. a principal frame attached to the vertex v , with the normal vector n_v as the third axis. Meusnier's theorem implies that the osculating circle of any curve in a surface lies in the sphere with radius $1/\kappa_n$ and center $(0, 0, 1/\kappa_n)$, where κ_n is the curve's normal curvature [do Carmo 1976]. We now use the following geometric trick: The inversion mapping $\iota(x) = x/\|x\|^2$ maps this sphere to the plane with equation $x_3 = \kappa_n/2$. Since the arc \widehat{vw} coincides with its own osculating circle, we can read off its normal curvature from the third coordinate of $\iota(w)$. This immediately yields the expression for the normal curvature, and Lemma 2 concludes the proof. \square

Even for a quadrilateral CAS, where curvatures are not compatible we may use the procedure suggested by Lemmas 2 and 3 to define a mean curvature.

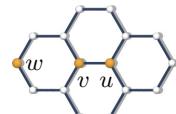
DEFINITION 4. If two edges $\widehat{uv}, \widehat{vw}$ join smoothly in the vertex v , having normal curvatures κ', κ'' there, we assign the average normal curvature $\frac{1}{2}(\kappa' + \kappa'')$ to this pair. In a right-angled CAS the arithmetic mean of the two occurring average curvatures is defined to be the mean curvature of the CAS – resulting in the formula

$$H_v = \frac{1}{\deg(v)} \sum_{e=\widehat{vw} \in E} \kappa_n(e).$$

Optimization involving curvatures. If we are to optimize a hexagonal CAS such that the mean curvature assumes a given value (see Figure 8), we create a sum of squares from Lemma 3 which penalizes deviation from this value and add it to the target function (1). Unfortunately this approach is capable only of small optimization tasks like the one shown by Figure 8, where we successively increment the value of H_v . It is not possible to solve more difficult problems this way, such as the boundary value problem for surfaces with constant mean curvature. It turns out that optimization needs an additional regularization term. We used

$$f_{\Delta, \text{hex}} = \sum_{(u,v,w)} \|2(u-v) - (v-w)\|^2,$$

where summation is over all triples u, v, w which are arranged as the figure shows.



Möbius-geometric properties of CAS. Circular arc structures are entities of Möbius geometry. This is because Möbius transformations preserve smoothness, angles, and the property of being a circular arc. It turns out that even the average normal curvature $\kappa = (\kappa' + \kappa'')/2$ according to Definition 4 is a Möbius invariant, if it is encoded as the “Meusnier sphere” of radius $1/\kappa$ (cf. the proof of Lemma 3). In the same sense the mean curvature, found by a further round of averaging, is a Möbius invariant if geometrically encoded as a sphere. This is an instance of a well known Möbius-invariant construction with spheres, based on the fact that three spheres which are in contact and which have radii $1/\kappa', 1/\kappa'', 1/(\kappa' + \kappa'')$ together with their common point of contact (with radius 0), constitute a harmonic quadruple [Hertrich-Jeromin 2003].

2.4 Supporting elements

In building construction statics is essential, and freeform designs require special care in this area. Even with straight load-bearing elements, both the necessary analysis and the assembly under partial load have to be done with care. The manufacturing of curved load-bearing elements is more difficult and costly than making straight ones, and it is therefore very important to know which kinds of simple shapes of support elements can be used. A closely related matter is the manner of connection of support elements in the vertices: We want to avoid torsion in the node (cf. Figure 2), which would entail the manufacturing of truly 3D node elements.

Symmetric edges. We show that by insisting on ‘symmetric’ edges we can give a solution to this problem. We define:

DEFINITION 5. An edge \widehat{vw} is symmetric if and only if reflection in the bisector plane of v, w maps the respective normal vectors n_v, n_w onto each other:

$$\rho_{vw}(n_v) = n_w, \quad \rho_{vw}(n_w) = n_v. \quad (3)$$

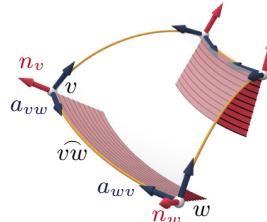


Figure 9: Symmetric edges (like \widehat{vw} in this picture) can be supported by a piece of right circular cone which is shown in red.

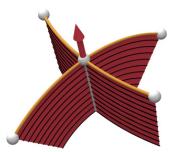


Figure 10: Avoiding node torsion with symmetric edges. Conical support elements intersect in a node axis spanned by the surface normal.

Such a situation is illustrated in Figure 9. The condition of symmetry of an edge \widehat{vw} means that the normals in vertices v, w intersect each other, and that there is a right circular cone containing the arc \widehat{vw} as well as both normals at the endpoints v, w (this manner of speaking includes the limit case that normals are parallel and the cone becomes a right circular cylinder). That cone serves as a support element of simple shape. Moreover, if such supports meet in a vertex v , they are going to intersect in a common axis spanned by the normal vector n_v (see Figure 10). An actual construction which is based on the existence of symmetric edges and conical support elements is shown in Figure 11.

Remark. The cone of normals which is associated with a symmetric edge defines a normal vector n in every point of that edge. As we proceed along a flow line, having a unit tangent vector t at each moment, the motion of the orthonormal frame $(t, n, t \times n)$ is the natural or rotation-minimizing one [Bishop 1975], because the surface traced out by the vector n is developable.

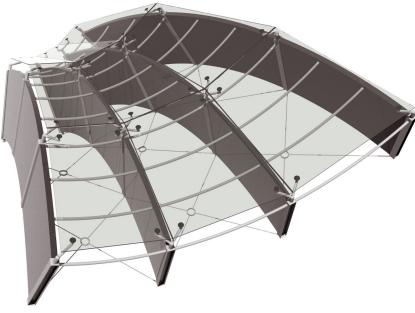


Figure 11: Supporting elements in the shape of cones along symmetric edges form part of this construction detail. This CAS is special and has an accompanying planar quad mesh (cf. Section 4) used for the mounting of glass panels. Construction and rendering courtesy M. Reis.

Optimization towards symmetric edges. We add condition (3) to our objective function (1) by adding the term

$$f_{\text{symm}} = \sum_{\text{symmetric edges } v \widehat{w}} \|\rho_{vw}(n_v) - n_w\|^2.$$

However, if vertices v, w are contained in a curvature-continuous surface, and their respective normals intersect, then the vector $v - w$ approximately follows a direction of principal curvature. This is one of the guiding principles in discretizing principal curvature lines, see e.g. [Sauer 1970] and [Liu et al. 2006]. We conclude: in order to approximate a reference surface Φ with a CAS with a family of symmetric edges, we must initialize optimization such that these edges follow Φ 's principal curvature lines. Figure 12 shows an example where this rule has not been observed.

3 CAS with repetitive elements

Repetition is a key ingredient in all considerations of manufacturing efficiency. Circular arc structures as given by Definition 1 already have a repetitive feature, namely congruence of nodes. In the interest of cost reduction we study further repetitions. It turns out that the radii of edges are both easy to handle computationally and important for manufacturing. This leads to the following definition.

DEFINITION 6. A quadrilateral CAS is radius-repetitive along a flow line, if the radius of its edges is constant. It is transversely radius-repetitive for a pair of neighbouring ‘parallel’ flow lines, if the edges which connect these flow lines have constant radius.

Application: double-curved concrete. The repetitivity properties mentioned by Definition 6 contribute to simpler manufacturing of edges. There is a particular application where it is hugely beneficial to have the reference surface densely covered by arcs of constant radius, namely the making of double-curved surfaces in concrete. Figure 13 shows an example of this. A transverse-repetitive CAS with a few dominant smooth flow lines and many

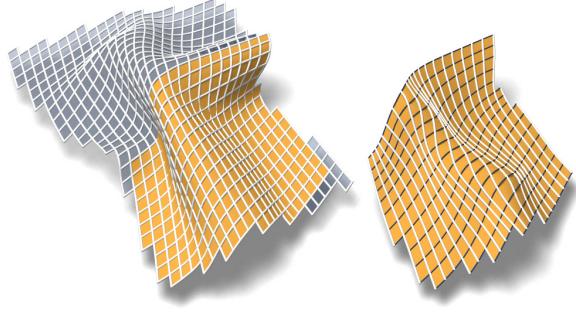


Figure 12: Symmetric edges. A CAS with edge angle 60° (yellow, at left, part of Figure 5) is optimized such that one family of flow lines becomes symmetric. The result is shown at right, with symmetric edges in blue. Since these flow lines did not follow the original principal curvature lines, optimization entails a significant shape change.

transverse edges of constant radius is optimized with relaxed angle constraints (i.e., a low weight for all angle terms involving the transverse edges). The many edges of constant radius now serve as curved supports for the pouring of concrete.

Computational issues: Initialization. The condition of certain edges having equal radii is easily incorporated into our objective function (1). The main computational problem here is initialization. We show a method based on level sets which in itself is an optimization problem. The first repetitivity property implies that the flow lines are arc splines of constant radius, i.e., curves of constant curvature. We therefore have to initialize optimization with curves of constant curvature which lie in the reference surface. A CAS with the transverse repetitivity property (constant radii across flow lines) is best initialized from a family of curves where the normal curvature across that curve does not change as we proceed along the curve.

For computational purposes the reference surface is represented as a triangle mesh, and a real-valued function ϕ is defined by its values on the vertices. We wish to find ϕ such that its level sets enjoy one of the curvature properties mentioned above. We follow [Pottmann et al. 2010] and express the normal curvature κ_n , the geodesic curvature κ_g , and the total curvature κ of level sets by

$$\kappa_n = \frac{\Pi(J\nabla\phi)}{\|\nabla\phi\|^2}, \quad \kappa_g = \text{div} \left(\frac{\nabla\phi}{\|\nabla\phi\|} \right), \quad \kappa^2 = \kappa_n^2 + \kappa_g^2.$$

Here Π is the second fundamental form of the reference surface, and J is rotation by 90 degrees. The normal curvature across a level set is given by the expression

$$\kappa_n^{\text{across}} = \Pi(\nabla\phi)/\|\nabla\phi\|^2.$$

For the example of Figure 13, the function ϕ was constrained to assume prescribed values in 2 vertices, and optimized such that the L^2 norm of $\kappa_n^{\text{across}} - \kappa_0$ is minimal (up to regularization; for more

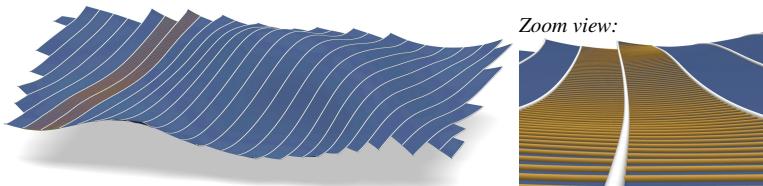


Figure 13: A surface densely covered by arcs of constant radius which stem from a transverse-repetitive CAS can be made in concrete, with wooden supports along the transverse edges (design surface by Mario Bellini Architects and Rudy Ricciotti, for the Museum of Islamic Arts, in the Louvre, Paris; realized in steel/glass).

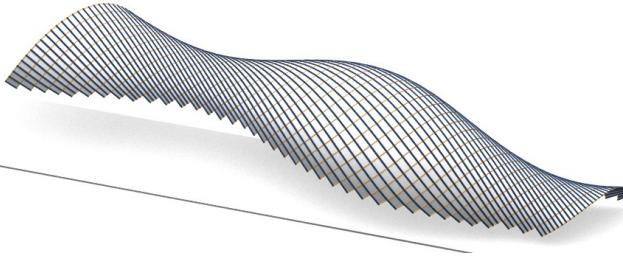


Figure 14: CAS which is radius-repetitive along flow lines (blue).

details of this optimization procedure we refer to [Pottmann et al. 2010]). The value κ_0 has been chosen interactively such that we get a suitable result. Note that in this special example the value κ_n^{across} is not only constant along flow lines, but constant for the entire surface. The level sets of ϕ then yield one family of curves which is needed for initializing CAS optimization. Other desired curvature properties lead to similar optimization procedures (cf. Figure 14).

If necessary the second family can also be found as level sets of a function ψ : The condition of constant intersection angle α reads $\langle \nabla \phi, \nabla \psi \rangle = \cos \alpha \cdot \|\nabla \phi\| \cdot \|\nabla \psi\|$. For the example in Figure 13, however, the ‘transverse’ family has been found as orthogonal trajectories of the first family which amounts to solving a first order ODE. It would be possible to use ODEs to obtain curves with the above-mentioned curvature properties, but this is rather unstable.

Computational issues: Optimization. Once initialization is done, we must incorporate the desired radius constraints into our optimization. Suppose all edges contained in some set $E_1 \subseteq E$ are to have the same radius r_1 , and similarly for edge sets E_2, \dots, E_k . We introduce the inverse radii as new variables and augment our target functional by

$$f_{\text{rad}} = \sum_{i=1}^k \sum_{\hat{uv} \in E_i} \left(\frac{1}{r_i} - 2 \frac{\|a_{uv} \times (u - v)\|}{\|a_{uv}\| \cdot \|u - v\|^2} \right)^2.$$

It is easy to see that the expression in brackets vanishes if and only if the edge \hat{uv} has radius r_i .

Application: CNC machining. Covering a freeform surface by circular arcs of constant radius is highly relevant for 5-axis CNC machining of such surfaces as the arcs directly correspond to the position of a flat-endmill. This application geometrically amounts to finding a transverse-repetitive CAS, with a few additional considerations specific to milling. We do not go into details.

4 Cyclidic structures

Dupin cyclides. The *Dupin cyclides* are an interesting class of surfaces which have sufficiently many degrees of freedom to make them useful for geometric design. This class is generated by applying Möbius transformations to tori (cf. Figure 15), and includes the cylinders and right circular cones. By definition, it also includes the limit cases of spheres and planes. Introduced to geometric modeling by [Martin et al. 1986] and studied in many papers, they have recently occurred in a discrete differential geometry context: [Bobenko and Huhnen-Venedey 2011] show how a circular quad mesh may be converted into a smooth surface consisting of cyclide patches. That paper also contains proofs of the following facts about cyclides (part of which are well known), and which are relevant for our purposes.

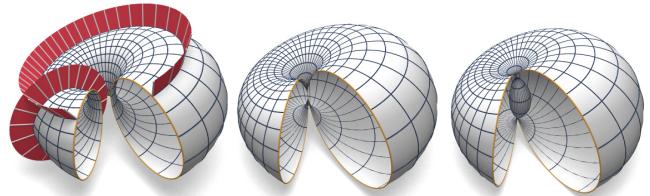


Figure 15: Dupin cyclides. Three different shapes of Dupin cyclides which by applying a Möbius transform can be mapped to a torus, a cylinder, or a cone. Left: The normals along a line of curvature lie in a right circular cone (two cones are shown in red).

- (a) The rectangular network of principal curvature lines consists of circles (illustrated in Figure 15).
- (b) The normals along such a principal circle constitute a right circular cone, including the limit cases of cylinder and plane (see Figure 15).
- (c) The vertices of a *principal* quadrilateral, whose edges are principal circles, possess a circumcircle (see Figure 16).
- (d) A quadrilateral with circular edges is a principal quadrilateral of a cyclide, if and only if edges intersect orthogonally and each edge is *symmetric* in the sense of Definition 5.
- (e) For each boundary edge of a cyclidic principal patch there is a right circular cone which is tangent to the patch along that edge (see Figure 20).

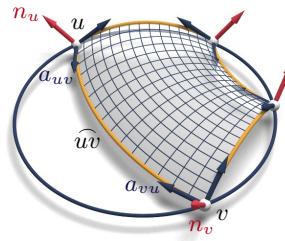


Figure 16: A cyclidic patch. The vertices have a circumcircle, and all normal vectors are symmetric w.r.t. reflections: $n_u = \rho_{uv}(n_v)$.

Cyclidic CAS. Properties (b) and (e) above imply that the union of two cyclidic principal patches with a common boundary arc and common normal vectors in its endpoints actually is a single smooth surface. Together with Property (d), this shows the following:

PROPOSITION 7. *A quadrilateral CAS with an edge angle of 90 degrees and symmetric edges is converted into a smooth surface by filling each face with its corresponding principal cyclidic patch. Moreover, the vertices of this CAS constitute a circular mesh.*

A freeform design whose skin is modelled as a cyclidic CAS possesses nicely shaped associated support elements: see Section 2.4 for the right circular cones orthogonal to the surface along edges, and Figure 11 for a construction detail based on this fact.

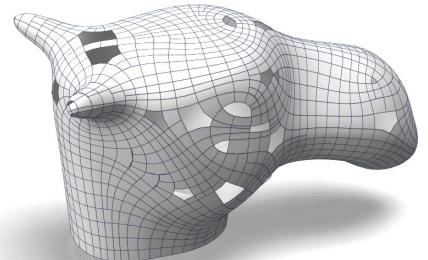


Figure 17: Cyclidic CAS which follows the principal curvature lines of a surface Φ . Optimization has been initialized from a circular mesh which approximates Φ .



Figure 18: Cyclidic CAS and tangentially circumscribed developable strip model. Left: Apart from isolated singularities, a freeform design is covered by principal patches derived from a cyclidic CAS. Right: Using tangent cones of cyclidic patches we obtain smooth developable strips consisting of right circular cones (the same cyclidic CAS is used in Figures 1 and 24, where the holes have been closed by non-cyclidic patches).

Computational issues. If a CAS with symmetric edges is to cover a reference surface Φ we must let the flow lines follow the principal curvature lines (cf. our previous remarks in Section 2.4). If all edges are symmetric, the CAS itself is a discrete version of the network of principal curvature lines, and we may initialize CAS optimization from any other discrete version of the same geometric entity. In particular a circular mesh or a conical approximating Φ serves this purpose [Liu et al. 2006; Bobenko and Suris 2008]. This is how Figures 17 and 18 have been produced.

Remark. The cyclidic CAS as a geometric object is essentially fully discussed by [Huhnen-Venedey 2007] and [Bobenko and Huhnen-Venedey 2011]. There it is also shown that taking the vertices from a circular quad mesh and choosing two edge vectors a_{vu} , a_{vv} uniquely determines a cyclidic CAS. The contribution of the present paper is to solve the approximation problem and to identify applications in freeform architecture.

Offsets and Lie geometry. It is well known that the offset at constant distance of a cyclidic patch is again a cyclidic patch, so the offsetting operation of cyclidic CAS is well defined. The construction details of Figures 1, 11, and 19 are based on this fact.

The above-mentioned geometric result about offsetting is a special case of a more general invariance property: Dupin cyclides are objects of Lie geometry, if they are interpreted as a set of oriented spheres which are in contact with the oriented surface. Any contact-preserving transformation in sphere-space maps cyclides to cyclides [Cecil 1992]. This is even true for cyclidic CAS, which follows from [Bobenko and Huhnen-Venedey 2011]. Thus cyclidic CAS are objects of Lie geometry.

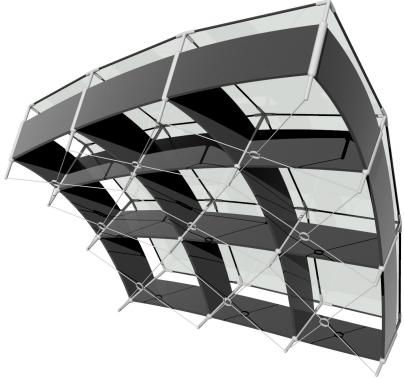


Figure 19: Combining smooth and polyhedral elements. This construction is based on 3 offset layers of a cyclidic CAS and exhibits conical support elements. The boundary vertices of any cyclidic patch are co-planar, which is exploited for the mounting of planar glass panels (courtesy M. Reis).

Developable strip models circumscribed to cyclidic CAS. A cyclidic patch possesses a right circular cone which is tangent to the patch along a boundary edge (this includes the limit case of cylinders and planes; see Property (e) above and Figure 20).

By joining the tangent cones of all edges of a smooth flow line we get a smooth developable surface. Doing this for an entire family of flow lines yields a semidiscrete surface – a developable strip model in the sense of [Pottmann et al. 2008], see Figure 18:

PROPOSITION 8. *A cyclidic CAS can be converted into a developable strip model, with each smooth developable being a union of right circular cones tangent to the cyclidic surface, and such that the common boundary of neighbouring strips consists of conics.*

Proof. Let us recap the well known rational Bézier representation of cyclidic patches. We use the symbol $B_i^k(t) = \binom{k}{i}(1-t)^{k-i}t^i$ ($0 \leq i \leq k$) for the degree k Bernstein polynomials. With control points b_{ij} and weights w_{ij} , a rational degree (m, n) Bézier surface reads $f(s, t) = \frac{\sum_{i,j} w_{ij} b_{ij} B_i^m(s) B_j^n(t)}{\sum_{i,j} w_{ij} B_i^m(s) B_j^n(t)}$, where $i \in \{0, \dots, m\}$, $j \in \{0, \dots, n\}$, and s, t run in the interval $[0, 1]$. Cyclidic patches have degree $(2, 2)$, with control points according to Figure 20. For the weights, see e.g. [Pratt 1995]. The control mesh with vertices b_{ij} has planar faces, with b_{11} as the intersection point of tangent planes in the boundary vertices $b_{00}, b_{20}, b_{22}, b_{02}$.

It follows from elementary properties of Bézier surfaces that the lines passing through $f(0, t)$ spanned by the derivative vectors $\frac{\partial}{\partial s} f(0, t)$ generate a rational degree $(1, 2)$ Bézier surface, which shares control points $\{b_{ij}\}_{i=0,1; j=0,1,2}$ and corresponding weights with the cyclidic patch. An analogous surface is tangent to the patch along the curve $f(1, t)$. Obviously these two surfaces are the right circular cones tangent to the cyclidic patch.

From the Bézier representation we know that their intersection is a rational Bézier curve defined by control points b_{10}, b_{11}, b_{12} and corresponding weights (i.e., a conic, see Figure 20). \square

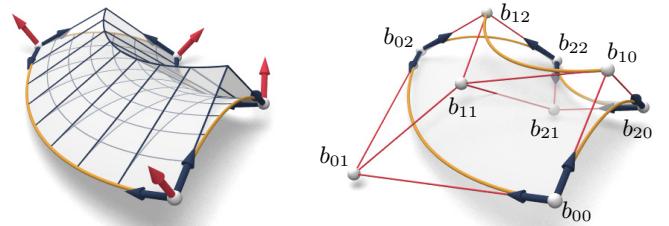


Figure 20: Left: Two right circular cones tangent to a cyclidic patch along opposite edges. Right: Control points $\{b_{ij}\}$ of the rational Bézier representation of a cyclidic patch. Note that b_{10}, b_{11}, b_{12} serve as control points of the intersection curve of tangent cones.

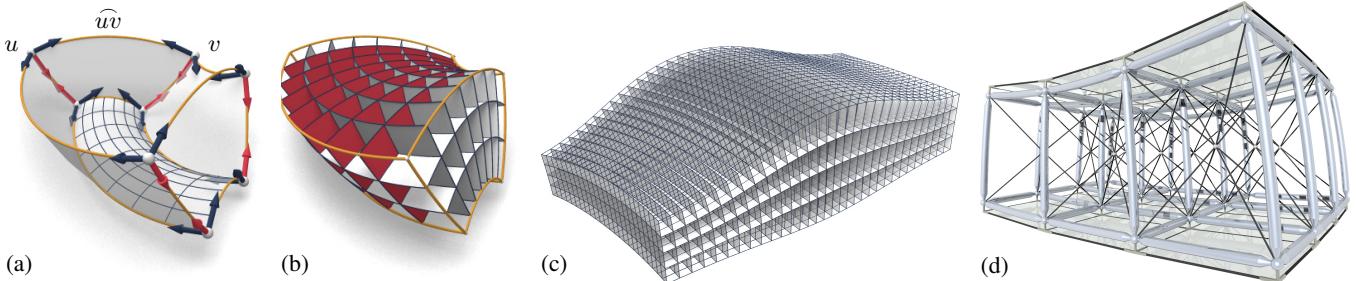


Figure 21: Cyclidic cell and cyclidic 3D structures. (a) A combinatorial cube where each of the 12 edges is symmetric, with cyclidic patches as faces. (b) Derived families of orthogonally intersecting cyclides. (c) Cyclidic 3D structures consist of cyclidic cells and can be interpreted as variable-distance offsets of a cyclidic CAS. (d) A construction detail of a variable-distance offset based on part of (c).

Remark. Since symmetric edges of a CAS follow the principal curvature lines of the underlying surface, it does not make too much sense to consider right-angled CAS where only one family of edges is symmetric.

5 Three-dimensional structures

Volumetric meshes in connection with freeform architecture play an important role in offset constructions and more general multilayer constructions [Liu et al. 2006; Pottmann et al. 2007]. The rigidity of the occurring geometric constraints, such as constant distances, makes truly three-dimensional geometric modeling difficult. This paper shows how cyclidic CAS may be extended to three-dimensional geometric structures, and that we are able to do geometric modeling to a limited extent.

Cyclidic cells and 3D cyclidic structures. It is well known that a Dupin cyclide can be non-uniquely embedded into three families of cyclides which mutually intersect at right angles, the intersection curves being the principal circles. Figure 21 shows a *cyclidic cell* bounded by 6 principal cyclidic patches which occur in this way. It has been observed by [Huhnen-Venedey 2007] that a combinatorial cube whose edges are orthogonally intersecting circular arcs is a cyclidic cell if and only if each edge is symmetric, i.e., the three edge vectors at either end are mapped into each other by a reflection (cf. Definition 5). Figure 21a illustrates such a symmetric collection of 24 edge vectors (two to each edge of the cube, and three to each vertex).

A 3D cyclidic structure is a combinatorial cube mesh where each face is a cyclidic patch, serving as a common boundary of two adjacent cyclidic cells. Figure 21 illustrates an example alongside a construction detail.

Computational issues. Apart from the 3D combinatorics, the optimization of a three-dimensional cyclidic structure is not different from the case of cyclidic CAS. If the 3D structure is an extension of a 2D cyclidic CAS which is to approximate a reference surface Φ , then we propose to initialize optimization from principal curvature lines of Φ together with those of appropriate offsets.

Available degrees of freedom. Unfortunately there are only few degrees of freedom in 3D cyclidic structures. We derive this information as follows. The vertices of such a structure constitute a 3D circular mesh (each 2D face has a circumcircle). Conversely a 3D circular mesh can be interpolated by a cyclidic structure, which is determined by the choice of a single orthonormal frame of edge vectors (by edge symmetry we can propagate the edge vectors to

the remaining vertices). The known degrees of freedom of circular meshes (cf. [Bobenko and Suris 2008]) show, for instance, a d.o.f. count for $3 + m + n$ for $m \times n$ cyclidic cells which extend a given cyclidic CAS consisting of $m \times n$ patches.

Remark. Any orthogonal 3D CAS consisting of combinatorial cubes is a discretization of three families of orthogonally intersecting surfaces. By Dupin's theorem their intersection curves are principal curvature lines. So it makes sense to consider only such right-angled three-dimensional CAS which are cyclidic.

6 Discussion

On the approximation problem. Throughout this paper we address *fitting* or *approximation* problems with CAS. It is essential for solving such problems that we do not try to find vertices first and subsequently interpolate them by arcs. To see that interpolation would be a bad idea, we perform sample d.o.f. counts: For a cyclidic CAS we can choose one edge vector and normal vector and propagate these data by reflection, which determines all edges. Thus, interpolating vertices by a cyclidic CAS has only 3 d.o.f. in total. For the weaker condition of orthogonally intersecting edges in a quadrilateral CAS, it is not difficult to see that there are in total 4 d.o.f. (because edge vectors in 2 neighbouring vertices determine the remaining edge vectors).

The degrees of freedom at our disposal are therefore essentially the vertex positions. Optimization moves both vertices and edges, thereby penalizing the vertices' deviation from the reference surface Φ . It does not seem to be necessary to penalize the edges' deviation from Φ as well; regularization by (2) proved sufficient. We emphasize again that the conceptually simple minimization of (1) succeeds only because we pay special attention to geometrically meaningful initialization. Once this is done, Gauss-Newton optimization has to move the vertices only a little bit.

Design problems. The two-stage process consisting of initialization and optimization moves interactive design to the initialization phase. We have mentioned to ways: For cyclidic CAS which are initialized e.g. from a circular or conical mesh, we propose to design the latter, e.g. using [Liu et al. 2006]. General CAS are initialized from a curve network in the reference surface, and design of the CAS amounts to design of this network, e.g. making use of conformal mappings.

Further constraints related to architecture. As usual, real-world applications may entail creative extension and modification of the methods described above. Often the freedom provided by the initialization phase allows us to introduce further constraints.



Figure 22: Flexibility of CAS optimization. We show a sequence of deformed reference surfaces and circular arc structures approximating them.

An example is Figure 6 where one family of flow lines in a quadrilateral CAS should correspond to floor levels and is therefore constrained to be ‘horizontal’. The other family is orthogonal to the first. Optimization includes a term which tries to make equal the z coordinates of vertices which belong to the same level.

Fig.	$ V $	s_{prox}	s_{pl}	s_{cons}	s_{symm}	s_{ang}	$s_{\Delta,v}$	$s_{\Delta,e}$	T
3a	6008	0.01	$1 \cdot 10^{-5}$	$1 \cdot 10^{-6}$	—	$7 \cdot 10^{-5}$	0.3	0.6	1200
3b	2304	0.004	$7 \cdot 10^{-6}$	$2 \cdot 10^{-7}$	—	$1 \cdot 10^{-4}$	—	—	100
22a	2304	0.003	$7 \cdot 10^{-6}$	$3 \cdot 10^{-7}$	—	$1 \cdot 10^{-4}$	—	—	100
22b	2304	0.007	$7 \cdot 10^{-6}$	$3 \cdot 10^{-7}$	—	$1 \cdot 10^{-4}$	—	—	100
22c	2304	0.02	$2 \cdot 10^{-5}$	$1 \cdot 10^{-6}$	—	$1 \cdot 10^{-4}$	—	—	100
3c	486	0.04	0.16	$2 \cdot 10^{-3}$	—	0.09	2.2	2.4	60
5	649	0.01	$3 \cdot 10^{-6}$	$7 \cdot 10^{-7}$	—	$2 \cdot 10^{-5}$	0.07	0.07	10
6	3845	0.01	$2 \cdot 10^{-4}$	$7 \cdot 10^{-7}$	—	$9 \cdot 10^{-6}$	1.3	1.5	450
12	312	0.3	$2 \cdot 10^{-13}$	$2 \cdot 10^{-7}$	$3 \cdot 10^{-7}$	$1 \cdot 10^{-6}$	0.06	0.08	5
13	3054	0.007	$3 \cdot 10^{-2}$	$5 \cdot 10^{-4}$	—	—	0.13	0.09	600
17	2091	.008	$2 \cdot 10^{-2}$	$1 \cdot 10^{-3}$	$5 \cdot 10^{-2}$	$2 \cdot 10^{-3}$	0.1	0.06	500
1,18	1535	—	$3 \cdot 10^{-4}$	$2 \cdot 10^{-6}$	$2 \cdot 10^{-5}$	$2 \cdot 10^{-5}$	1.4	1.7	200
25	3097	0.002	$1 \cdot 10^{-4}$	$5 \cdot 10^{-5}$	—	$2 \cdot 10^{-4}$	0.01	0.007	225

Figure 23: Optimization details. This table uses the symbols s_{prox} , s_{cons} , s_{symm} , s_{ang} , which mean the respective maximum values of terms whose squares form the contributions f_{prox} , f_{cons} , f_{symm} , f_{ang} to the target function (1). We define $s_{pl} = \max |\frac{\pi}{2} - \angle(a_{vw}, n_v)|$ as a smoothness measure (given in degrees). As to the regularization terms, we give $s_{\Delta,v} = \max \|u - 2v + w\|$, $s_{\Delta,s} = \max \|s_{uv} - 2s_{vw} + s_{wz}\|$ in the notation of (2). Objects are scaled for bounding box size 1. T means computing time in seconds on a laptop with a dual core CPU with 2.4 GHz and 3GB memory.

Computation details. Figure 23 gives a detailed account of CAS quality. One can clearly see that optimization of triangular CAS does not work as well as the quad and hex cases, but still well enough for purposes of freeform architecture. For each CAS, optimization took 10–20 iterations. The weights of the individual contributions to (1) equal 100 for f_{prox} , f_{Δ} , while the other weights, which correspond to geometric constraints, are set to 1 in the first round of iteration and are multiplied by 10 in every round.

Limitations. The main limitation in computing CAS is the loss of shape flexibility when additional geometric conditions are imposed. While general CAS can approximate general curve networks in surfaces, provided they have the right connectivity and the right intersection angles, edge symmetry implies that flow lines of the CAS must follow principal curvature lines. This can lead to uneven panel sizes or alternatively to the introduction of T-junctions (for a similar problem, see Figure 6). 3D cyclidic structures have only a few more d.o.f. than 2D cyclidic CAS, so their ‘spatial’ component is rather too inflexible to allow freeform modeling.

In order to demonstrate the flexibility of circular arc structures, we

study a deformation sequence (Figures 3b and 22abc). We cover each surface by a triangle mesh combinatorially equivalent to the others and optimize it for the circle-packing property [Schiftner et al. 2009], which defines a discrete conformal mapping between those surfaces. By dualizing we generate hexagonal meshes which serve as initial values for circular arc structure optimization. The quality of these CAS, as seen in the table of Figure 23, is almost constant. In the final stage of deformation the surface exhibits curvature radii comparable to mesh edgelengeth and approximation quality reaches only 2% of the bounding box diameter, as opposed to the previous 0.4%.

Conclusion and future research. We have shown the applicability of circular arc structures in freeform architecture (congruent nodes, smooth skin, simple edge elements) and we have demonstrated that certain special circular arc structures have even more properties which are relevant for freeform building construction (support elements, offsets, derived developable strip models). Computation of circular arc structures is done by nonlinear optimization with geometrically meaningful initial guesses.

As to future research, we are convinced that circular arc structures represent an interesting object of discrete differential geometry, since they provide a discrete Möbius-invariant version of orthogonal parametrizations. CAS represent a certain kind of ‘surfaces of circles’, which have already proved their geometric capability (see e.g. [Bobenko and Schröder 2005]). Regarding applications and geometry processing, the architectural aspects have not yet been fully explored (we mention better initialization, design, further repetitions). The connection to CNC milling mentioned above definitely deserves further study.



Figure 24: Cyclidic CAS with offset, interior view. This CAS is also employed by Figures 1 and 18. Here the combinatorial singularities have been closed by non-cyclidic patches (courtesy H. Schmidhofer).

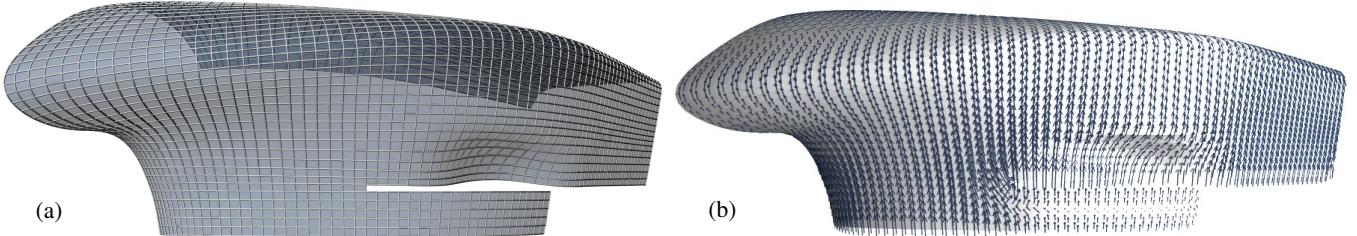


Figure 25: Statics properties. (a) Panel boundaries (as built) of Dongdaemun Design Plaza (Seoul) by Zaha Hadid Architects, have been optimized to become an orthogonal CAS. (b) Directions of panel boundaries suggested by statics properties [Schriftner and Balzer 2010] agree with (a) apart from the obvious notch disturbance.

Acknowledgments This research has been supported by the Austrian Science Fund (FWF) under grants No. S92-06 and S92-09 (National Research Network *Industrial Geometry*). We want to express our thanks to Zaha Hadid Architects, London, to Waagner Biro Stahlbau, Vienna, and to Evolute GmbH, Vienna, to be able to work on data which come from some of their recent projects. We are grateful to Martin Reis and Heinz Schmiedhofer for their help with construction details and illustrations.

References

- ALEXA, M., COHEN-OR, D., AND LEVIN, D. 2000. As-rigid-as-possible shape interpolation. In *Proceedings of SIGGRAPH 2000*, K. Akeley, Ed., 157–164.
- BISHOP, R. L. 1975. There is more than one way to frame a curve. *Amer. Math. Monthly* 82, 246–251.
- BOBENKO, A. I., AND HUHNEN-VENEDEY, E. 2011. Curvature line parametrized surfaces and orthogonal coordinate systems. Discretization with Dupin cyclides. arXiv preprint 1101.5955.
- BOBENKO, A., AND SCHRÖDER, P. 2005. Discrete Willmore flow. In *Symp. Geometry Processing*, M. Desbrun and H. Pottmann, Eds., Eurographics, 101–110.
- BOBENKO, A., AND SURIS, YU. 2008. *Discrete differential geometry: Integrable Structure*. American Math. Soc.
- CECIL, T. 1992. *Lie Sphere Geometry*. Springer.
- DEGEN, W. 2002. Cyclides. In *Handbook of Computer Aided Geometric Design*, G. Farin, J. Hoschek, and M.-S. Kim, Eds. Elsevier, 575–601.
- DO CARMO, M. 1976. *Differential Geometry of Curves and Surfaces*. Prentice-Hall.
- EIGENSATZ, M., KILIAN, M., SCHIFTNER, A., MITRA, N., POTTMANN, H., AND PAULY, M. 2010. Paneling architectural freeform surfaces. *ACM Trans. Graphics* 29, 4, #45, 1–10.
- FARIN, G., HOSCHEK, J., AND KIM, M.-S., Eds. 2002. *Handbook of Computer Aided Geometric Design*. Elsevier.
- FU, C.-W., LAI, C.-F., HE, Y., AND COHEN-OR, D. 2010. K-set tilable surfaces. *ACM Trans. Graphics* 29, 4, #44, 1–6.
- GU, X. D., AND YAU, S.-T. 2008. *Computational Conformal Geometry*. International Press.
- HERTRICH-JEROMIN, U. 2003. *Introduction to Möbius differential geometry*. Cambridge University Press.
- HUHNEN-VENEDEY, E. 2007. *Curvature line parametrized surfaces and orthogonal coordinate systems. Discretization with Dupin cyclides*. Master's thesis, TU Berlin.
- KELLEY, C. T. 1999. *Iterative Methods for Optimization*. SIAM.
- LEOPOLDSEDER, S. 2001. Algorithms on cone spline surfaces and spatial osculating arc splines. *Comput. Aided Geom. Design* 18, 505–530.
- LIU, Y., POTTMANN, H., WALLNER, J., YANG, Y.-L., AND WANG, W. 2006. Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graphics* 25, 3, 681–689.
- MARTIN, R. R., DE PONT, J., AND SHARROCK, T. J. 1986. Cyclide surfaces in computer aided design. In *The Mathematics of Surfaces*, J. A. Gregory, Ed. Clarendon Press, Oxford, 253–268.
- POTTMANN, H., LIU, Y., WALLNER, J., BOBENKO, A., AND WANG, W. 2007. Geometry of multi-layer freeform structures for architecture. *ACM Trans. Graphics* 26, 3, #65, 1–11.
- POTTMANN, H., SCHIFTNER, A., BO, P., SCHMIEDHOFER, H., WANG, W., BALDASSINI, N., AND WALLNER, J. 2008. Freeform surfaces from single curved panels. *ACM Trans. Graphics* 27, 3, #76, 1–10.
- POTTMANN, H., HUANG, Q., DENG, B., SCHIFTNER, A., KILIAN, M., GUIBAS, L., AND WALLNER, J. 2010. Geodesic patterns. *ACM Trans. Graphics* 29, 4, #43, 1–10.
- PRATT, M. J. 1995. Cyclides in computer aided geometric design II. *Comput. Aided Geom. Design* 12, 131–152.
- SAUER, R. 1970. *Differenzengeometrie*. Springer.
- SCHIFTNER, A., AND BALZER, J. 2010. Statics-sensitive layout of planar quadrilateral meshes. In *Advances in Architectural Geometry 2010*, C. Ceccato et al., Eds. Springer, 221–236.
- SCHIFTNER, A., HÖBINGER, M., WALLNER, J., AND POTTMANN, H. 2009. Packing circles and spheres on surfaces. *ACM Trans. Graphics* 28, 5, #139, 1–8.
- SINGH, M., AND SCHAEFER, S. 2010. Triangle surfaces with discrete equivalence classes. *ACM Trans. Graphics* 29, 4, #46, 1–7.
- SONG, X., AIGNER, M., CHEN, F., AND JÜTTLER, B. 2009. Circular spline fitting using an evolution process. *J. Comp. Appl. Math.* 231, 423–433.
- TOLEDO, S., 2003. TAUCS, a library of sparse linear solvers. C library, <http://www.tau.ac.il/~stoledo/taucs/>.
- WALTON, D. J., AND MEEK, D. S. 1995. Approximating smooth planar curves by arc splines. *J. Comp. Appl. Math* 59, 221–231.
- ZENG, W., YIN, X., ZHANG, M., LUO, F., AND GU, X. D. 2009. Generalized Koebe's method for conformal mapping multiply connected domains. In *Proc. Geom. Phys. Modeling*, 89–100.