

Geodesic Patterns for free-form Architecture

MPDA'18 Master Thesis — UPC-ETSAV

Alan Rynne Vidal

Abstract

In this publication, we implement and analyze different existing techniques to obtain cladding solutions using exclusively straight planks, either wood or metal, and analyze their benefits and weaknesses from an architectural standpoint. We will concentrate on the generation of *geodesic curves*, more specifically *straightest geodesics*, which minimize distance between two points on curved surface. For cost and manufacturing reasons, geodesic planks are favorable for cladding solutions, as they can be manufactured and assembled using inexpensive construction techniques. We will also explain some basic structural analysis concepts to check the admissibility of the generated solutions.

Contents

1	Introduction	1
2	Background	2
3	Geometric properties	5
4	Algorithmic strategies	8
5	Quality Assessment	22
6	Conclusions	24
	References	24
A	Appendix	26

1 Introduction

With the development of advanced 3D CAD tools in the last decades, architects are now able to incorporate complex freeform surfaces into their designs. This new capabilities have given birth to new architectural possibilities and, more importantly, new needs and challenges for the industry that can often be solved by the means of *Differential Geometry* (Carmo 2016) and, more specifically, *Discrete Differential Geometry* (Keenan Crane 2013), which studies the discrete counterparts of *Differential Geometry*. This new fusion between architecture and mathematics gave birth to the field known as *Architectural Geometry* (Pottmann, Schiftner, and Wallner 2008; Pottmann 2010), which has been further developed during the past years.

Given the scale architectural projects, this smooth freeform surfaces cannot be built as designed in 3D CAD applications and, therefore, must be divided in smaller, simpler parts that could be easily manufacturable. This process is called *Rationalization*, and is not a trivial task; since the *Rationalization* of a surface implies a loss of smoothness in the resulting shape, it must be done in such a way that minimizes this loss. The selected shape of the generated panels also plays a key role when approaching this problems, since it is clear the final appearance of the design will vary greatly depending on whether the panels are triangular, quad, hexagonal or long strips.

Constructibility of the selected design (panel planarity, curvature, error margins) is of great concern in rationalization methods for architectural purposes, looking for robust and precise algorithms, as opposed to related Computer Graphics methods, whose main focus is usually related with execution times & speed.

In this publication we center on the generation of *Geodesic Panels on Freeform Surfaces*, meaning, seamlessly covering any given surface with a pattern of **planks**; *thin, long, straight panels*, which must be *developable* and rectangular (or quasi-rectangular) in shape and must approximate the surface by being bent only on their weak axis. We will introduce the basic geometric properties behind this concept and the algorithmic implementation of several of the methods. We will also explain some concepts developed for optimizing any given shape towards *developability*.

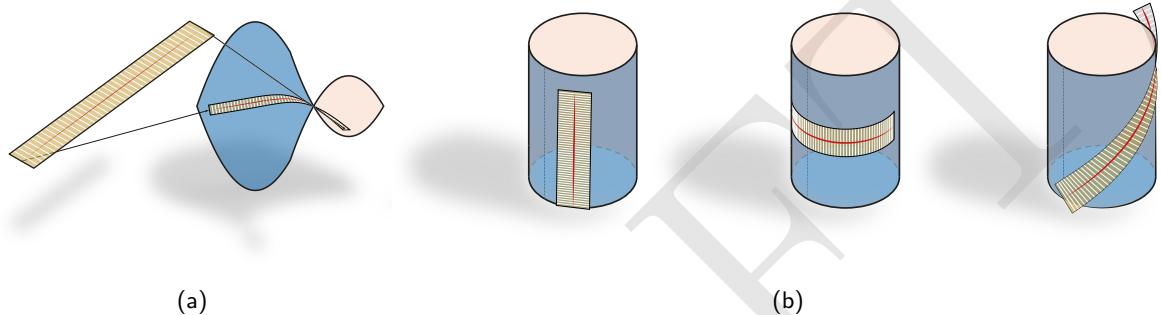


Fig. 1: Straight planks bent on a hyperbolic paraboloid (a) and a cylindrical surface (b).

2 Background

Examples of the application of *Architectural Geometry* are abundant in today's architecture. Architect Frank Ghery is known for his work ground-breaking work on the design of architectural shapes based purely on *developable* patches (Stein, Grinspun, and Crane 2018), which can be easily covered by a variety of panel patterns. Many of his buildings have become iconic landmarks of the cities they are built in



Fig. 2: Famous Ghery projects (right to left): Guggenheim Museum (Bilbao, 1997), Walt Disney Concert Hall (Los Angeles, 2003) and the Guggenheim Abu Dhabi winning proposal (Abu Dhabi, -)

Covering surfaces with long rectangular, or quasi-rectangular panels of equal width, is a topic widely explored in traditional boat building, and has also been recently applied in several architectural projects, such as Toyo Ito's Minna No Mori (Japan), or the interior cladding at Frank Ghery's Burj Khalifa (Dubai) (Meredith and Kotronis 2013).



Fig. 3: Interior view of Toyo Ito's Mina No Mori wooden lattice roof (a) and exterior view of the lattice under construction (b).



Fig. 4: Burj Khalifa interior ceiling constructed from wooden planks (a) and final paneling solution (b).

NOX Architects, an award winning architecture firm, is famous for building complex shapes using planks. Their design methodology included the creation of very precise models, and manually covering these models with paper strips until a final cladding solution could be achieved fig. 5

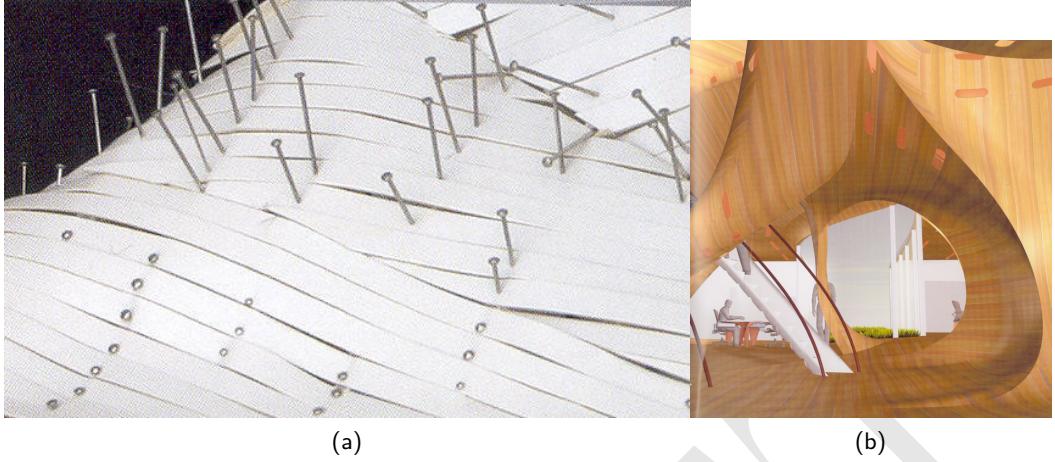


Fig. 5: NOX Architects paper models & final rendered design solution.

Some recently finished buildings would have also benefitted from the optimization of paneling and/or beam layout like the Yeoju Golf project by Shigeru Ban, which involved the construction of complex laminated wood shapes fig. 6



Fig. 6: Shigeru Ban's Yeoju Golf Resort. Right: Pre-assembled complex lattice. Left: Aerial view of lattice under construction.

Geodesic curves on manifolds and their calculation is also important for this topic (V. Surazhsky et al. 2005; Cheng et al. 2016, Polthier and Schmies (1998)) as well as the concept of *developability* of surfaces and their discrete counterparts (see Carmo 2016, and more recently Stein, Grinspun, and Crane (2018)). In this publication, all of the geodesics will be generated with the *initial value problem*.

The concept of *mesh developability* has also been widely studied in the past, although strictly speaking, it will not be the object of this publication. Previous attempts at mesh developability can be viewed in fig. 7. The latest and more robust implementation was developed by Stein, Grinspun, and Crane (2018) and is based on reducing the angle defect on each vertex on a mesh, by forcing the normals of the faces adjacent to each vertex to be parallel; this causes that each vertex on the given mesh becomes either flat or a *hinge* fig. 8a, effectively concentrating the gaussian curvature of the original mesh on the hinges. This procedure provides very smooth developable results on any given mesh, but most of the time, the resulting *developable* will exhibit

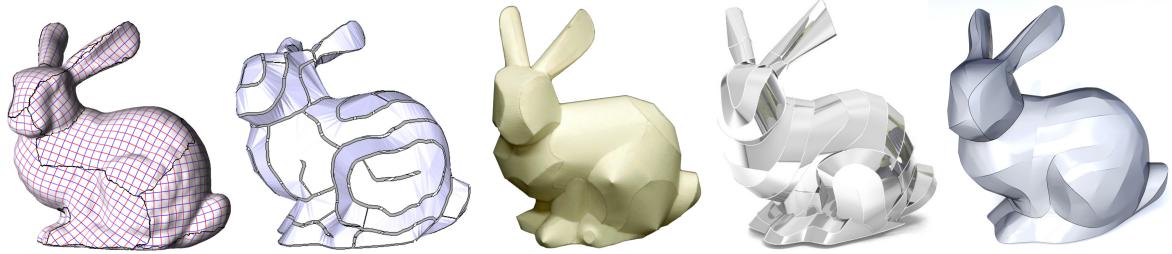


Fig. 7: Mesh developability techniques. Left to right: Julius, Kraevoy, and Sheffer (2005), Mitani and Suzuki (2004), Shatz, Tal, and Leifman (2006) Y. Liu et al. (2006) and Stein, Grinspun, and Crane (2018) [^KraneDevelop]

creases and sharp edges in some areas, which are heavily influenced by the topology of the starting mesh. Also, the resulting *flat* configuration of the mesh usually contains self-intersections and complex boundary shapes that would increase complexity on the construction and assembly stages in an architectural scale project.



Fig. 8: Left: Any given vertex on a mesh (left) will become either flat (centre) or a hinge (right). Right: Developalizing a sphere. Results highly depend on the initial mesh topology[^KraneDevelop]

3 Geometric properties

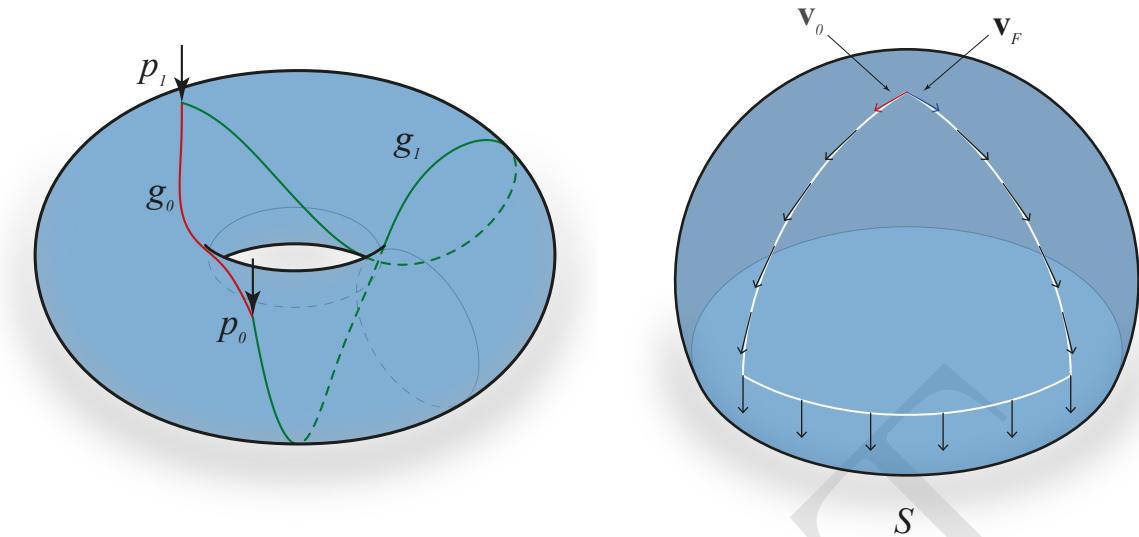
3.1 Geodesic curves

In differential geometry, a *geodesic curve* is the generalization of a straight line into curved spaces. It is usually understood as the shortest path between two points on a given surface, although this may not always be the case.

Also, in the presence of an *affine connection*, a geodesic is defined to be a curve whose tangent vectors remain parallel if they are transported along it. We will explore the notion of vector *parallel transport* in the following sections.

For triangle meshes, shortest polylines cross edges at *equal angles*.

Finding the truly shortest geodesic paths requires the computation of distance fields (see Carmo 2016; Kimmel and Sethian 1998)



- (a) The concept of ‘shortest geodesics’: curves g_0 (red) and g_1 (green) are both geodesic curves of a torus, although g_1 is more than double the length of g_0 . g_0 was computed solving the *initial value problem*, while g_1 was computed using the boundary value problem* and (b) parallel transport of a vector \mathbf{v}_0 on a ‘piecewise geodesic’ path.

Algorithmic ways of generating geodesics

The computation of geodesics on smooth surfaces is a classical topic, and can be reduced to two different solutions, depending on the initial conditions of the problem (Deng 2011):

Initial value problem given a point $p \in S$ and a vector $v \in T_p S$, find a geodesic which is incident with p , such that the tangent vector to g at p is v .

Boundary value problem given two points $p_1, p_2 \in S$ find a geodesic g connecting p_1 and p_2 .

Both problems have different ways of being solved either numerically, graphically or by the means of simulations. The initial value problem can be solved using the concept of *straightest geodesics* (Polthier and Schmies 1998), whereas the boundary value problem has a very close relation with the computation of *shortest paths* between two points on a surface.

3.2 Developable surfaces

It is also important to introduce the concept of the *developable surface*, a special kind of surface that have substantial and variable normal curvature while guaranteeing zero gaussian curvature, and as such, this surfaces can be *unrolled* into a flat plane with no deformation of the surface. This surfaces have been an extremely important design element in the practice of known architect Frank Ghery and explained in (Shelden 2002).

There are several ways of generating a developable surface using a curve in space. The most simple way is the following.

1. Select the starting point of the curve
2. Obtain the perpendicular frame of the curve at the specified parameter.
3. Deconstruct the frame into it's X, Y & Z vectors.
4. Select one of the vectors at all the sample points
5. Draw a line using the selected vector at the start point of the curve.

6. Create a surface using the rulings obtained in step 5.

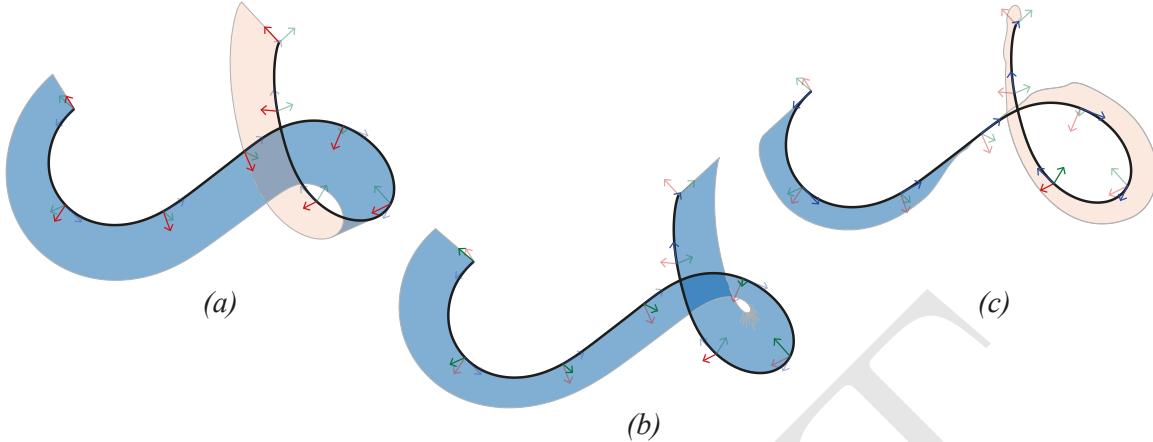


Fig. 9: Developable surface generated using curve a geodesic curve. (a) using the X component of the curve's perp frame; (b) using the Y component & (c) using the Z component (tangent of the curve).

Not all developable surfaces are of interest in an architectural sense. We must make a distinction between *noisy* and *smooth* developable surfaces, as can be seen in Fig. 10. We are specifically interested in generating *smooth* developable surfaces, which can be easily manufactured, as opposed to the complex bending operations that manufacturing a *noisy* developable surface will require



Fig. 10: Noisy to smooth sheet (Stein, Grinspun, and Crane 2018)[^KraneDevelop]

3.3 Properties to aim for

We will first introduce the properties we aim for when generating geodesic panels:

Geodesic property

- Long Thin panels that bend about their weak axis
- Zero geodesic curvature
- Represent the shortest path between two points on a surface

Constant width property

- Panels whose original, unfolded shape is a rectangle.
- The only way this can happen is if the entire surface is developable.
- For all other surfaces:
 - Assuming no gaps between panels
 - Panels will not be exactly rectangular when unfolded
 - **Requirement:** Geodesic curves that guide the panels must have approximately constant distance from their neighborhood curves.

Developable (or ‘pure-bending’) property

- Bending panels on surfaces changes the distances in points only by a small amount so,
- A certain amount of twisting is also present in this applications.
Some methods in this chapter do not take into account this property.

Objectives

Look for a system of geodesic curves that covers a freeform surface in a way that:

1. They have approximate constant distance with it’s neighbours.
2. This curves will serve as guiding curves for the panels.
3. The panels are to cover the surface with ***no overlap*** and ***only small gaps***

4 Algorithmic strategies

In this section, we will introduce the different existent methods for generating geodesic curve patterns that will be used to model the final panels, using methods indicated in sec. 4.6.

4.1 Parallel Transport Method

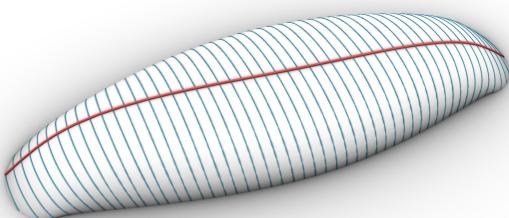


Fig. 11: Example of parallel transport method. Generatrix geodesic g (red) and geodesics g^\perp generated from a parallel transported vector (blue) computed given a point and a vector \mathbf{v} tangent to the surface, in both positive and negative directions.

This method, described in (Pottmann et al. 2010), allows for the generation of a system of geodesic curves where either the maximum distance or the minimum distance between adjacent points occurs at a prescribed location.

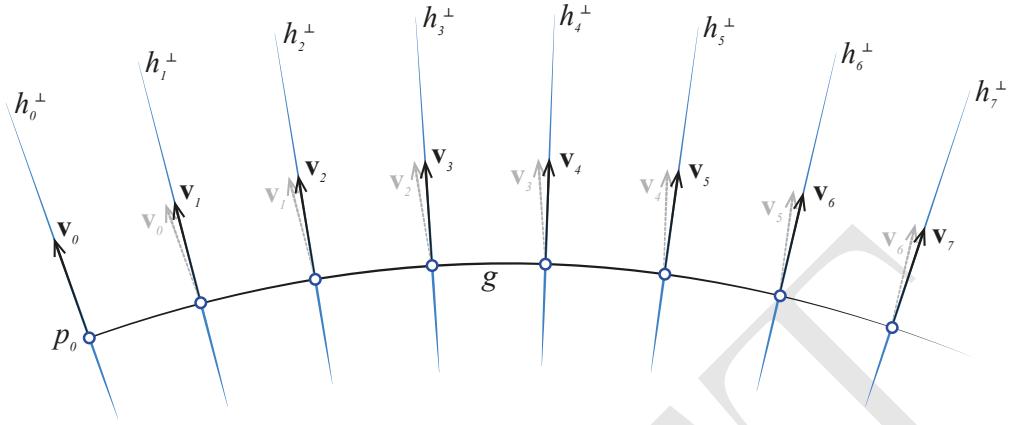


Fig. 12: Parallel transport method: We can parallel transport a vector along a curve placed on a surface by iteratively projecting the previous vector \mathbf{v}_{i-1} over the tangent plane of v_i and normalizing it's length.

In differential geometry, the concept of *parallel transport* (see fig. 11) of a vector \mathbf{V} along a curve S contained in a surface means moving that vector along S such that:

1. It remains tangent to the surface
2. It changes as little as possible in direction
3. It's length remains unchanged

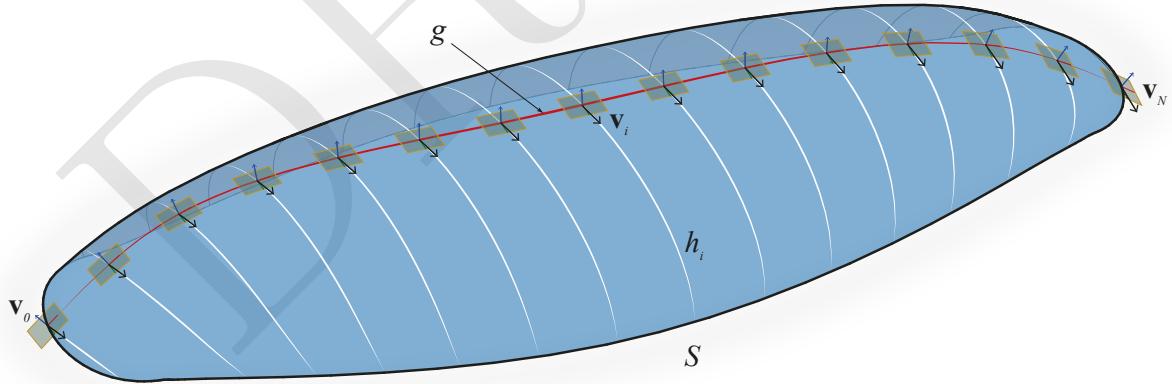


Fig. 13: Parallel transport method. Main geodesic g , tangent planes at each point p_i , transported vectors \mathbf{v}_i , generated geodesics h_i .

Procedure

Following this procedure, *extremal distances between adjacent geodesics occur near the chosen curve*. Meaning:

Algorithm 1 Parallel Transport Method

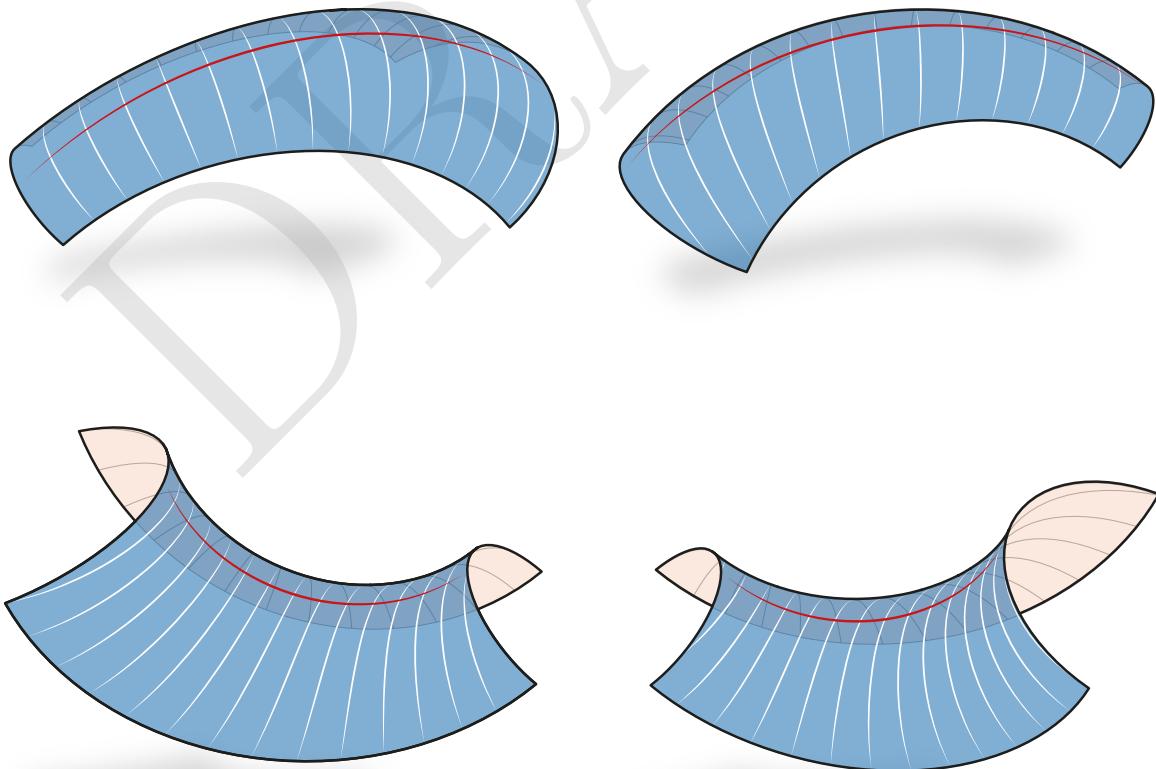
Require: A surface Φ , represented as a triangular mesh (V, E, F)

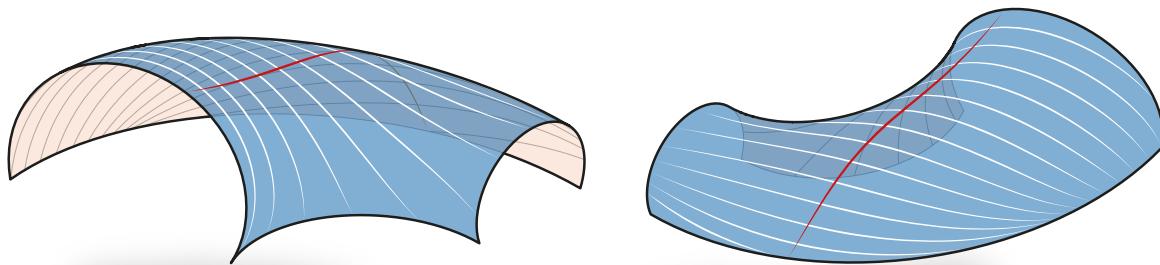
- 1: Place a geodesic curve g_x along S such that it divides the surface completely in 2.
- 2: Divide the curve into N equally spaced points p with distance W .
- 3: Place a vector v onto p_0
- 4: Parallel transport that vector along g_x as described in [fig:parTransProc].
- 5: **for all** points p_i where $i = 0, \dots, M$ **do**
- 6: Generate geodesic curve $+g_i$ and $-g_i$ using vector v_i and $-v_i$ respectively.
- 7: Join $+g_i$ and $-g_i$ together to obtain g_i
- 8: Add g_i to output.
- 9: **end for**

Ensure: Set of geodesic curves g_i , where $i = 0, \dots, M$

1. For surfaces of **positive curvature**, the parallel transport method will yield a 1-geodesic pattern on which the *maximum distance* between curves will be W .
2. On the other hand, for surfaces of **negative curvature**, the method will yield a 1-geodesic pattern with W being the *closest (or minimum)* distance between them.

The placement of the first geodesic curve and the selection of the initial vector are not trivial tasks. For surfaces with high variations of surfaces, the results might be unpredictable and, as such, this method is only suitable for surfaces with nearly constant curvature. Other solutions might involve cutting the surface into patches of nearly-constant curvature, and applying the *parallel transport method* independently on each patch.





Results of the parallel transport method applied to surfaces of different curvature: (a) positive, (b) negative and (c) doubly curved.

4.2 Evolution Method

Two main concepts are covered in this section, both proposed by (Pottmann et al. 2010): the first, what is called the *evolution method*, and a second method based on *piecewise-geodesic* curves, curves that are not geodesic, but are composed of several connected geodesic segments.

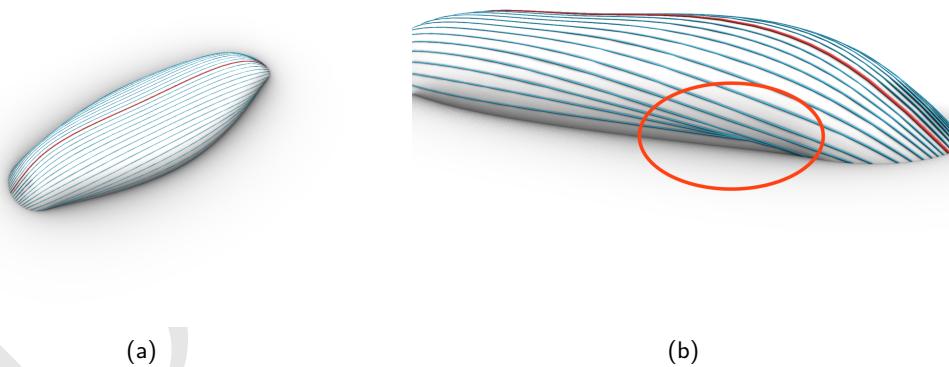


Fig. 14: Surface covered by a 1-geodesic pattern using the evolution method without introducing breakpoints. (a) shows an overview of the result; while (b) highlights the intersection point of several geodesic curves. This problem will be addressed by introducing the concept of ‘piece-wise’ geodesic curves; which are curves that are not geodesics, but are composed of segments of several connected geodesic curves.

The evolution method

Starting from a source geodesic somewhere in the surface:

- Evolve a pattern of geodesics iteratively computing ‘next’ geodesics.
- ‘Next’ geodesics must fulfill the condition of being at approximately constant distance from its predecessor.

- ‘Next geodesics’ are as described in fig. 15

Distances between geodesics

1. No straight forward solution.
 1. Only for rotational surfaces (surfaces with evenly distributed meridian curves).
 2. But a first-order approximation of this distance can be approximated:

The method proposed in Pottmann et al. (2010) is based on the calculation of *Jacobi Fields*, vector fields perpendicular to a geodesic that approximate the distance between one geodesic and the next in an infinitesimal distance.

Starting at time $t = 0$ with a geodesic curve $g(s)$, parametrized by arc-length s , and let it move within the surface.

A snapshot at time $t = \varepsilon$ yields a geodesic g^+ near g .

$$g^+(s) = g(s) + \varepsilon \mathbf{v}(s) + \varepsilon^2(\dots) \quad (1)$$

The derivative vector field \mathbf{v} is called a *Jacobi field*. We may assume it is orthogonal to $g(s)$ and it is expressed in terms of the geodesic tangent vector g' as:

$$\mathbf{v}(s) = w(s) \cdot R_{\pi/2}(g'(s)), \quad \text{where } w'' + Kw = 0 \quad (2)$$

Since the distance between infinitesimally close geodesics are governed by Eq. 2, that equation also governs the width of a strip bounded by two geodesics at a small finite distance.

We propose a simpler approach to this problem, since the computation of the jacobi field can give very high margins of error and ‘next’ geodesic g_{i+1} will still need to be computed using the initial value problem. The modification is the following: we can substitute the length of the jacobi field for the user’s specified width W and find the best fit geodesic that fits this *signed distances*.

Obtaining the ‘next’ geodesic

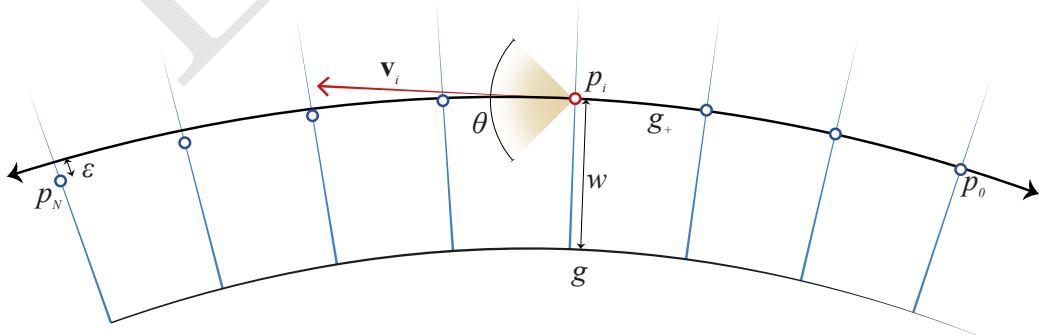


Fig. 15: Calculating the ‘next’ geodesic

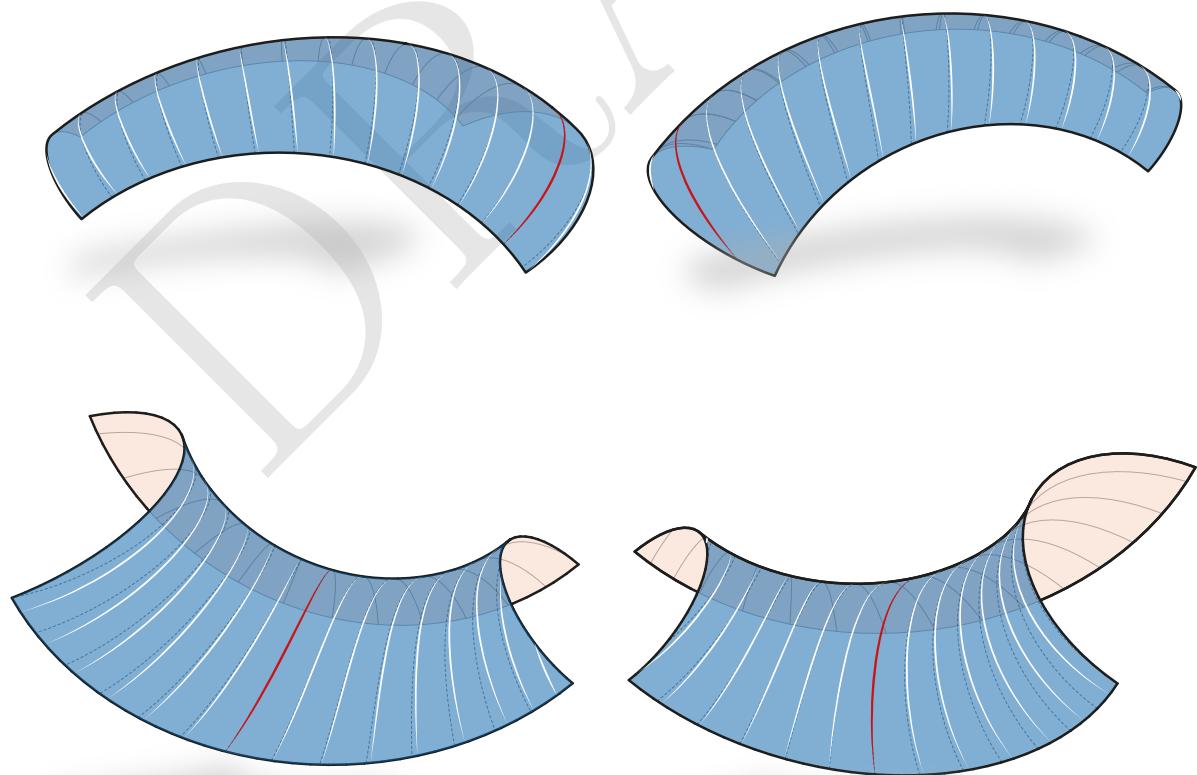
Algorithm 2 Calculate the best-fit geodesic

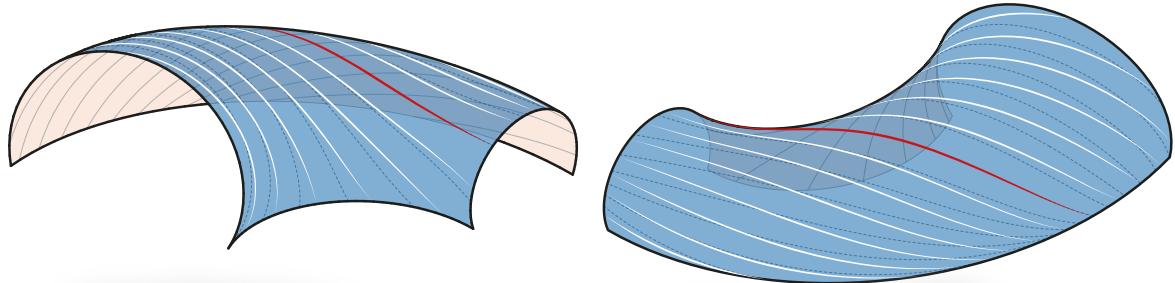
Require: Curve g_i , perp geodesics h^\perp , desired width W and an angle threshold α .

- 1: Obtain a point p_i at distance W from the starting point of each g_i^\perp
- 2: Select any point p_i as the start point and name it p_0 .
- 3: Select the tangent vector \mathbf{v}_T of g_i^\perp at p_0 .
- 4: Rotate \mathbf{v}_T 90° around the normal of Φ at p_0 .
- 5: Generate an initial geodesic g_i .
- 6: Obtain error measure ϵ as the least squares difference between the desired distance W and the actual distance D
- 7: Find the geodesic that has the least error by rotating v_T by a small amount each step.

Ensure: The next geodesic curve that best fits W .

Any given surface can be completely covered in this manner by recursively computing the next geodesic using the previous, given some margin of error.





Results of the evolution method applied to surfaces of different curvature: (a) positive, (b) negative and (c) doubly curved.

Algorithm 3 Evolution Method

Require: A surface Φ , represented as a triangular mesh (V,E,F), a desired width W , and a starting geodesic curve g_0 .

- 1: Place a geodesic curve g_i along Φ and name it g_0 .
- 2: Divide the curve into equally N number of sample points.
- 3: **for all** points p_i where $i = 0, \dots, N$ **do**
- 4: Find tangent vector \mathbf{v}_i^T of g
- 5: Rotate \mathbf{v}_i^T by $\frac{\pi}{4}$
- 6: Generate geodesic h_i^\perp from p_i and \mathbf{v}_g^T
- 7: **end for**
- 8: Compute BEST FIT GEODESIC g_{i+1} ??
- 9: **if** No best fit is found **then**
- 10: BREAK
- 11: **end if**
- 12: Make g_{i+1} the current geodesic for next iteration.

Ensure: Set G of geodesic curves g_i , where $i = 0, \dots, M$

Piecewise evolution method

We can modify the implementation of the evolution method to be able to control the distance between geodesics even in areas with a high rate of change in curvature. In order to prevent this unwanted width variations, instead of looking for the best overall geodesic, we will look for the best geodesic that fits the largest interval of h_i 's possible fig. 16, given a specified threshold ϵ . Once this interval is obtained, the current geodesic is split and a new geodesic will be generated. This process will be repeated until the curve crosses all h_i 's (some exceptions apply).

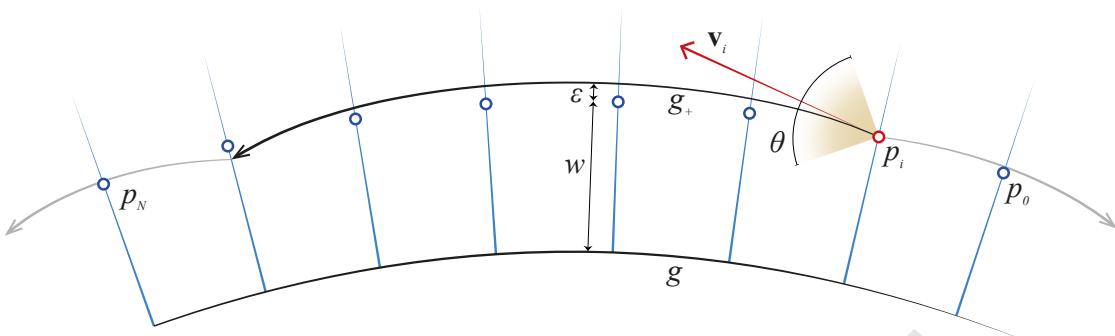


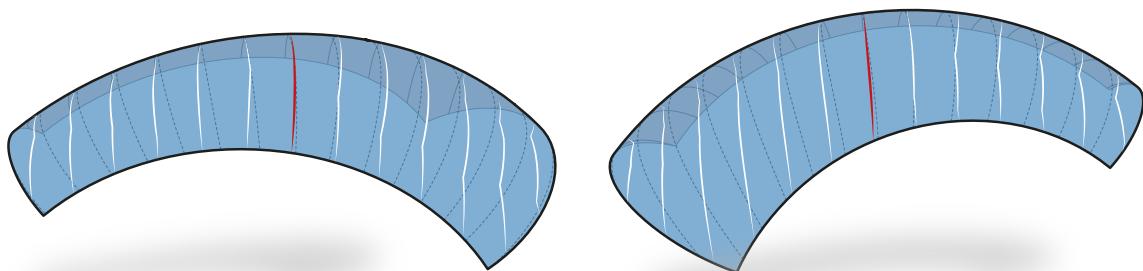
Fig. 16: Calculating the best piece-wise geodesic

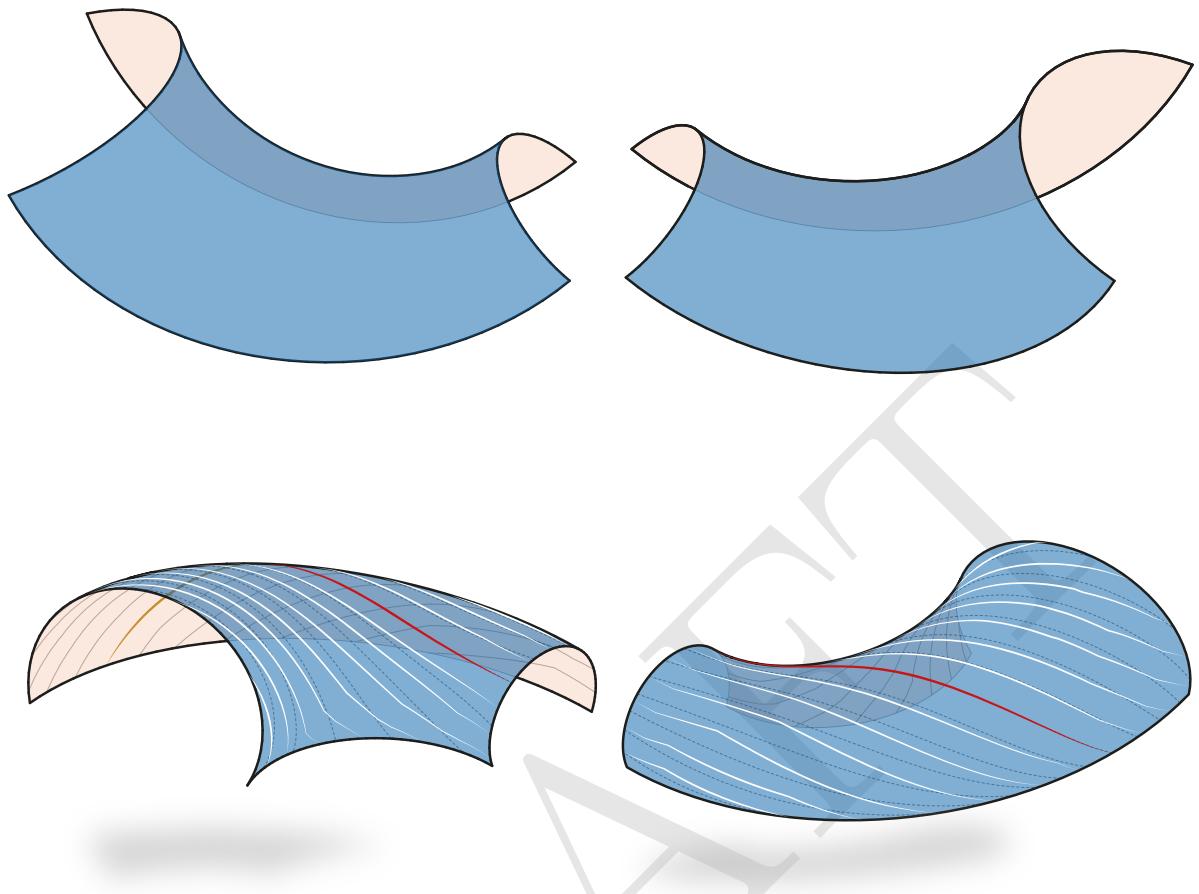
Algorithm 4 Calculate the best-fit piecewise geodesic

Start from: Step ?? of Algorithm 3.

- 1: Find the largest interval I of g_{i+1} that does not exceed ϵ by a given
- 2: **if** g_i satisfies the ϵ for all h^\perp 's **then**
- 3: BREAK
- 4: **end if**
- 5: Split g_i
- 6: Remove used h^\perp from list.
- 7: Start Algorithm 3 again using the new h^\perp , and the start/end point of g_i

On end: Continue from Step ?? of Algorithm 3.





Piecewise evolution method applied to: (a) a positive curvature surface, (b) a negative curvature surface and (c) a doubly curved surface.

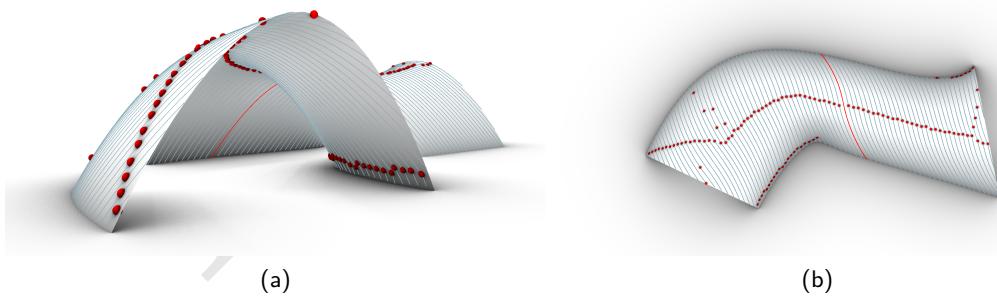


Fig. 17: (a) Perspective view; (b) Top View. In red, the initial geodesic curve; in blue, the generated piecewise geodesic pattern; the red dots are breakpoints in the piecewise geodesic curves.

4.3 Level-Set Method

We can also compute geodesic curve patterns on a surface as a series of levels (or level-sets) of a scalar-valued function assigned on each vertex of a mesh. This concept was first introduced in Wallner et al. (2010) and is based on concepts developed in the geodesic active contour method.

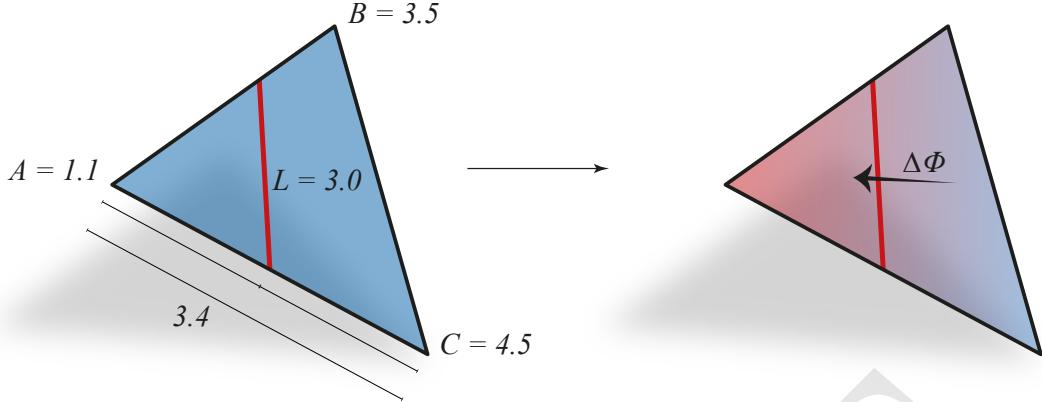


Fig. 18: Calculating a level set on a single face

The computation of level-sets on a triangle mesh is a fairly simple task. Since the values of the function are assigned per vertex, the line of a constant value of the function is obtained by linear interpolation on each of the mesh faces fig. 18. The gradient of the scalar-valued function also needs to be calculated, which results in a vector field which is constant on each face. Once the gradient has been computed, the geodesic curvature of a level set can be obtained as the divergence of the gradient vector field over each vertex:

$$K_g = \operatorname{div}\left(\frac{\nabla\phi}{\|\nabla\phi\|}\right)$$

In order that the obtained levels approximate the surface, we can approach the problem as the minimization of a combination of error measurements:

$$F_{min} = F_k + \lambda F_\nabla + \nu F_w$$

where F_k measures the deviation of the scalar function geodesic property, F_∇ is computed for regularization purposes and F_w measures the constant-width deviation; λ and ν are user defined weighting values. F_k and F_w will only vanish at the same time in the case of developable surfaces. These values can be computed on a mesh (V, E, F) using the following formulas, explained with more detail in Pottmann et al. (2010):

$$\begin{aligned} F_k &= \sum_{\mathbf{v} \in V} \mathcal{A}(\mathbf{v}) \left(\operatorname{div}\left(\frac{\nabla\phi}{\|\nabla\phi\|}(\mathbf{v})\right) \right)^2 \\ F_\Delta &= \operatorname{area}(S) \sum_{\mathbf{v} \in V} \mathcal{A}(\mathbf{v}) \Delta\phi(\mathbf{v})^2 \\ F_w &= \sum_{f \in F} \operatorname{area}(f) \left(\|\nabla\phi(f)\| - \frac{h}{w} \right)^2 \end{aligned}$$

The algorithm would be started with a set of randomly placed values on each vertex of the mesh fig. 19. The implementation used in this paper differs from the originally proposed in Pottmann et al. (2010), in which the Gaussian-Newton method is used for minimization and all high order derivatives are computed analytically. In our implementation, we explored the application

of gradient-less minimization algorithms, in particular BOBYQA (Bound Optimization BY Quadratic Approximation) although this decision causes computation times to increase severely.

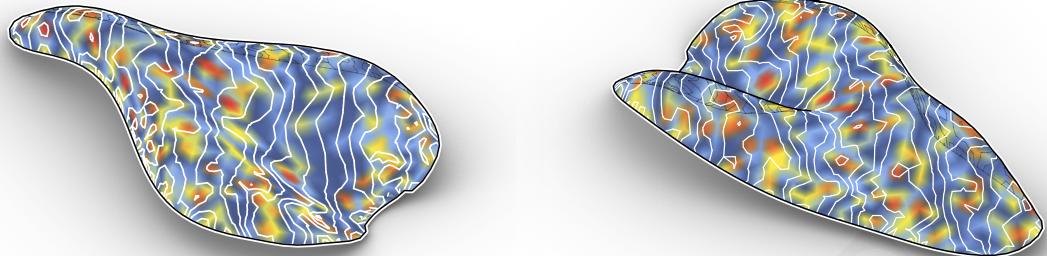


Fig. 19: Starting conditions for the level set method on a double curve surface

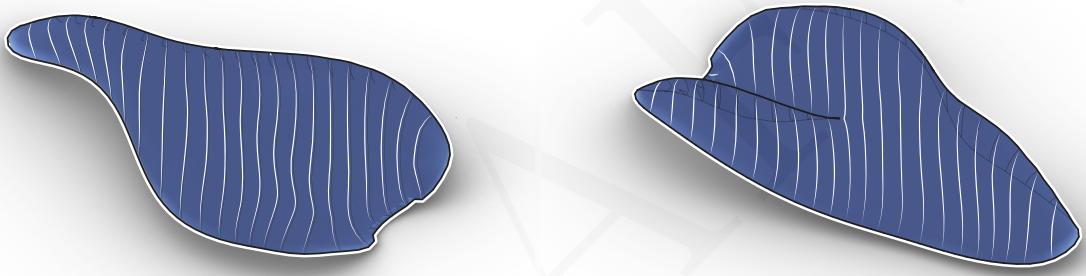


Fig. 20: Final result of the level set method over a complex double curve surface.

4.4 Geodesic Webs

We can also expand the Level-Set method explained in the previous section from one family of curves to several families of interconnected quasi-geodesic curves by simply computing several sets of scalar functions on each vertex of the surface.

This can be easily done by adding another equation to our minimization problem.

$$\nu \dot{F}_{angle} = \nu \sum_{f \in F} \frac{\text{area}(f)}{\text{area}(S)} \langle \frac{\nabla \phi_i}{\|\nabla \phi_i\|} \cdot R_{\frac{\pi}{2}-\alpha} \left(\frac{\nabla \phi_j}{\|\nabla \phi_j\|} \right) \rangle^2 \quad (3)$$

where ν is the weight given to F_{angle} , $R_{\frac{\pi}{2}-\alpha}$ means the rotation by the angle $\frac{\pi}{2} - \alpha$ in the respective face.

4.5 Geodesic Vector fields

The level-set method described in Sec. 4.3 is not suitable for arbitrary surfaces, and therefore it must be adapted to achieve the desired result. In this section, we introduce the concept of

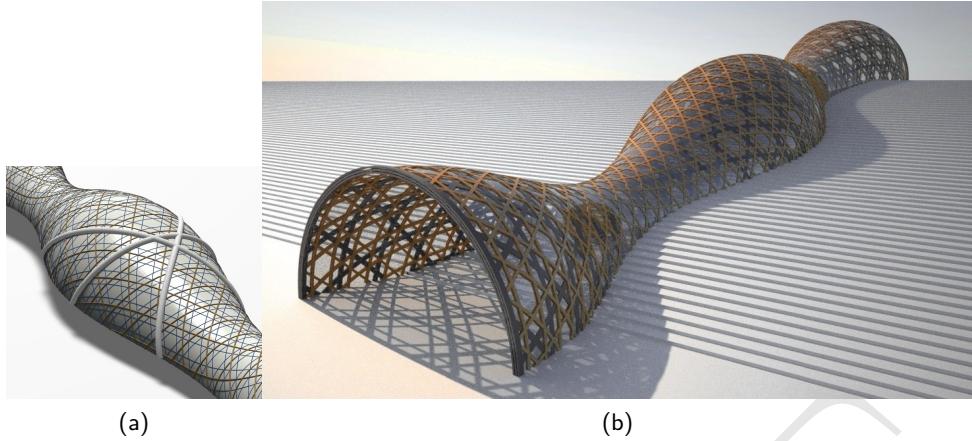


Fig. 21: Beam structure generated by the geodesic web method.

Geodesic Vector-fields (Pottmann et al. 2010) to divide the surface into patches that could be easily covered by equal width panels using the level-set method.

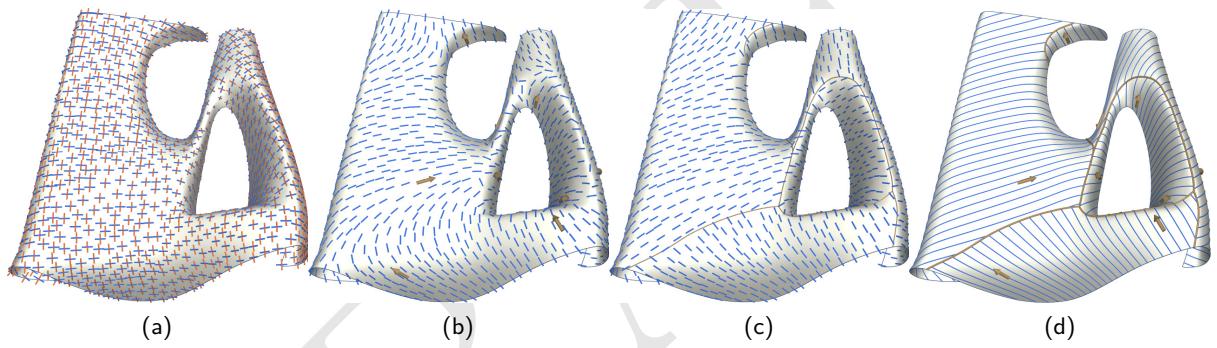


Fig. 22: Geodesic vector fields

A geodesic vector field on a surface S is geodesic if it consists of tangent vectors of a 1-parameter family of geodesics covering S (Pottmann et al. 2010).

Considering the surface S as a triangle mesh (V, E, F), the vector field is represented by unit vectors \mathbf{v}_f attached to the incenters of face $f \in F$.

$$\mathbf{v}_{f_2} = \mathbf{v}_{f_1} + J_{f_1} \dot{(m_{f_2} - m_{f_1})} + r_e \quad (4)$$

$$\text{where } J_f = XXXX$$

The vector field $\mathbf{v} = (\mathbf{v}_f)_{f \in F}$ is a geodesic vector field if its length is constant for all faces f , and we can find a collection of coefficients $\mathbf{g} = (\mathbf{g}_f)_{f \in F}$ with $\mathbf{g}_f = (g_{1,f}, g_{2,f}, g_{3,f})$, such that eq. 4 holds with \mathbf{r} 's below some threshold.

The following equation quantifies how well \mathbf{v} satisfies the geodesic property:

$$Q(\mathbf{v}) = \min_{g \in R^{3|F|}} \quad (5)$$

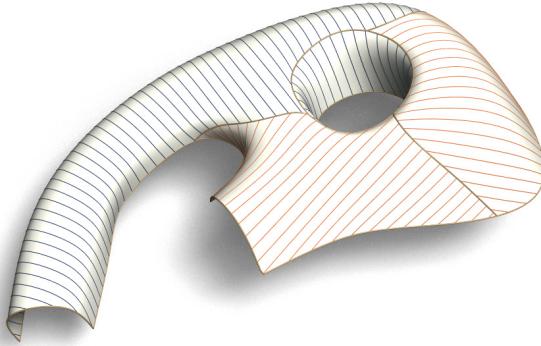


Fig. 23: Final result of geodesic vector fields method

4.6 Panel modeling

In this section, we will discuss several ways to generate panels from curves lying on a given surface:

The Bi-Normal Method

The second method for defining panels, once an appropriate system of geodesics has been found on Φ , works directly with the geodesic curves.

Assume that a point $P(t)$ traverses a geodesic s with unit speed, where t is the time parameter. For each time t there is:

- a velocity vector $T(t)$
- the normal vector $N(t)$
- a third vector $B(t)$, the *binormal vector*

This makes $T.N.B$ a *moving orthogonal right-handed frame*

Although this method generates mathematically correct developable surfaces, Pottmann et al. (2010) describes it is unclear that the boundaries of the panels generated with this technique will correspond to real constructive requirements for the design, so we concentrate our efforts on method explained in the following section.

Tangent-developable method

The notion of *Conjugate tangents* on smooth surfaces must be defined:

- Strictly related to the **Dupin Indicatrix**
- In negatively curved areas, the Dupin Indicatrix is an hyperbola whose asymptotic directions (A_1, A_2)
- Any parallelogram tangentially circumscribed to the indicatrix defines two conjugate tangents \mathbf{T} and \mathbf{U} .
- The asymptotic directions of the Dupin indicatrix are the diagonals of any such parallelogram.

Initial algorithm is as follows:

For all geodesics s_i in a given pattern:

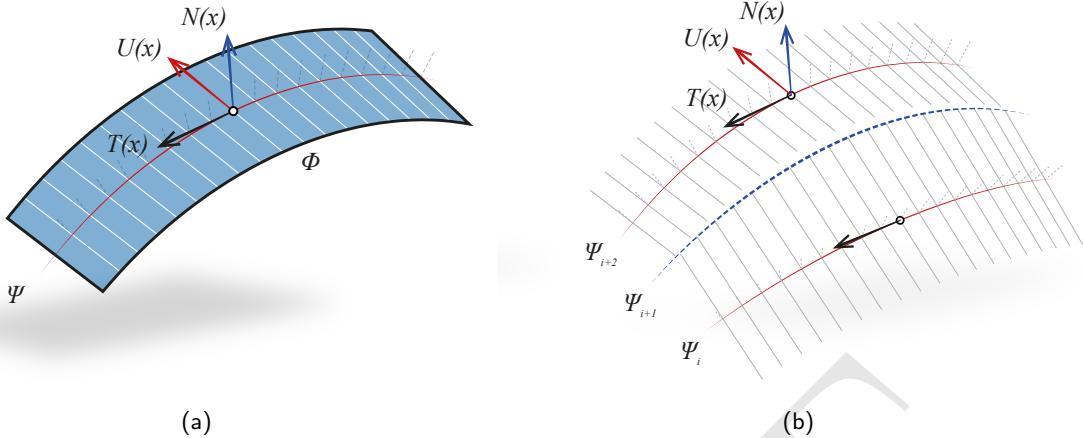


Fig. 24: Computing panels using the tangent developable method: (a) generation of one panel and (b) modification of the procedure proposed in Pottmann et al. (2010).

1. Compute the *tangent developable surfaces* $\rightarrow \Psi_i$
2. Trim Ψ_i along the intersection curves with their respective neighbours.
3. Unfold the trimmed Ψ_i , obtaining the panels in flat state.

Unfortunately, this method needs to be refined in order to work in practice because:

1. The rulings of tangent developables may behave in weird ways
2. The intersection of the neighboring Ψ_i 's is often *ill-defined*.

Therefore, the procedure was modified in the following way:

Algorithm 5 Panels: Modified Tangent-developable method

Require: The reference surface Ψ , represented as a mesh (V,E,F) and a geodesic pattern of curves $g_0\dots g_N$

- 1: Compute the *tangent developable surfaces* Ψ_i for all surfaces $s_i \rightarrow i = \text{even numbers}$
- 2: Delete all rulings where the angle enclose with the tangent α is smaller than a certain threshold (i.e. 20°).
- 3: **for all Rulings do**
- 4: Determine points $A_i(x)$ and $B_i(x)$ which are the closest to geodesics s_{i-1} and s_{i+1} . This serves for trimming the surface Ψ_i .
- 5: **end for**
- 6: Trim Ψ_i with the curve generated from all points $A_i(x)$ and $B_i(x)$ respectively.

Ensure: Trimmed developable surface Ψ_i .

This adjustments to the algorithm allow for the modeling of panels that meet the requirements of developability and approximately constant width, although it must be noted computation times increase, as double the amount of geodesic curves need to be generated, and subsequently, the desired width function needs to be half the desired width of the panels. Fig. 24 shows the result of computing such panels and the subsequent gaps between the generated panels; these gaps need to be kept within a certain width in order to produce a successful watertight panelization of the original surface. Furthermore, material restrictions such as bending or torsion where not taken into account during the construction of the panels.

5 Quality Assessment

5.1 Panel gaps

Since we are generating the panels using the tangent developable method, it is inevitable that some gaps appear between panels, especially when there is a high degree of curvature in the area. We can take this into account when calculating the trim curves, and obtain the maximum gap values of the model. The gap width is strictly related to the thickness of the panel, so if the maximum width of the gaps exceeds a desired a specified distance, computing narrower panels under the same conditions should guarantee the watertightness of the solution.

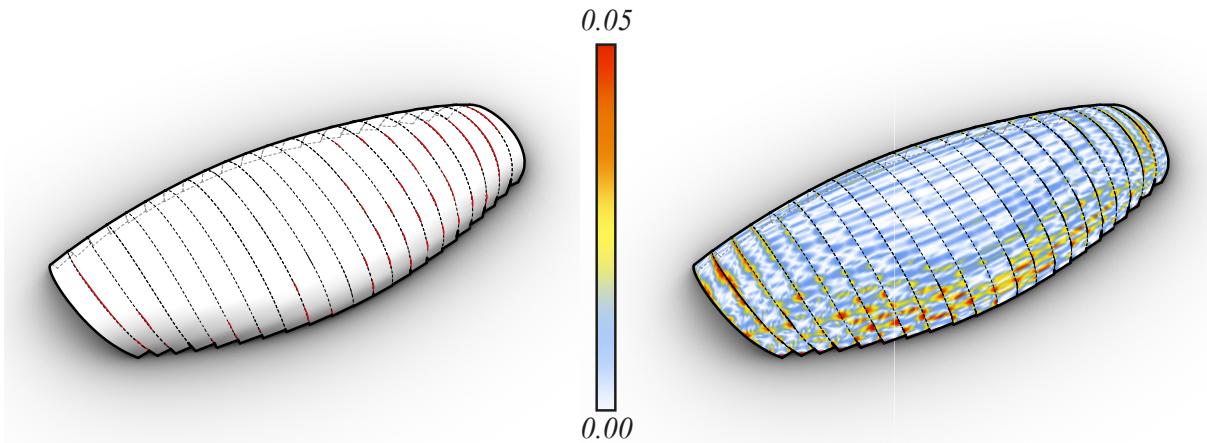


Fig. 25: . On the left, the computed panels using the *tangent developable method* with the corresponding panel gaps highlighted in red. On the right, panels colored by distance to reference mesh. Maximum gap size =

5.2 Stress and strain in panels

The following section investigates the behavior of a rectangular strip of elastic material when it is bent to the shape of a ruled surface Ψ un such way that the central line m of the strip follows the ‘middle geodesic’ g in Ψ . This applies to both methods defining panels.fig. 25

- Lines parallel to m at distance $d/2$ are not only bent but also stretched.
- If you introduce the radius of Gaussian curvature $\rho = 1/\sqrt{|K|} \dots$
- ... the relative increment in length of the strip is:

$$\varepsilon = \frac{1}{2}(d/2\rho)^2 + \dots \quad (6)$$

Tensile stress

Tensile stress is estimated by expressing stress as $\sigma = E\varepsilon$, which yields

$$d/2\rho \leq C, \quad \text{with} \quad C = \sqrt{\sigma_{max}/E} \quad (7)$$

where ρ_{max} is the maximum admissible stress and is Young’s modulus. Since this calculation is an approximation, a safety factor must be used when choosing ρ_{max} .

C is a material constant, from which we can obtain an upper width bound for the panels by $d_{max} = 2\rho_{min}C$, which will be the maximum admissible width for that design configuration.

Bending and shear stress

Bending (σ) and shear stresses (τ) for panels with a thin rectangular cross-section depend on the panel thickness h , but not on the panel width d *if* $h/d \ll 1$, and maximum values occur on the outer surface of the panel (Wallner et al. 2010).

These values depend on the curvature κ of the central geodesic and the rate of torsion θ of the panel. We have:

$$\sigma = E\kappa h/2 \quad \text{and} \quad \tau = hG\theta \quad (8)$$

It is important to note that τ , measured by arc per meter, does not exceed $\sqrt{|K|} = 1/\rho$, where the maximum value occurs in case the central geodesic's tangent happens to be an asymptotic direction of the panel surface (Carmo 2016).

It is standard procedure to combine all stresses (tension, shear, bending) and use this information for checking panel admissibility.

Also, as one might conclude straight away, panels generated using the tangent developable method will experience less shear than the ones created by the ‘binormal’ method.

Tab. 1: Computing the constant for each material by measuring some sample values on the surface.

Material	Young modulus E[N/mm ²]	Maximum stress ∇_{max} [N/mm ²]	constant $C = \sqrt{2\nabla_{max}/E}$
Wood	13000	80	0.11
Steel	200000	250	0.05
Titanium			

From here we can also check the admissibility of the computed models using the formulas described above.

Tab. 2: Panelization solution admissibility check.

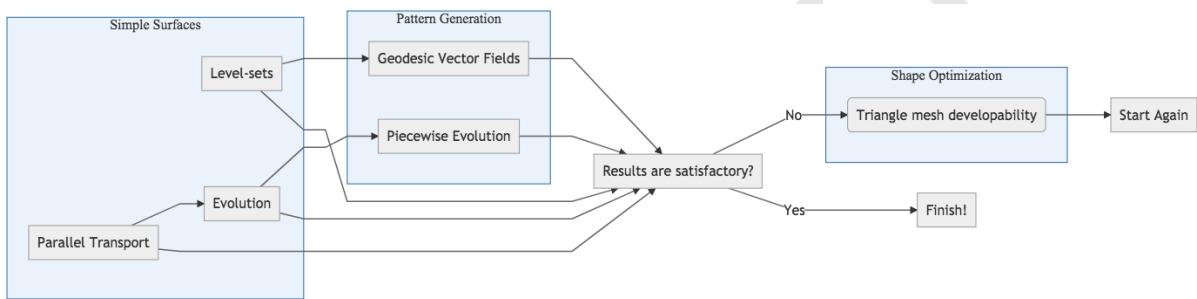
Figure no.	Material	Panel Width (m)	$ K _{max}$ [m ⁻²]	ρ_{min}	Admissible width (m)	Admissible?

Since the examples in this publication were created for illustration purposes, many of the models will not be admissible for fabrication. However, once a max panel width is calculated, if the design is not admissible, one can calculate *narrower* panels using the maximum calculated width in order to have an admissible model.

6 Conclusions

TEMPORARY

1. Parallel transport method and evolution method is only useful for simple surfaces of constant curvature, where the behavior of the generated geodesics can be easily predicted.
2. Piecewise evolution method is useful for any kind of surface, but there is a lack of control over the breakpoint positions, which might be undesirable depending on the design intention of the architect.
3. Level-set method, coupled with the geodesic vector fields, is a perfect solution to enable user interaction during the pattern design process, although in our implementation, we did not achieve sufficiently low computation times for this to be a possibility. It will also be necessary to check the deviation of the resulting curves' geodesic property, since depending on the reference surface and the user specified weights, the generated pattern can deviate from true geodesics.



References

- Arsan, Güler Gürpinar, and Abdulkadir Özdeger. 2015. "Bianchi Surfaces Whose Asymptotic Lines Are Geodesic Parallels." *Adv. Geom.* 15 (1). <https://doi.org/10.1515/advgeom-2014-0020>.
- Azencot, Omri, Maks Ovsjanikov, Frédéric Chazal, and Mirela Ben-Chen. 2015. "Discrete Derivatives of Vector Fields on Surfaces—An Operator Approach." *ACM Transactions on Graphics (TOG)* 34 (3). ACM: 29.
- Carmo, Manfredo P do. 2016. *Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition*. Courier Dover Publications.
- Chen, Jindong, and Yijie Han. 1996. "Shortest Paths on a Polyhedron, Part I: Computing Shortest Paths." *Int. J. Comput. Geom. Appl.* 06 (02). <https://doi.org/10.1142/s0218195996000095>.
- Cheng, Peng, Chunyan Miao, Yong-Jin Liu, Changhe Tu, and Ying He. 2016. "Solving the Initial Value Problem of Discrete Geodesics." *Computer-Aided Design* 70. Elsevier: 144–52. <http://dx.doi.org/10.1016/j.cad.2015.07.012>.
- Deng, Bailin. 2011. "Special Curve Patterns for Freeform Architecture." Edited by Helmut Pottmann and Niloy J. Mitra. <http://resolver.obvsg.at/urn:nbn:at:at-ubtuw:1-40363>.
- Deng, Bailin, Helmut Pottmann, and Wallner Johannes. n.d. "Functional Webs for Freeform Architecture." *Computer Graphics Forum* 30 (5): 1369–78. doi:[10.1111/j.1467-8659.2011.02011.x](https://doi.org/10.1111/j.1467-8659.2011.02011.x).
- Eigensatz, Michael, Martin Kilian, Alexander Schiftner, Niloy J Mitra, Helmut Pottmann, and Mark Pauly. 2010. "Paneling Architectural Freeform Surfaces." *ACM Transactions on Graphics (TOG)*. doi:[10.1145/1833349.1778782](https://doi.org/10.1145/1833349.1778782).

- Jakob, Wenzel, Marco Tarini, Daniele Panozzo, and Olga Sorkine-Hornung. 2015. “Instant Field-Aligned Meshes.” *ACM Trans. Graph.* 34 (6). Citeseer: 189–1.
- Jia, Yan-Bin. 2017. “Curves on a Surface.”
- Julius, Dan, Vladislav Kraevoy, and Alla Sheffer. 2005. “D-Charts: Quasi-Developable Mesh Segmentation.” In *Computer Graphics Forum*, 24:581–90. 3. Wiley Online Library.
- Kahlert, Joe, Matt Olson, and Hao Zhang. 2010. “Width-Bounded Geodesic Strips for Surface Tiling.” *Vis. Comput.* 27 (1). <https://doi.org/10.1007/s00371-010-0513-3>.
- Keenan Crane, Mathieu Desbrun, Fernando de Goes. 2013. “Digital Geometry Processing with Discrete Exterior Calculus.” In *ACM Siggraph 2013 Courses*. SIGGRAPH ’13. New York, NY, USA: ACM.
- Kensek, Karen, Judith Leuppi, and Douglas Noble. 2000. “Plank Lines of Ribbed Timber Shell Structures.” In *Proceedings of the 22nd Annual Conference of the Association for Computer-Aided Design in Architecture, Acadia 2000: Eternity, Infinity and Virtuality in Architecture*, 261–66.
- Kimmel, R., and J A Sethian. 1998. “Computing Geodesic Paths on Manifolds.” *Proc. Natl. Acad. Sci. U. S. A.* 95 (15). <https://doi.org/10.1073/pnas.95.15.8431>.
- Liu, Yang, Helmut Pottmann, Johannes Wallner, Yong-Liang Yang, and Wenping Wang. 2006. “Geometric Modeling with Conical Meshes and Developable Surfaces.” In *ACM Transactions on Graphics (TOG)*, 25:681–89. 3. ACM.
- Meredith, Neil, and James Kotronis. 2013. “Self-Detailing and Self-Documenting Systems for Wood Fabrication: The Burj Khalifa.” In *Advances in Architectural Geometry 2012*, edited by Lars Hesselgren, Shrikant Sharma, Johannes Wallner, Niccolo Baldassini, Philippe Bompas, and Jacques Raynaud, 185–98. Vienna: Springer Vienna. https://doi.org/10.1007/978-3-7091-1251-9_14.
- Mitani, Jun, and Hiromasa Suzuki. 2004. “Making Papercraft Toys from Meshes Using Strip-Based Approximate Unfolding.” *ACM Transactions on Graphics (TOG)* 23 (3). ACM: 259–63.
- Polthier, Konrad, and Markus Schmies. 1998. “Straightest Geodesics on Polyhedral Surfaces.” In *Mathematical Visualization*, 135–50. <https://doi.org/10.1145/1185657.1185664>.
- Pottmann, Helmut. 2010. “Architectural Geometry as Design Knowledge.” *Architectural Design* 80 (4). Wiley Online Library: 72–77. <https://doi.org/10.1002/ad.1109>.
- Pottmann, Helmut, Michael Eigensatz, Amir Vaxman, and Johannes Wallner. 2015. “Architectural Geometry.” *Computers and Graphics* 47: 145–64. doi:[10.1016/j.cag.2014.11.002](https://doi.org/10.1016/j.cag.2014.11.002).
- Pottmann, Helmut, Qixing Huang, Bailin Deng, Alexander Schiftner, Martin Kilian, Leonidas Guibas, and Johannes Wallner. 2010. “Geodesic Patterns.” In *ACM SIGGRAPH 2010 Papers on - SIGGRAPH ’10*. <https://doi.org/10.1145/1833349.1778780>.
- Pottmann, Helmut, Alexander Schiftner, and Johannes Wallner. 2008. “Geometry of Architectural Freeform Structures.” In *Symposium on Solid and Physical Modeling*, 9. <https://doi.org/10.1145/1364901.1364903>.
- Rose, Kenneth, Alla Sheffer, Jamie Wither, Marie-Paule Cani, and Boris Thibert. 2007. “Developable Surfaces from Arbitrary Sketched Boundaries.” In *SGP ’07 - 5th Eurographics Symposium on Geometry Processing*, edited by Alexander G. Belyaev and Michael Garland. SGP ’07 Proceedings of the Fifth Eurographics Symposium on Geometry Processing. Eurographics Association. <http://dl.acm.org/citation.cfm?id=1281991.1282014>.

- Rusinkiewicz, Szymon. 2004. “Estimating Curvatures and Their Derivatives on Triangle Meshes.” In *Null*, 486–93. IEEE.
- Shatz, Idan, Ayellet Tal, and George Leifman. 2006. “Paper Craft Models from Meshes.” *The Visual Computer* 22 (9-11). Springer: 825–34.
- Shelden, Dennis Robert. 2002. “Digital Surface Representation and the Constructibility of Gehry’s Architecture.” PhD thesis, Massachusetts Institute of Technology. <http://hdl.handle.net/1721.1/16899>.
- Stein, Oded, Eitan Grinspun, and Keenan Crane. 2018. “Developability of Triangle Meshes.” *ACM Trans. Graph.* 37 (4). New York, NY, USA: ACM.
- Surazhsky, Vitaly, Tatiana Surazhsky, Danil Kirsanov, Steven J Gortler, and Hugues Hoppe. 2005. “Fast Exact and Approximate Geodesics on Meshes.” *ACM Trans. Graph.* 24 (3). <https://doi.org/10.1145/1186822.1073228>.
- Tong, Yiying, Santiago Lombeyda, Anil N Hirani, and Mathieu Desbrun. 2003. “Discrete Multiscale Vector Field Decomposition.” In *ACM Transactions on Graphics (ToG)*, 22:445–52. 3. ACM.
- Wallner, Johannes, Alexander Schiftner, Martin Kilian, Simon Flöry, Mathias Höbinger, Bailin Deng, Qixing Huang, and Helmut Pottmann. 2010. “Tiling Freeform Shapes with Straight Panels: Algorithmic Methods.” In *Advances in Architectural Geometry 2010*, edited by Cristiano Ceccato, Lars Hesselgren, Mark Pauly, Helmut Pottmann, and Johannes Wallner. Vienna: Springer Vienna. https://doi.org/10.1007/978-3-7091-0309-8_5.
- Weinand, Yves, and Claudio Pirazzi. 2006. “Geodesic Lines on Free-Form Surfaces - Optimized Grids for Timber Rib Shells.” *World Conference in Timber Engineering WCTE*. <https://infoscience.epfl.ch/record/118623>.

A Appendix

A.1 Implementation Details

Methods used in this paper were implemented for the Rhinoceros+Grasshopper software as a series of Grasshopper components.

All related code and compiled versions, including example files, can be found at:

<https://github.com/AlanRynne/Geodesic-Patterns>

A.2 Indexes

List of Tables

1	Computing the constant for each material by measuring some sample values on the surface.	23
2	Panelization solution admissibility check.	23

List of Algorithms

1	Parallel Transport Method	10
2	Calculate the best-fit geodesic	13

3	Evolution Method	14
4	Calculate the best-fit piecewise geodesic	15
5	Panels: Modified Tangent-developable method	21

List of Figures

1	Straight planks bent on a hyperbolic paraboloid (a) and a cylindrical surface (b).	2
2	Famous Ghery projects (right to left): Guggenheim Museum (Bilbao, 1997), Walt Disney Concert Hall (Los Angeles, 2003) and the Guggenheim Abu Dhabi winning proposal (Abu Dhabi, –)	2
3	Interior view of Toyo Ito's Mina No Mori wooden lattice roof (a) and exterior view of the lattice under construction (b).	3
4	Burj Khalifa interior ceiling constructed from wooden planks (a) and final paneling solution (b).	3
5	NOX Architects paper models & final rendered design solution.	4
6	Shigeru Ban's Yeoju Golf Resort. Right: Pre-assembled complex lattice. Left: Aerial view of lattice under construction.	4
7	Mesh developability techniques. Left to right: Julius, Kraevoy, and Sheffer (2005), Mitani and Suzuki (2004), Shatz, Tal, and Leifman (2006) Y. Liu et al. (2006) and Stein, Grinspun, and Crane (2018) [^KraneDevelop]	5
8	Left: Any given vertex on a mesh (left) will become either flat (centre) or a hinge (right). Right: Developalizing a sphere. Results highly depend on the initial mesh topology[^{^KraneDevelop}]	5
9	Developable surface generated using curve a geodesic curve. (a) using the X component of the curve's perp frame; (b) using the Y component & (c) using the Z component (tangent of the curve).	7
10	Noisy to smooth sheet (Stein, Grinspun, and Crane 2018)[^{^KraneDevelop}]	7
11	Example of parallel transport method. Generatrix geodesic g (red) and geodesics g^\perp generated from a parallel transported vector (blue) computed given a point and a vector \mathbf{v} tangent to the surface, in both positive and negative directions.	8
12	Parallel transport method: We can parallel transport a vector along a curve placed on a surface by iteratively projecting the previous vector \mathbf{v}_{i-1} over the tangent plane of v_i and normalizing it's length.	9
13	Parallel transport method. Main geodesic g , tangent planes at each point p_i , transported vectors \mathbf{v}_i , generated geodesics h_i	9
14	Surface covered by a 1-geodesic pattern using the evolution method without introducing breakpoints. (a) shows an overview of the result; while (b) highlights the intersection point of several geodesic curves. This problem will be addressed by introducing the concept of 'piece-wise' geodesic curves; which are curves that are not geodesics, but are composed of segments of several connected geodesic curves.	11
15	Calculating the 'next' geodesic	12
16	Calculating the best piece-wise geodesic	15
17	(a) Perspective view; (b) Top View. In red, the initial geodesic curve; in blue, the generated piecewise geodesic pattern; the red dots are breakpoints in the piecewise geodesic curves.	16
18	Calculating a level set on a single face	17
19	Starting conditions for the level set method on a double curve surface	18
20	Final result of the level set method over a complex double curve surface.	18
21	Beam structure generated by the geodesic web method.	19
22	Geodesic vector fields	19

23	Final result of geodesic vector fields method	20
24	Computing panels using the tangent developable method: (a) generation of one panel and (b) modification of the procedure proposed in Pottmann et al. (2010).	21
25	. On the left, the computed panels using the <i>tangent developable method</i> with the corresponding panel gaps highlighted in red. On the right, panels colored by distance to reference mesh. Maximum gap size =	22

DRAFT