SIGGRAPH2008

Curved Folding

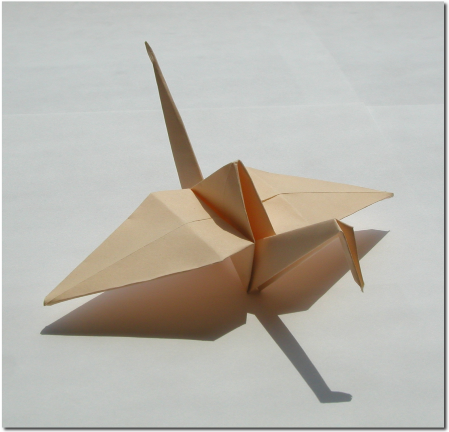
M. Kilian, S. Flöry, Z. Chen, N. J. Mitra,
A. Sheffer, H. Pottmann

evolute. research and consulting

UBC

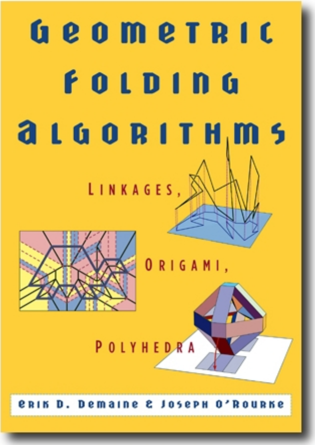
TU VIENNA
Technische Universität Wien
Vienna University of Technology

Traditional Origami sculptures are produced according to simple rules/principles:



- only straight folds are allowed
- no tearing, cutting, gluing

Resulting surfaces are developable – they can be unfolded. Mathematically speaking they are isometric to a planar domain.

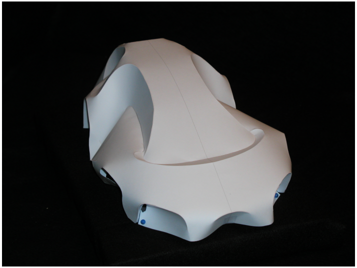


More information on the algorithmic treatment of straight folds in the book by Demaine and O'Rourke.

Motivation

Curved Folding – Curved Crease Origami – Curvigami

Adding curved creases to the set of allowable folds complex and elegant shapes can be designed with a small number of folds.



Models created by David Huffman and Gregory Epps. All models are folded from a single sheet of paper.

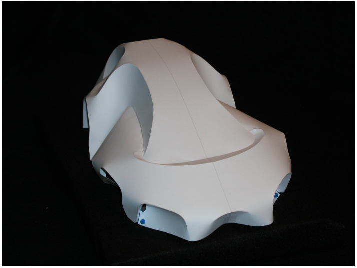
[0] D. Huffman 76: *Curvature and Crease: A Primer on Paper*

Motivation

Curved Folding – Curved Crease Origami – Curvigami

Adding curved creases to the set of allowable folds complex and elegant shapes can be designed with a small number of folds.





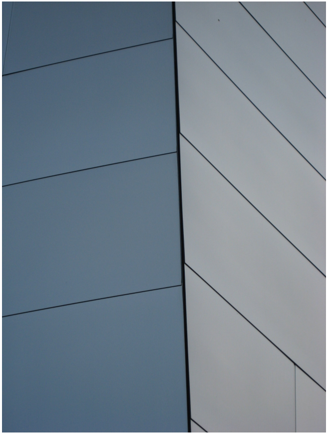
Models created by David Huffman and Gregory Epps. All models are folded from a single sheet of paper.

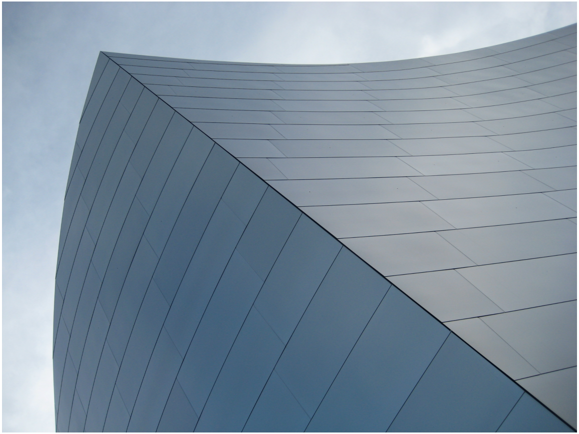
[0] D. Huffman 76: *Curvature and Crease: A Primer on Paper*

There is great interest in developable surfaces in architecture. The Disney Concert Hall designed by Frank Gehry is a popular example.

Motivation

Piecewise Developable Surfaces in Architecture



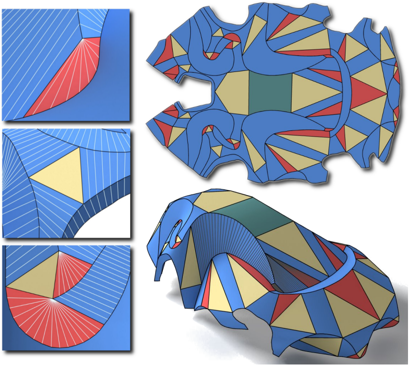


Assembling developable surfaces at a common crease leads to the tiling problem if the crease is not developable.

Aid the user in the

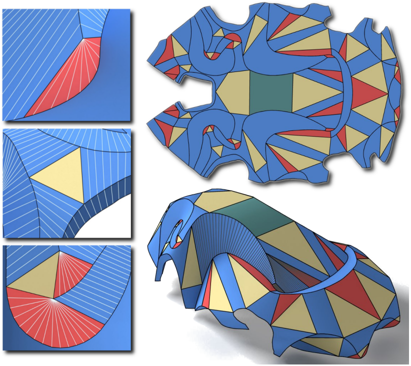
- design,
- optimization,
- and approximation

with surfaces that can be produced by curved folding.



Torsal ruled surfaces can be decomposed into patches lying on

- planar regions
- cones
- cylinders
- tangent surfaces of space curves



Torsal ruled surfaces can be decomposed into patches lying on

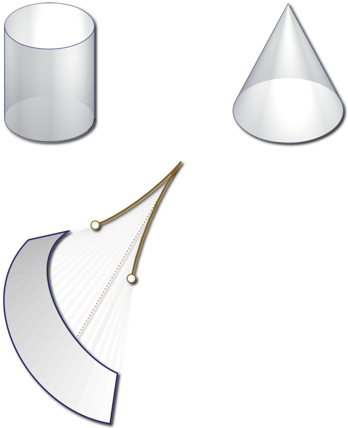
- planar regions
- cones
- cylinders
- tangent surfaces of space curves

Pottmann and Wallner: *Computational Line Geometry*

Smooth vs. Discrete

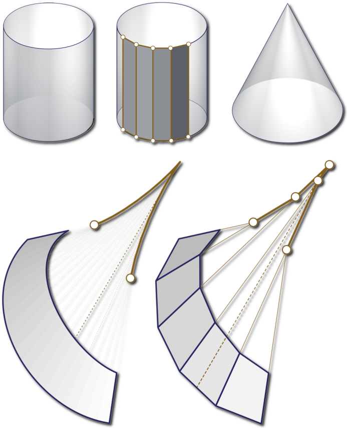
Each patch just described has a natural representation as a discrete surface.

- PQ strips
- triangle fans
- planar polygons



Smooth vs. Discrete

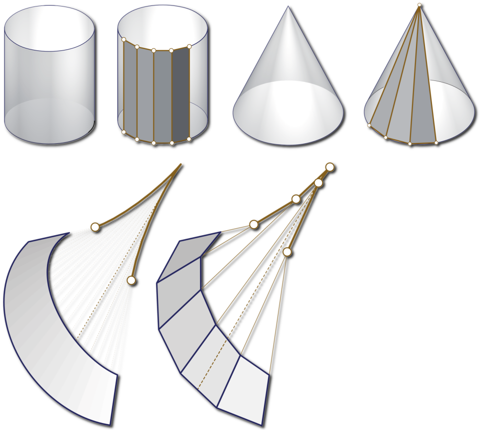
Each patch just described has a natural representation as a discrete surface.



- PQ strips
- triangle fans
- planar polygons

Smooth vs. Discrete

Each patch just described has a natural representation as a discrete surface.

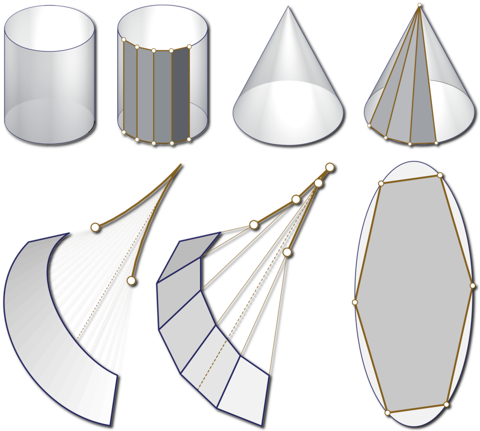


- PQ strips
- triangle fans
- planar polygons

Smooth vs. Discrete

Each patch just described has a natural representation as a discrete surface.

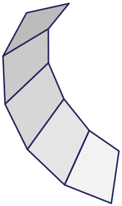
- PQ strips
- triangle fans
- planar polygons



Discrete Developable Surfaces

Surface Representation II

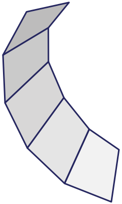
We use quad dominant meshes with planar faces (PQ meshes) to model discrete developable surfaces.



- developability guaranteed
- rulings and curve of regression are explicit
- maintaining planarity of quads during subdivision generates a developable surface

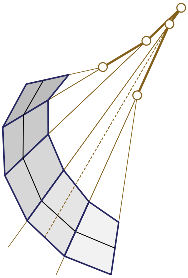
Discrete Developable Surfaces

We use quad dominant meshes with planar faces (PQ meshes) to model discrete developable surfaces.



- developability guaranteed
- rulings and curve of regression are explicit
- maintaining planarity of quads during subdivision generates a developable surface

We use quad dominant meshes with planar faces (PQ meshes) to model discrete developable surfaces.

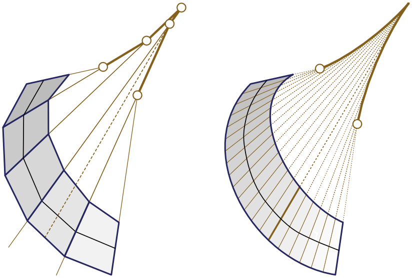


- developability guaranteed
- rulings and curve of regression are explicit
- maintaining planarity of quads during subdivision generates a developable surface

Discrete Developable Surfaces

Surface Representation II

We use quad dominant meshes with planar faces (PQ meshes) to model discrete developable surfaces.



- developability guaranteed
- rulings and curve of regression are explicit
- maintaining planarity of quads during subdivision generates a developable surface

Discrete Developable Surfaces

Surface Representation III

A discrete developable surface is a collection of

- PQ-strips,
- triangle fans,
- planar polygons.

Each edge of such a mesh is either a

- ruling direction,
- part of a crease,
- part of a boundary curve.

Discrete Developable Surfaces

Surface Representation III

A discrete developable surface is a collection of

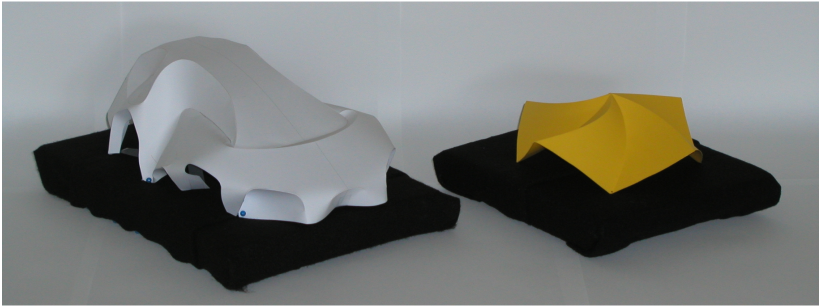
- PQ-strips,
- triangle fans,
- planar polygons.

Each edge of such a mesh is either a

- ruling direction,
- part of a crease,
- part of a boundary curve.

Problem

Approximate an almost developable surface (e.g. obtained by 3D scanning of folded models made of paper-like materials) by a discrete developable surface



How to generate patches from measurement data

1. Estimate rulings, creases, and planar regions
2. (Approximately) unfold to the plane
3. Map rulings and creases to plane using the development
4. Generate a quad mesh aligned to rulings and creases
5. Map quads back to space using inverse development
6. Register corresponding faces

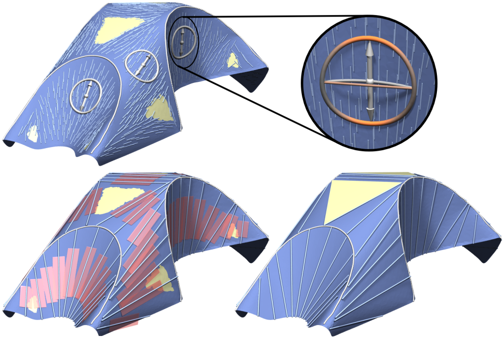
How to generate patches from measurement data

1. Estimate rulings, creases, and planar regions
2. (Approximately) unfold to the plane
3. Map rulings and creases to plane using the development
4. Generate a quad mesh aligned to rulings and creases
5. Map quads back to space using inverse development
6. Register corresponding faces

Curved Folding

Ruling estimation I

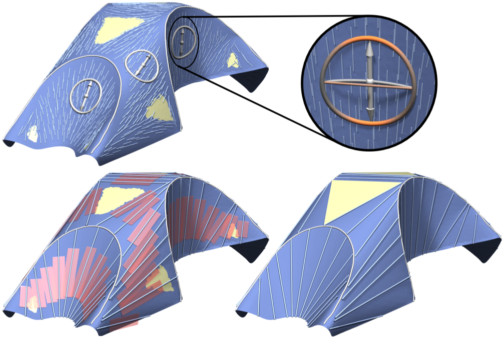
Rulings are characterized as lines with constant surfaces normals. For any two points on a ruling the geodesic distance and the spatial distance are equal.



Compute creases with [1]. Estimate ruling directions in vertices away from creases. Integrate these directions and find a sparse set of good rulings.

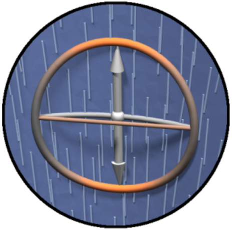
[1] Ohtake et al: *Ridge-valley lines on meshes via implicit surface fitting*

Rulings are characterized as lines with constant surfaces normals. For any two points on a ruling the geodesic distance and the spatial distance are equal.



Compute creases with [1]. Estimate ruling directions in vertices away from creases. Integrate these directions and find a sparse set of good rulings.

[1] Ohtake et al: *Ridge-valley lines on meshes via implicit surface fitting*



Ruling directions

Compute geodesic circle of radius r_p around each vertex p. A maximum of the score

$$\sigma(p) := n_p \cdot n_q + \nu \|p - q\| / r_p$$

characterizes a ruling direction if the geodesic disc is developable (compare area to determine this).

Ruling extension

Extend previously computed ruling directions as long as the deviation of surface normals is below a predefined threshold

Pruning

Use the mean deviation of normals along extended rulings as quality measure. Keep best ruling. Discard all rulings inside a certain neighborhood. Repeat exhaustively.

Ruling extension

Extend previously computed ruling directions as long as the deviation of surface normals is below a predefined threshold

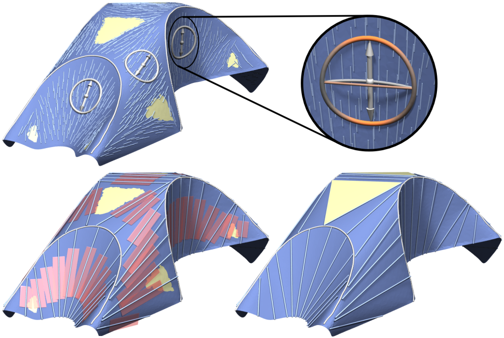
Pruning

Use the mean deviation of normals along extended rulings as quality measure. Keep best ruling. Discard all rulings inside a certain neighborhood. Repeat exhaustively.

Curved Folding

Ruling estimation I

Rulings are characterized as lines with constant surfaces normals. For any two points on a ruling the geodesic distance and the spatial distance are equal.



Compute creases with [1]. Estimate ruling directions in vertices away from creases. Integrate these directions and find a sparse set of good rulings.

[1] Ohtake et al: *Ridge-valley lines on meshes via implicit surface fitting*

How to generate patches from measurement data

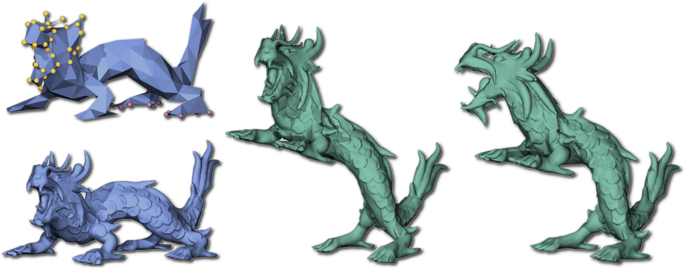
1. Estimate rulings, creases, and planar regions
2. (Approximately) unfold to the plane
3. Map rulings and creases to plane using the development
4. Generate a quad mesh aligned to rulings and creases
5. Map quads back to space using inverse development
6. Register corresponding faces

Use constrained shape deformation tool of [2] to unfold the model.

The z-coordinate of vertices are constrained to be zero.

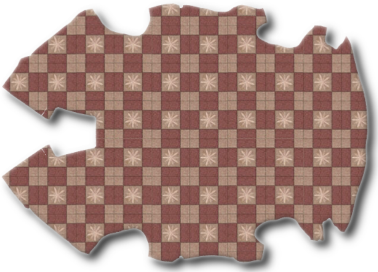
[2] Kilian et al. 07: *Geometric Modeling in Shape Space*

[3] Liu et al. 08: *A Local/Global Approach to Mesh Parametrization*



Use constrained shape deformation tool of [2] to unfold the model.
The z-coordinate of vertices are constrained to be zero.

[2] Kilian et al. 07: *Geometric Modeling in Shape Space*
[3] Liu et al. 08: *A Local/Global Approach to Mesh Parametrization*

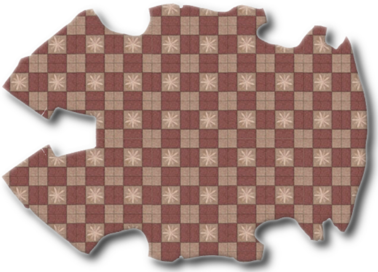


Use constrained shape deformation tool of [2] to unfold the model.
The z-coordinate of vertices are constrained to be zero.

[2] Kilian et al. 07: *Geometric Modeling in Shape Space*
[3] Liu et al. 08: *A Local/Global Approach to Mesh Parametrization*

Curved Folding

Unfolding



Use constrained shape deformation tool of [2] to unfold the model.
The z-coordinate of vertices are constrained to be zero.

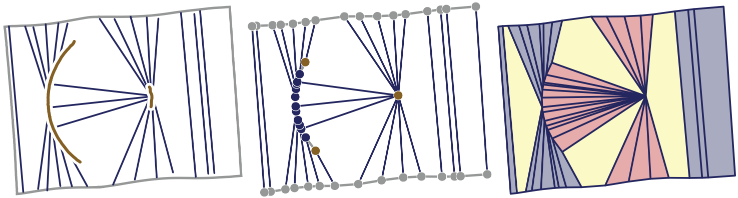
[2] Kilian et al. 07: *Geometric Modeling in Shape Space*
[3] Liu et al. 08: *A Local/Global Approach to Mesh Parametrization*

How to generate patches from measurement data

1. Estimate rulings, creases, and planar regions
2. (Approximately) unfold to the plane
3. Map rulings and creases to plane using the development
4. Generate a quad mesh aligned to rulings and creases
5. Map quads back to space using inverse development
6. Register corresponding faces

Curved Folding

Quad Mesh Initialization



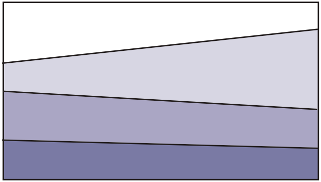
- Extend rulings to boundary/crease.
- Coalesce close ruling endpoints.
- Remove T-junctions at creases by inserting a ruling on the other side.

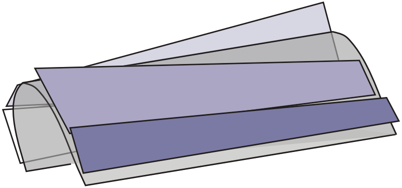
How to generate patches from measurement data

1. Estimate rulings, creases, and planar regions
2. (Approximately) unfold to the plane
3. Map rulings and creases to plane using the development
4. Generate a quad mesh aligned to rulings and creases
5. Map quads back to space using inverse development
6. Register corresponding faces

Curved Folding

Result of Initialization

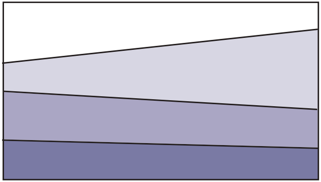


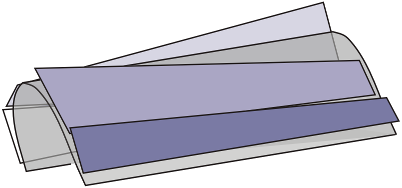


- planar mesh with tagged edges (ruling, crease, boundary),
- polygon soup in space,
- correspondence of faces.

Curved Folding

Result of Initialization

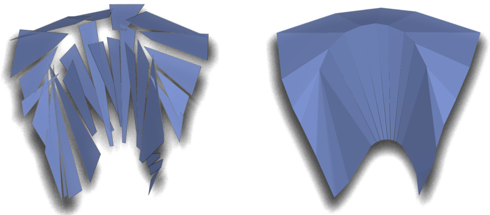




- planar mesh with tagged edges (ruling, crease, boundary),
- polygon soup in space,
- correspondence of faces.

Curved Folding

Result of Initialization



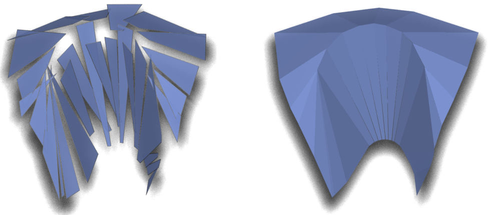
Optimization

Optimize both the shape of planar faces and the spatial position and orientation of corresponding congruent faces to make the polygon soup a mesh. We use a PriMo-like approach to solve this problem.

[4] Botsch et al. *PriMo: Coupled Prisms for Intuitive Surface Modeling*

Curved Folding

Result of Initialization



Optimization

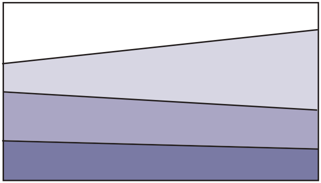
Optimize both the shape of planar faces and the spatial position and orientation of corresponding congruent faces to make the polygon soup a mesh. We use a PriMo-like approach to solve this problem.

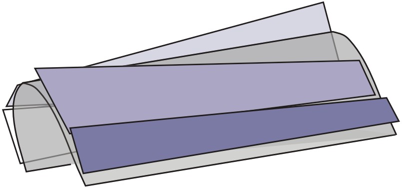
[4] Botsch et al. *PriMo: Coupled Prisms for Intuitive Surface Modeling*

Main goal during optimization

Reduce the distance of corresponding edges and vertices of the quad soup to make it a mesh. Optimization is subject to suitable fairness conditions.

$$F = F_{vert} + \lambda F_{fit} + \mu F_{fair}$$





The objective function in more detail

The objective function consists of vertex agreement, fairness, and fitting terms.

$$F_{vert} := \sum_{\mathbf{p} \in P} (\mathbf{m}_p^i - \mathbf{m}_p^j)^2$$

$$F_{fit} := \sum_{\mathbf{m} \in M} ((\mathbf{m} - \mathbf{m}_c) \cdot \mathbf{n}_c)^2$$

$$F_{fair} := \sum_{\mathbf{e}_{ij} \in E} w_{ij} (\mathbf{n}^i - \mathbf{n}^j)^2$$

Vertices **m** belong to the polygon soup. Those vertices are related to vertices **p** of the planar mesh by a rigid body motion.

The objective function in more detail

The objective function consists of vertex agreement, fairness, and fitting terms.

$$F_{vert} := \sum_{\mathbf{p} \in P} (\mathbf{m}_p^i - \mathbf{m}_p^j)^2$$

$$F_{fit} := \sum_{\mathbf{m} \in M} ((\mathbf{m} - \mathbf{m}_c) \cdot \mathbf{n}_c)^2$$

$$F_{fair} := \sum_{\mathbf{e}_{ij} \in E} w_{ij} (\mathbf{n}^i - \mathbf{n}^j)^2$$

Vertices $\mathbf{m}$ belong to the polygon soup. Those vertices are related to vertices $\mathbf{p}$ of the planar mesh by a rigid body motion.

Bending energy

The bending energy of a surface patch S is defined as

$$E = \int_S \kappa_1^2 + \kappa_2^2 dA$$

with principal curvatures κ_1, κ_2.

- Rulings constitute principal curvature lines corresponding to principal curvature 0.
- Define the other family of curvature lines motivated by the theory of circular meshes as orthogonal trajectories of ruling bisectors (see Bobenko and Suris, *Discrete Differential Geometry. Consistency as Integrability*).

Bending energy

The bending energy of a surface patch S is defined as

$$E = \int_S \kappa_2^2 dA$$

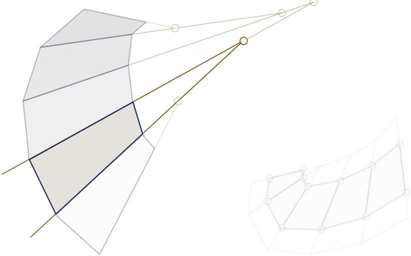
with principal curvatures κ_1, κ_2.

- Rulings constitute principal curvature lines corresponding to principal curvature 0.
- Define the other family of curvature lines motivated by the theory of circular meshes as orthogonal trajectories of ruling bisectors (see Bobenko and Suris, *Discrete Differential Geometry. Consistency as Integrability*).

Discrete Developable Surfaces

Some Discrete Differential Geometry

We compute the bending energy of a PQ strip bounded by two discrete principal curvature lines C and $\bar{C}$.

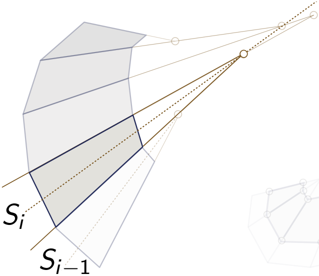


- $L_i = \|\mathbf{m}_i - \mathbf{m}_{i-1}\|$
- $N_i = \|\mathbf{n}_i - \mathbf{n}_{i-1}\|$
- $\kappa_2 = N_i / L_i$
- $w_i = h \frac{\log(\bar{L}_i) - \log(L_i)}{\bar{L}_i - L_i}$
- $E_{bend} = \sum w_i \|\mathbf{n}_i - \mathbf{n}_{i-1}\|^2$

Discrete Developable Surfaces

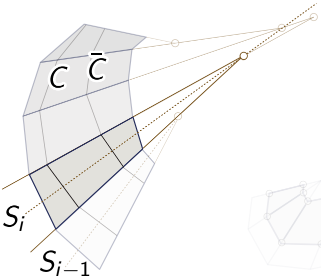
Some Discrete Differential Geometry

We compute the bending energy of a PQ strip bounded by two discrete principal curvature lines C and $\bar{C}$.



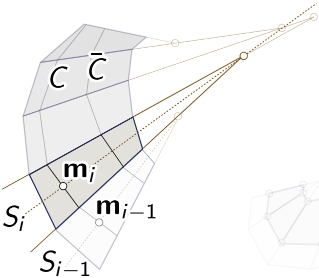
- $L_i = \|\mathbf{m}_i - \mathbf{m}_{i-1}\|$
- $N_i = \|\mathbf{n}_i - \mathbf{n}_{i-1}\|$
- $\kappa_2 = N_i / L_i$
- $w_i = h \frac{\log(\bar{L}_i) - \log(L_i)}{\bar{L}_i - L_i}$
- $E_{bend} = \sum w_i \|\mathbf{n}_i - \mathbf{n}_{i-1}\|^2$

We compute the bending energy of a PQ strip bounded by two discrete principal curvature lines C and $\bar{C}$.



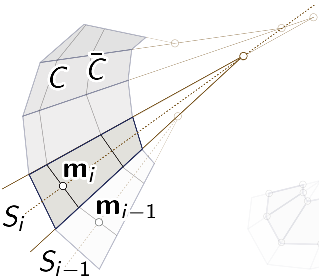
- $L_i = \|\mathbf{m}_i - \mathbf{m}_{i-1}\|$
- $N_i = \|\mathbf{n}_i - \mathbf{n}_{i-1}\|$
- $\kappa_2 = N_i / L_i$
- $w_i = h\frac{\log(\bar{L}_i) - \log(L_i)}{\bar{L}_i - L_i}$
- $E_{bend} = \sum w_i \|\mathbf{n}_i - \mathbf{n}_{i-1}\|^2$

We compute the bending energy of a PQ strip bounded by two discrete principal curvature lines C and $\bar{C}$.



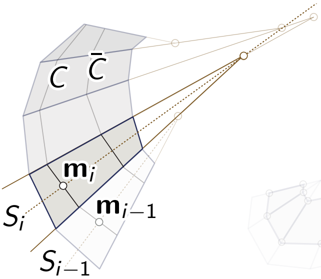
- $L_i = \|\mathbf{m}_i - \mathbf{m}_{i-1}\|$
- $N_i = \|\mathbf{n}_i - \mathbf{n}_{i-1}\|$
- $\kappa_2 = N_i / L_i$
- $w_i = h\frac{\log(\bar{L}_i) - \log(L_i)}{\bar{L}_i - L_i}$
- $E_{bend} = \sum w_i \|\mathbf{n}_i - \mathbf{n}_{i-1}\|^2$

We compute the bending energy of a PQ strip bounded by two discrete principal curvature lines C and $\bar{C}$.



- $L_i = \|\mathbf{m}_i - \mathbf{m}_{i-1}\|$
- $N_i = \|\mathbf{n}_i - \mathbf{n}_{i-1}\|$
- $\kappa_2 = N_i / L_i$
- $w_i = h \frac{\log(\bar{L}_i) - \log(L_i)}{\bar{L}_i - L_i}$
- $E_{bend} = \sum w_i \|\mathbf{n}_i - \mathbf{n}_{i-1}\|^2$

We compute the bending energy of a PQ strip bounded by two discrete principal curvature lines C and $\bar{C}$.

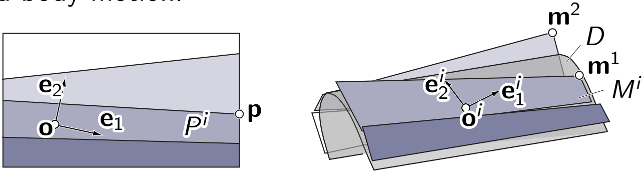


- $L_i = \|\mathbf{m}_i - \mathbf{m}_{i-1}\|$
- $N_i = \|\mathbf{n}_i - \mathbf{n}_{i-1}\|$
- $\kappa_2 = N_i / L_i$
- $w_i = h\frac{\log(\bar{L}_i) - \log(L_i)}{\bar{L}_i - L_i}$
- $E_{bend} = \sum w_i \|\mathbf{n}_i - \mathbf{n}_{i-1}\|^2$

Pick a frame $(\mathbf{o}, \mathbf{e}_1, \mathbf{e}_2)$ in the plane. Then

$$\mathbf{p} = \mathbf{o} + p_x \mathbf{e}_1 + p_y \mathbf{e}_2$$
$$\mathbf{m}_p^i = \mathbf{o}^i + p_x \mathbf{e}_1^i + p_y \mathbf{e}_2^i$$

since corresponding faces are congruent and the frames are related by a rigid body motion.



- Optimizing M changes the rigid body motion
- Optimizing P changes the coordinates p_x and p_y

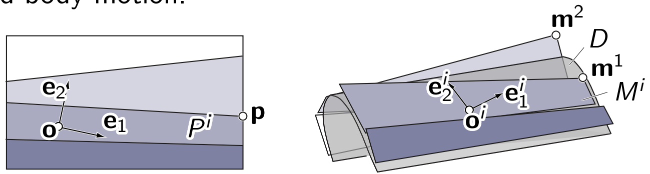
Curved Folding

The Basic Optimization

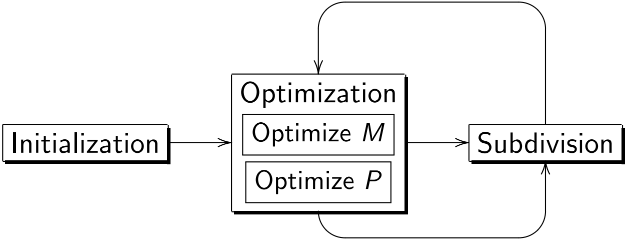
Pick a frame $(\mathbf{o}, \mathbf{e}_1, \mathbf{e}_2)$ in the plane. Then

$$\mathbf{p} = \mathbf{o} + p_x \mathbf{e}_1 + p_y \mathbf{e}_2$$
$$\mathbf{m}_p^i = \mathbf{o}^i + p_x \mathbf{e}_1^i + p_y \mathbf{e}_2^i$$

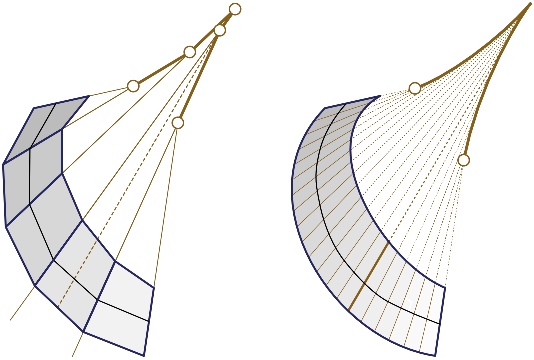
since corresponding faces are congruent and the frames are related by a rigid body motion.



- Optimizing M changes the rigid body motion
- Optimizing P changes the coordinates p_x and p_y



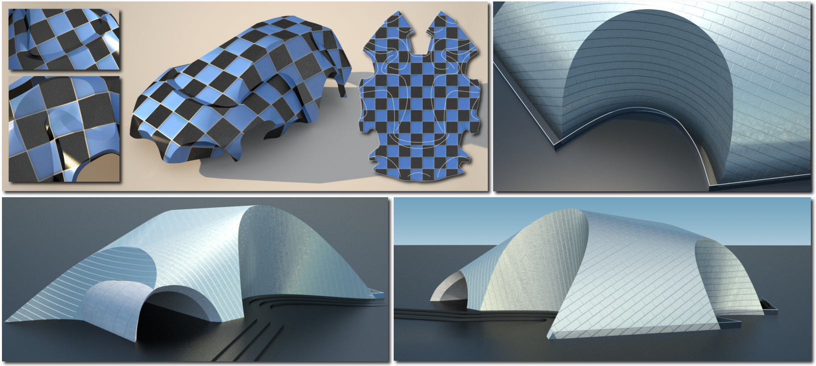
We optimize the spatial position of the faces of M and the geometry of the faces of P in an alternating fashion.



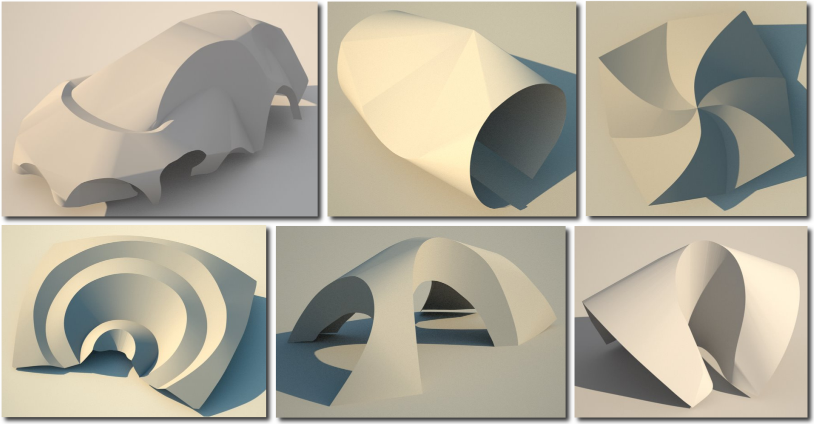
Subdivision occurs only in ruling direction. Only creases and boundary edges are split during subdivision.

Curved Folding

Results



Curved Folding
Digital Paper Models



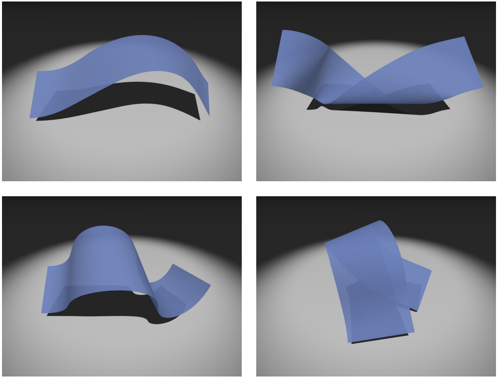
Acknowledgments

We thank
Heinz Schmiedhofer for rendering and scanning and Martin Peternell and Johannes Wallner for their thoughts and comments on the subject.

This work is supported by the Austrian Science Fund (FWF) under grants S92 and P18865 and a Microsoft outstanding young faculty fellowship.

FWF

Der Wissenschaftsfonds.

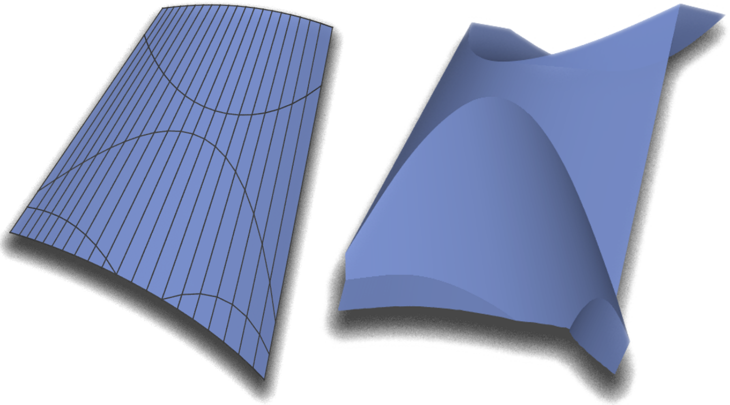


No reference surface but boundary conditions on tangent planes.

[5] Bo and Wang 07: *Geodesic-controlled developable surfaces for modeling paper bending*

Curved Folding

Surface Design



Timings for models seen in the gallery of digital paper models. A 50K triangles reference surface was used in all examples.

Ruling extraction	160 sec
Mesh layout	20 sec
Optimization	140 sec

Three rounds of subdivision were performed. The objective function was reduced to order 10^{-4}.