Institute of Science and Technology Austria (IST Austria)
About This Course

• Introduction and general challenges
• Computational tools and design tools for
  – Deformable Shapes
  – Foldable Shapes
• Latest research
• Advanced manufacturing
• Inspiration

Source: The Economist (Cover)
Other 3D Printing Courses at Siggraph/Siggaph Asia

• Siggraph Asia 2014
  – 3D printing oriented design: geometry and optimization
    http://staff.ustc.edu.cn/~lgliu/Courses/SigAsia_2014_course_3Dprinting/index.html

• Siggraph 2015
  – Modeling and Toolpath Generation for Consumer-Level 3D Printing
    http://webloria.loria.fr/~slefebvr/sig15fdm/

• Siggraph 2016
  – Computational Tools for 3D Printing
    http://computational-fabrication.com/2016/

• Eurographics 2017
  – Topology Optimization for Computational Fabrication
    https://topopt.weblog.tudelft.nl/
Course Schedule

16:00 - 16:15, Welcome & Introduction **Bernd**
16:15 - 16:45, Inverse Design – Deformable Shapes **Bernd**
16:45 - 17:15, Inverse Design – Foldable Shapes **Niloy**
17:15 - 17:25, Advanced Manufacturing and Open Challenges, **Bernd**
17:25 – 17:30 Q&A
Introduction
Computational Fabrication

Engineering / Customized Products

Bioprinting / Medicine

Robotics / Nanofabrication
«Game Changer» 3D Printer

3D Printers

[Stratasys]
Complexity (almost) for Free

Arabic Icosahedron
(Carlo H. Séquin)
Benefits of Additive Manufacturing

- Very flexible
- Rapid fabrication
- Excellent for customization
- Complexity for free
- AM has minimal material waste
Limitations

- Limited part sizes
- Limited fabrication speed
- Limited materials
- Poor surface finish
- Inconsistent part quality
- High cost (machine, material, pre- and postprocessing)
Computational Fabrication

Physical World
Information
Ideas

Virtual World

Output
State of the Art

- Shape modeling
- Simulation
- Triangle Surface Mesh
State of the Art

Shape modeling: Trillions of voxels

Simulation: Finite Element method too slow

Triangle Surface Mesh: Prescriptive, no semantics
Direct Specification

• Decompose into regions
• Assign one material for each region
Functional Specification

Appearance Properties

Texture
From Functional 2 Direct Specification

Base Materials → Mapping / Optimization → Fabrication

Target Object → Output
Questions?
Inverse Design of Deformable Shapes
Bernd Bickel
Deformable Objects

[shapeways.com]

[RBO Hand, TU Berlin]

[Festo]
Manufacturing Deformable Materials

intrinsic

[Hiller and Lipson 2009]

extrinsic

[Bickel et al. 2010]
[Skouras et al. 2013]

[Chen et al. 2013]

[Bickel et al. 2012]
[Chen et al. 2014]

[Schumacher et al. 2015]
[Panetta et al. 2015]
[Martínez et al. 2016] [Konakovic et al. 2016]
The 4th dimension is the set of behavioral rules that are pre-programmed into a 3D-printed shape. Based on any number of stimuli, the object can be set to respond differently. These responses can take a near limitless number of forms including (but definitely not limited to) changes in color, temperature, shape, movement, ...
From Functional 2 Direct Specification

Base Materials → Search & Simulation → Fabrication → Output

Target Object
From Functional 2 Direct Specification

Base Materials → Search & Simulation → Fabrication → Output

Target Object
Manufacturing Deformable Materials

- **Intrinsic**
  - [Hiller and Lipson 2009]
  - [Schumacher et al. 2015]
  - [Panetta et al. 2015]
  - [Martínez et al. 2016]

- **Extrinsic**
  - [Bickel et al. 2010]
  - [Skouras et al. 2013]
  - [Chen et al. 2013]
  - [Bickel et al. 2012]
  - [Chen et al. 2014]
Goal
Approach 1: Topology Optimization

[Sigmund 2009]
[Schumacher et al. 2015]
Classes of structural optimization methods

Initial

Optimized

Size

Shape

Topology

[courtesy Aage 2017]
Generating Optimal Topologies

[Bendsøe and Kikuchi 1988]

Design domain

Optimal material redistribution

Interpretation

[courtesy Aage 2017]
Discrete Topopt Formulation

\[
\begin{align*}
\min_{\rho} & \quad \Phi(\rho, U(\rho)) \\
\text{s.t.} & \quad \sum_{e=1}^{N} v_e \rho_e = v^T \rho \leq V^* \\
& \quad g_i(\rho, U(\rho)) \leq g_i^*, \quad i = 1, \ldots, M \\
& \quad \rho_e = \begin{cases} 
0 & \text{(void)} \\
1 & \text{(material)} 
\end{cases}, \quad e = 1, \ldots, N \\
& \quad K(\rho)U = F
\end{align*}
\]

0/1 Integer problem

Huge number of combinations!

[courtesy Aage 2017]
[Sigmund 2015]
SIMP-approach
(Simplified Isotropic Material with Penalization)

\[
\begin{align*}
\min_{\rho} & : \Phi(\rho, U(\rho)) \\
\text{s.t.} & : \sum_{e=1}^{N} v e \rho_e = v^T \rho \leq V^* \\
& : g_i(\rho, U(\rho)) \leq g_i^*, \quad i = 1, \ldots, M \\
& : 0 < \rho_{\text{min}} \leq \rho \leq 1 \\
& : K(\rho) U = F
\end{align*}
\]

Stiffness interpolation:
\[
E(\rho_e) = \rho_e^p E_0
\]

\[p \geq 1\]
[courtesy Aage 2017]
Sensitivity Analysis by Adjoint Method

• A general function and a general residual
  \( \Phi = \Phi(\rho, \mathbf{u}(\rho)), R(\rho, \mathbf{u}(\rho)) = 0 \)

• Step 1: differentiate using the chainrule
  \[
  \frac{d\Phi}{d\rho_e} = \frac{\partial \Phi}{\partial \rho_e} + \left( \frac{\partial \Phi}{\partial \mathbf{u}} \right) \frac{d\mathbf{u}}{d\rho_e} \quad \frac{dR}{d\rho_e} = \frac{\partial R}{\partial \rho_e} + \left( \frac{\partial R}{\partial \mathbf{u}} \right) \frac{d\mathbf{u}}{d\rho_e} = 0
  \]

• Use the residual eqs:
  \[
  \frac{d\mathbf{u}}{d\rho_e} = - \left( \frac{\partial R}{\partial \mathbf{u}} \right)^{-1} \frac{\partial R}{\partial \rho_e}
  \]

[Slide from Aage 2017]
Sensitivity Analysis by Adjoint Method

- **Step 2:** Insert trouble term into derivative
  \[ \frac{d\Phi}{d\rho_e} = \frac{\partial \Phi}{\partial \rho_e} + \frac{\partial \Phi}{\partial \mathbf{u}} \left( - \frac{\partial \mathbf{R}}{\partial \mathbf{u}} \right)^{-1} \frac{\partial \mathbf{R}}{\partial \rho_e} \]

- **Step 3:** Adjoint problem
  \[ \lambda^T = - \frac{\partial \Phi}{\partial \mathbf{u}} \left( \frac{\partial \mathbf{R}}{\partial \mathbf{u}} \right)^{-1} \Rightarrow \frac{\partial \mathbf{R}^T}{\partial \mathbf{u}} \lambda = - \frac{\partial \Phi}{\partial \mathbf{u}} \]

- **Final sensitivity**
  \[ \frac{d\Phi}{d\rho_e} = \frac{\partial \Phi}{\partial \rho_e} + \lambda^T \frac{\partial \mathbf{R}}{\partial \rho_e} \]

[Slide from Aage 2017]
Mesh-dependence

[Slide from Aage 2017]
Regularization

Checkerboards

Mesh refinement

[Slide from Aage 2017]
TopOpt Scheme

1. Initialize FEM
2. Finite Element Analysis (Elastic, Thermal, Electrical, ...)
3. Sensitivity Analysis
4. Regularization (filtering)
5. Optimization
6. $\rho_e$ converged?
7. STOP
8. $KU = F$ or $R(U) = 0$
“TopOpt App” from DTU

http://www.topopt.dtu.dk
Approach 1: Topology Optimization

\[ x_j = x_1^{\text{base}} + x_i - x_0^{\text{base}} \]

[Smit et al. 1998]
[Kharevych et al. 2009]
Approach 1: Topology Optimization

\[ O(\alpha) = \| C_{goal} - \tilde{C}(\rho_i) \|_F^2 + R \]
Approach 1: Topology Optimization

\[ O(\alpha) = \| C_{\text{goal}} - \tilde{C}(\rho_i) \|_F^2 + R \]
TopOpt – Family of Methods

• Density-based (what we have seen before)
• Implicit methods
• Topological derivatives
• Discrete approaches (Evolutionary methods)
• Combined shape and topology approaches
Approach 2: Systematic Topology Enumeration

[Panetta et al. 2015]
Approach 2: Systematic Topology Enumeration

[Panetta et al. 2015]
Microstructure Shape Optimization

- **Thickness** and **offset** parameters continuously control microstructure’s **shape**, $\omega$
- Fit the microstructure to an elasticity tensor:

\[
J(\omega) = \left|\left| C^H(\omega) - C^* \right|\right|^2
\]

[Panetta et al. 2015]
Shape Optimization Results

[Panetta et al. 2015]
Shape Optimization Results

[Panetta et al. 2015]
Manufacturing Deformable Materials

intrinsic

[Hiller and Lipson 2009]
[Schumacher et al. 2015]
[Panetta et al. 2015]
[Martínez et al. 2016]

extrinsic

[Bickel et al. 2010]
[Skouras et al. 2013]
[Chen et al. 2013]

[Bickel et al. 2012]
[Chen et al. 2014]
From Functional 2 Direct Specification

**Input**

- Shape with assigned Material Parameters
- Deformation Specification

**Output**

- Spatially-varying material structure
From Functional 2 Direct Specification

Input

• Shape with assigned Material Parameters

• Deformation Specification

Output

• Spatially-varying material structure

mapping
Challenge: Find optimal combinations

[Schumacher et al. 2015]
Synthesis

Challenge: Find optimal combinations

[Schumacher et al. 2015]
Periodic Tiling: Challenges

• Mapping? Possible, but difficult.

Hexahedral-dominant meshing [Sokolov et al. 2015]

• Gradation? Possible, but transitions?

[Schumacher et al. 2015] – solves an optimization problem for finding compatible tilings
Procedural Synthesis for Fabrication

\[ F(x,y,z) \]

Slice

Fill with microstructure

[Martínez et al. 2016]
Procedural Synthesis for Fabrication

Printed with Autodesk Ember

[Martínez et al. 2016]
Procedural Synthesis for Fabrication: Result Cute Octopus

Printed with B9 Creator

[Martínez et al. 2016]
From Functional 2 Direct Specification

**Input**
- Shape with assigned Material Parameters
- Deformation Specification

**Output**
- Spatially-varying material structure
Problem Formulation

\[ E(x, p) = E_{match}(x, x_{target}) \quad \text{subject to} \quad f^{total}(x, p) = 0 \]
Problem Formulation

\[ E(x, p) = E_{\text{match}}(x, x_{\text{target}}) \quad \text{subject to} \quad f^{\text{total}}(x, p) = 0 \]

Optimization Strategies:
- Discrete ([Bickel et al. 2010])
- Continuous ([Skouras et al. 2013])
Result

Flipflop: side view

Flipflop: top view

[Bickel et al. 2010]
Ready to Wear

[Bickel et al. 2010]
Material Distribution Optimization

pose 1

pose 2

pose 3

stiff

soft
Material Distribution Optimization
Results

Rest Pose

Target Pose

Stiff  Soft
CurveUps: Shaping Objects from Flat Plates with Tension-Actuated Curvature

R. Guseinov, E. Miguel, B. Bickel

ACM Transactions on Graphics (Proc. SIGGRAPH 2017)
Fabrication
Conclusion

• Summary
  – Control at various levels
  – Techniques could be combined
  – Interactive vs. Specification-Based Design

• Limitations / Future Work
  – Scaling
  – Non-linear material behavior can be very complex
  – Fabrication constraints often quite specific
  – 3D printer
    • Durability of materials
    • Handling of materials
Thank you!

Acknowledgements
Niels Aage, Melina Skouras, Jeremie Dumas, Sylvain Lefebvre, and Julian Panetta for providing slides/image material.

References


Simplifying Design of Foldable Shapes

Niloy J. Mitra
Origami
Origami

Simplifying Making of Foldable Shapes
Curved Folding

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

[Disney Concert Hall, F. Gehry (w/Kilian, Sheffer, Pottmann, et al.)]
Piecewise Developable Surface

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

created by: David Huffman, Gregory Epps
How to Create One?

Curved Folding

Problem Formulation

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.
Developable Surface
Developable Surface

- vanishing Gaussian curvature
Developable Surface

- vanishing Gaussian curvature
- non-flat $\Rightarrow$ 1-parameter family of tangent planes
Developable Surface

- vanishing Gaussian curvature
- non-flat $\Rightarrow$ 1-parameter family of tangent planes
  - cones, cylinders, tangent surfaces
Developable Surface

- vanishing Gaussian curvature
- non-flat $\Rightarrow$ 1-parameter family of tangent planes
  - cones, cylinders, tangent surfaces
- $\Rightarrow$ ruled surface
Developable Surface

• vanishing Gaussian curvature

• non-flat $\Rightarrow$ 1-parameter family of tangent planes
  – cones, cylinders, tangent surfaces

• $\Rightarrow$ ruled surface
  – same tangent plane along same generator (ruling)
Developable Patches

• PQ strips
• triangle fans
• planar polygons
Developable Patches

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

- PQ strips
- triangle fans
- planar polygons
Developable Patches

- PQ strips
- triangle fans
- planar polygons

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

- Simplifying Making of Foldable Shapes

- Smooth vs. Discrete

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

  - PQ strips
  - triangle fans
  - planar polygons
Step 1: Estimate Rulings

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.

Step 1: Estimate Rulings

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
Step 2: Unfold to a Plane

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
Step 3: 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.
Step 4: 2D-3D Optimization

Curved Folding

Result of Initialization

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.

Botsch et al. PriMo: Coupled Prisms for Intuitive Surface Modeling
Representation
Representation
Representation
Discretizing Curvature
Discretizing Curvature
Discretizing ‘Folds’
Discretizing ‘Folds’
Discretizing ‘Folds’
Step 4: 2D-3D 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.

[Botsch et al. 2008]

PriMo: Coupled Prisms for Intuitive Surface Modeling
Step 4: 2D-3D 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.

[Botsch et al.] PriMo: Coupled Prisms for Intuitive Surface Modeling
Step 4: 2D-3D Optimization

The objective function in more detail

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

\[
F_{\text{vert}} := \sum_{p \in P} (m^i_p - m^i_p)^2
\]

\[
F_{\text{fit}} := \sum_{m \in M} ((m - m_e) \cdot n_c)^2
\]

\[
F_{\text{fair}} := \sum_{e_{ij} \in E} w_{ij} (n'_i - n'_j)^2
\]
Step 4: 2D-3D Optimization

Curved Folding

The Objective Function

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

\[ F_{\text{vert}} := \sum_{p \in P} (m_{p}^j - m_{p}^i)^2 \]

\[ F_{\text{fit}} := \sum_{m \in M} ((m - m_{c}) \cdot n_{c})^2 \]

\[ F_{\text{fair}} := \sum_{e_{ij} \in E} w_{ij} (n_{i} - n_{j})^2 \]

\[ E = \int_{S} \kappa_{1}^2 + \kappa_{2}^2 \, dA \]
Step 4: 2D-3D Optimization

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

\[ F_{\text{vert}} := \sum_{p \in P} (m_p^i - m_p^j)^2 \]

\[ F_{\text{fit}} := \sum_{m \in M} ((m - m_c) \cdot n_c)^2 \]

\[ F_{\text{fair}} := \sum_{e_{ij} \in E} w_{ij} (n_i' - n_j')^2 \]

\[ E = \int_S \kappa_1^2 dA \]
Final Output

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
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
Results

Simplifying Making of Foldable Shapes
Results: Paper Models
(Constrained) Shape Space

- Mesh maps to a point in $\mathbb{R}^d$
- Deformation maps to a path in that space
(Constrained) Shape Space

- Mesh maps to a point in \( \mathbb{R}^d \)
- Deformation maps to a path in that space

[w/ Yang, Yang, Pottmann]
(Constrained) Shape Space

- Mesh maps to a **point** in $\mathbb{R}^d$
- Deformation maps to a **path** in that space

Simplifying Making of Foldable Shapes

[w/ Yang, Yang, Pottmann]
(Constrained) Shape Space

- Mesh maps to a **point** in $\mathbb{R}^d$
- Deformation maps to a **path** in that space
Constrained Meshes
Constrained Meshes

- Given:
  
  single input mesh with a set of *non-linear constraints* in terms of mesh vertices
Constraining Meshes

Given:

- single input mesh with a set of **non-linear constraints** in terms of mesh vertices

Goal:

- explore *neighboring* meshes respecting the prescribed constraints
- based on different application requirements, navigate only the *desirable* meshes according to given quality measures
Map Mesh to Point
Map Mesh to Point

- The family of meshes with *same combinatorics*
Map Mesh to Point

- The family of meshes with **same combinatorics**
- Mesh to point

\[ \mathbf{x} := (v_1, \ldots, v_n) \in \mathbb{R}^D \]
Map Mesh to Point

- The family of meshes with **same combinatorics**
- Mesh to point

\[ \mathbf{x} := (v_1, \ldots, v_n) \in \mathbb{R}^D \]

- Displacement vector to update the current mesh

\[ \mathbf{d} \Rightarrow \mathbf{x}_0 + \mathbf{d} \]
Map Mesh to Point

- The family of meshes with **same combinatorics**
- Mesh to point
  \[ \mathbf{x} := (v_1, \ldots, v_n) \in \mathbb{R}^D \]
- Displacement vector to update the current mesh
  \[ \mathbf{d} \Rightarrow \mathbf{x}_0 + \mathbf{d} \]
- Distance measure
  \[ d(\mathbf{x}_1, \mathbf{x}_2) := \| \mathbf{x}_1 - \mathbf{x}_2 \| \]
Constrained Mesh Manifold

- Constrained mesh manifold $M$:
  - represents all the meshes under a set of non-linear constraints

- Individual constraint
  - $E(x_i) = 0$ defines a hypersurface in $\mathbb{R}^D$
Constrained Mesh Manifold

- Involving $m$ constraints in $\mathbb{R}^D$

$$\Gamma_i = \{ \mathbf{x} \in \mathbb{R}^D | E_i(\mathbf{x}) = 0 \}, \quad i = 1, \ldots, m$$
Constrained Mesh Manifold

- Involving $m$ constraints in $\mathbb{R}^D$
  \[ \Gamma_i = \{ x \in \mathbb{R}^D | E_i(x) = 0 \}, i = 1, \ldots, m \]

- $M$ is the intersection of $m$ hypersurfaces
  - dimension $D-m$ (tangent space)
  - codimension $m$ (normal space)
Example: PQ Mesh Manifold

- PQ mesh manifold $M$: $f_i \rightarrow E_i$
- Constraints (planarity per face)
  - each face $|E_i| \leq \epsilon$ (signed diagonal distance)

- deviation from planarity
- 10mm allowance for 2m x 2m panels
Tangent Space

- Starting mesh $x_0$ is PQ

$$T_M(x_0) := \{ x_0 + t \mid \nabla E_i^T(x_0) \cdot t = 0 \forall i = 1, \ldots, m \}.$$

- Geometrically, intersection of all the tangent planes of the hypersurfaces.
Walking on the Tangent Space

PQ mesh manifold
Walking on the Tangent Space

PQ mesh manifold

deformation field
Walking on the Tangent Space
Walking on the Tangent Space
Walking on the Tangent Space
Better Approximation?

- Tangent space - 1st order approximant

straight path ignores the curvature of the manifold

constrained mesh manifold
Better Approximation?

- Better approximation - 2\textsuperscript{nd} order approximant

\[
\begin{align*}
\mathbf{x}_0 & \quad \mathbf{p}(u) \\
t & \quad \mathbf{n}(t)
\end{align*}
\]

curved path

\textit{considers the curvature of the manifold}
Compute Osclulant

- Generalization of the osculating paraboloid in 3D
- Has the following form:

\[ S(u) = x_0 + \sum_{i=1}^{D-m} u_i e_i + \frac{1}{2} \sum_{j=1}^{m} (u^T \cdot A_j \cdot u) n_j \]

- Second order contact with each of the constraint
Walking on the Osculant
Walking on the Osculant

Simplifying Making of Foldable Shapes
Walking on the Osculant

Simplifying Making of Foldable Shapes
Walking on the Osculant

Simplifying Making of Foldable Shapes
Mesh Quality?

- Osculant concerns only constraints
- Quality measures based on application
  - fairness – meshes with beautiful structure
- Extract the *meaningful* part of the manifold
What Do We Gain?

Simplifying Making of Foldable Shapes
Spectral Analysis

- Good (desirable) subspaces to explore
- 2d-slice of design space
PQ Mesh Shape Space Exploration

Exploration 2D
model: Six  # v: 720  # f: 656

Geometry  Planarity

Tangent  Osculant

live capture
Handle Driven Exploration

\[ \min_{t} F(x_0 + t) \text{ such that, } \]

\[ \nabla E^T_i \cdot t = 0, \quad \forall i = 1, \ldots, m; \]

\[ t_j = v_j' - v_j \]
Handle Driven Exploration

Simplifying Making of Foldable Shapes
Other Constrained Meshes
Circular Mesh Manifolds

- Circular Meshes
  - Each face has a circumcircle

\[
(\alpha_1, \alpha_2, \alpha_3, \alpha_4) \\
\Rightarrow \alpha_1 + \alpha_3 = \pi
\]
Circular Mesh Manifolds

- Circular Meshes
  - Each face has a circumcircle

\[ (\alpha_1, \alpha_2, \alpha_3, \alpha_4) \]
\[ \Rightarrow \alpha_1 + \alpha_3 = \pi \]

\[ E^c_i : \alpha_1 + \alpha_3 - \pi \]
Circular Mesh Shape Space
Circular Mesh Shape Space

$E^c = \alpha_1 + \alpha_3 - \pi$

max$_i |E^c_i|$

average displacement/vertex (mm)

zoom A

A

B

Simplifying Making of Foldable Shapes
Circular Mesh Shape Space

\[ E^C = \alpha_1 + \alpha_3 - \pi \]

Simplifying Making of Foldable Shapes
Combined Constraints Manifolds

- Input circular mesh
- Circular polyline
- Planar polyline (floor constraint)
- Mesh obtained via constrained shape space exploration

Simplifying Making of Foldable Shapes
Combined Constraints Manifolds

Exploration with floor and circle constraints
model: Roof
#faces: 75
Combined Constraints Manifolds

Exploration with floor and circle constraints
model: Roof
#faces: 75
Flat Circular Mesh Exploration
What can be Folded?
Curved Folds

[w/ Kilian, Monszpart]
Curved Folds

What happens when folding along curved creases?

[w/ Kilian, Monszpart]
How to Fold?

[ © RoboFold]
How to Fold?

[© RoboFold]
String Actuated Folding

Given a folding sequence $S: [0, 1] \times \mathbb{R}^3$, we are looking for a set of actuation points and network of strings in order to reproduce the given deformation.
String Actuated Folding

Given a folding sequence $S: [0, 1] \times U!^3$ we are looking for a set of actuation points and network of strings in order to reproduce the given deformation.
String Actuated Folding

\[ S : [0, 1] \times U \rightarrow \mathbb{R}^3 \]
String Actuated Folding

\[ S : [0, 1] \times U \rightarrow \mathbb{R}^3 \]

\[ S(0) \]

\[ S(1) \]
String Actuated Folding

\( S : [0, 1] \times U \rightarrow \mathbb{R}^3 \)

\( S(0) \quad S(t_0) \quad S(1) \)
String Actuated Folding

\[ S : [0, 1] \times U \rightarrow \mathbb{R}^3 \]

\( S(0) \quad S(t_0) \quad S(1) \)
String Actuated Folding

\[ S : [0, 1] \times U \rightarrow \mathbb{R}^3 \]
Questions
Questions

• Which points to connect?
Questions

• **Which** points to connect?

• **How** to connect the points?
Deformation in Shape Space

At every instant $t \in [0, 1]$ of time the infinitesimal deformation of the surface $S(t)$ is described by the vector field $X(t, u) = \partial_t S(t, u)$. 
Deformation in Shape Space

\[ X(t_0, u) := \frac{\partial}{\partial t} S(t_0, u) \]
Deformation in Shape Space

\[ X(t_0, u) := \frac{\partial}{\partial t} S(t_0, u) \]
Main Idea

- Express deformation in terms of actuation modes

\[ X_i(t_0, u) := \frac{\partial}{\partial s} S_i(t_0)(0, u) \]
Deformations using ‘Actuation Modes’

Figure 3: A folding sequence describes a curve \( t \) in shape space, \( \gamma \), such that all intermediate surfaces \( S(t) \) are isometric to \( S(0) \). This curve is described by the vector field \( \partial S \gamma(t) \). We show an example of such a deformation field into a simpler, finite set of modes of manual interactions into a simple set of modes.

Figure 4: For now the means of discretization are not important and we argue that transforms the planar sheet into an isometric deformation, there is a strong regularization given by the bending energy \( E \) of the deforming sheet. Hence, the deformation energy is optimized with parameter domain \( t \) as a curve in shape space, such that the folded target surface \( S(1) \) is discovered by increasing the fold magnitudes while respecting their orientation of a fold the folded shape is then known to the right. Because of the physical nature of curved crease Origami, we relieve the practitioner of this feature of curved crease Origami. We relieve the practitioner of this feature of curved crease Origami.
Deformations using ‘Actuation Modes’

\[ X(t_0) \approx \sum_i \lambda_i(t_0)X_i(t_0) \]
Deformations using ‘Actuation Modes’

\[ X(t_0) \approx \sum_i \lambda_i(t_0) X_i(t_0) \]

\[ \min_{\Lambda} \| X(t_0) - \sum_i \lambda_i(t_0) X_i(t_0) \|^2 \]
Example: Actuation Modes

![Diagram of actuation modes](image)

S1

S2

Section

In this section, we discuss the computation of actuation modes for a given structure. The modes are calculated through a least norm least squares solution, ensuring that the required data is obtained. Typically, strings with similar endpoints give rise to similar modes, which we refer to as the mode induced by the string. This induces a deformation in terms of the local modes.

After introducing our computational framework to compute the deformation, we can prescribe face transformations to rigid body motions. After applying this framework, we have a set of admissible transformations that minimize the energy of the system. In a nutshell, we represent surfaces as adaptive triangle meshes subject to a non-linear surface energy cost function. The model is based on the concepts introduced previously.

In the following section, we present our approach to discretize the deforming model. We use a local approach to adaptively refine and coarsen the mesh where necessary. This allows for efficient computation of the deformation field.

Online Submission ID: papers

Simplifying Making of Foldable Shapes
Example: Actuation Modes
Example: Actuation Modes

To measure this gap a prism face along the edge and recover the planar state. To measure this gap a prism face along the edge and recover the planar state.

Throughout this paper we will use the convention that any triangle isometric deformations of triangle meshes are characterized by (isometry) and bending. Conversely, we can prescribe face transformations that are not chosen carefully. To penalize the amount of stretch we want this will still incur arbitrarily large distortion if the transformations from a planar reference mesh. Hence we restrict the set of admissibles that is crease aware and accounts for both, stretch minimization and recover.

We achieve this by penalizing the gap between neighboring triangles and their transformations are isometry and bending.

Throughout this paper we will use the convention that any triangle isometric deformations of triangle meshes are characterized by (isometry) and bending. Conversely, we can prescribe face transformations that are not chosen carefully. To penalize the amount of stretch we want this will still incur arbitrarily large distortion if the transformations from a planar reference mesh. Hence we restrict the set of admissibles that is crease aware and accounts for both, stretch minimization and recover.

We achieve this by penalizing the gap between neighboring triangles and their transformations are isometry and bending. Conversely, we can prescribe face transformations that are not chosen carefully. To penalize the amount of stretch we want this will still incur arbitrarily large distortion if the transformations from a planar reference mesh. Hence we restrict the set of admissibles that is crease aware and accounts for both, stretch minimization and recover.
Example: Actuation Modes

Figure 5: Simplifying Making of Foldable Shapes
Example: Actuation Modes
Example: Actuation Modes

We present a solution to this problem in our setting in the form of a set of pairs over time. In summary, we are looking for a sparse set of strings, which can result in a different sparsity pattern in the coefficient vector. This will also minimize the number of non-zeros in the coefficient vector. 

Actuation modes. Top row (left to right): a network of strings with similar endpoints, giving rise to similar modes, i.e., the corresponding matrix is rank deficient. Additionally, strings with similar endpoints give rise to similar modes. In general, the set of modes will neither form a basis nor a set of generators, i.e., the corresponding matrix is rank deficient.

We are looking for a representation of the global deformation field to be joined by a string. Tightening the string reduces the distance of the planar state. To measure this gap, we produce a gap between prism faces that are perfectly aligned in the planar state. To approximate the deformation behavior of paper, or any other sheet material with high in-plane stiffness, we need a notion of bending energy. The above idea can be generalized by enclosing the sheet material with high in-plane stiffness, we need a notion of bending energy. The above idea can be generalized by enclosing the sheet material with high in-plane stiffness, we need a notion of bending energy. The above idea can be generalized by enclosing the sheet material with high in-plane stiffness, we need a notion of bending energy.

Conversely, we can prescribe face transformations to rigid body motions. After applying $X_i(t_0)$, we can prescribe face transformations to rigid body motions. After applying $X_i(t_0)$, we can prescribe face transformations to rigid body motions. After applying $X_i(t_0)$, we can prescribe face transformations to rigid body motions. After applying $X_i(t_0)$, we can prescribe face transformations to rigid body motions.

Throughout this paper, we will use the convention that any triangle is adaptive triangle meshes subject to a non-linear surface energy minimization, i.e., $X_i(t_0)$, $X_i(t_0)$, $X_i(t_0)$, $X_i(t_0)$. We are used to define the shape of a mesh in terms of vertex displacements $u_i$ and let $\Delta X_i(t_0)$ be as close as possible before computing $X_i(t_0)$ to $X_i(t_0)$. We are used to define the shape of a mesh in terms of vertex displacements $u_i$ and let $\Delta X_i(t_0)$ be as close as possible before computing $X_i(t_0)$ to $X_i(t_0)$. We are used to define the shape of a mesh in terms of vertex displacements $u_i$ and let $\Delta X_i(t_0)$ be as close as possible before computing $X_i(t_0)$ to $X_i(t_0)$. We are used to define the shape of a mesh in terms of vertex displacements $u_i$ and let $\Delta X_i(t_0)$ be as close as possible before computing $X_i(t_0)$ to $X_i(t_0)$.
Globally Consistent Solution

\[ S(0), X_1(t_0), X_2(t_0), X(t_0), S(t_0), S(1) \]
Globally Consistent Solution

\[ S(0) \xrightarrow{X_1(t_0)} S(t_0) \xrightarrow{X(t_0)} X(t_1) \xrightarrow{X_2(t_0)} S(1) \]
Globally Consistent Solution

Simplifying Making of Foldable Shapes
strings such that shortening of those strings lifts the planar CP to a set of actuation points and how to connect them with a sequence of at once. The most basic variation of precreasing can be achieved by and does not require any coordinated folding along multiple creases would just crumple arbitrarily. Practically, precreasing is much eas- process of precreasing, folding a crease according to its moun- Once precreased, a CP can be brought from its flat state to a folded point where our string actuated folding process can simplify the cre- it does not help with the synchronized folding process. This is the practioner uses precreasing mainly to produce sharp and exact folds only consider precreasing according to fold orientation. While a crease feature of curved crease Origami. We relieve the practioner of this task by providing a simple means to simultaneously actuate creases and creases which have not been given a fold orientation are shown in gray. We validate our method computationally and pratically by realiz- A sequence of poses along a deformation Online Submission ID: papers
Global Solution

\[ g_{ij}(\xi, \lambda) := \xi_i \lambda_{ij}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n, \]

\[ g_{ij}(\xi, \lambda) = 0 \]
Global Solution

\[ g_{ij}(\xi, \lambda) := \xi_i \lambda_{ij}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n, \]

\[ g_{ij}(\xi, \lambda) = 0 \]

\[
\begin{align*}
\min \left[ w \sum_{i=1}^{m} (1 - \xi_i) + \sum_{j=1}^{n} \| X_j - \sum_{i=1}^{m} \lambda_{ij} X_{ij} \|^{2} \right]
\end{align*}
\]

\[ 0 \leq \xi_i \leq 1, \quad g_{ij}(\xi, \lambda) = 0, \quad 0 \leq \lambda_{ij}. \]
Sequence: QUAD
Sequence: QUAD
Implementation Details
Implementation Details

- **Isometric deformation** framework
  - driven by string lengths
  - searching for fold angles
Implementation Details

- **Isometric deformation** framework
  - driven by string lengths
  - searching for fold angles

- Prevent **self-intersection**
Implementation Details

- **Isometric deformation** framework
  - driven by string lengths
  - searching for fold angles

- Prevent **self-intersection**

- **Pruning** initial string candidates
Implementation Details

- **Isometric deformation** framework
  - driven by string lengths
  - searching for fold angles

- Prevent **self-intersection**

- **Pruning** initial string candidates

- **Dynamic** triangulation
Concept Chair

\[ S \]
Concept Chair

S

#1

b

a

Simplifying Making of Foldable Shapes
Concept Chair

$S$  

#1  
$\begin{align*}
    a \\
    b
\end{align*}$

#2  
$\begin{align*}
    A \\
    c \\
    d \\
    e \\
    f \\
    g
\end{align*}$

Simplifying Making of Foldable Shapes
Sequence: CONCEPT CHAIR
Sequence: CONCEPT CHAIR
Folded Canopy
Folded Canopy
Apricot: Multiple Solutions
Apricot: Multiple Solutions

Simplifying Making of Foldable Shapes
Sequence: APRICOT
Sequence: APRICOT
thank you

http://vecg.cs.ucl.ac.uk/Projects/SmartGeometry/
Open Challenges

Bernd Bickel
Niloy Mitra
• Giga voxels/inch$^3$, Tera voxels/foot$^3$
• What are good exchange formats? Standards?
• Smart and reusable material definitions
• Dithering strategies to obtain halftones representations
• Resolution and printer independence
• Simulate printing processes
• Predict quality
• Help users to deal with parameters
• Promoting open-source environment vs. commercial products
Cabin bracket for the Airbus A350 XWB made of Ti manufactured by Concept Laser GmbH [image from 3ders.org]
Going Beyond What an Engineer Can Do
Empowering Everyday Users

[Image of mechanical template, target shape, and functional model connected by an arrow labeled "RETARGETING"]

[mechanical template]

[target shape]

[functional model]

[Zhang et al. 2017]
Code For Construction - Self-Assembly

[Self-Assembly Lab, MIT]
Code For Construction - Self-Assembly

[Self-Assembly Lab, MIT]