3D Printing Oriented Design: Geometry and Optimization

Siggraph Asia 2014 Course

Dec. 5, 2014 , Shenzhen

SIGGRAPH ASIA 2014 SHENZHEN

Ligang Liu , Charlie Wang , Ariel Shamir, Emily Whiting

Web: http://staff.ustc.edu.cn/~lgliu/Courses/SigAsia_2014_course_3Dprinting/index.html

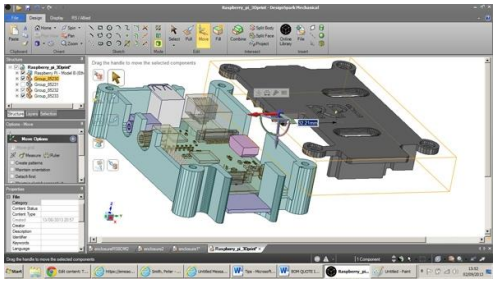
Part 4: Design Tools

- **Ariel Shamir**

- School of Computer Science
- The Interdisciplinary Center, Israel
- http://www.faculty.idc.ac.il/arik

Designing 3D Objects

- A difficult problem on its own:
  - 3D world on 2D displays
  - 3D manipulation using 2D (or 2.5D) devices
  - Complex: need expert knowledge
  - What is the digital representation?



Challenges Specifically for Fabrication & Printing

- Need to be actually constructed or printed:
  - Fitting parts
  - Finding intersections
  - Defining connectors
  - Checking printability

Geometry

- Need to be physically plausible:
  - Forces
  - Materials
  - Appearance

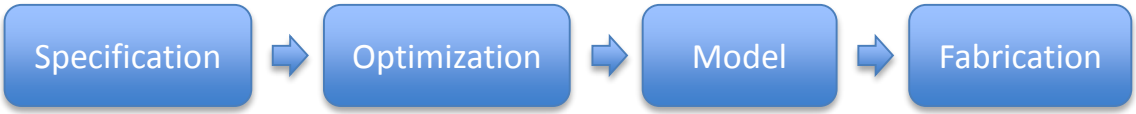
Physics

Part 4: Emily Whiting

Two Approaches

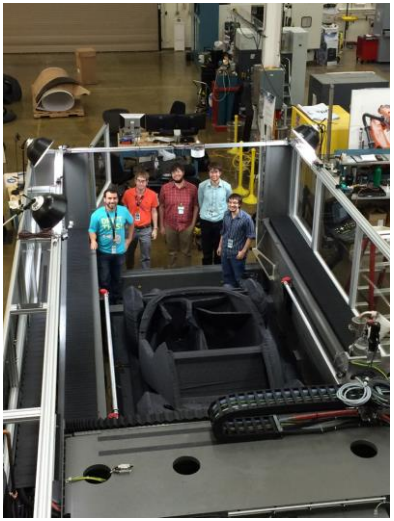
"Inverse Modeling"

1. Specification based design:



2. Interactive modeling:

Strati: First 3D Printed Car



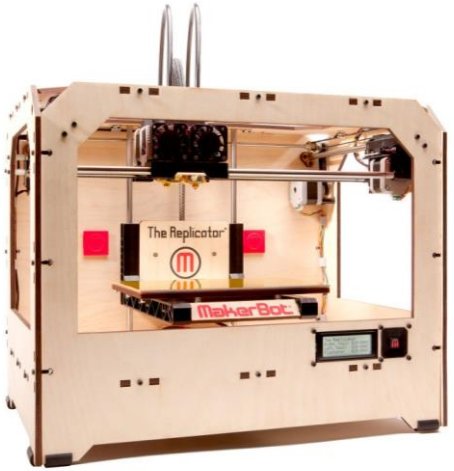
- Made by **Local Motors** for the International Manufacturing Technology Show (**IMTS**)
- Strati = “layers” in Italian
- 18 months of design
- 44 hours of printing

Usually Print Volume is Limited

- Typically for under $200k:

  10cm x 10cm x 10cm up to 50cm x 40cm x 30cm

- Printing large objects requires chopping and assembly

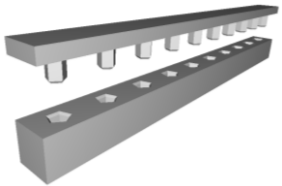


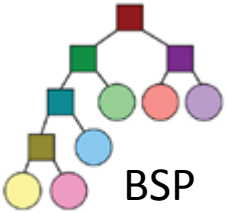
Two Challenges

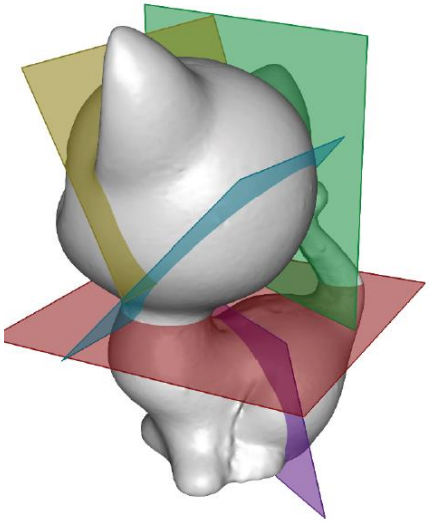
- How to segment?
- How to connect?

Two Challenges

- How to segment?
  - Use planes to define parts
  - Cut top down recursively
- How to connect?
  - Use male/female connectors:

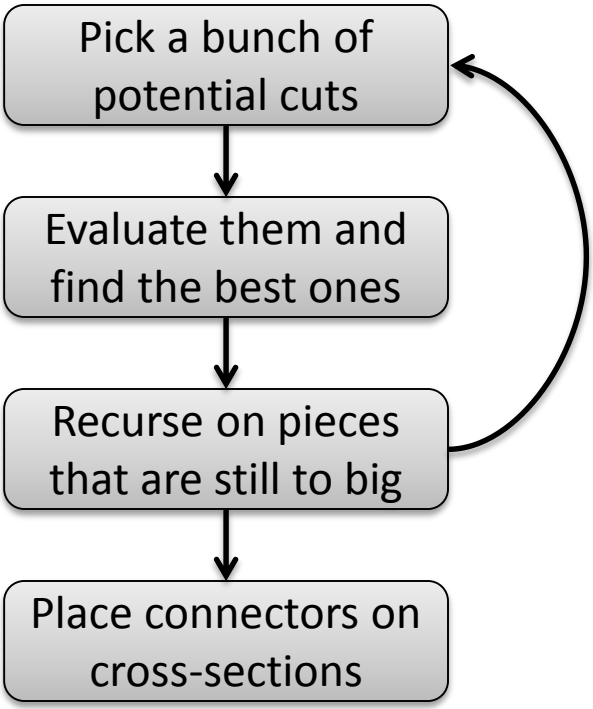






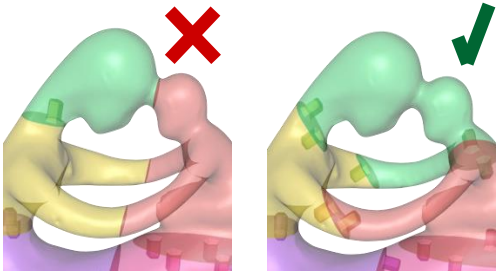
Chopper: Partitioning Models into 3D-Printable Parts
Linjie Luo, Ilya Baran, Szymon Rusinkiewicz, Wojciech Matusik
ACM Transactions on Graphics, 31(6), (SIGGRAPH Asia), 2012

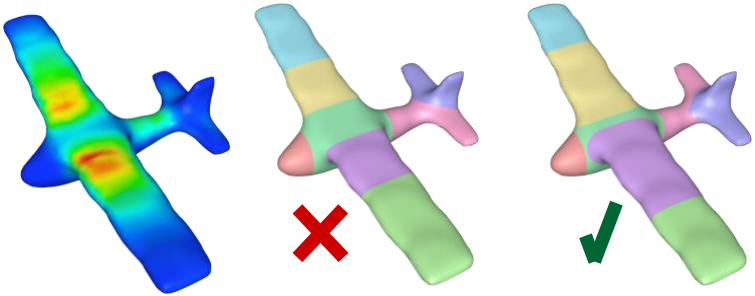
Chopper algorithm outline

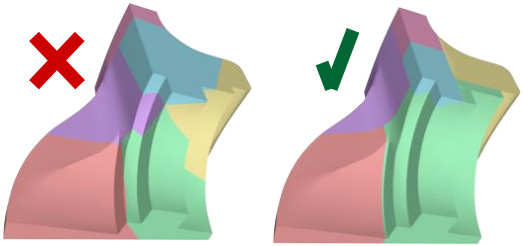


Optimization Objective

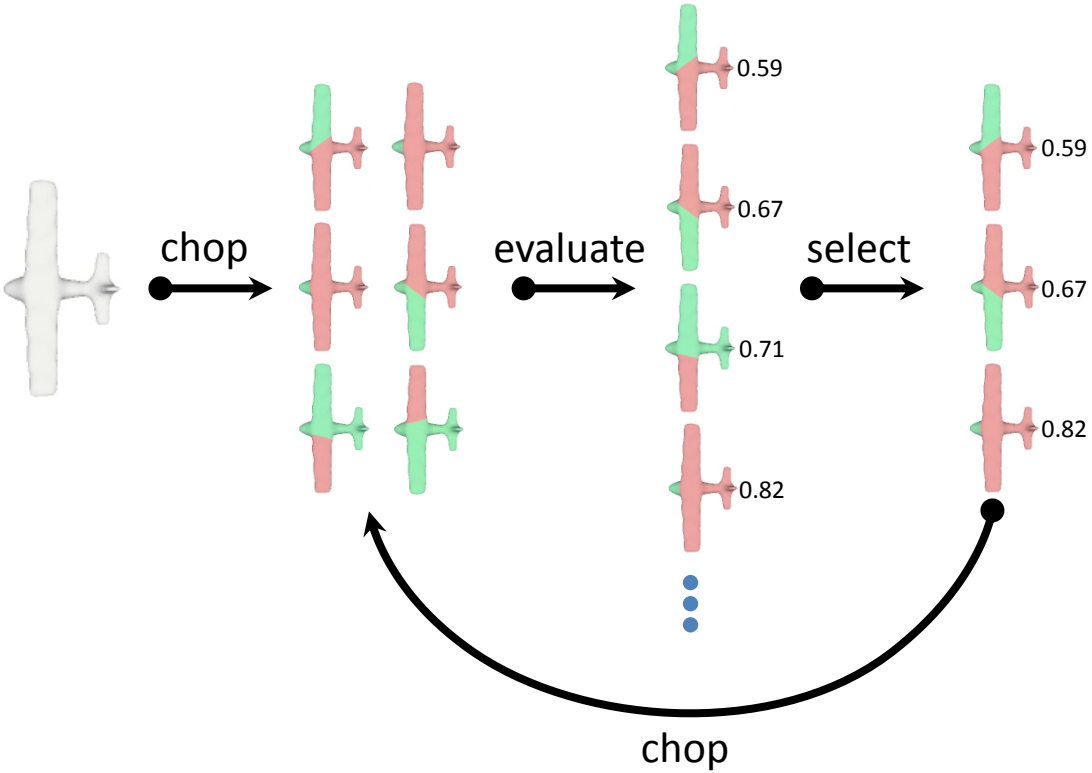
- Connector feasibility
- Part fragility
- Structural soundness when assembled
- Number of parts
- Printing volume utilization
- Seam aesthetics



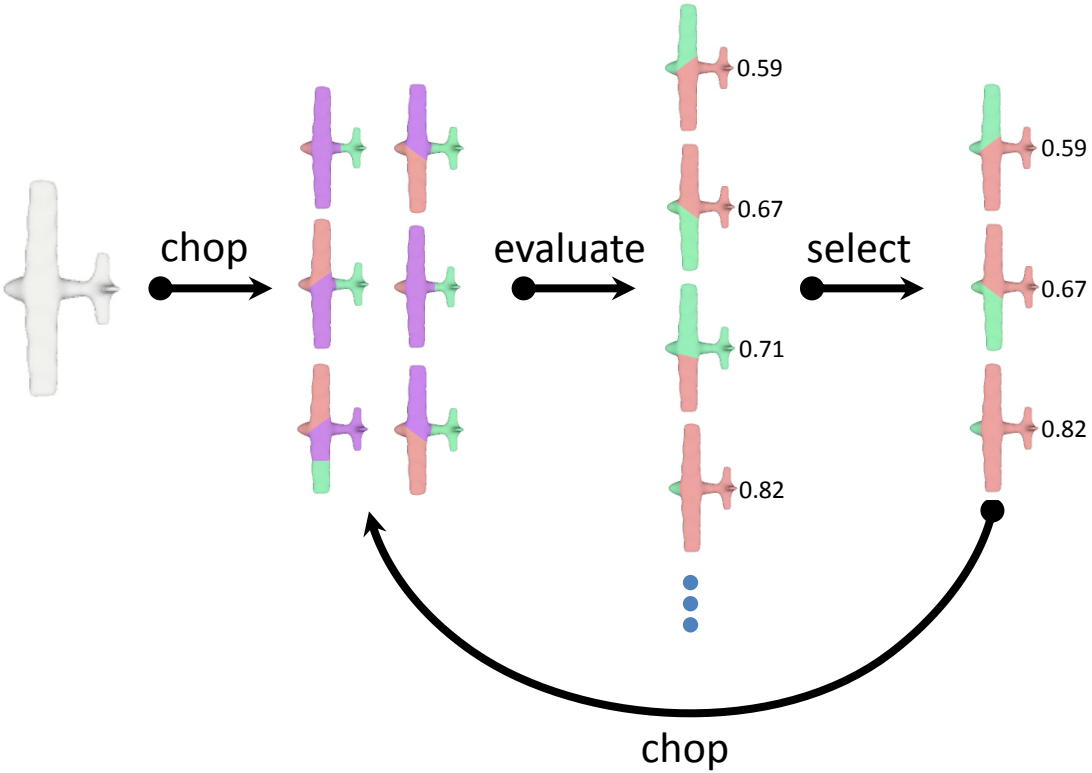




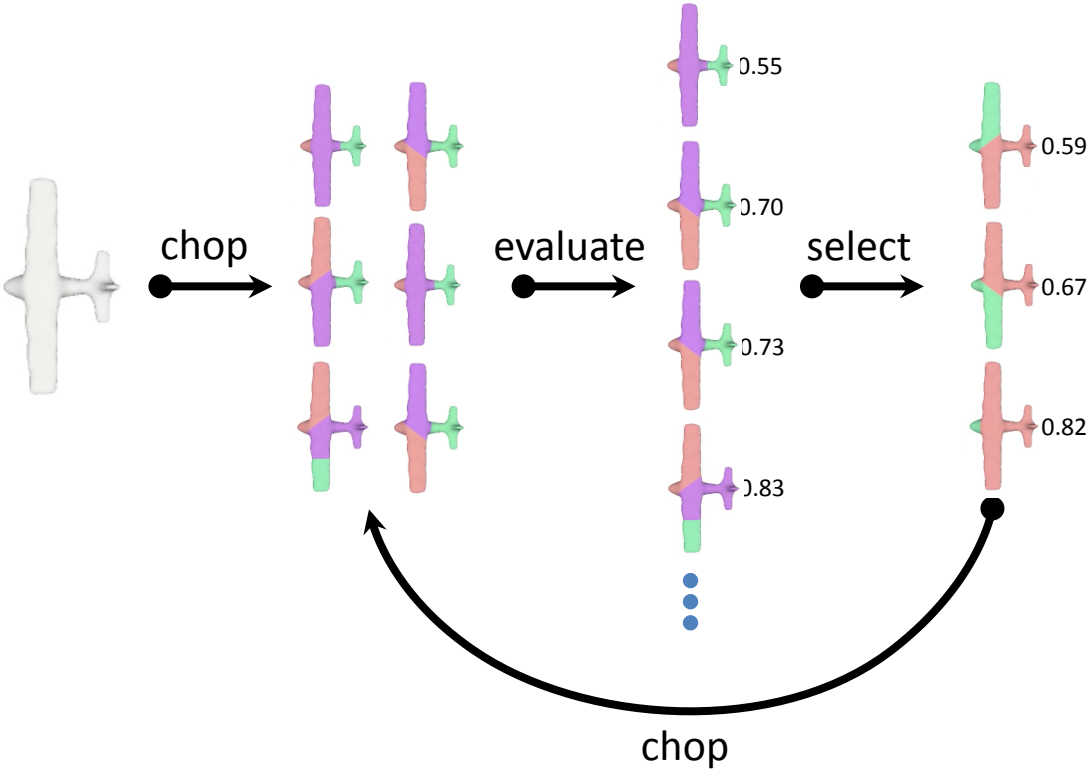
Optimization: Beam Search [Lowerre 1976]



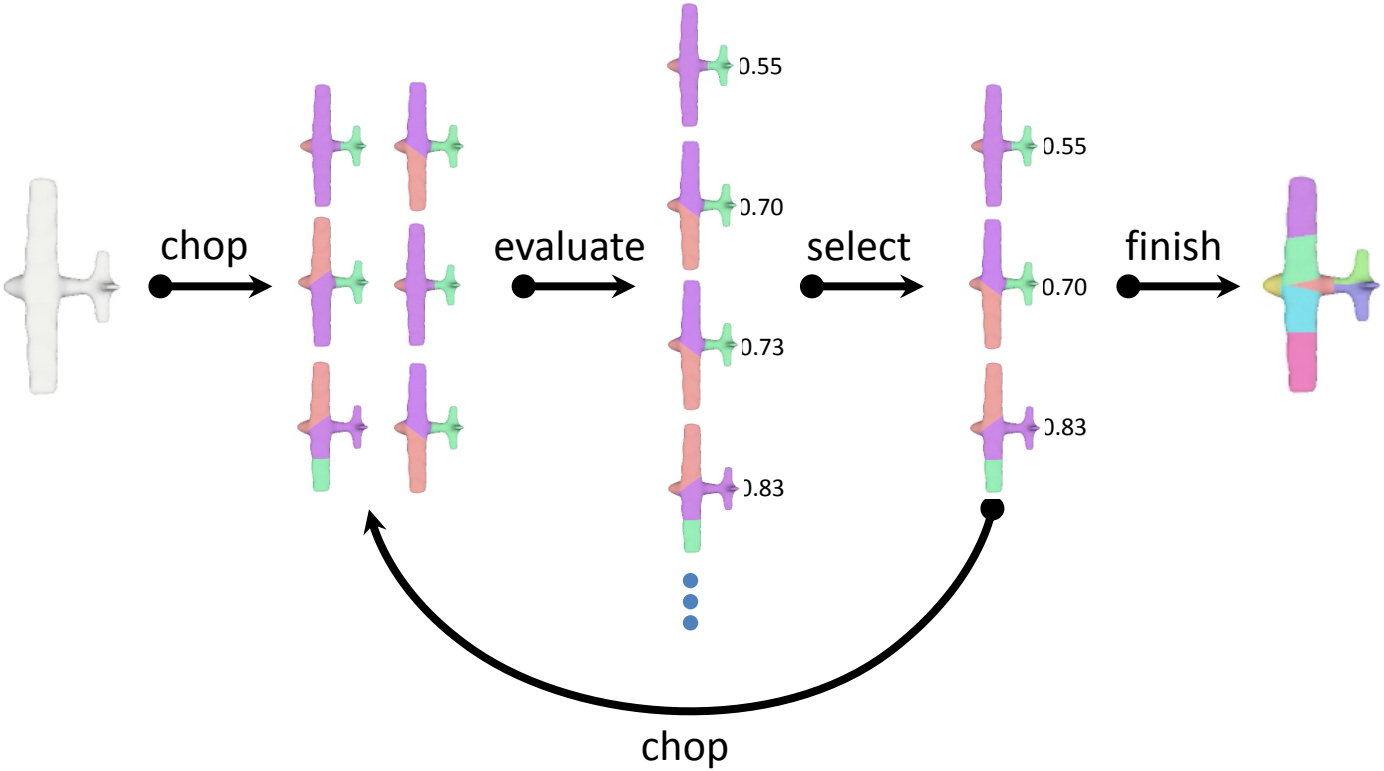
Optimization: Beam Search



Optimization: Beam Search

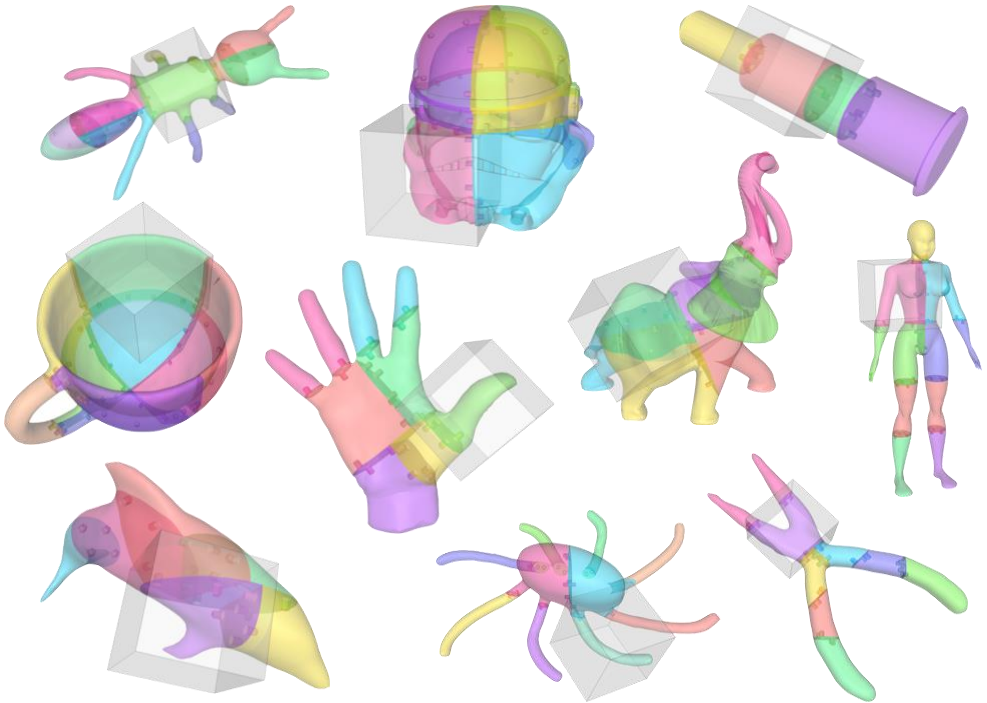


Optimization: Beam Search



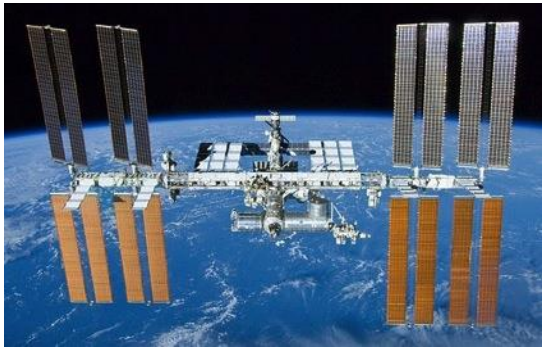
Results (a little sped up)

Some More Results



A New Challenge: Folding!

- One connected component
- Optimized printing
- Saving space



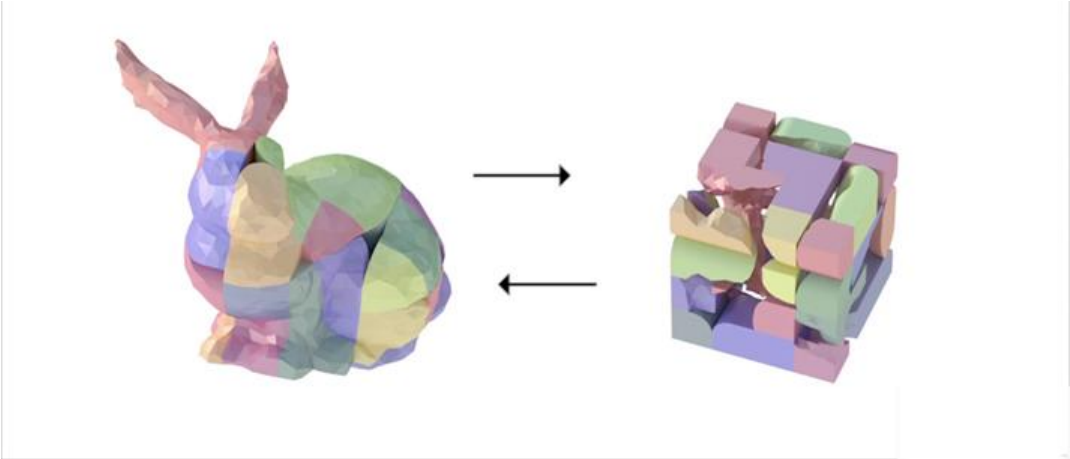
Transformers & Puzzles

[Song et al. 2012]

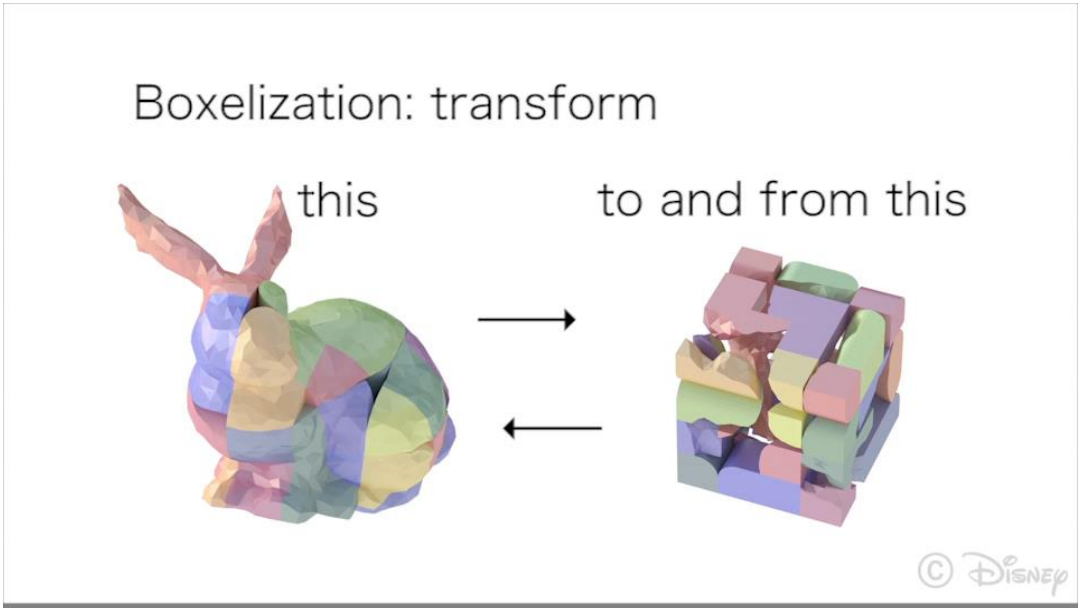
Boxelization

Convert this…

…to this



In One Piece…



Boxelization: Folding 3D Objects into Boxes
Yahan Zhou, Shinjiro Sueda, Wojciech Matusik, Ariel Shamir
ACM Transactions on Graphics, 33(4), (SIGGRAPH), 2014

…and Fabricate It

Why is it hard?

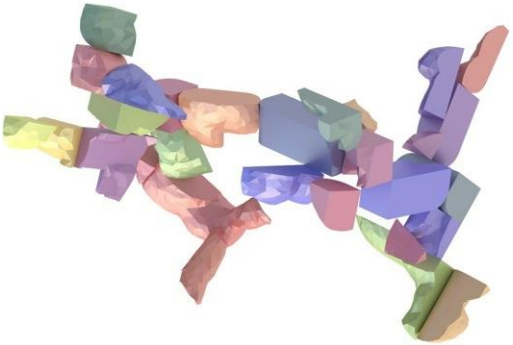
1. How to segment into parts?
   (endless possibilities)

Why is it hard?

1. How to segment into parts?
2. How to fit the parts to two different shapes?
   (complex combinatorial problem)

Why is it hard?

1. How to segment into parts?
2. How to fit the parts to two different shapes?
3. How to guarantee continuous foldability?
   (exponential configuration space)


Sub-Problems

How to segment?

How to fit?

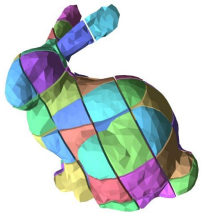
How to fold?

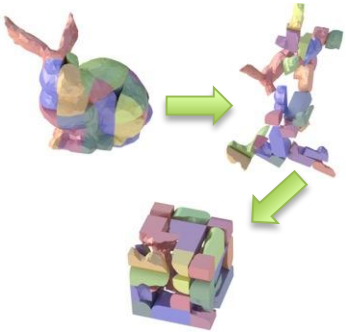
Boxelization Approach

Voxelization

How to fit?

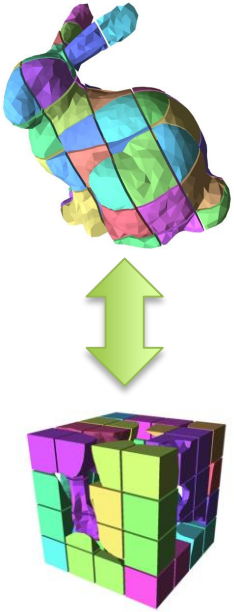
How to fold?



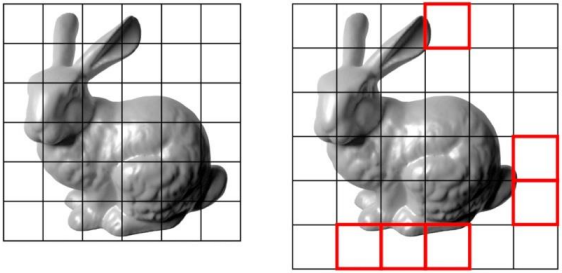


Voxelization

- Reduces the search to a known space
- Fits better to a cube shape



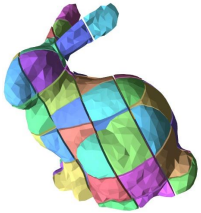
- Try various directions and positioning by rotation, scale and translation to minimize empty voxels:

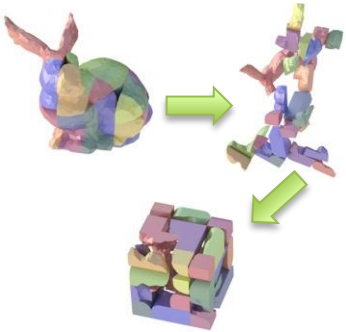


Boxelization Approach

Boxelization Approach

Voxelization → Tree fitting → How to fold?





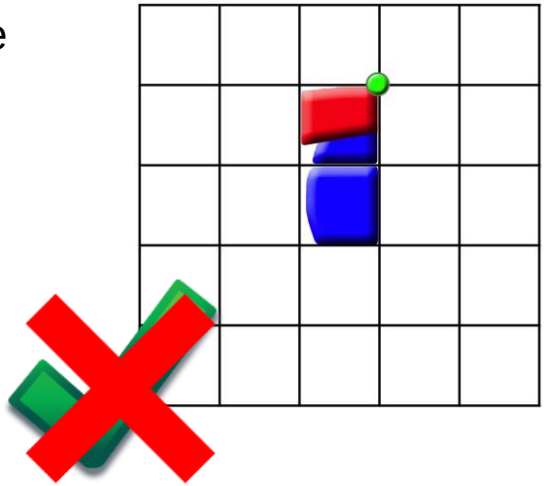
Tree Fitting

- Voxels → Nodes
- Joints → Edges

- Turn the graph into an undirected tree in the two configurations simultaneously

Optimization Objective

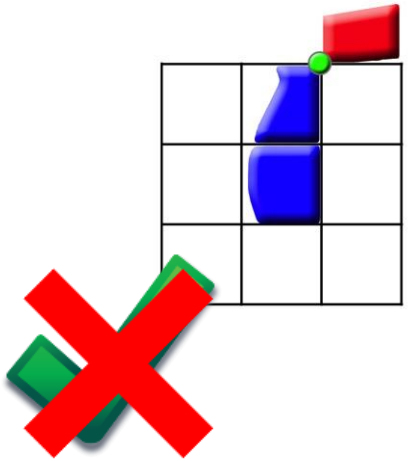
- No two voxels/joints should collide



$$E(\mathbf{joints}) = E_{collision} + E_{template} + E_{surface} + E_{count}$$

Optimization Objective

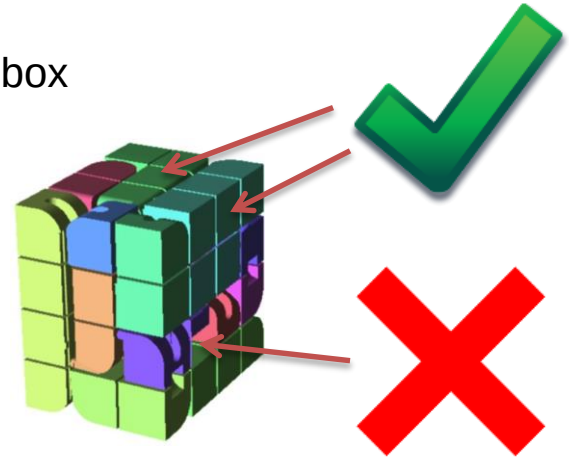
- No two voxels/joints should collide
- Shape must be located inside target box



$$E(\mathbf{joints}) = E_{collision} + E_{template} + E_{surface} + E_{count}$$

Optimization Objective

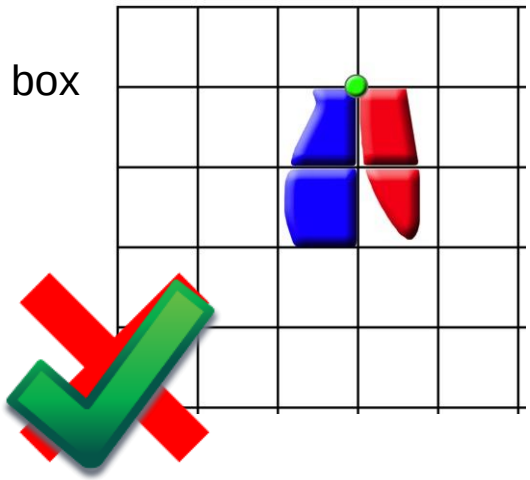
- No two voxels/joints should collide
- Shape must be located inside target box
- Prefer complete outer facets



$$E(\mathbf{joints}) = E_{collision} + E_{template} + E_{surface} + E_{count}$$

Optimization Objective

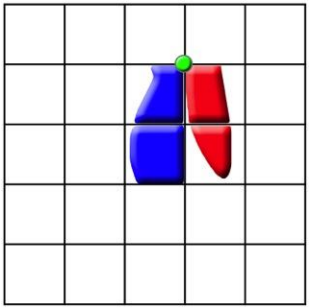
- No two voxels/joints should collide
- Shape must be located inside target box
- Prefer complete outer facets
- Reduce the number of joints



$$E(\mathbf{joints}) = E_{collision} + E_{template} + E_{surface} + E_{count}$$

Random Greedy Search

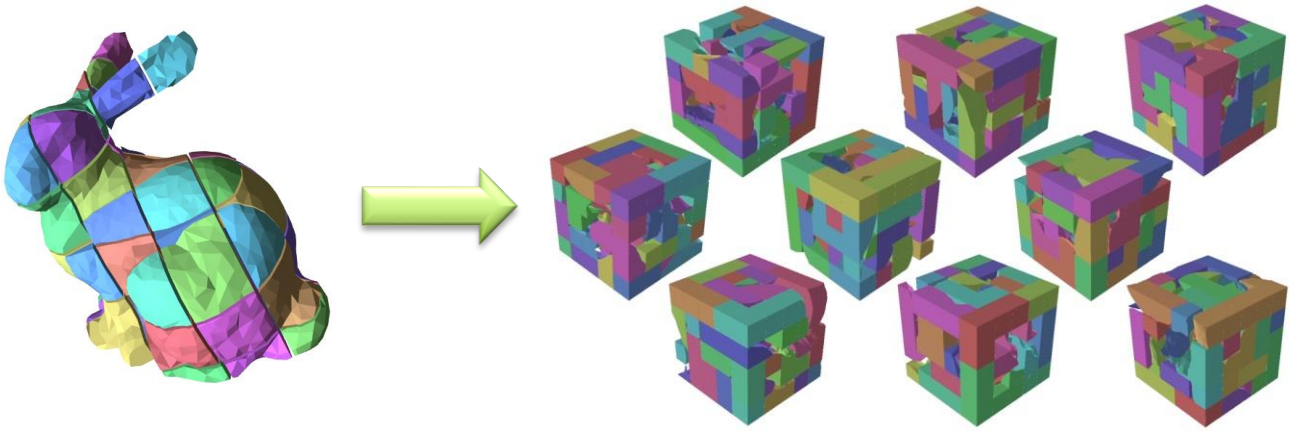
1. Pick a random root voxel
2. Repeat:
   - Add voxels/joints one by one
   - Evaluate



$$E(\mathbf{joints}) = E_{collision} + E_{template} + E_{surface} + E_{count}$$

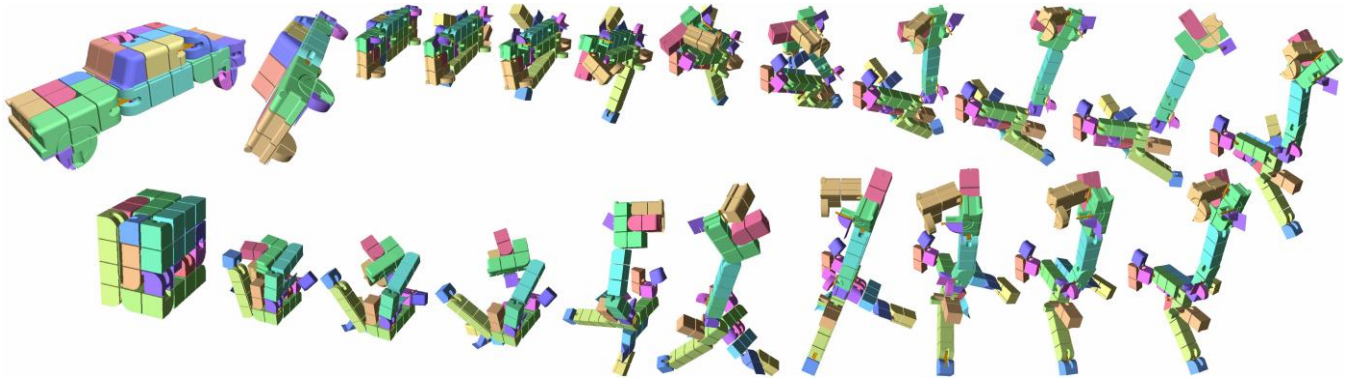
Random Greedy Search

- Returns many results



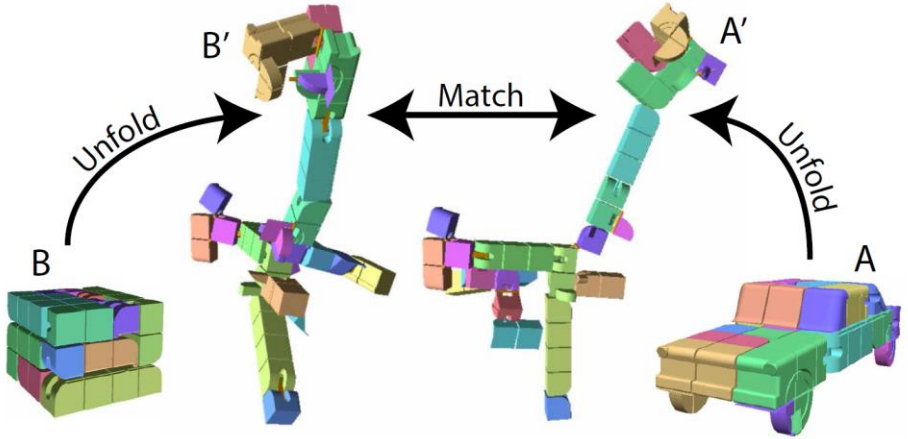
Can we actually fold it?

- Tree fitting guarantees fitting the two static configurations (initial and final)
- What about *during* the folding sequence?



Key Insight

- Unfolding is much simpler than folding



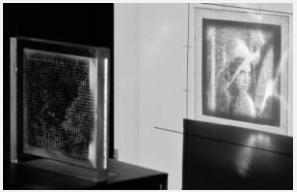
- We you use physics & interaction

Boxelization Approach

Boxelization Approach

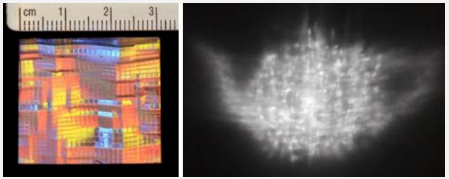
Other Specification Design Objectives

[Finckh *et al.*, 2010]



[Papas *et al.*,2011]

Caustic Images

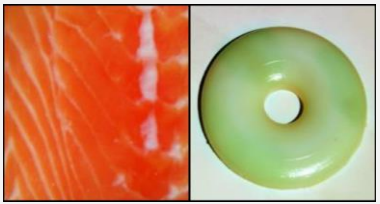


[Weyrich *et al.*, 2009]

Surface reflectance

[Hašan *et al.*,2010]

Subsurface scattering



[Dong *et al.*,2010]

Subsurface reflectance

[Bickel *et al.*,2012]

synthetic skin

Other Specification Design Objectives

[Papas *et al.*, 2012]

Refractive steganography

[Baran *et al.*, 2012]

[Bermano *et al.*, 2012]

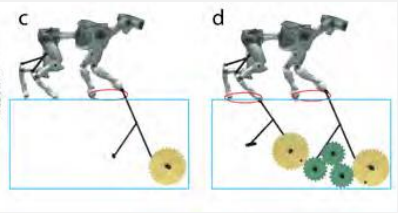
[Mitra and Pauly.,2009]

Shadows

Other Specification Design Objectives

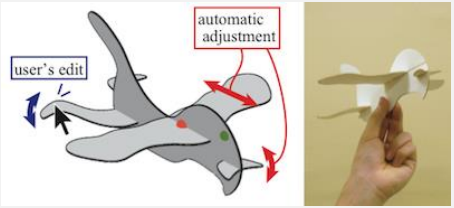
[Bickel *et al.*,2010]

Deformation Behavior



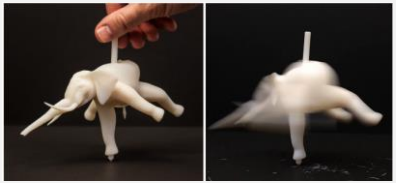
[Coroset *et al.*, 2012]

Dynamics & Motion



[Umetani *et al.*, 2014]

Flight ability

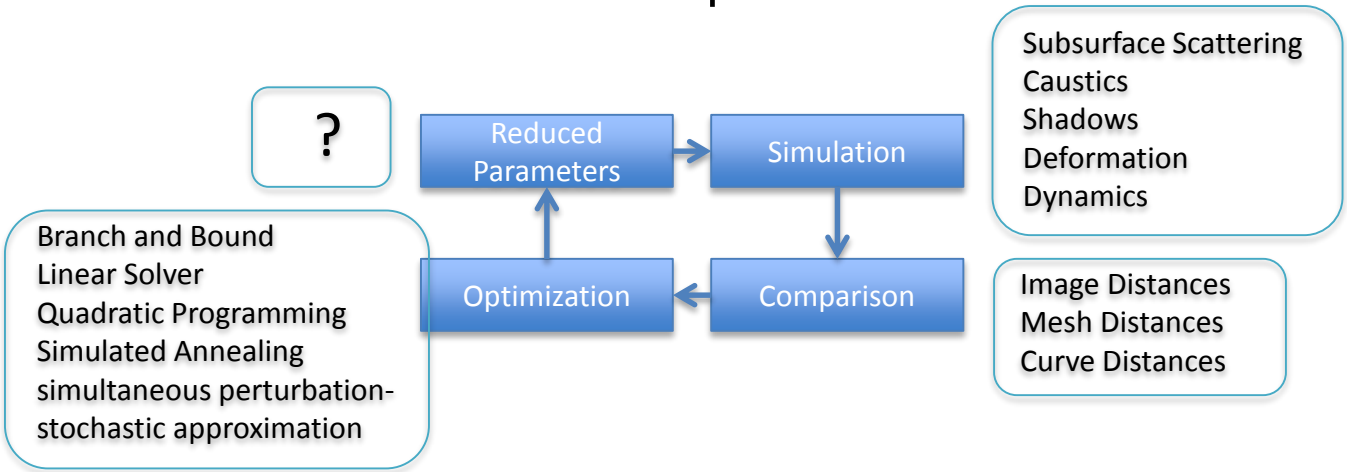


[Baecher *et al.*, 2014]

Spin ability

Key Observations

- Specification design processes use a similar structure
- Small set of common components:

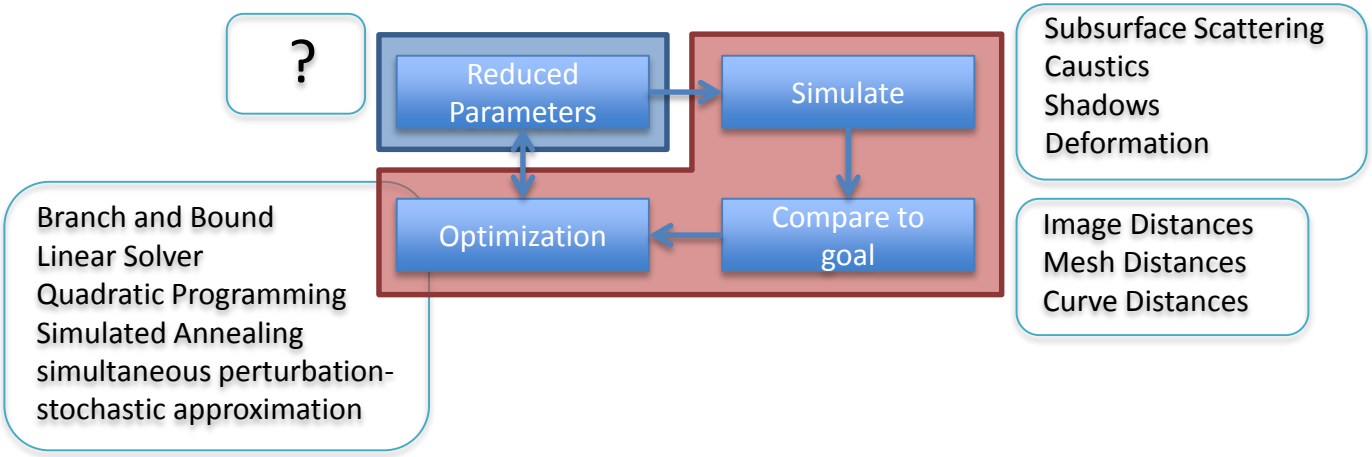


Spec2Fab

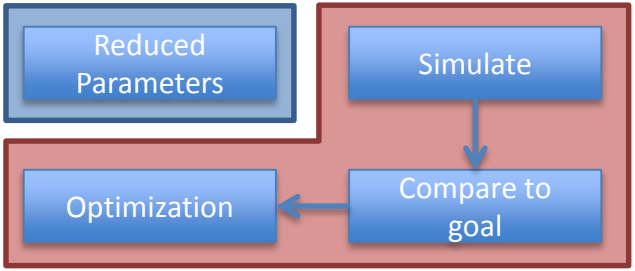
Spec2Fab: A Reducer-Tuner Model for Translating Specifications to 3D Prints
Desai Chen David I.W. Levin Piotr Didyk Pitchaya Sitthi-Amorn Wojciech Matusik
SIGGRAPH 2013

- A unified model for Spec2Fab translation
  - Modular
  - Extensible
  - Geometry Independent
  - Device Independent

Common Components



Common Components



Reducer Tree

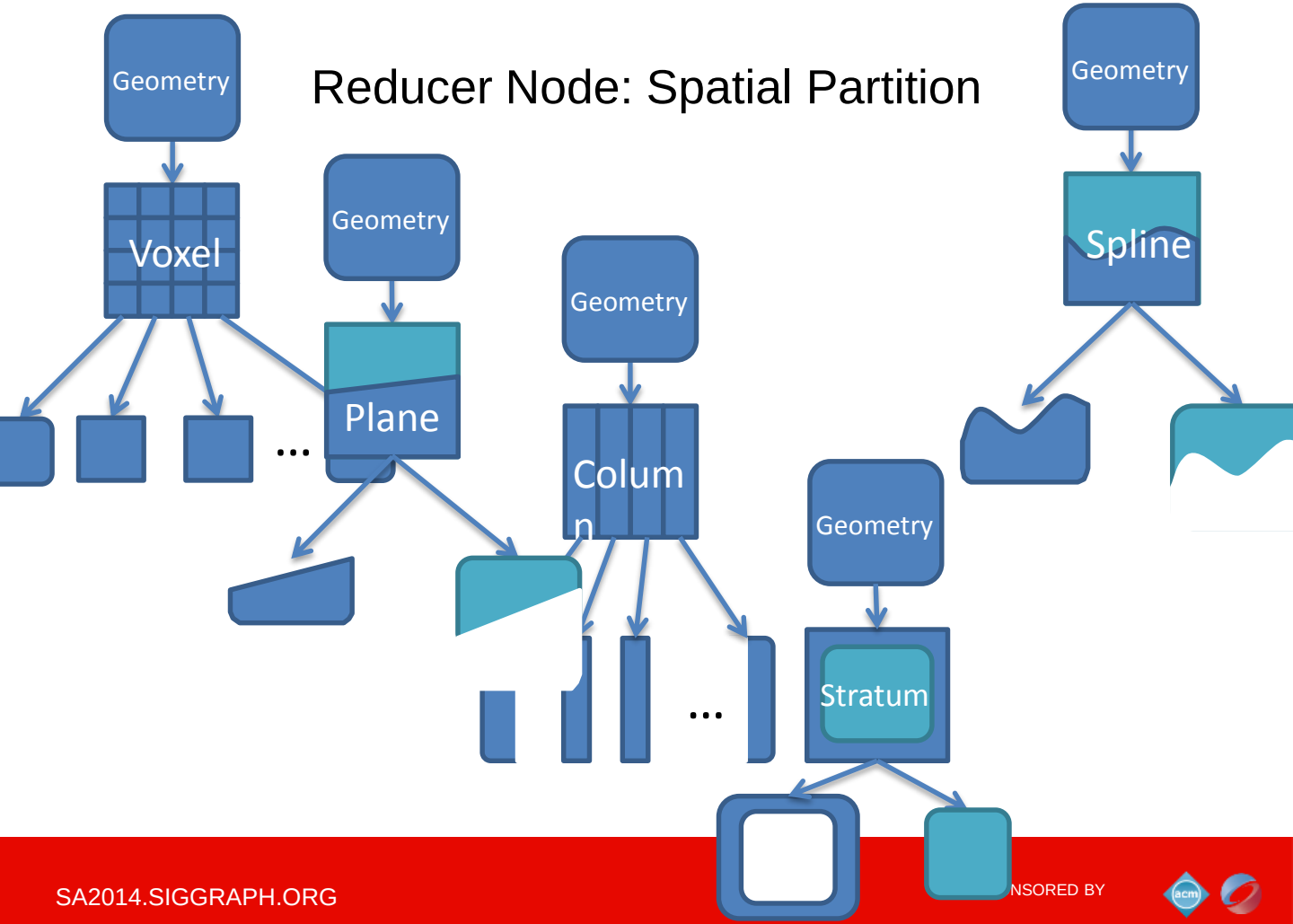
Tuner Network

Reducer Tree

- Internal Nodes
  - Spatial Partition

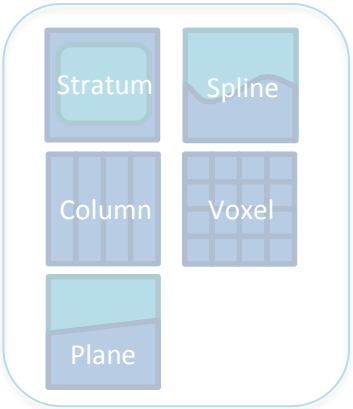
- Leaf Nodes
  - Material Assignment

Reducer Node: Spatial Partition

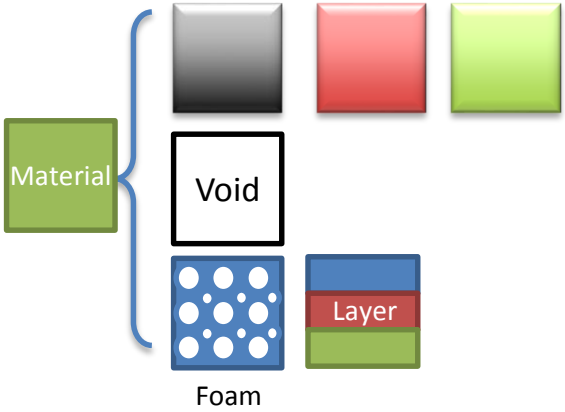


Reducer Node

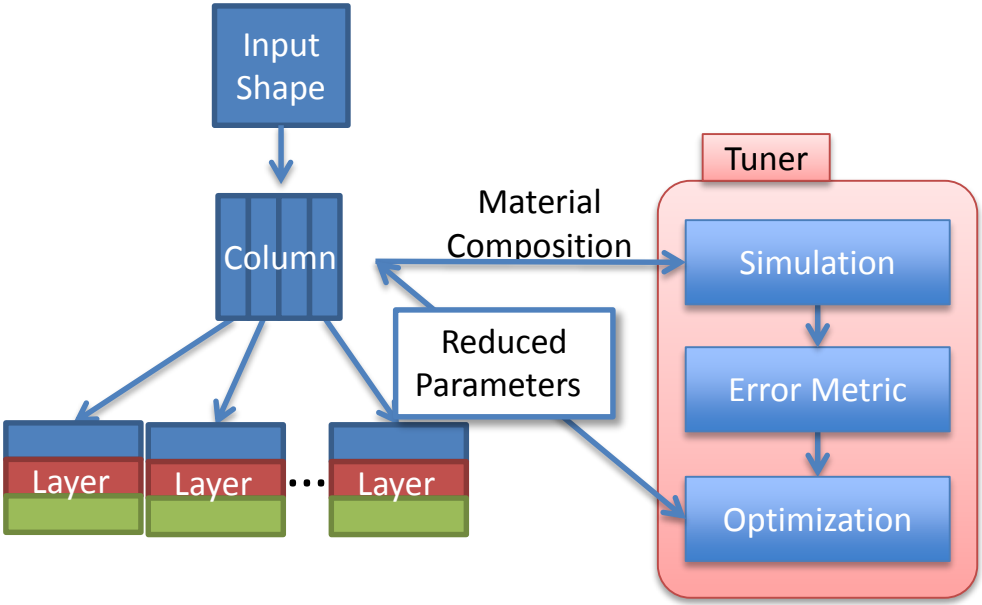
- Internal Nodes
  - Spatial Partition



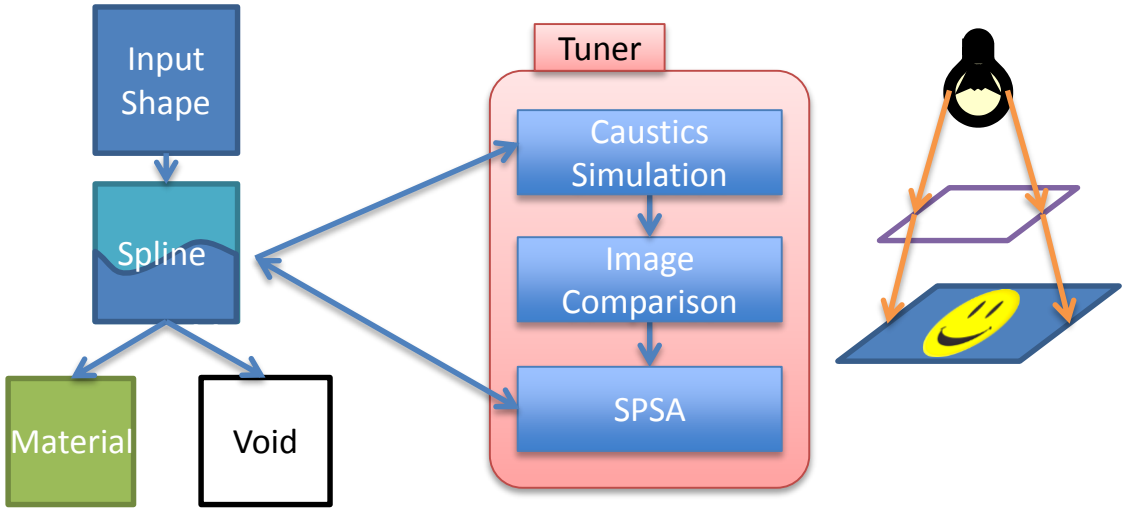
- Leaf Nodes
  - Material Assignment



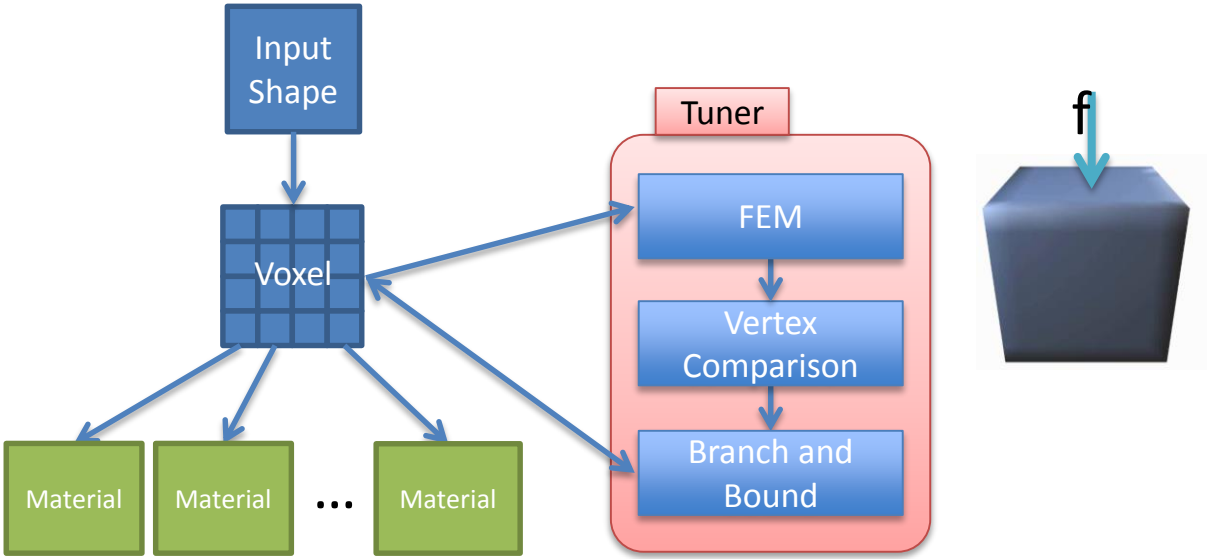
Tuner


Input Shape
Column
Material Composition
Reduced Parameters
Layer
Layer
Layer
Tuner
Simulation
Error Metric
Optimization

Tuner for Caustics Inspired by Finckh *et al.*, 2010

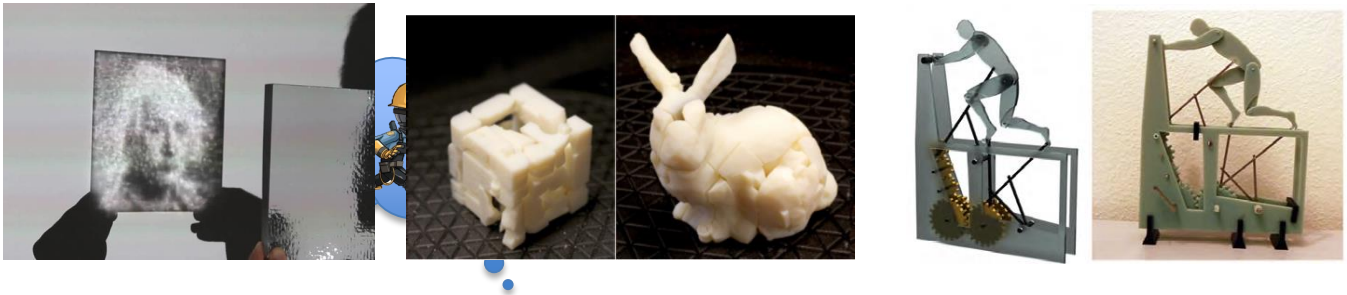


Tuner for Deformation Inspired by Bickel *et al.*, 2010



Specification vs. Interactive Modeling

- Most of these involve simulation and high domain expert knowledge in Caustics, Physics, Geometry etc.



Specification vs. Interactive Modeling

- Most of these involve simulation and high domain expert knowledge in Caustics, Physics, Geometry etc.
- How can we design an interactive tool (for the layman) that will still involve expert knowledge?

?

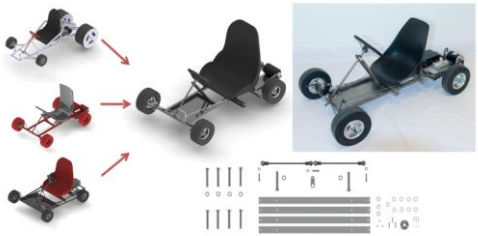
“Intelligent” Interactive Tools for Design

1. 3Sweep: modeling from images

(copying shape and appearance)

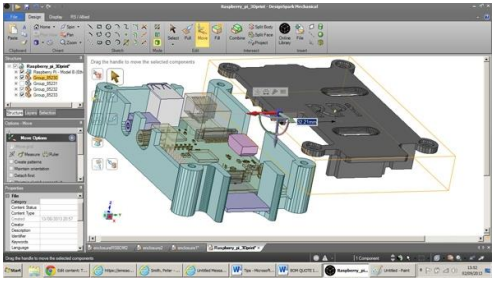
2. Fab-by-Example: modeling from examples

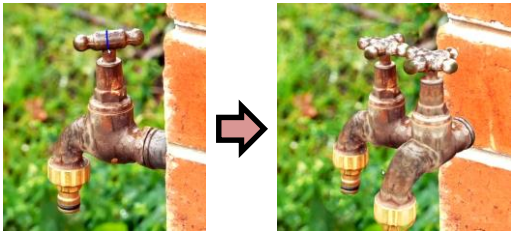
(keeping things fabricable)



3Sweep Motivations

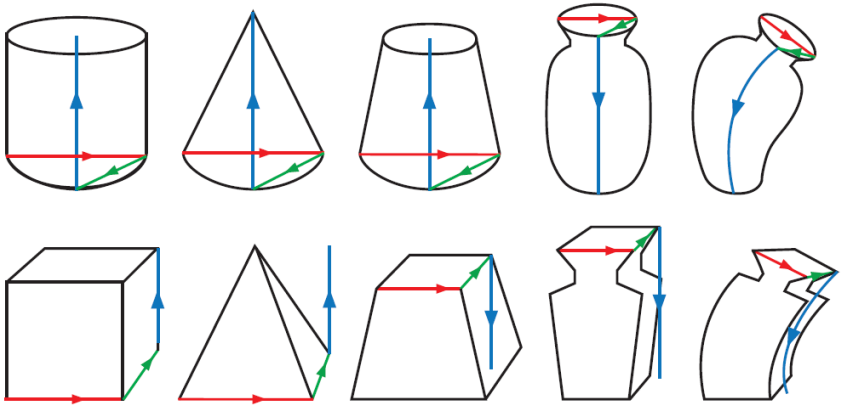
- Modeling is difficult and tedious for both designers and unskilled users
- Use Images?
- Modeling from an image is challenging for automatic computer algorithms





3-Sweep: Image Based Object Modeling

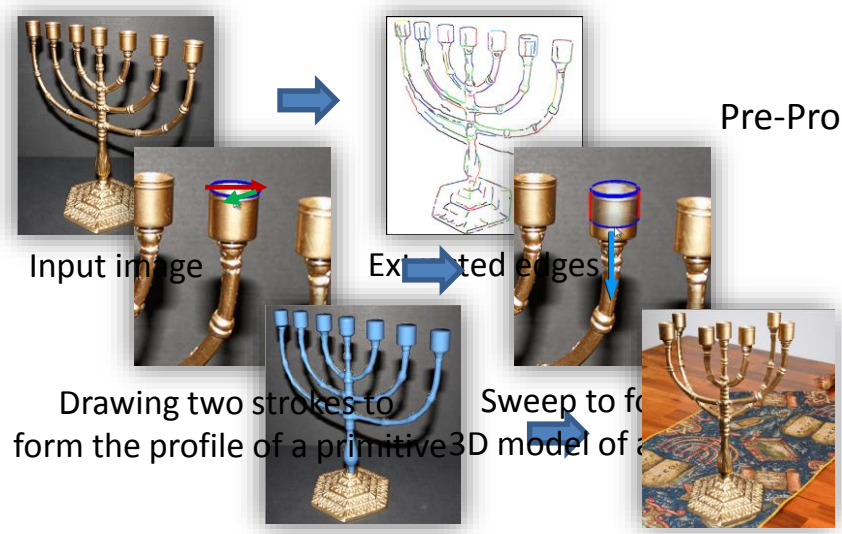
- A simple intuitive gesture to define 3D primitives using 3 mouse sweeps:



Modeling One Primitive

3-Sweep: Extracting Editable Objects from a Single Photo
Tao Chen, Zhe Zhu, Ariel Shamir, Shi-Min Hu, Daniel Cohen-Or
ACM Transactions on Graphics, 32(6), (SIGGRAPH Asia) 2013

3Sweep Overview



Pre-Processing

3-Sweep:
Modeling One Primitive

Optimization:
Modeling Composite

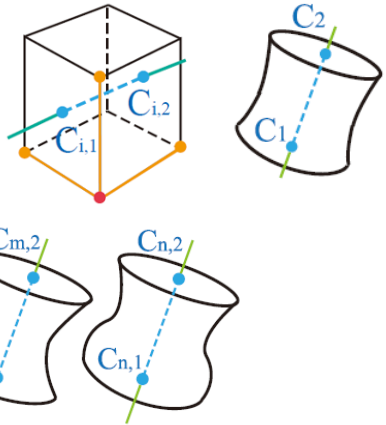
Behind the Scene: Outline Snapping Rules

- Prefer the outline that is parallel to the 3rd stroke
- Stick to the same outline, jump to the one with minimal angle change if necess
- Use symm sing outline
- Fit to unifo and use diamete can only be scaled u

Behind the Scene: Geo-Semantic Constraints

Expert Knowledge: Geo-Semantic Constraints

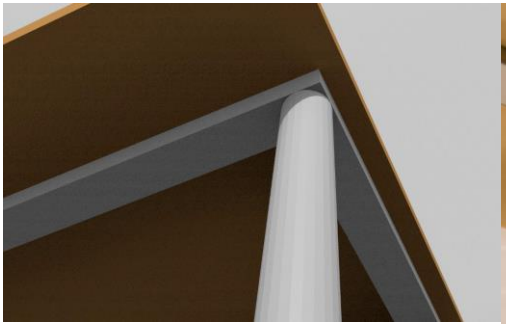
- Defined in terms of major axes of the primitives.
- Support six constraint types:
  - parallelism
  - orthogonality
  - collinear anchors
  - overlapping anchors
  - coplanar anchors
  - coplanar axes

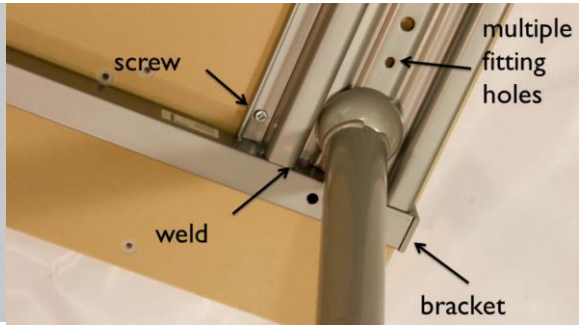


$$(\mathbf{C}_{m,1} - \mathbf{C}_{m,2}) \times (\mathbf{C}_{n,1} - \mathbf{C}_{n,2}) = 0$$

Results in Editable Objects

Virtual vs. Real

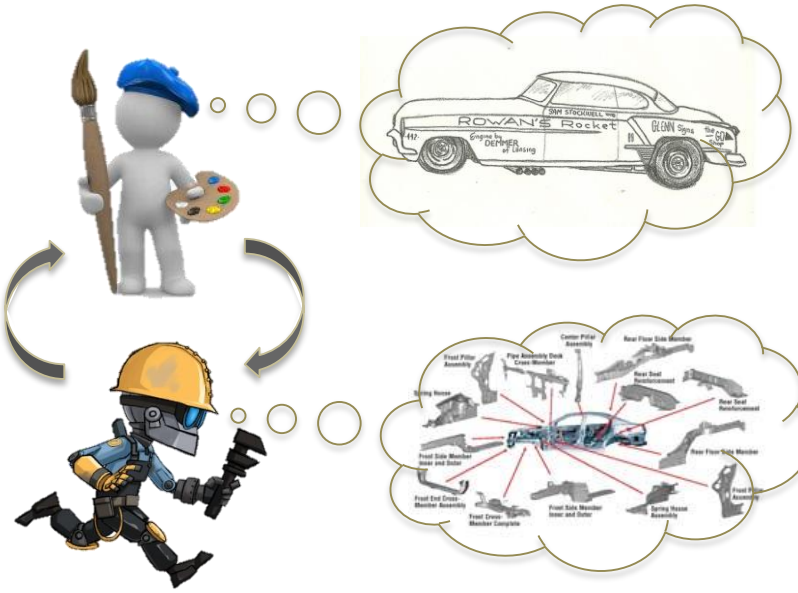




Many Parts (~300)

The Design for Fabrication Process

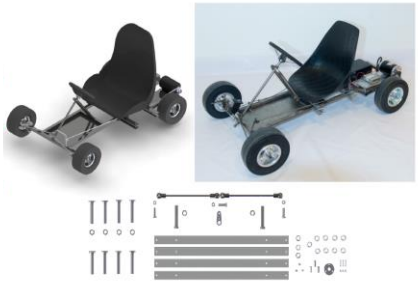
Designer

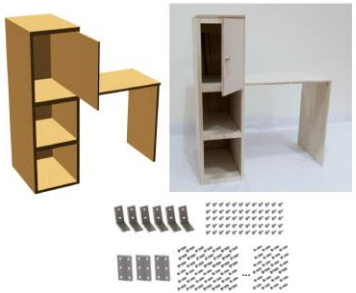


Engineer

Fabrication by Example Motivation

- Designing objects that can be really fabricated:
  - Requires many small details
  - Includes how to connect parts
  - Involves materials, physics…

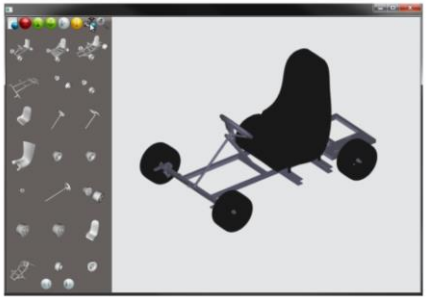




Key Idea: Database of Example

hierarchical parametric objects (templates)

Modeling System



Design and Fabrication by Example
Adriana Schulz, Ariel Shamir, David Levin, Pitchaya Sitthi-Amorn, Wojciech Matusik
ACM Transactions on Graphics, Volume 33(4), (SIGGRAPH) 2014

Approach

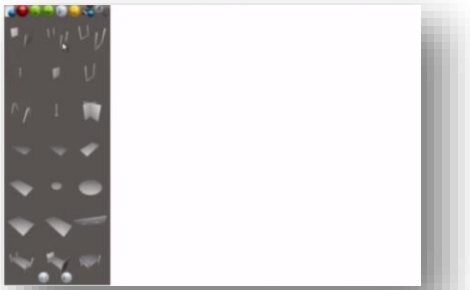
Input

Approach

Input

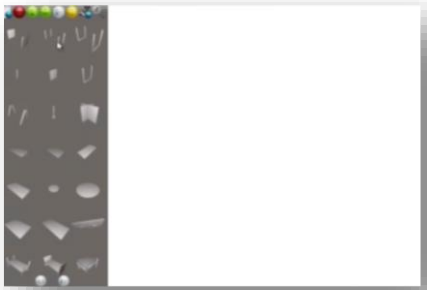
Expert Data

Approach



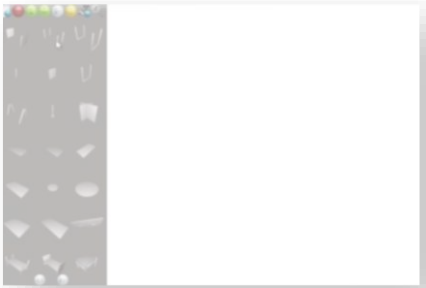
Input Expert Data Composition Tool

Approach



Input Expert Data Composition Tool Fabrication

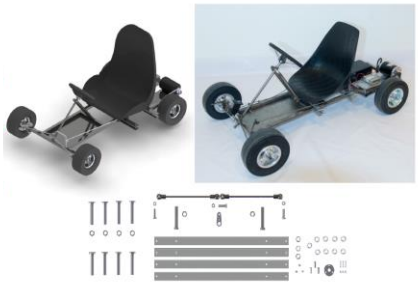
Approach

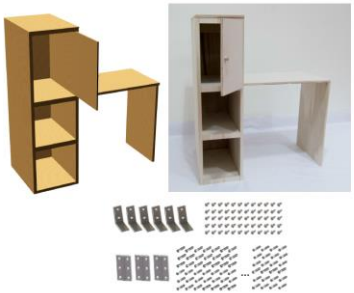


Input Expert Data Composition Tool Fabrication

Flexible Models: Parameters & Constraints

Part Based Modeling





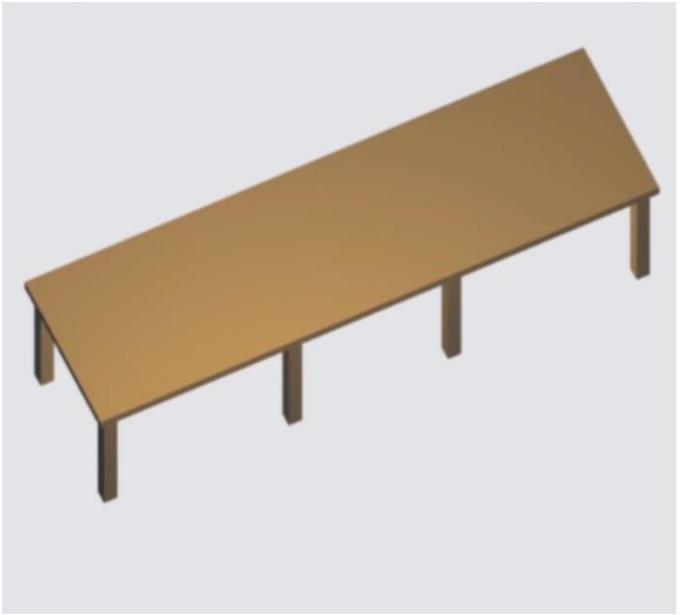
Snapping

- User interaction

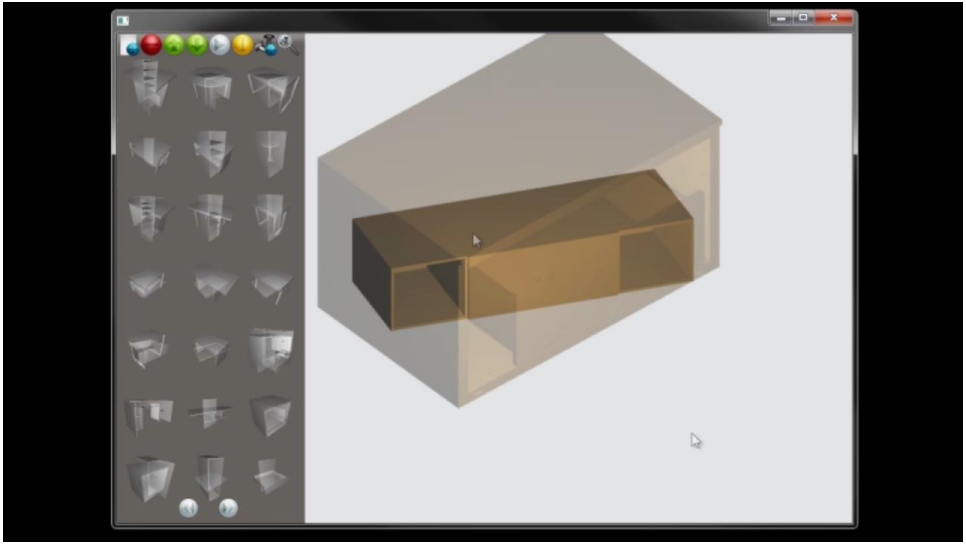
- Preserve alignment constraints

Quadratic Program

$$\min \|q - q_{\text{current}}\|^2 \quad \text{s.t.} \quad \begin{matrix} Aq = B \\ q \in \mathcal{A} \end{matrix}$$

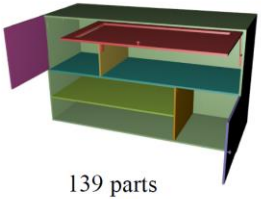


Connectors

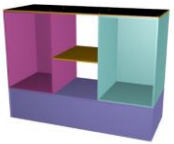


Examples

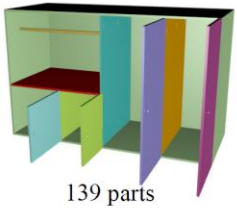
179 parts
(163 connectors)



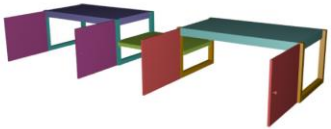
139 parts
(122 connectors)



217 parts
(197 connectors)

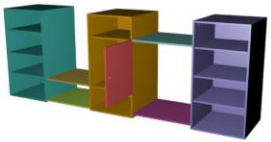


139 parts
(121 connectors)



147 parts
(124 connectors)

156 parts
(140 connectors)



128 parts
(99 connectors)

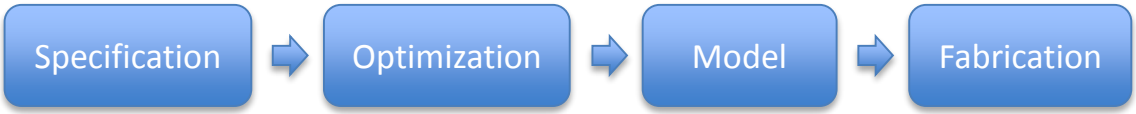
101 parts
(90 connectors)

Examples

Example Modeling Session

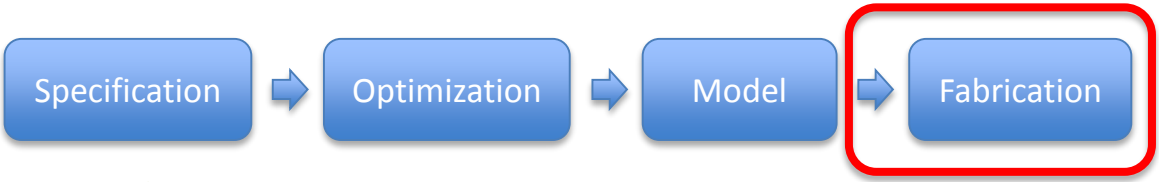
Two Approaches

1. Specification based design:

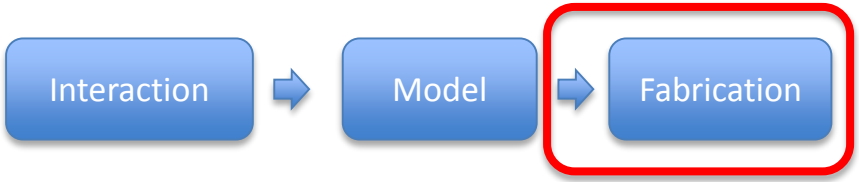


2. Interactive modeling:

Key for Both:



Include expert domain knowledge



Main Papers

Chopper: Partitioning Models into 3D-Printable Parts
Linjie Luo, Ilya Baran, Szymon Rusinkiewicz, Wojciech Matusik
ACM Transactions on Graphics, 31(6), (SIGGRAPH Asia), 2012

Boxelization: Folding 3D Objects into Boxes
Yahan Zhou, Shinjiro Sueda, Wojciech Matusik, Ariel Shamir
ACM Transactions on Graphics, 33(4), (SIGGRAPH), 2014

Spec2Fab: a reducer-tuner model for translating specifications to 3D prints
D. Chen, D. I. W. Levin, P. Didyk, P. Sitthi-amorn, W. Matusik
ACM Transactions on Graphics, 32(4), (SIGGRAPH) 2013

3-Sweep: Extracting Editable Objects from a Single Photo
Tao Chen, Zhe Zhu, Ariel Shamir, Shi-Min Hu, Daniel Cohen-Or
ACM Transactions on Graphics, 32(6), (SIGGRAPH Asia) 2013

Design and Fabrication by Example
Adriana Schulz, Ariel Shamir, David Levin, Pitchaya Sitthi-Amorn, Wojciech Matusik
ACM Transactions on Graphics, Volume 33(4), (SIGGRAPH) 2014

Thank you!

SIGGRAPH ASIA 2014 SHENZHEN

Comment and feedback via course webpage