Selective Padding for Polycube-Based Hexahedral Meshing

Pierre Alliez

Inria Sophia Antipolis – Méditerranée

Titane project-team: https://team.inria.fr/titane/

pierre.alliez@inria.fr

and…Toward (pure-)Hexahedral Meshing

Pierre Alliez

Inria Sophia Antipolis – Méditerranée

Titane project-team: https://team.inria.fr/titane/

pierre.alliez@inria.fr

Join Work

COMPUTER GRAPHICS *forum*

Volume 0 (1981), Number 0 pp. 1–12

Selective Padding for Polycube-based Hexahedral Meshing

G. Cherchi[1], P. Alliez[2], R. Scateni[1], M. Lyon[3] and D. Bommes[4]

[1]University of Cagliari, Italy.
[2]Inria Sophia Antipolis - Méditerranée and Université Côte d'Azur, France.
[3]RWTH Aachen University, Germany.
[4]University of Bern, Switzerland.

(a) (b) (c) (d)

Figure 1: *Overview. Our pipeline takes as input a model and its polycube mapping (a); we compute the relative hex-mesh and locate the surface areas in need of padding analyzing the mapping quality (b); we set and solve a binary problem to find a set of facets to extrude in order to create a selective padding layer (c); we compute and analyze the mapping with the new hex-mesh structure (d).*

Abstract

Hexahedral meshes generated from polycube mapping often exhibit a low number of singularities but also poor quality elements located near the surface. It is thus necessary to improve the overall mesh quality, in terms of the minimum Scaled Jacobian (MSJ) or average Scaled Jacobian (ASJ). Improving the quality may be obtained via global padding (or pillowing), which pushes the singularities inside by adding an extra layer of hexahedra on the entire domain boundary. Such a global padding operation suffers from a large increase of complexity, with unnecessary hexahedra added. In addition, the quality of elements near the boundary may decrease. We propose a novel optimization method which inserts sheets of hexahedra so as to perform selective padding, where it is most needed for improving the mesh quality. A sheet can pad part of the domain boundary, traverse the domain and form singularities. Our global formulation, based on solving a binary problem, enables us to control the balance between quality improvement, increase of complexity and number of singularities. We show in a series of experiments that our approach increases the MSJ value and preserves (or even improves) the ASJ, while adding fewer hexahedra than global padding.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

Volumetric meshes are an ideal tool for physically-based simulations, especially in the fields of mechanical simulation and computational fluid dynamics. While tetrahedral meshes are very simple to obtain and flexible in adapting to the computational domain, hexahedral meshes have better numerical properties and lower complexity. One straightforward way to generate a hexahedral mesh on a given domain passes through polycube mapping.

Polycubes are three-dimensional domains formed by connecting axis-aligned cuboids face to face. The simplicity of their structure explains their popularity for several computer graphics problems such as quadrilateral and hexahedral meshing (e.g., [HXH10,GSZ11]), efficient texture mapping (e.g., [THCM04,CL*10]), morphing (e.g., [FJFS05]), spline fitting (e.g., [WHL*07]), etc.

When the volumetric domains to partition in hexahedra are regular enough (e.g., with limited range of levels of details like mechanical and CAD models), polycubes are an ideal tool for generating hexahedral meshes, by computing volumetric mappings with their corresponding domains. However, the quality of the hexahedra directly depends on the mapping distortion. While the distortion is

Gianmarco Cherchi
Riccardo Scateni
University of Cagliari

Max Lyon
David Bommes
Aachen University

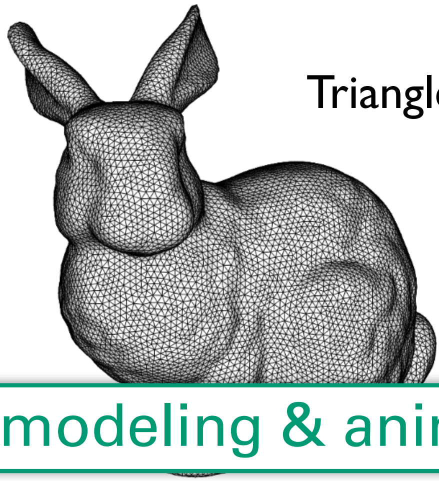
Outline

- Polycubes & Hex Meshing
- Padding
  - Global
  - Selective
- Toward Pure-Hex Meshing
- Summary & Outlook

Meshes

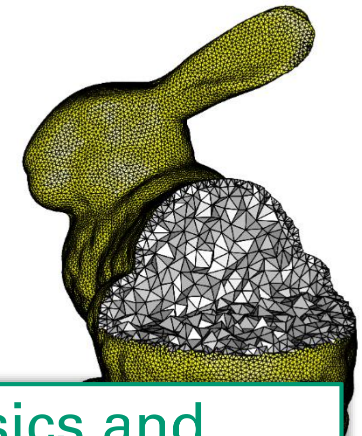
Surface meshes

Volume meshes



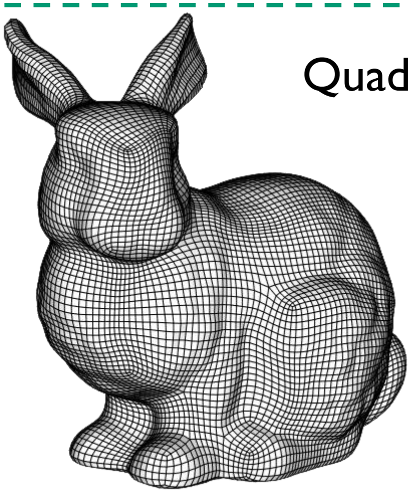
Triangle mesh

Tetrahedral mesh



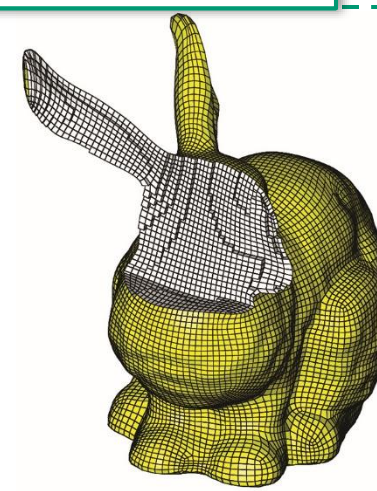
modeling & animation

physics and material simulation



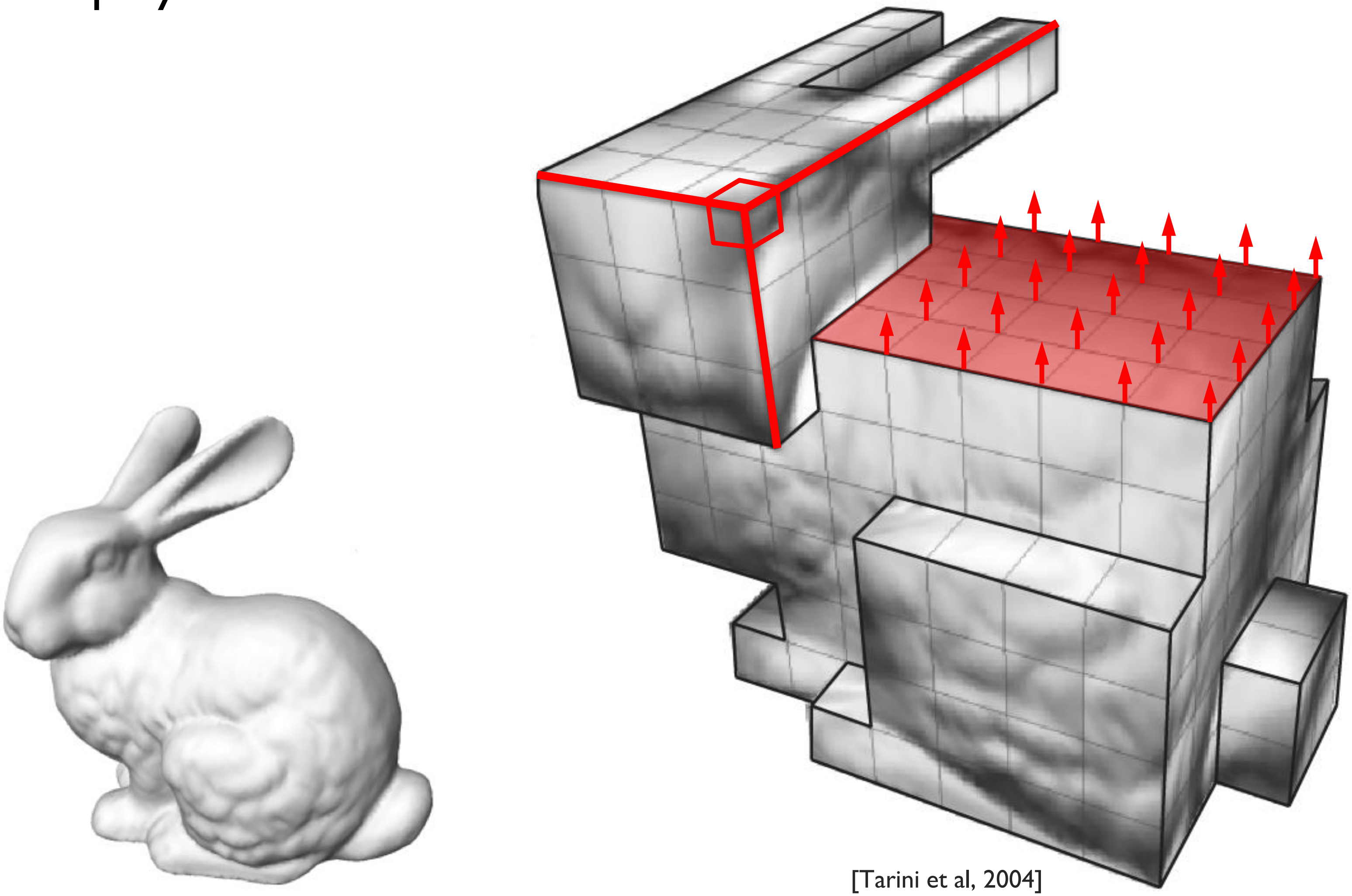
Quadrilateral mesh

Hexahedral mesh



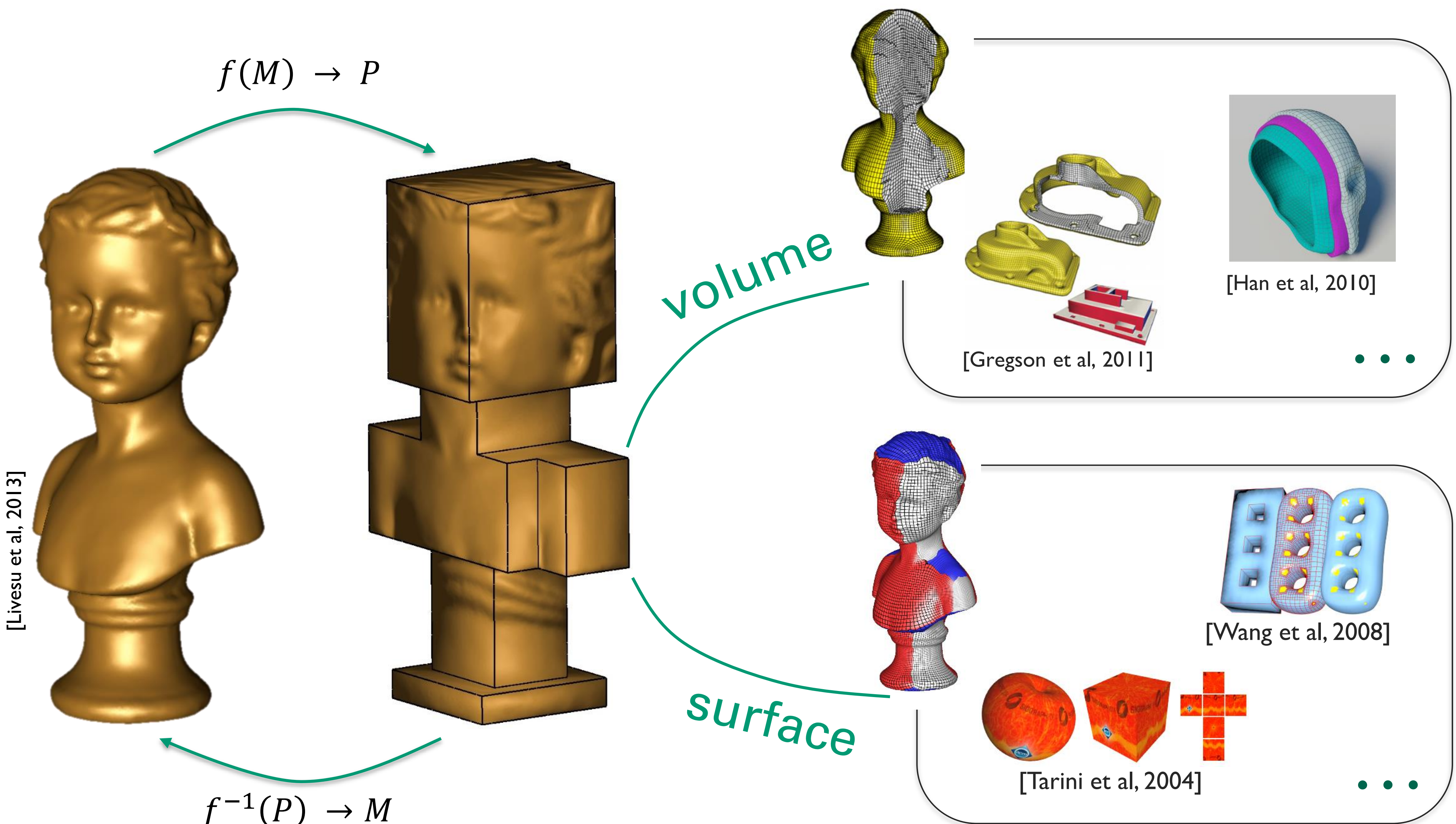
Polycube?

Orthogonal polyhedron of a 3D model



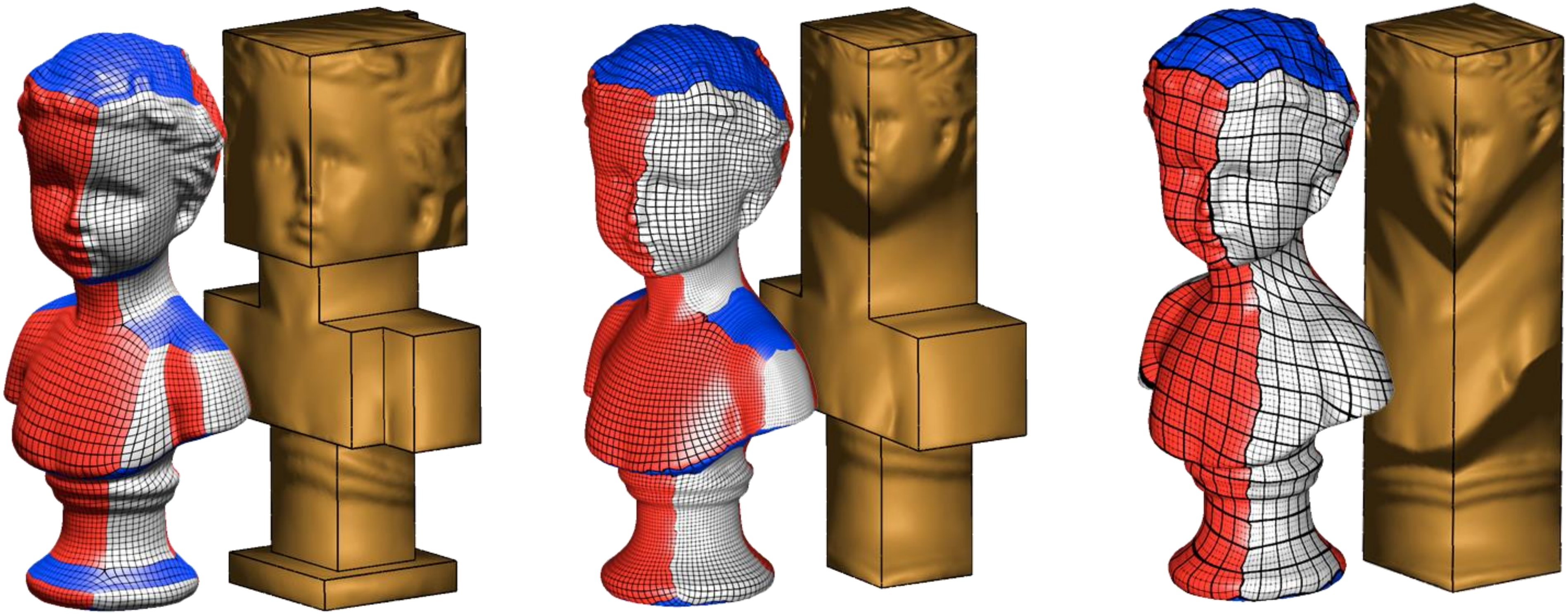
[Tarini et al, 2004]

Polycubes



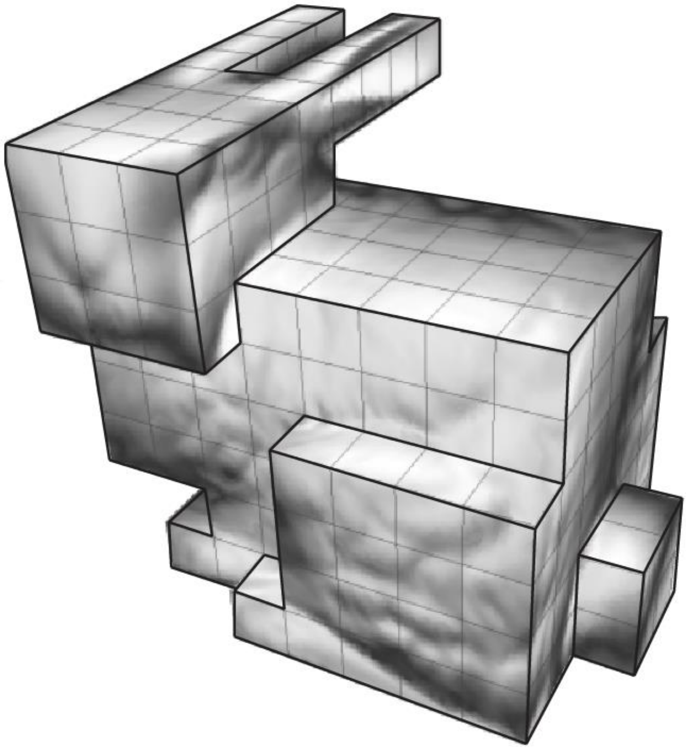
$f(M) \rightarrow P$
$f^{-1}(P) \rightarrow M$
[Livesu et al, 2013]
volume
surface
[Gregson et al, 2011]
[Han et al, 2010]
[Wang et al, 2008]
[Tarini et al, 2004]

Compactness vs Fidelity

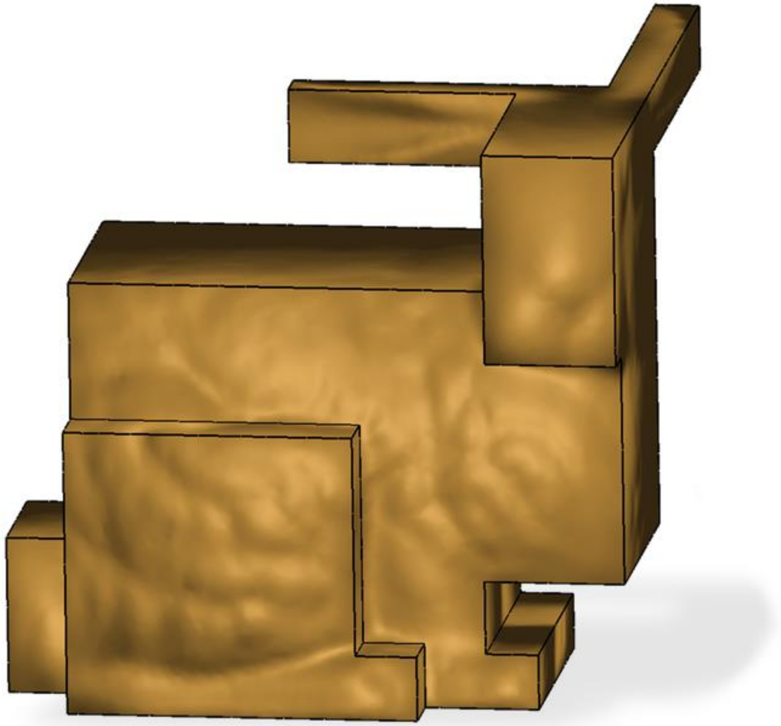


Low……………………………………………………………..large mapping distortion

The Bunny Ear Dilemma…

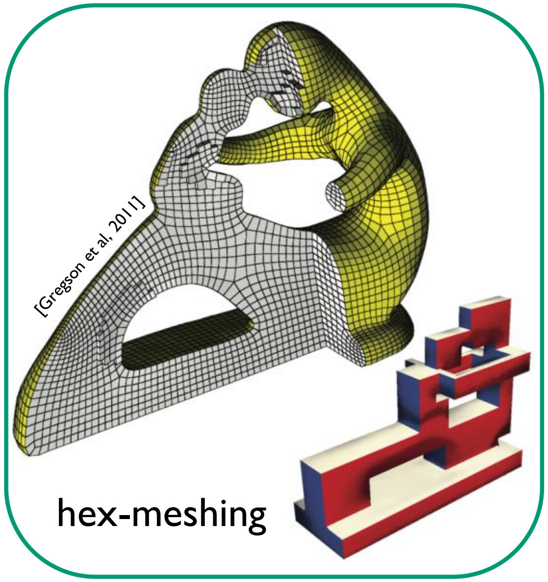


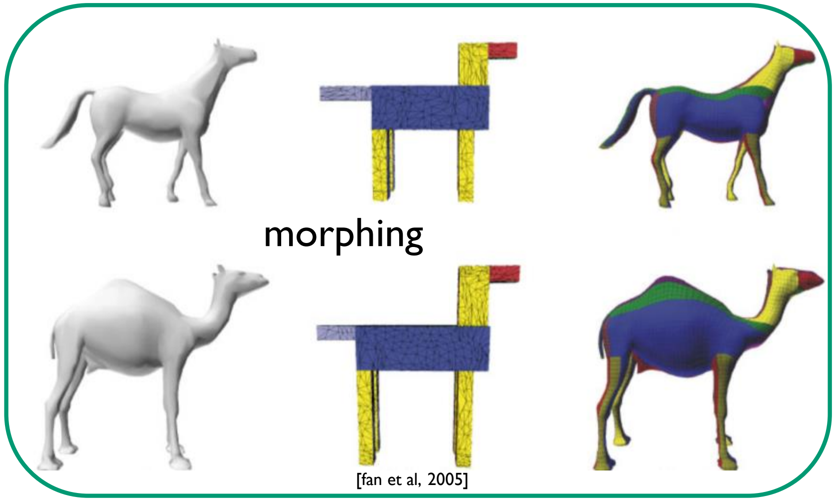
[Tarini et al. 2004]

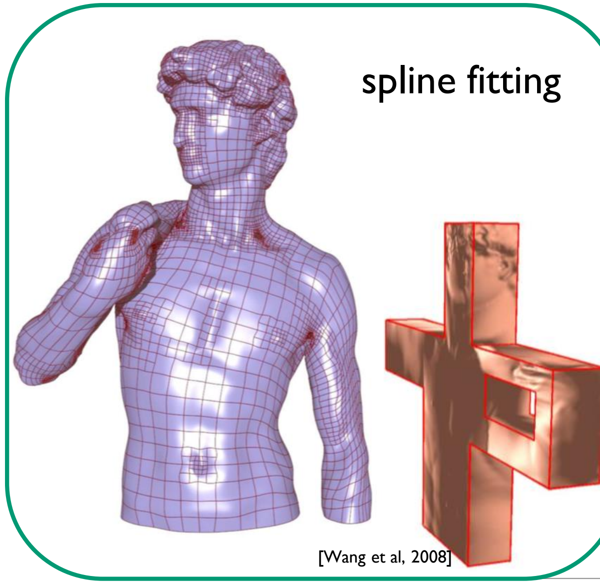


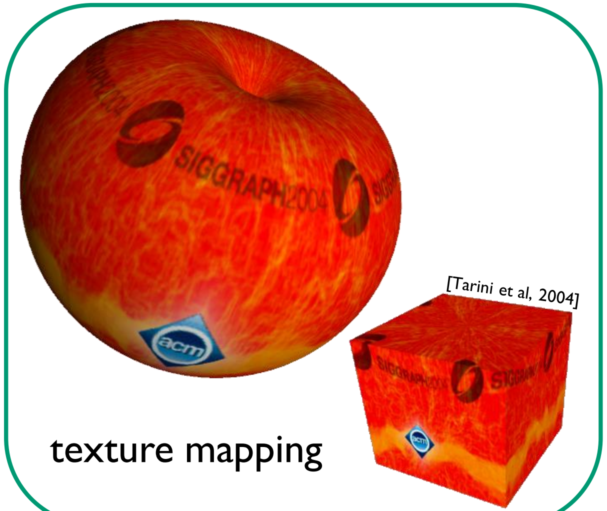
[Livesu et al. 2013]

Why polycubes?

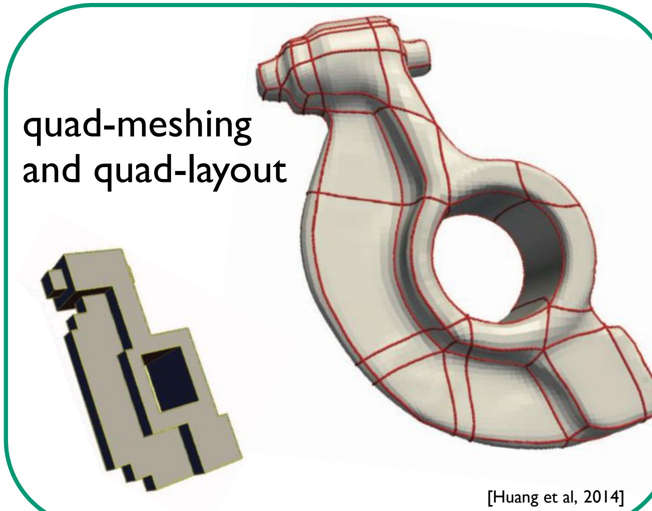




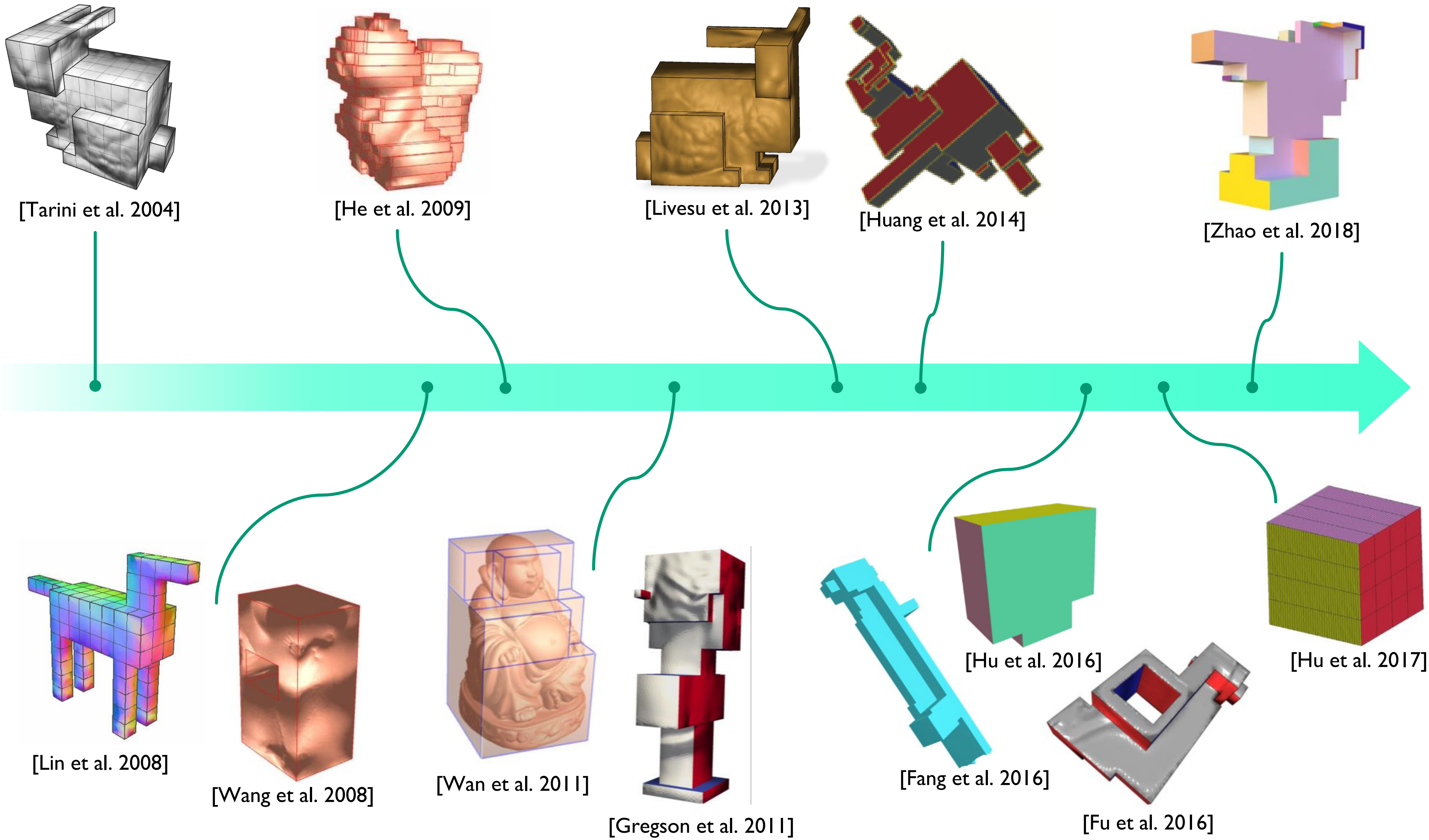




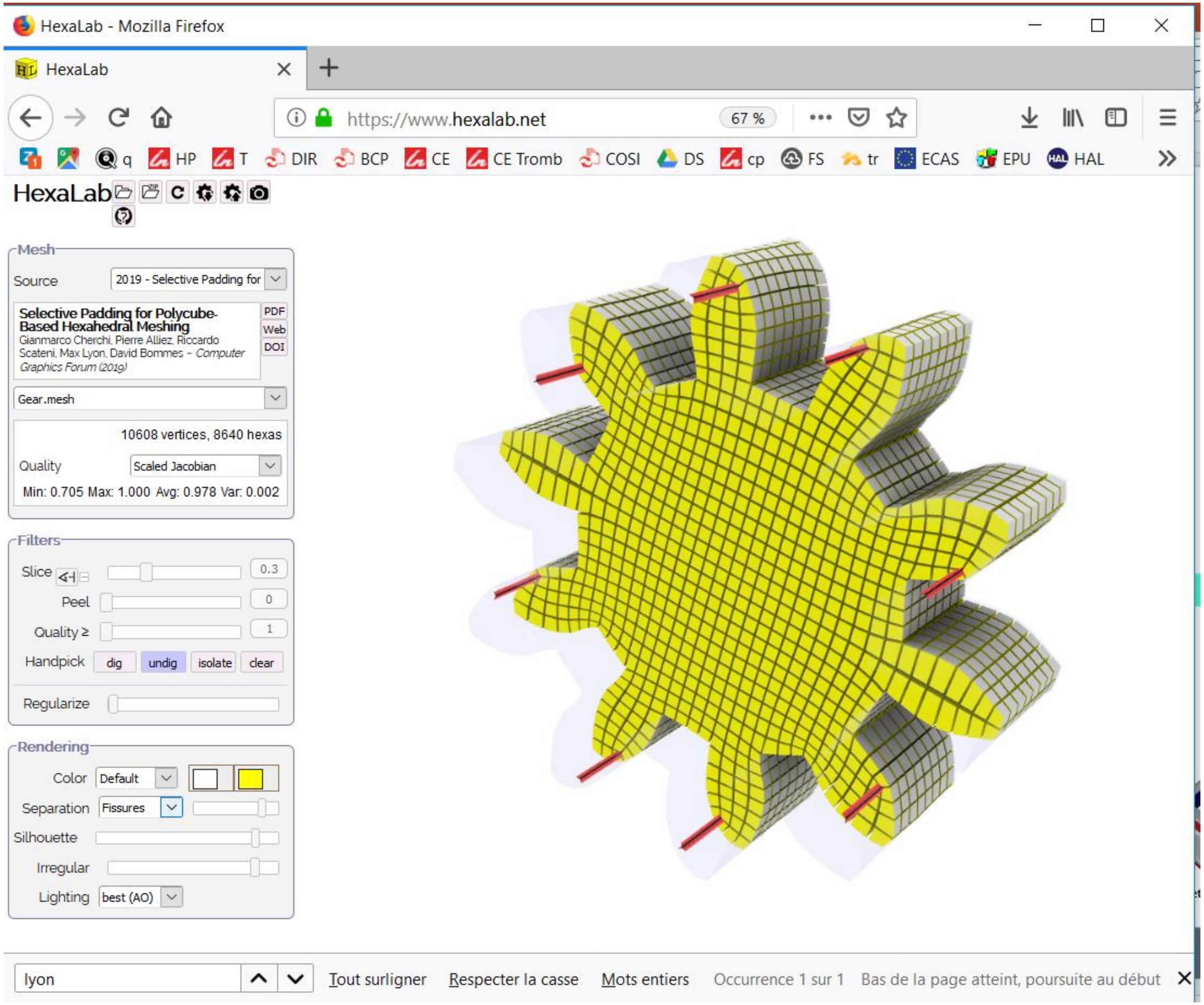
and others…



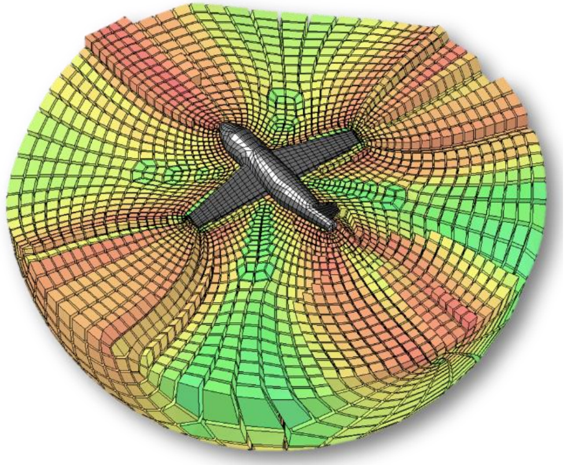
Polycubes – State of the art

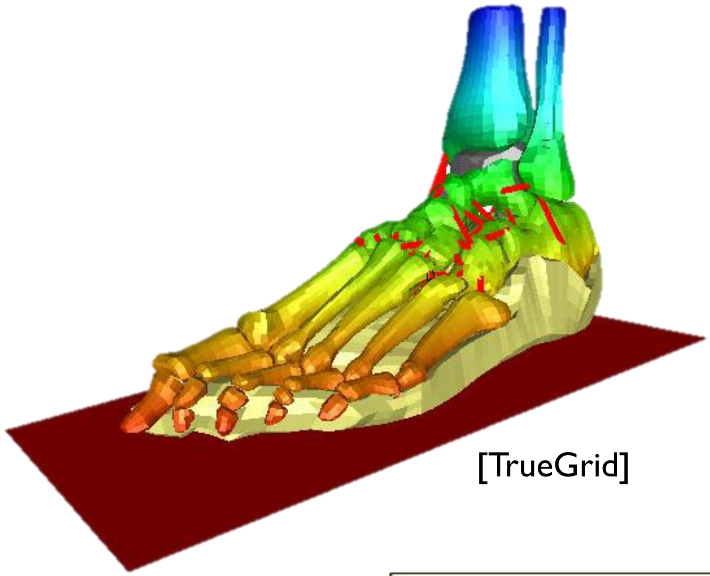


https://www.hexalab.net/

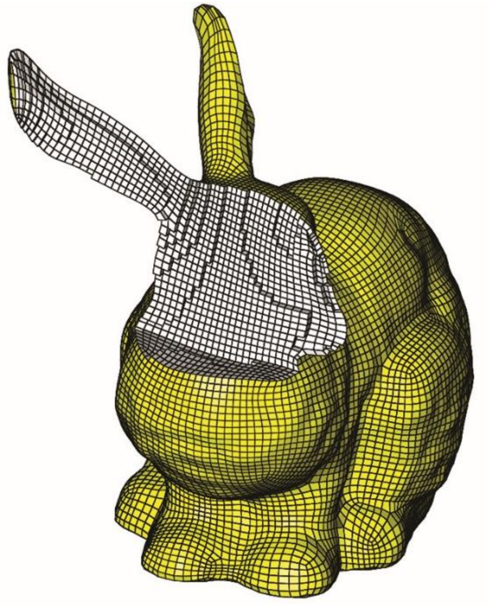


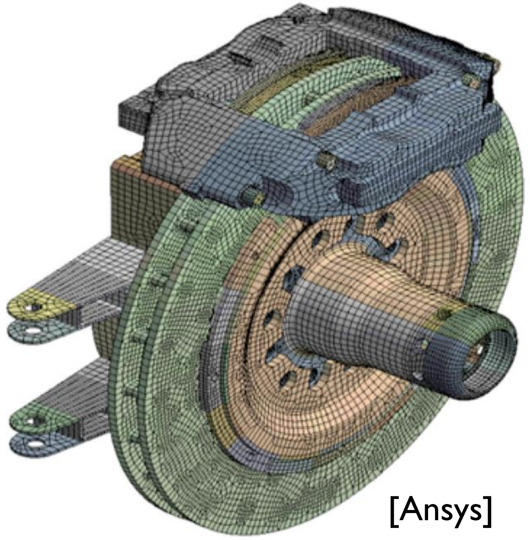
Hexahedral Meshes



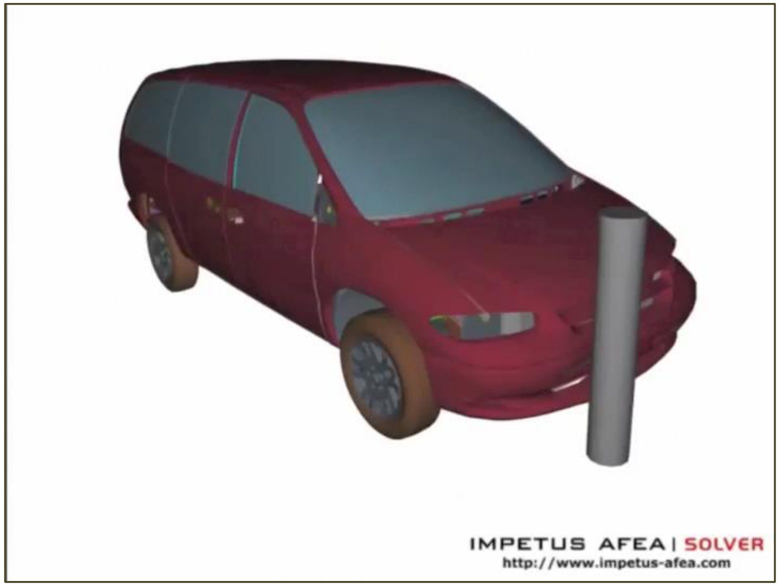


[TrueGrid]



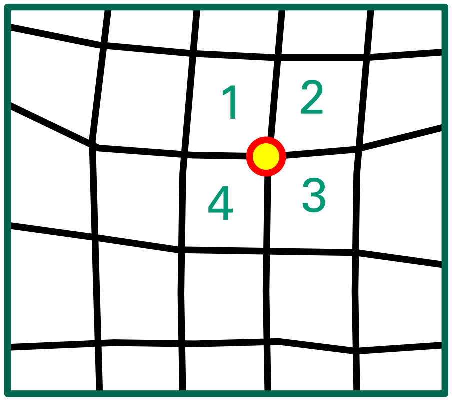


[Ansys]

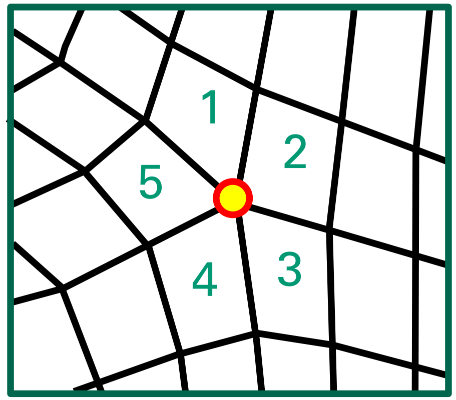


Mesh Singularities

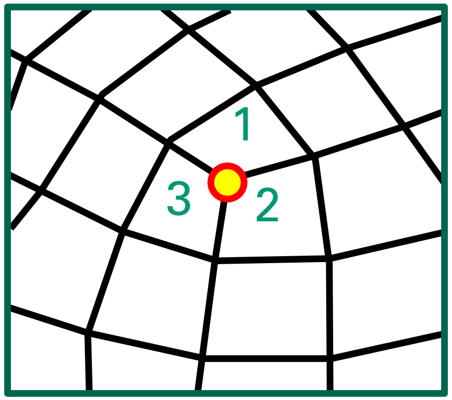
Quad-mesh example



regular vertex (val. 4)

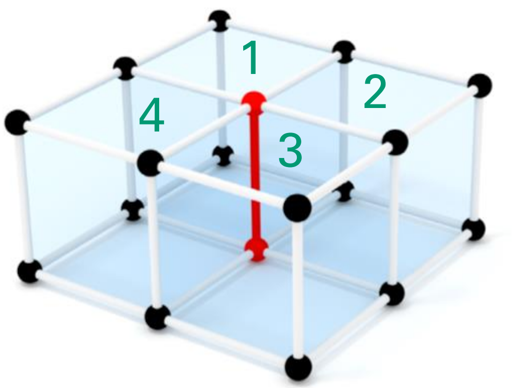


singular vertex (val. 5)

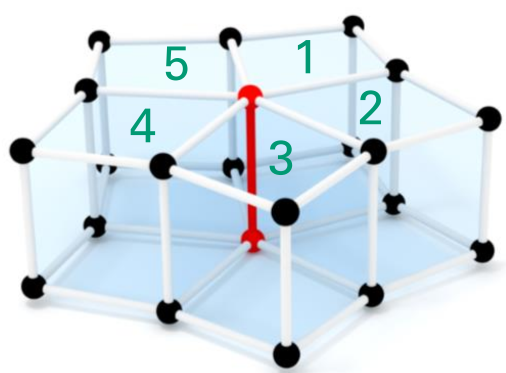


singular vertex (val. 3)

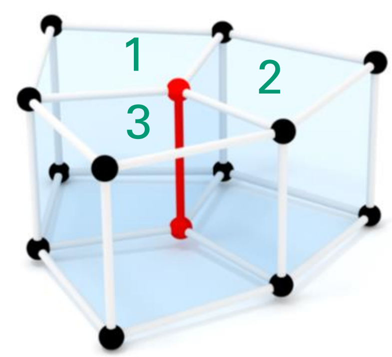
[Liu et al, 2018]



regular edge (val. 4)



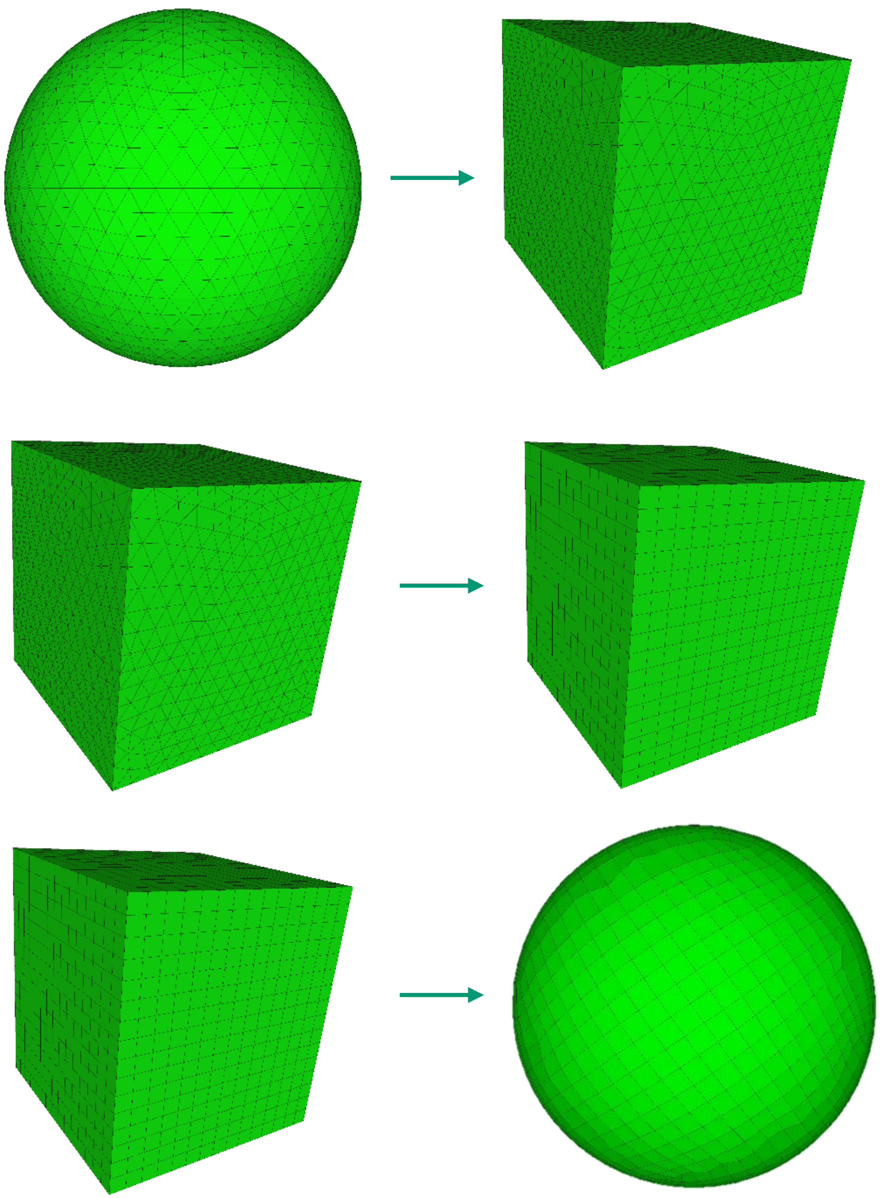
singular edge (val. 5)



singular edge (val. 3)

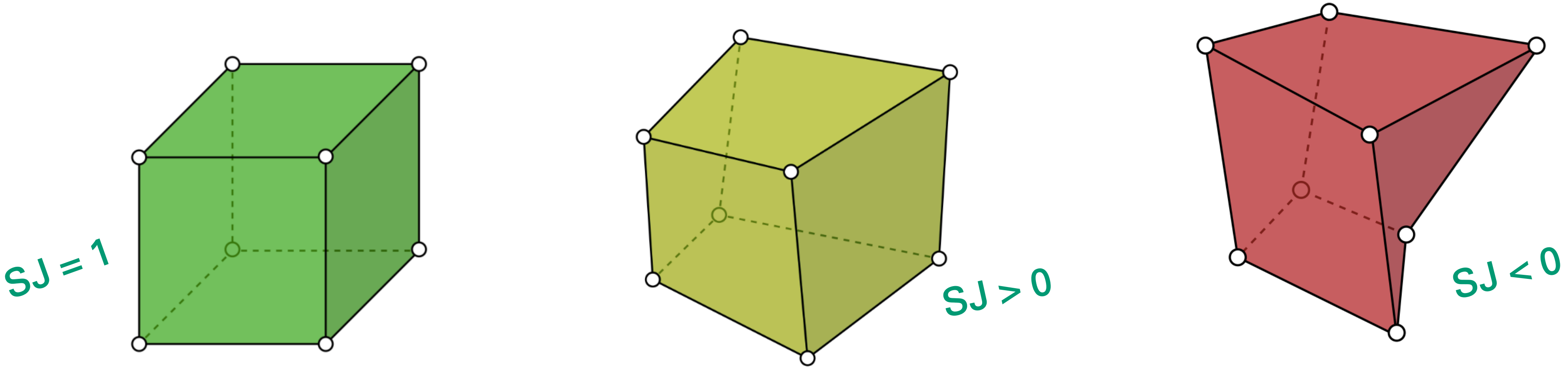
Polycube-based Meshing

- Map volume to polycube space
- Mesh (grid) in polycube space
- Use inverse mapping to bring hex-mesh back to $\mathbb{R}^3$

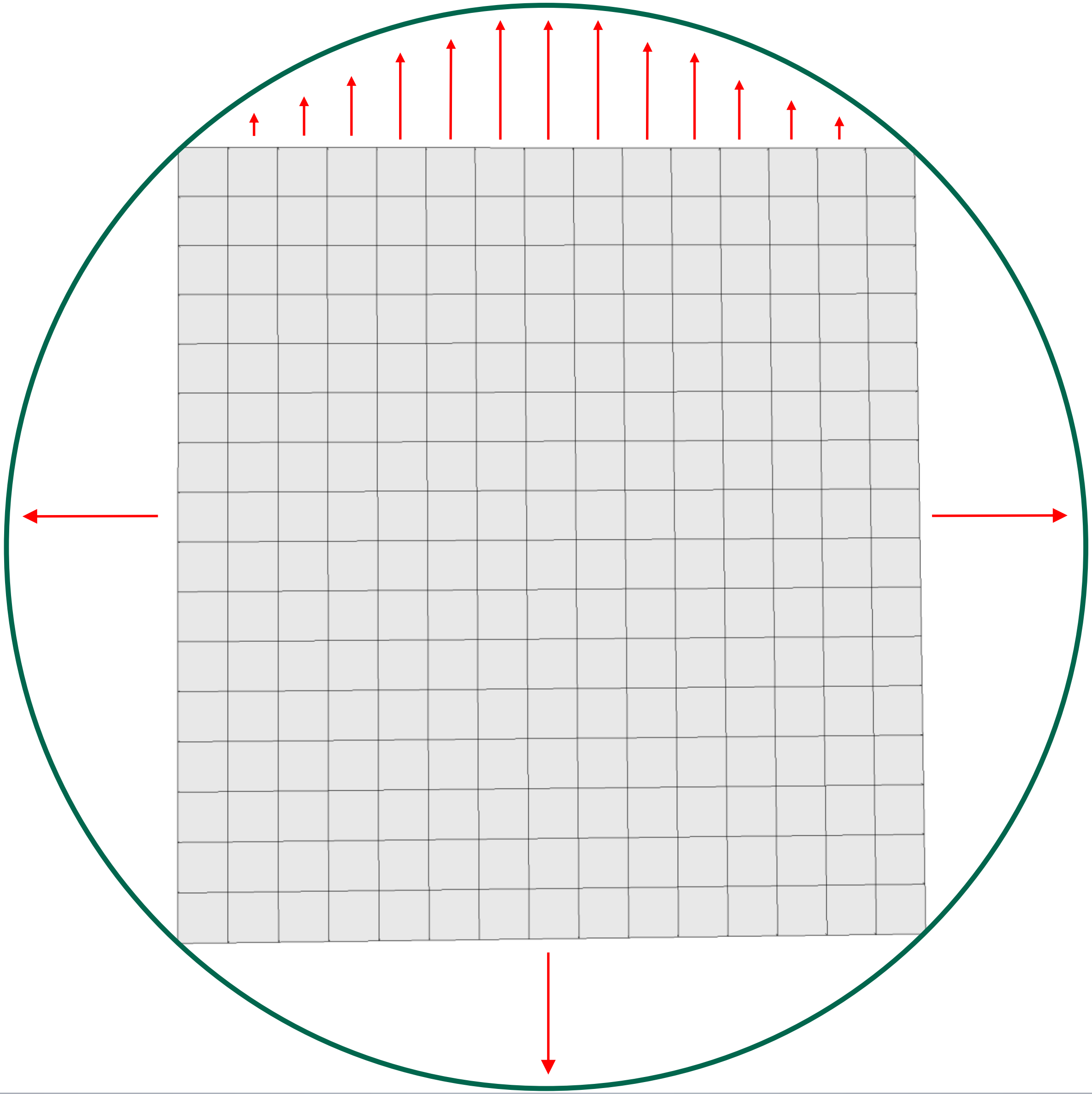


Measuring Quality via Scaled Jacobian (SJ)

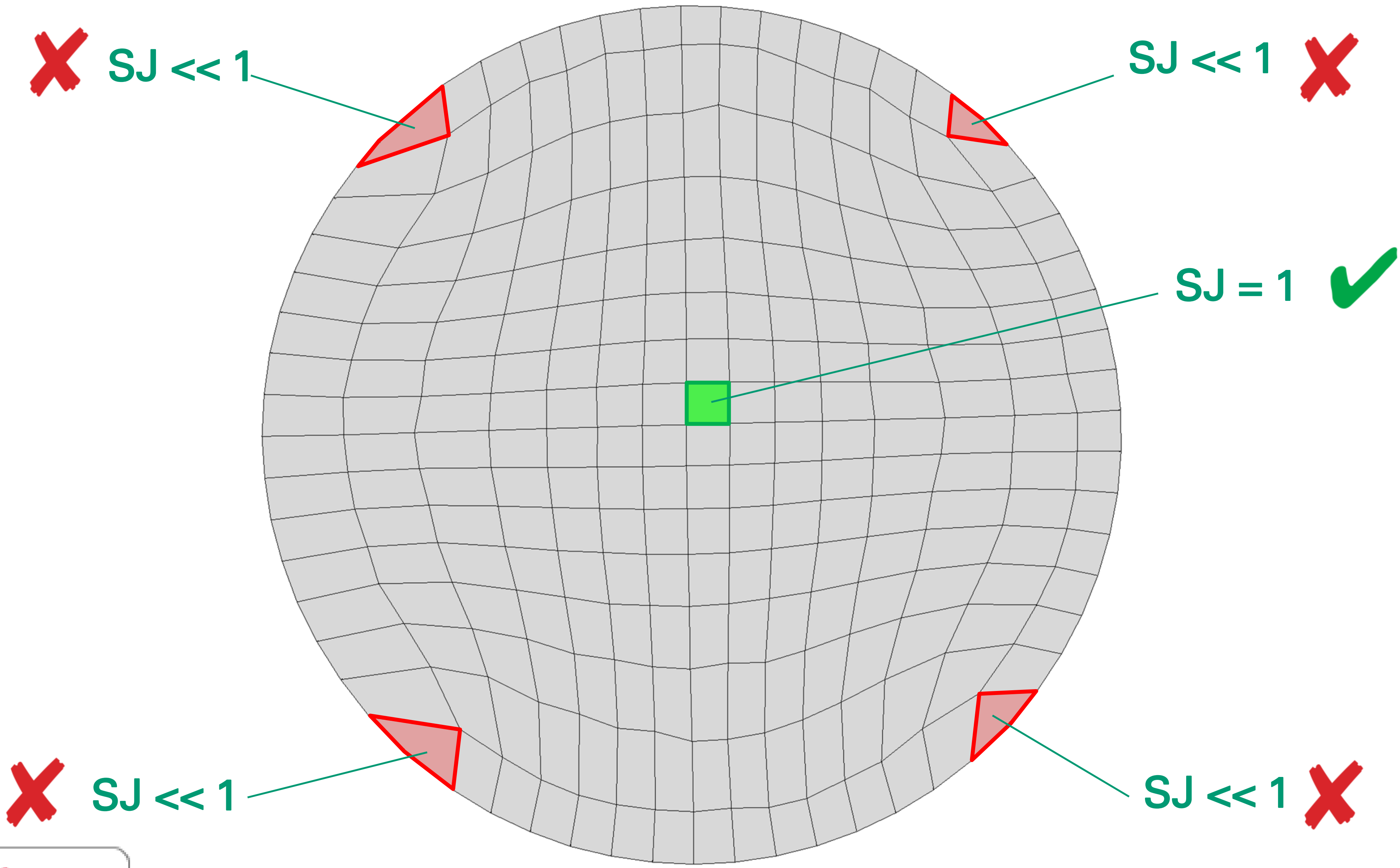
- SJ := min(det(Jacobian))
- Measures quality of a hexahedron, within [-1,1] :
  - Ideal hexahedron = perfect cube (SJ = 1)
  - Convex hexahedron -> SJ > 0
  - Concave hexahedron -> SJ < 0



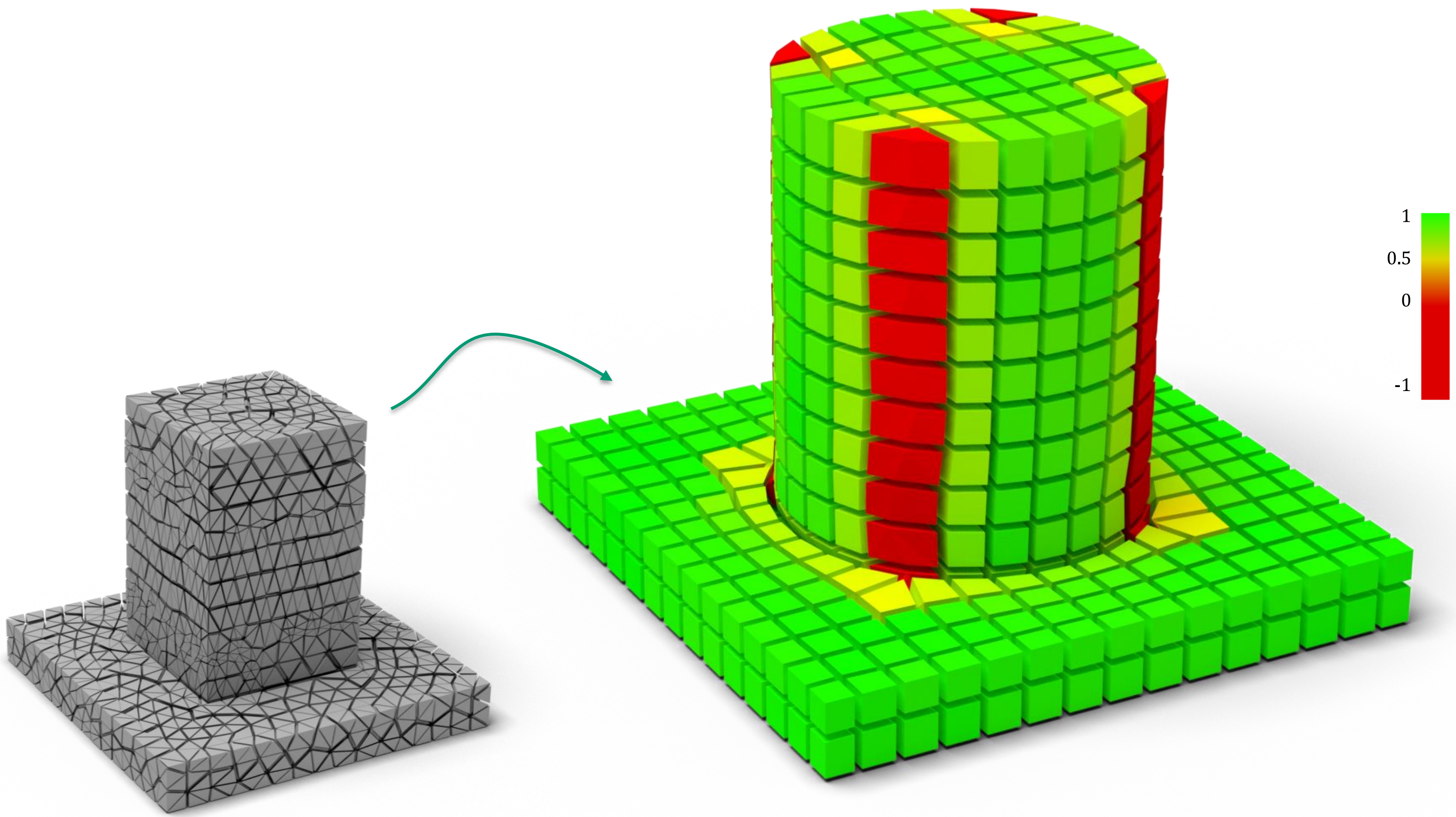
Distortion



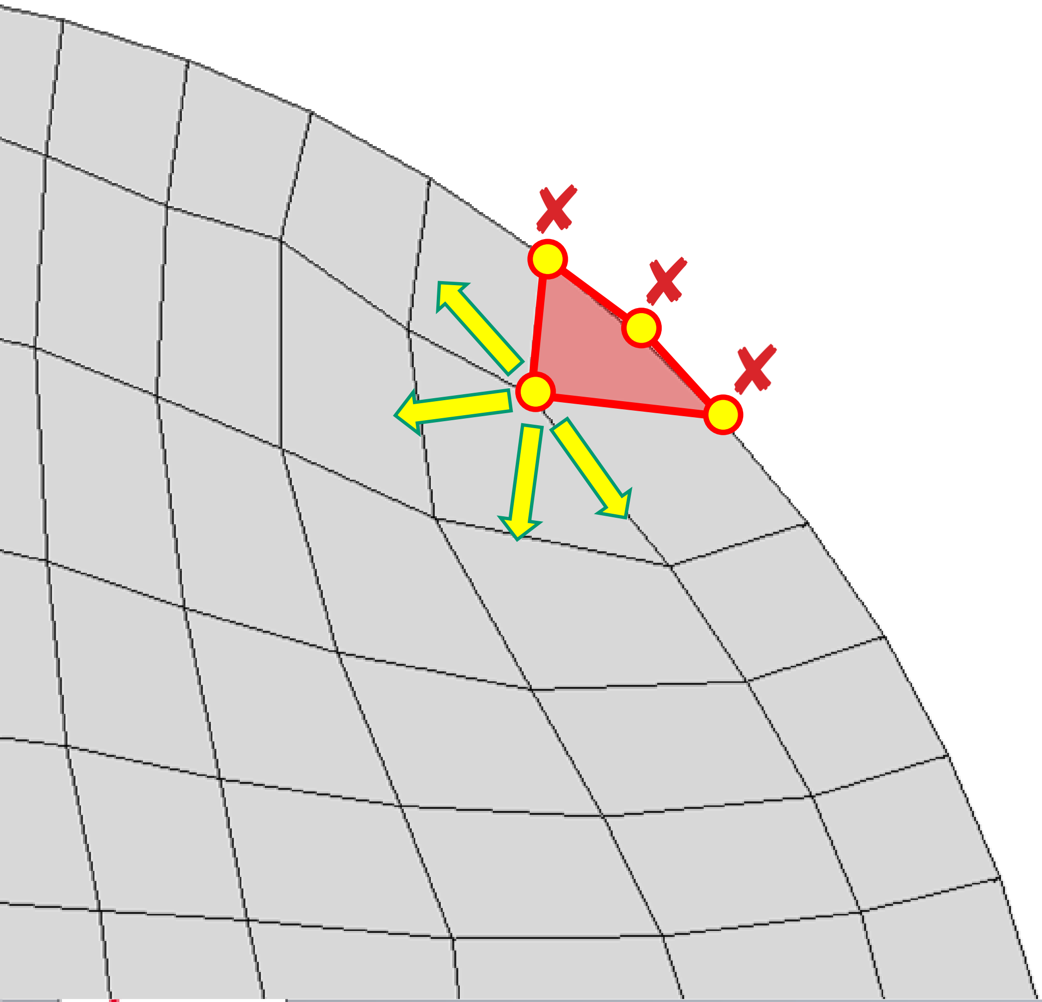
Quality



Distortion

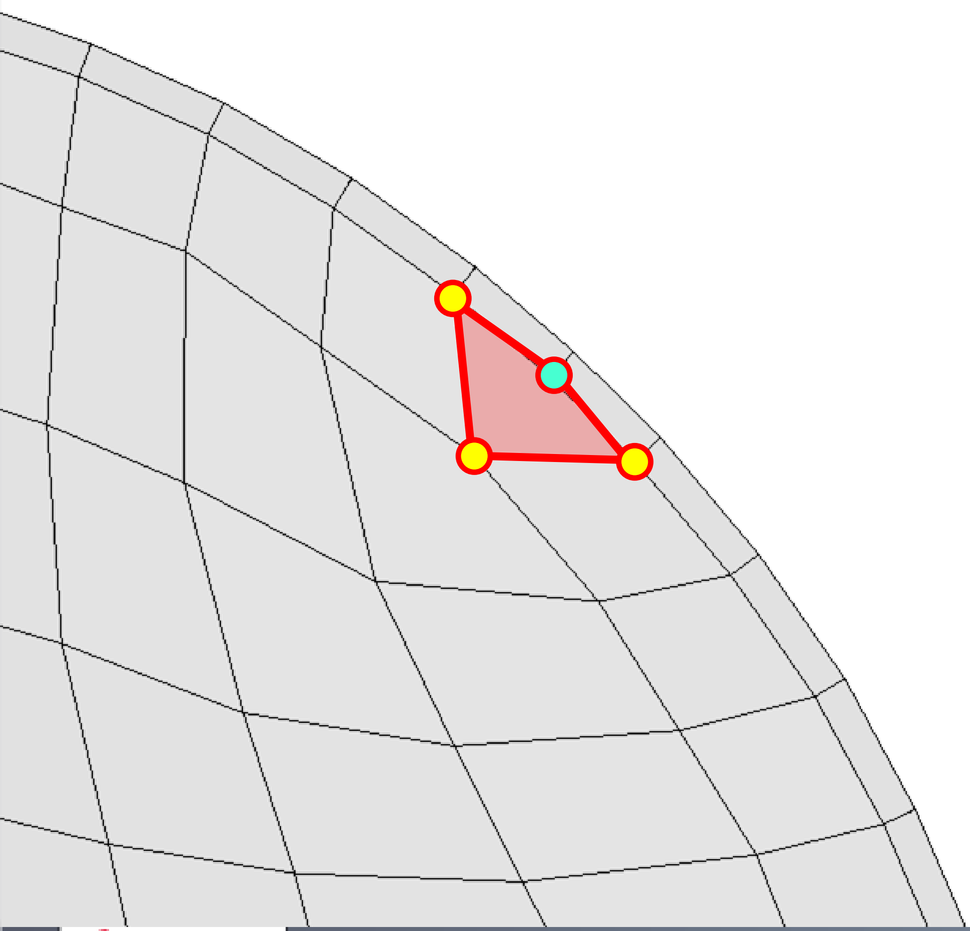


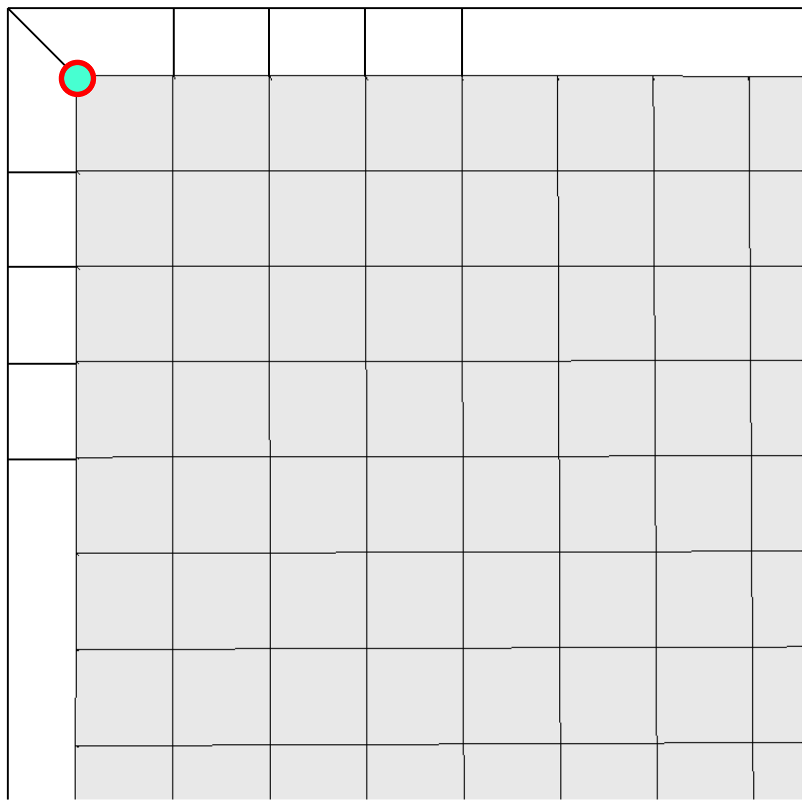
Quality Improvement?



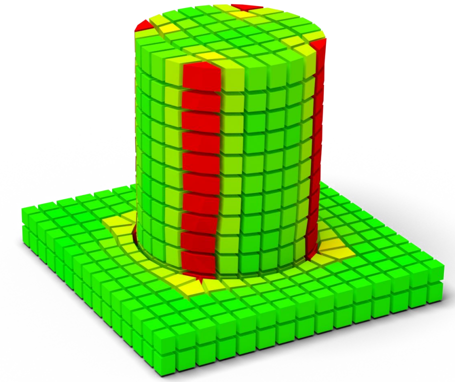
Global Padding

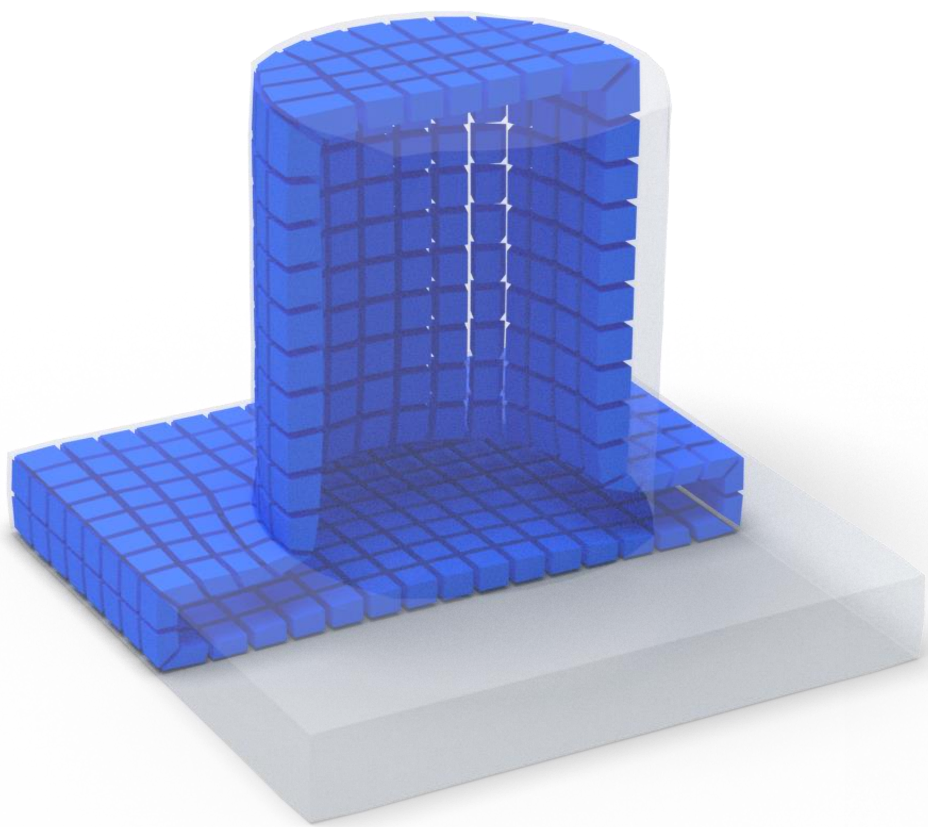
- A layer of new hexahedra
- Global operation

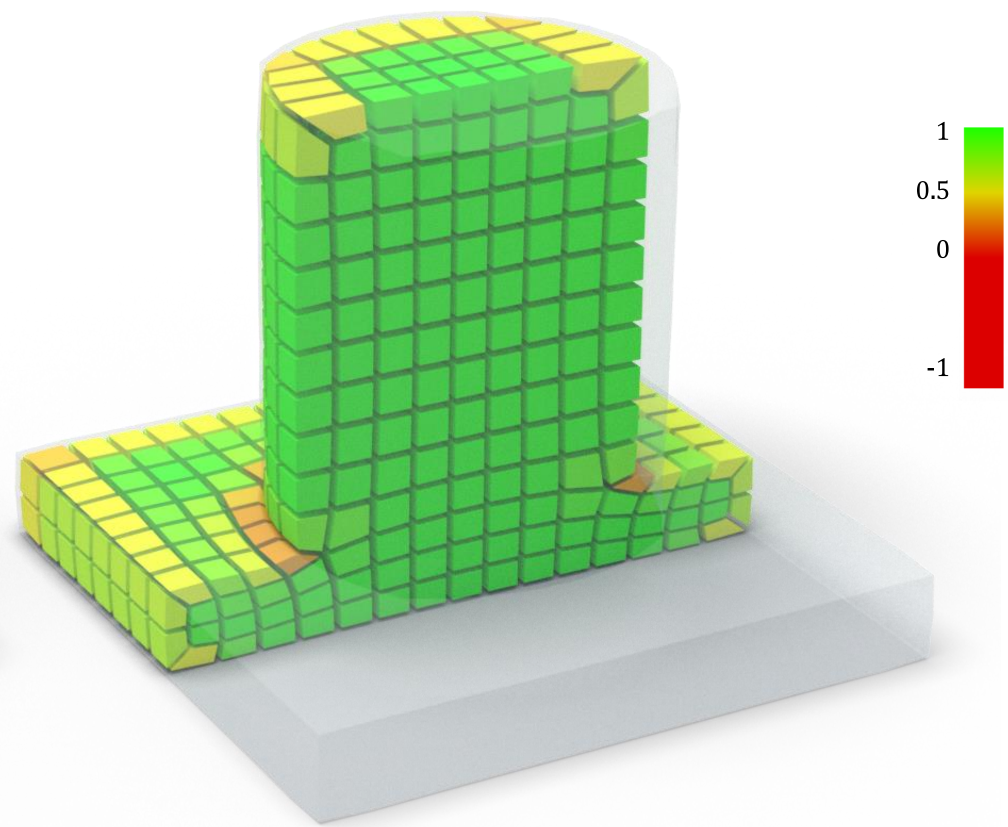




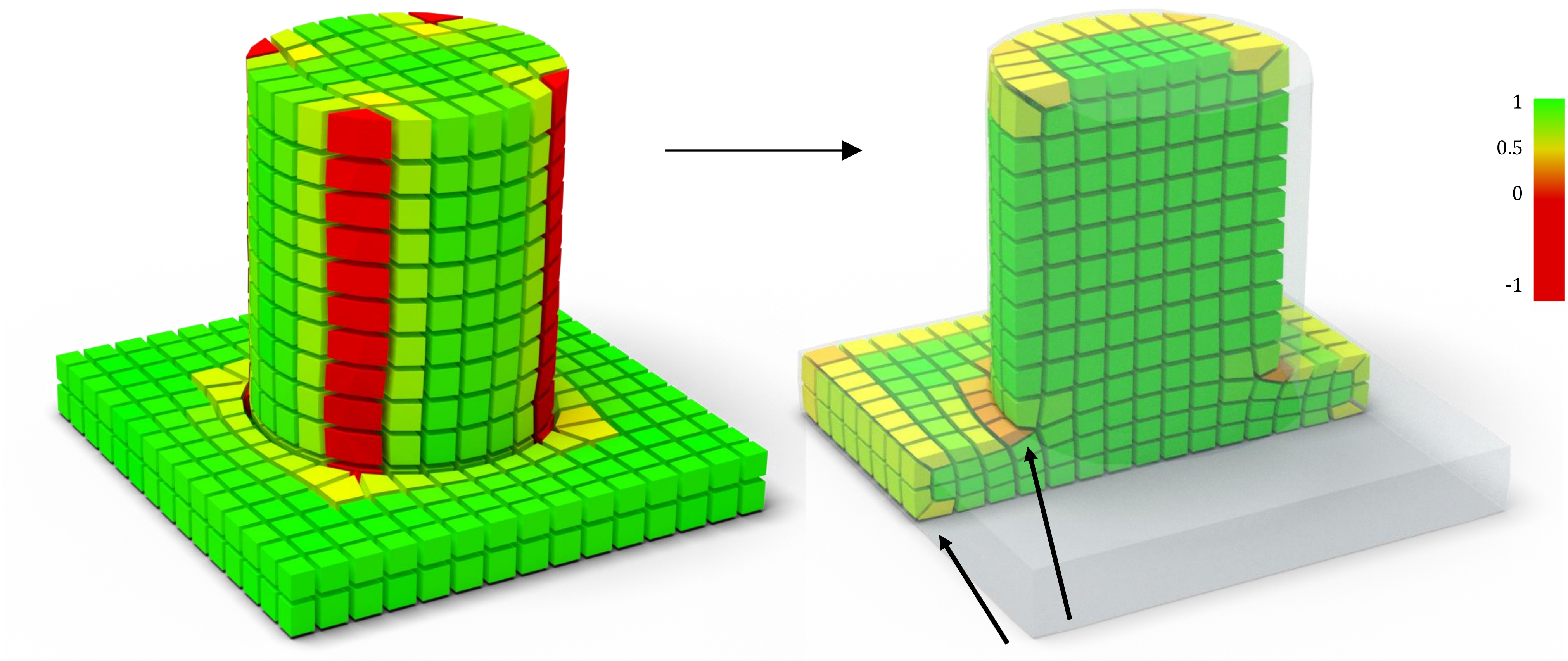
Global Padding







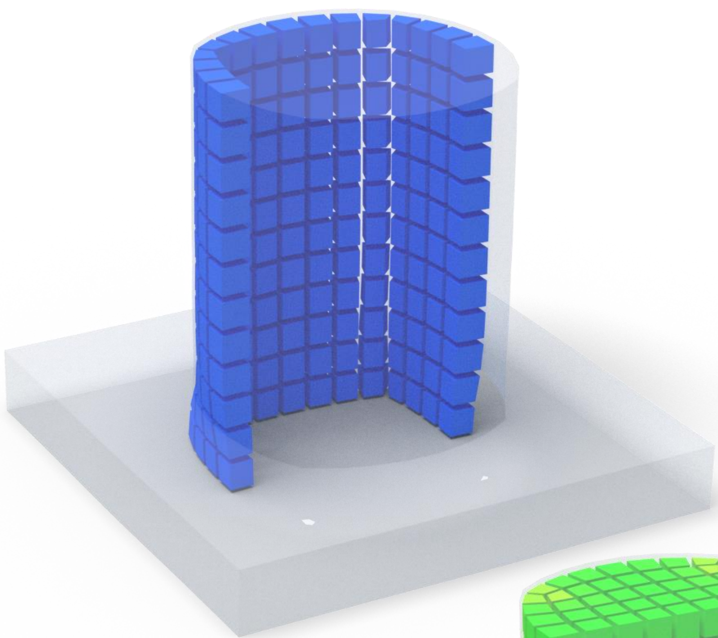
Global Padding

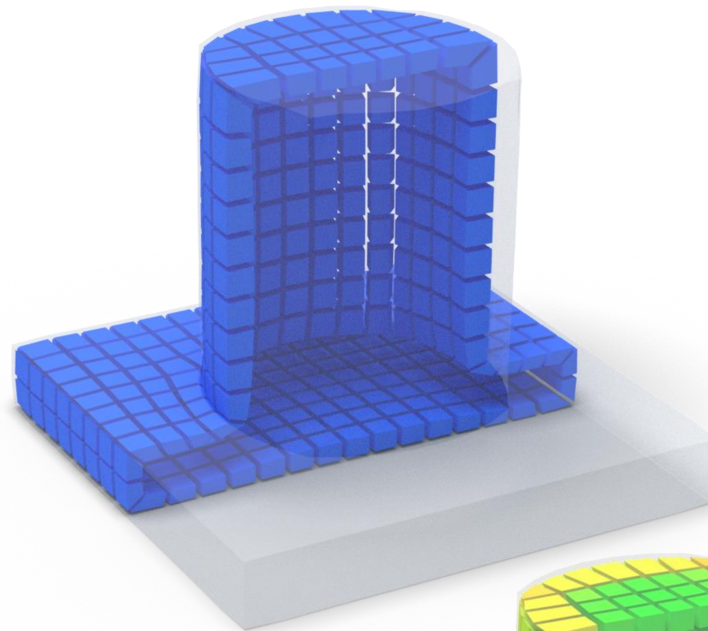


Increased distortion!

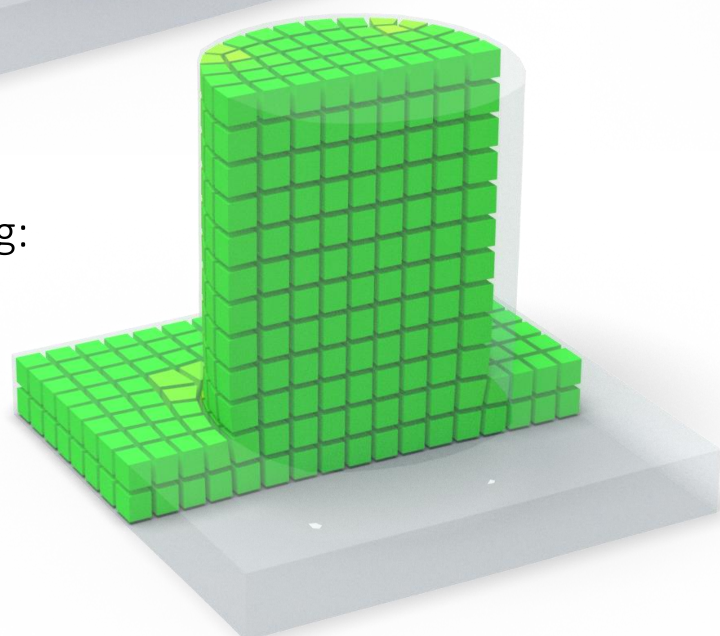
SELECTIVE PADDING

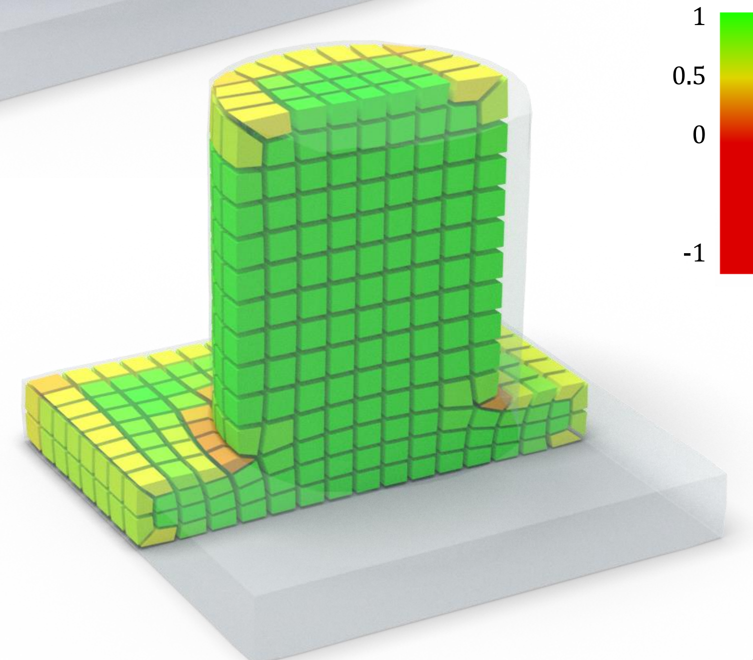
Selective vs Global Padding





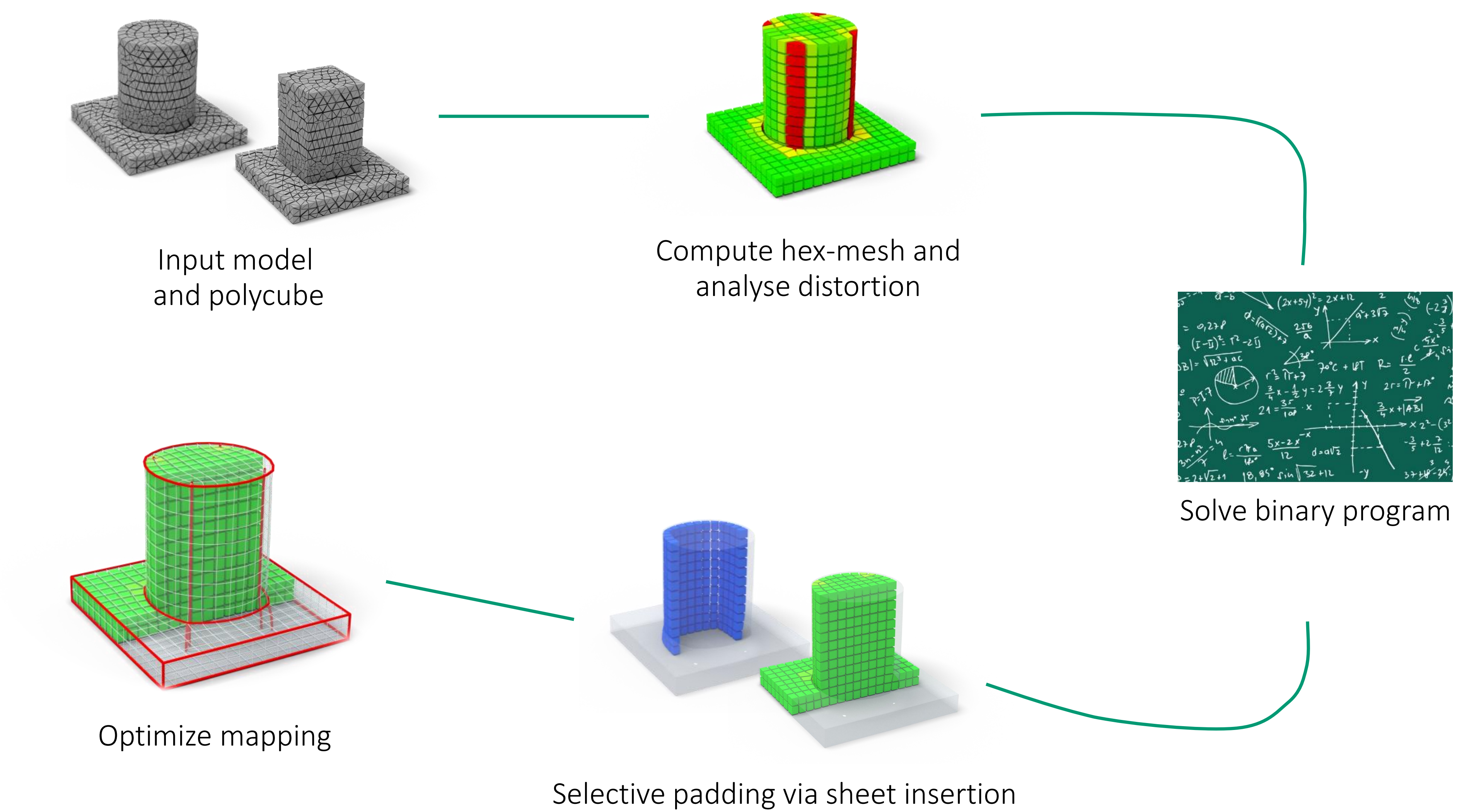
Global padding:
+ elements
+/- singularities
- quality





Selective padding:
- elements
+/- singularities
+ quality

Selective Padding Pipeline

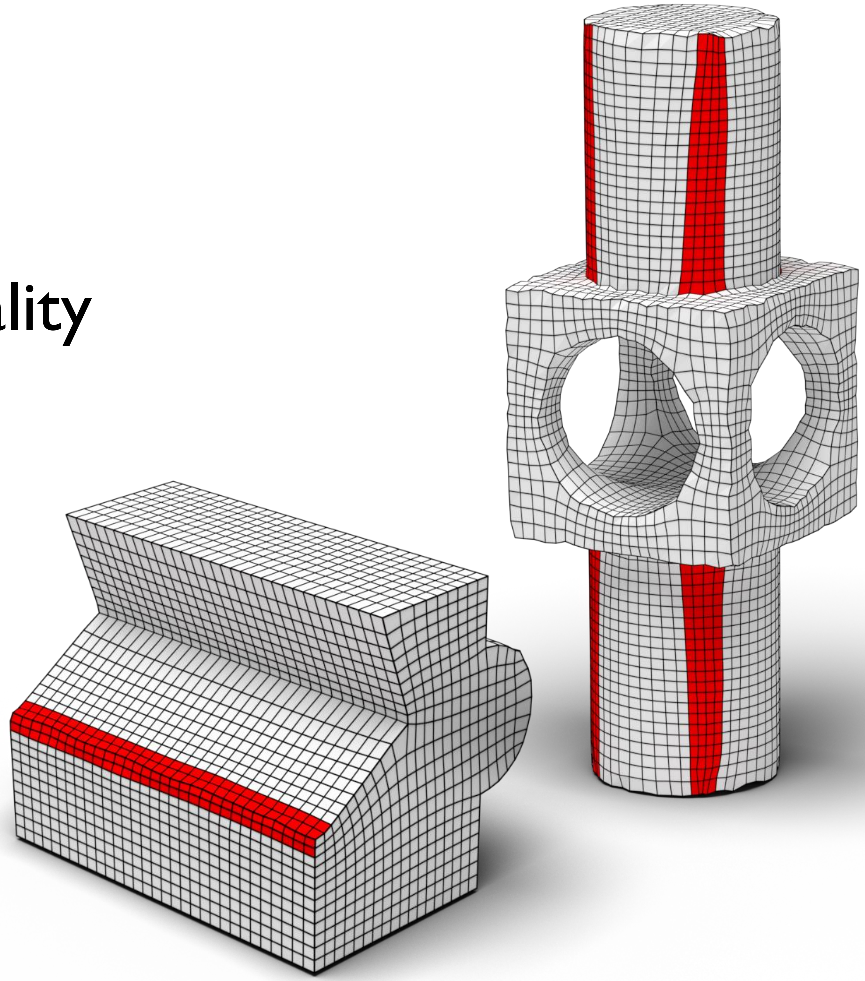


Analyzing Distortion

Find low quality elements

We obtain the final set of low quality facets
(*HF* set)

HF: set of hard constraints



Optimization Problem

- INPUT: *set of faces that require a padding (HF set)*

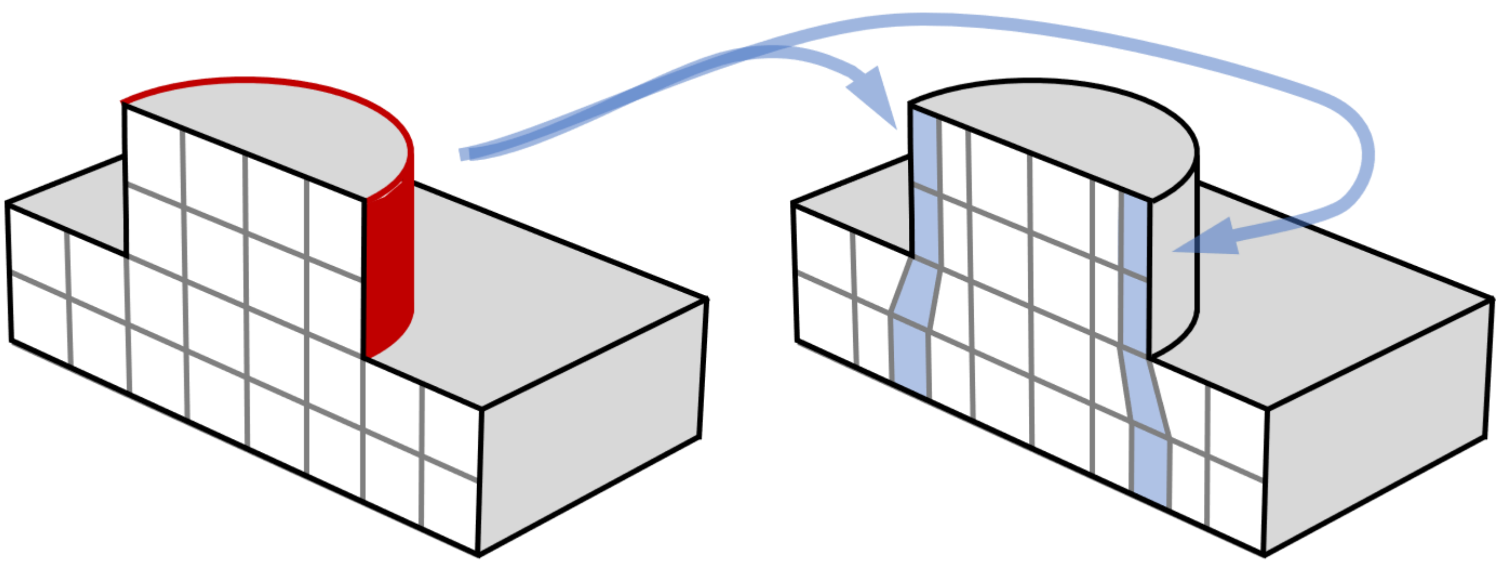
- OUTPUT: *set of faces to pad*
  - Low number of new hexahedra
  - Low number of new singularities
  - Preserve topological consistency (structural constraints)

Find padding facets

A mathematical model with binary and integer variables

$$\min_{\substack{s.t.\\ structural\ constraints}} E = \boxed{E_{padding}} + \lambda \cdot E_{complexity}$$

$$E_{padding} = \sum_{f_i \in F \setminus \mathrm{HF}} xf_i$$



NB: simplified formulas. Extended version of formulas in the thesis.

Find padding facets

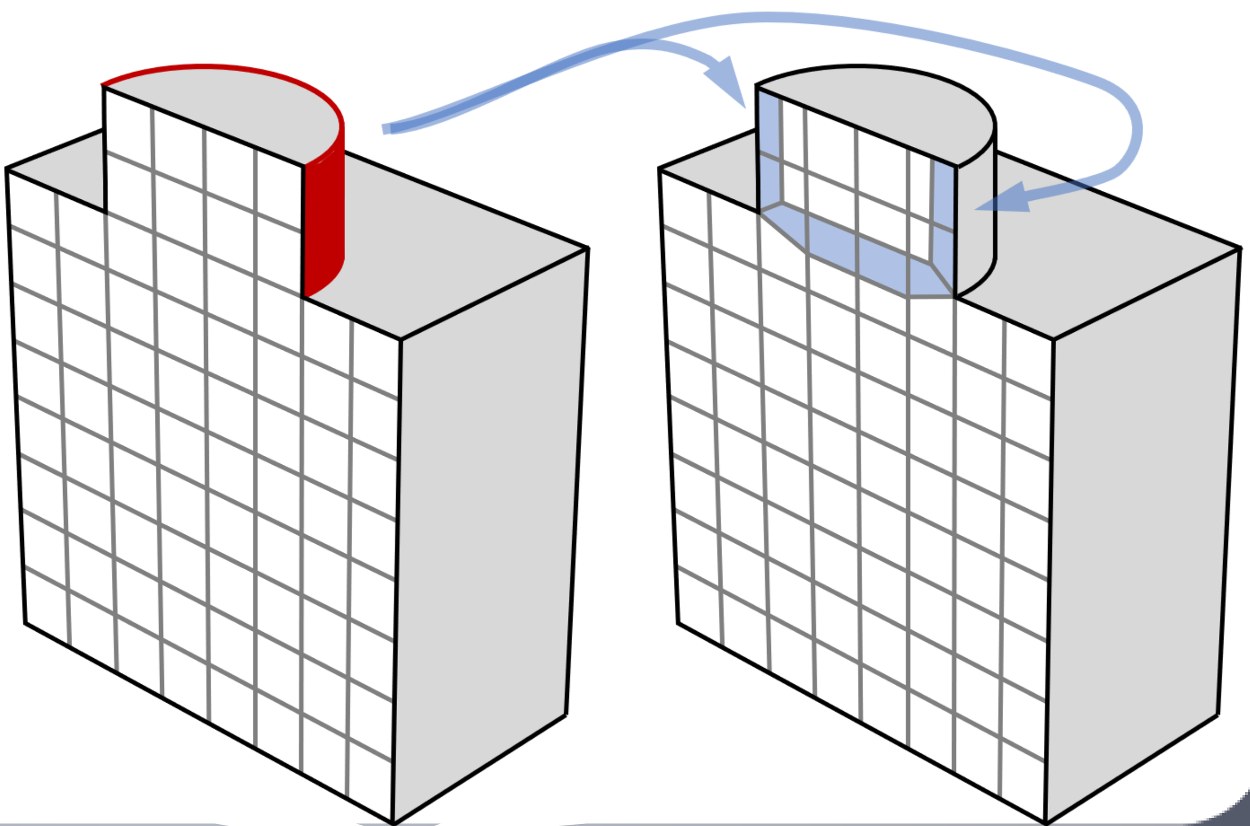
A mathematical model with binary and integer variables

$$\min E = E_{padding} + \lambda \cdot \boxed{E_{complexity}}$$

s.t.

structural constraints

$$E_{complexity} = \sum_{e_j \in E^*} te_j + \sum_{v_l \in V^*} tv_l$$



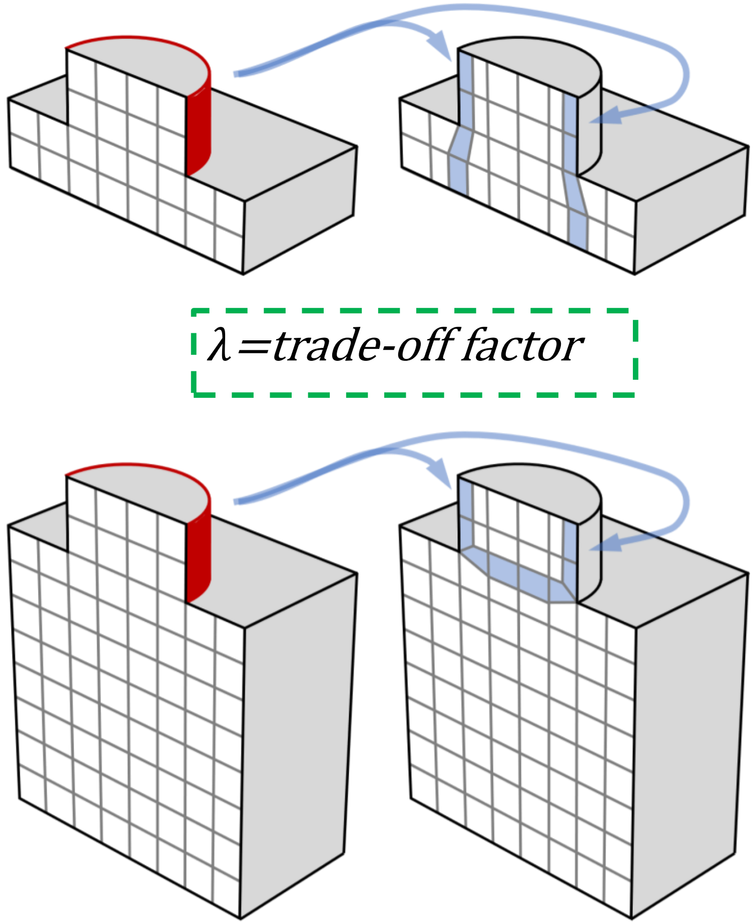
NB: simplified formulas. Extended version of formulas in the thesis.

Find padding facets

A mathematical model with binary and integer variables

$$\min E = E_{padding} + \boxed{\lambda} \cdot E_{complexity}$$

s.t.
structural constraints



λ=*trade-off factor*

$$E_{padding} = \sum_{f_i \in F \setminus \mathrm{HF}} xf_i$$

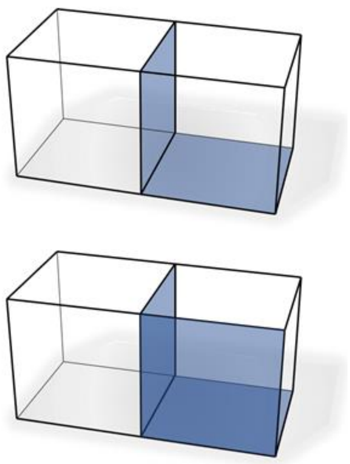
$$E_{complexity} = \sum_{e_j \in E^*} te_j + \sum_{v_l \in V^*} tv_l$$

NB: simplified formulas. Extended version of formulas in the thesis.

Find padding facets

A mathematical model with binary and integer variables

$$\min E = E_{padding} + \lambda \cdot E_{complexity}$$
$$s.t.$$
$$\boxed{structural\ constraints}$$



$$E_{padding} = \sum_{f_i \in F \setminus \mathrm{HF}} xf_i$$

$$E_{complexity} = \sum_{e_j \in E^*} te_j + \sum_{v_l \in V^*} tv_l$$

λ=*trade-off factor*

- Correct propagation of the new layer
- Counting vertex turns
- Counting edge turns

NB: simplified formulas. Extended version of formulas in the thesis.

Padding constraints

$$\min E_{padding} = \sum_{f_i \in F \setminus \mathrm{HF}} \boxed{xf_i}$$

A binary variable for each facet:

$$xf_i = \begin{cases} 1 \; if \; f_i \; need \; padding \\ 0 \; otherwise \end{cases}$$

$$xf_i = 1 \quad \forall f_i \in HF$$

Hard constraints for facets in HF

A binary variable for each facet

$$\sum_{f_i \in F(e_j)} xf_i = 2k_i \qquad \forall e_j \in E$$

Constraint to avoid topological inconsistencies during the new layer propagation

Complexity constraints

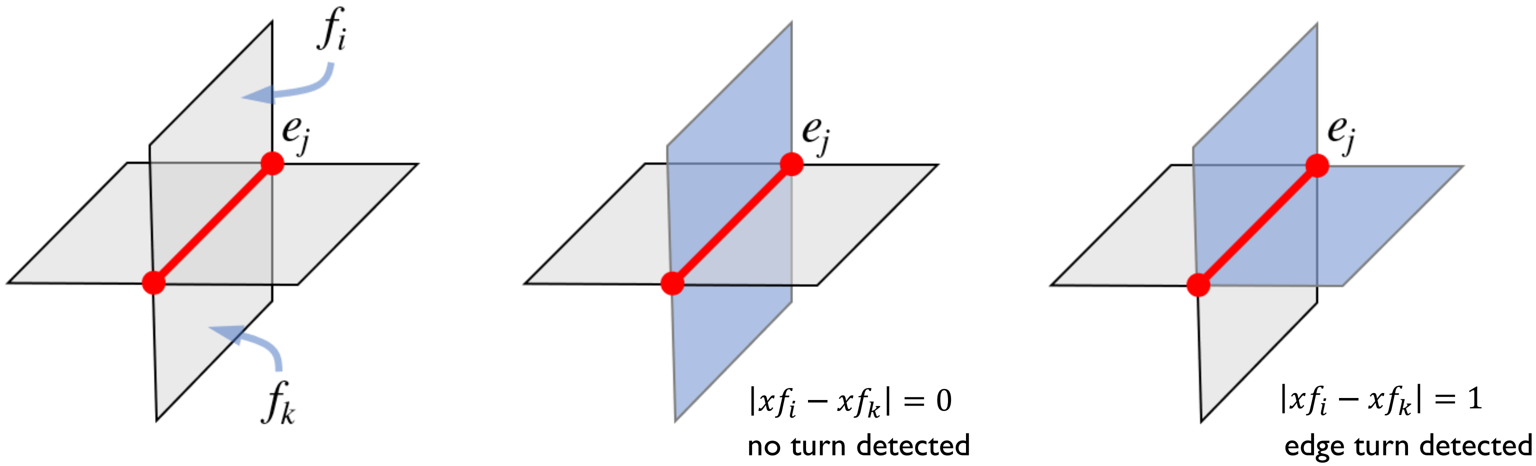
$$E_{complexity} = \sum_{e_j \in E^*} \boxed{te_j} + \sum_{v_l \in V^*} \boxed{tv_l}$$

A binary variable for each edge and for each vertex:

$$t_i = \begin{cases} 1 \; if \; there \; is \; a \; turn \\ 0 \; otherwise \end{cases}$$

Edge turn:

$$te_j = |xf_i - xf_k| \qquad \forall e_j \in E^*, \qquad \vec{f_i} = \vec{f_k} \quad and \quad f_i, f_k \in F(e_j)$$

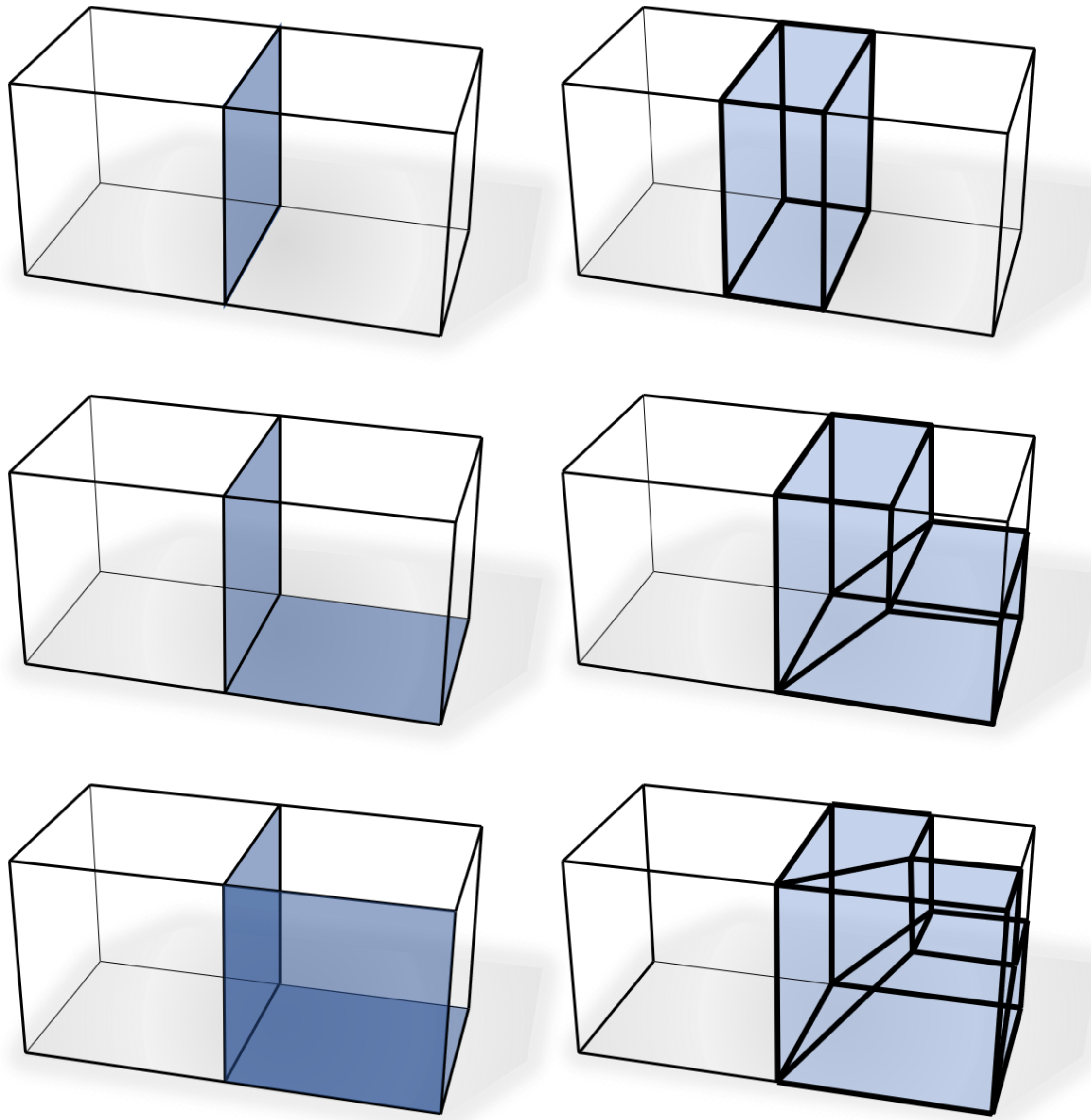


Same reasoning for vertex turns

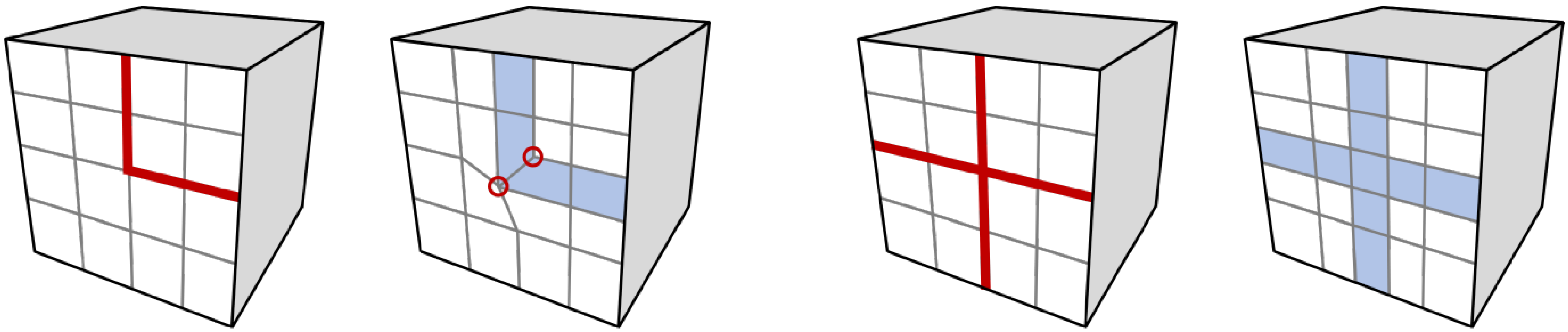
Sheet Insertion

Once we know the set of facets to “pad” to create the padding layer

Padding == facet extrusion

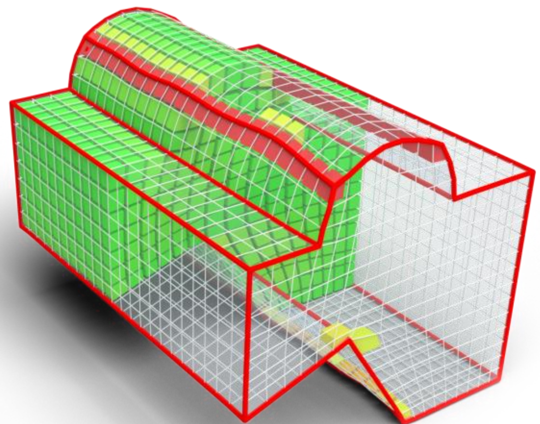


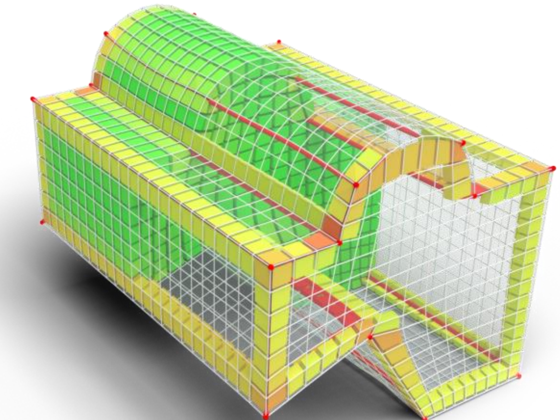
Sheet Insertion



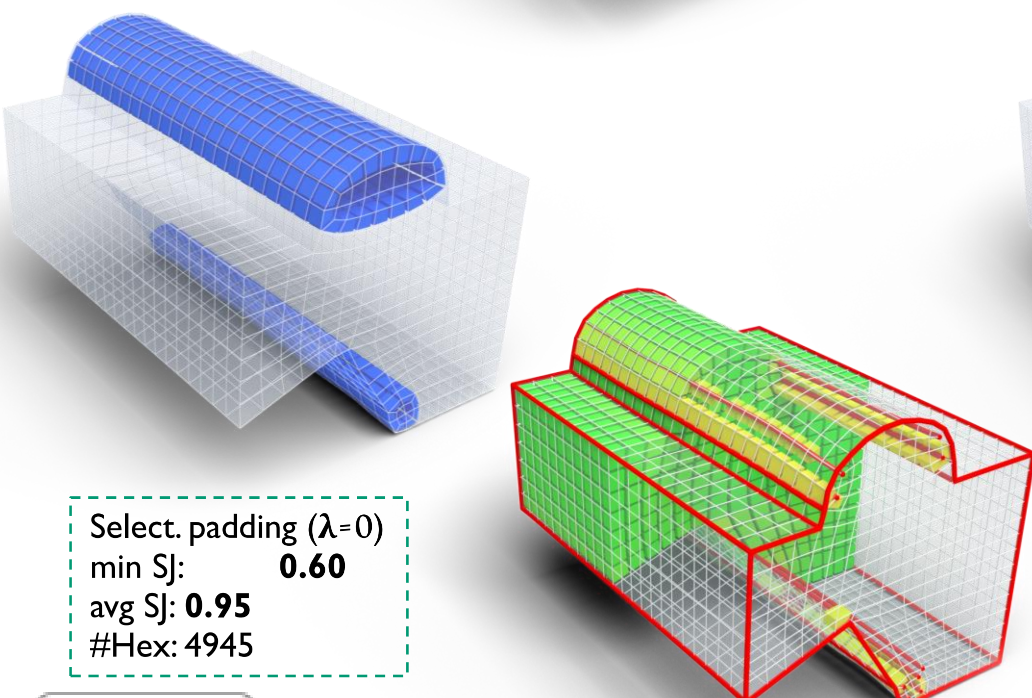
Results

Original model
min SJ: 0.10
avg SJ: 0.96
#Hex: 4347



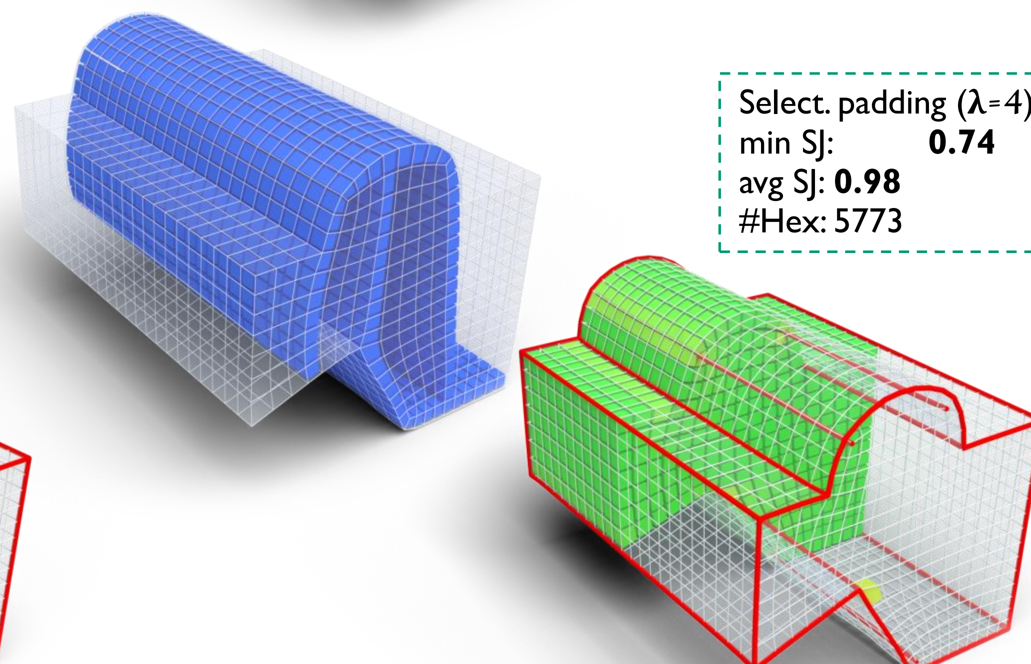


Global padding
min SJ: 0.10
avg SJ: 0.94
#Hex: 6197

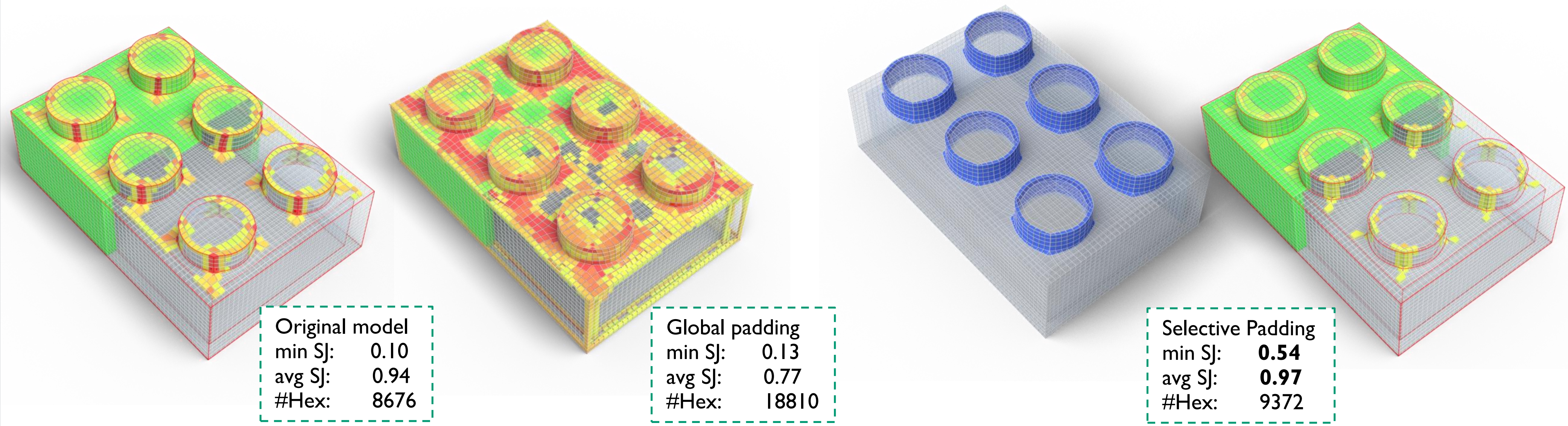


Select. padding (λ=0)
min SJ: **0.60**
avg SJ: **0.95**
#Hex: 4945

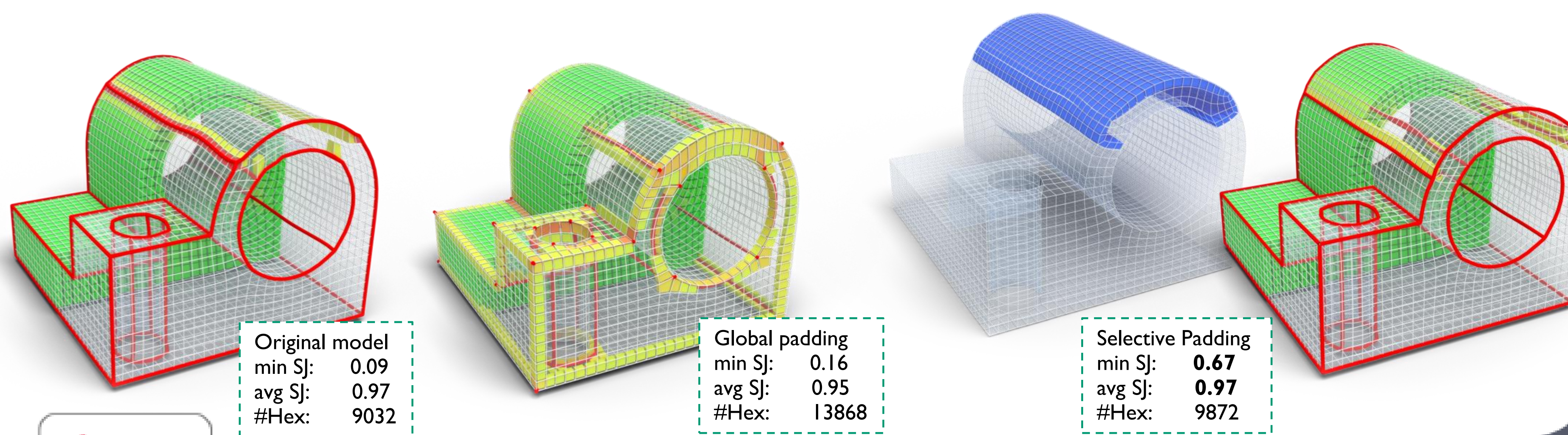
Select. padding (λ=4)
min SJ: **0.74**
avg SJ: **0.98**
#Hex: 5773



Results

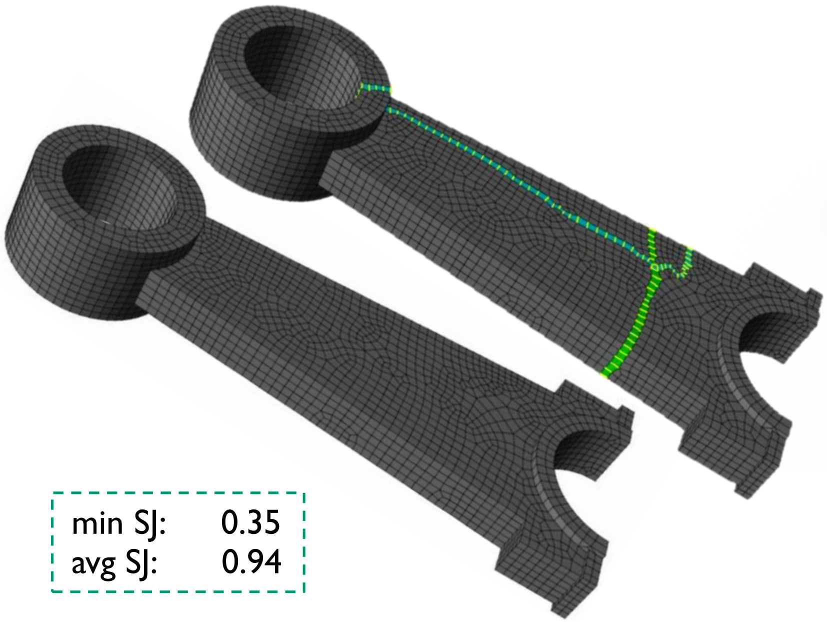


Original model
min SJ: 0.10
avg SJ: 0.94
#Hex: 8676
Global padding
min SJ: 0.13
avg SJ: 0.77
#Hex: 18810
Selective Padding
min SJ: **0.54**
avg SJ: **0.97**
#Hex: 9372



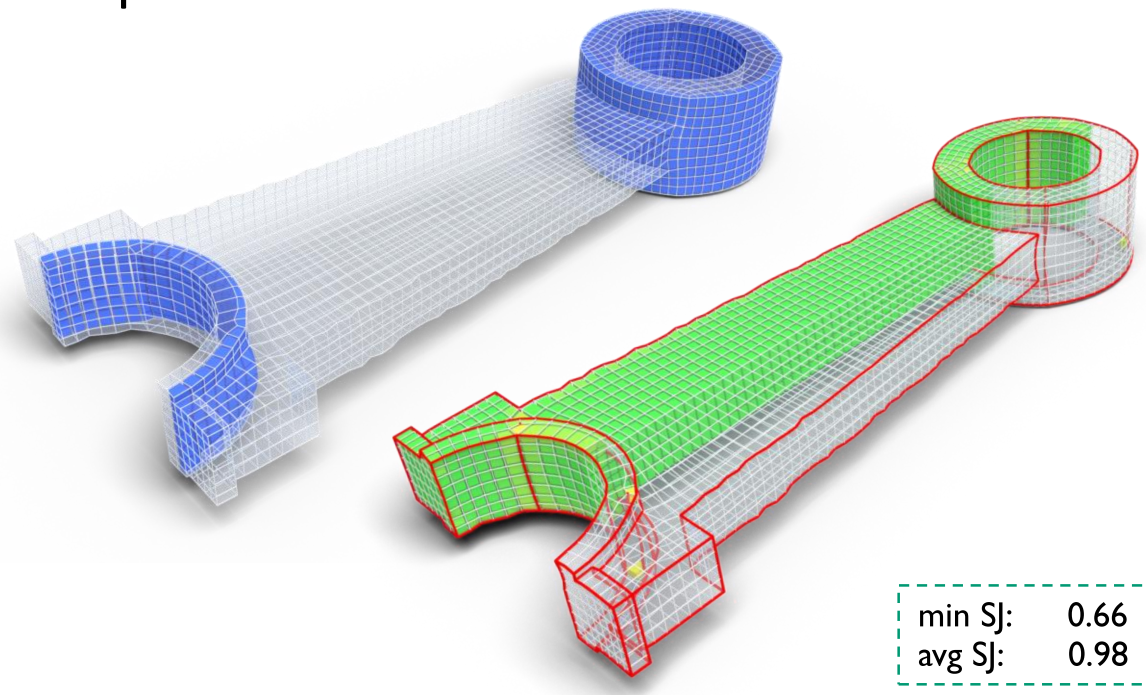
Original model
min SJ: 0.09
avg SJ: 0.97
#Hex: 9032
Global padding
min SJ: 0.16
avg SJ: 0.95
#Hex: 13868
Selective Padding
min SJ: **0.67**
avg SJ: **0.97**
#Hex: 9872

Comparison



min SJ: 0.35
avg SJ: 0.94

[Wang et al. 2018]



min SJ: 0.66
avg SJ: 0.98

[Ours]

- Comparable results (usually better)
- More regular inner structure (and less singularities)
- Analysis only on model surface

Limitations

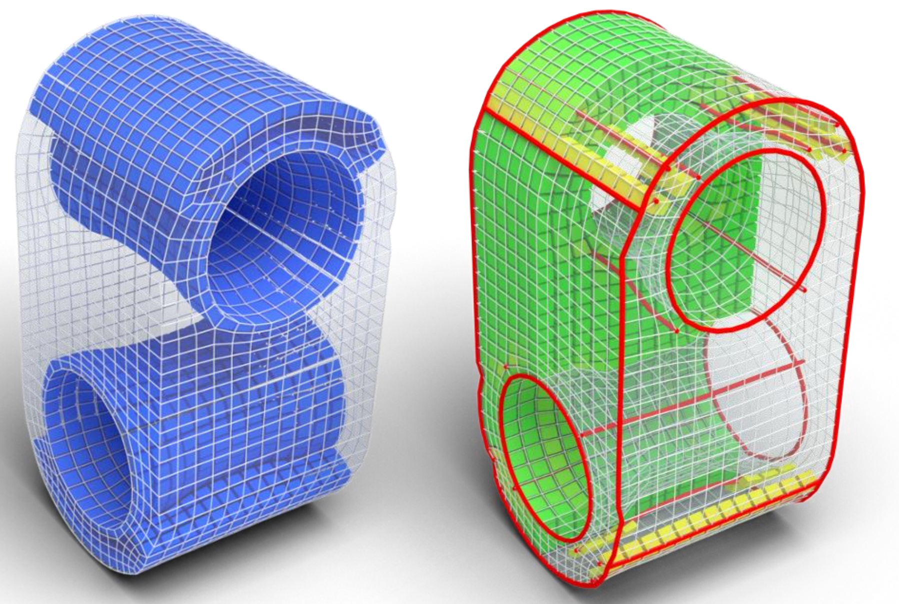
- Padding “holes”

object shape

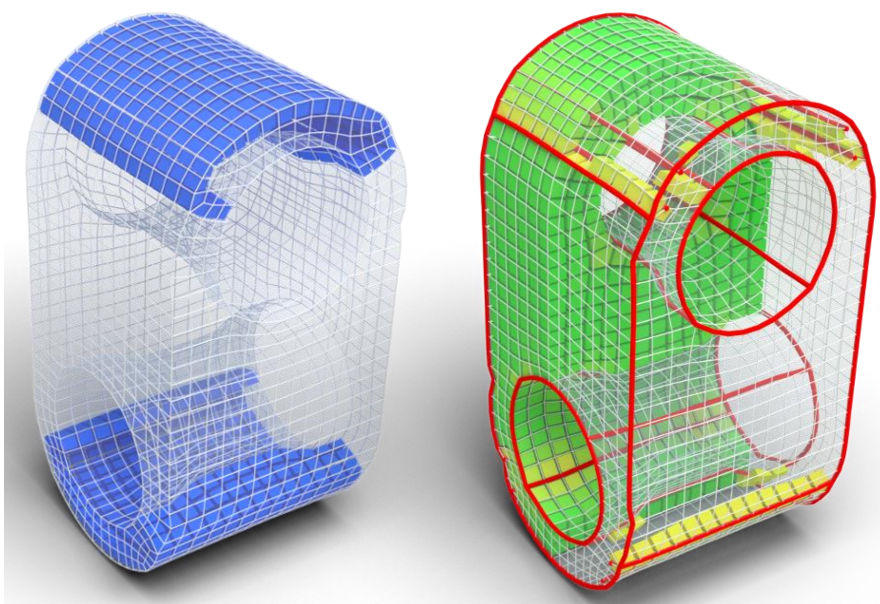
object final use

object resolution

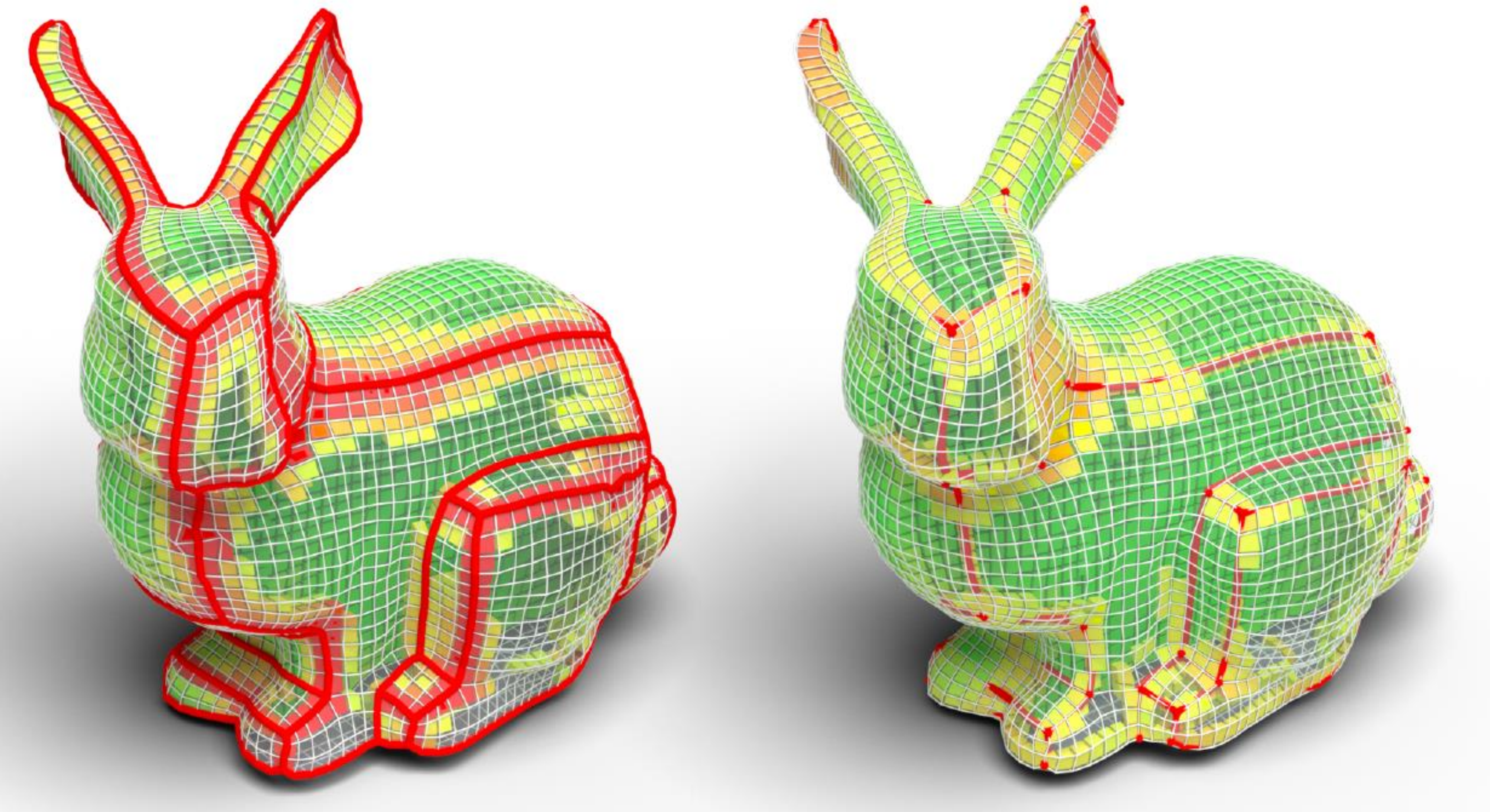
Selec. Padding **(holes)**
min SJ: 0.63
avg SJ: 0.95
#Hex: 4550



Selec. Padding **(NO holes)**
min SJ: **0.65**
avg SJ: 0.95
#Hex: **3770**



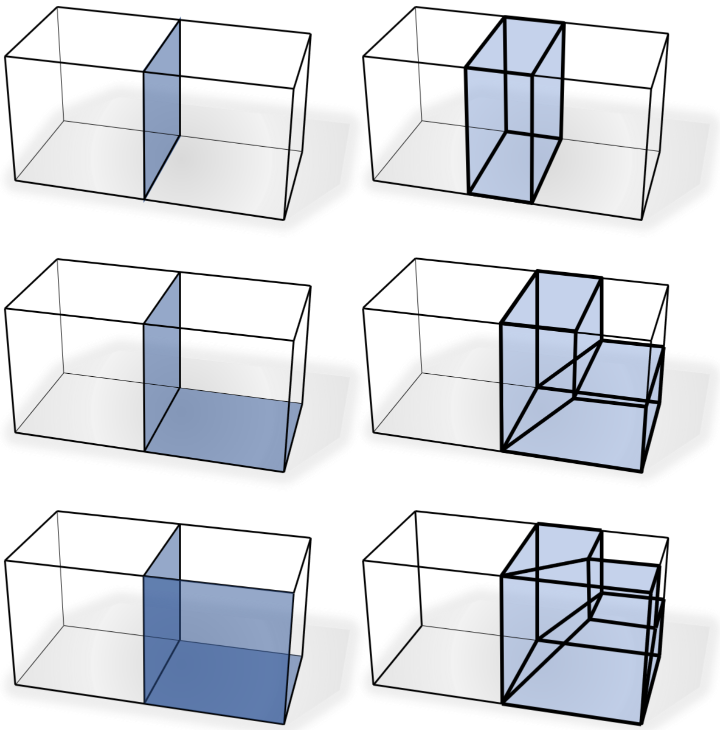
Free-form Shape



Strengths

Strengths

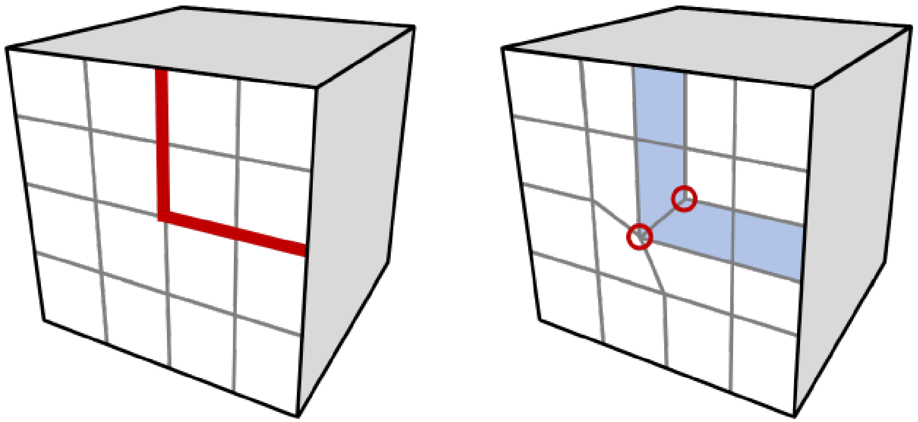
- Pure hex by construction
- Complexity/distortion tradeoff
  - Automatic parameter selection?
- Padding: increase DOFs
  - Sizing (multi-padding?)
  - Anisotropy
  - Orientation
  - Singularities & layout [Kobbelt et al.]



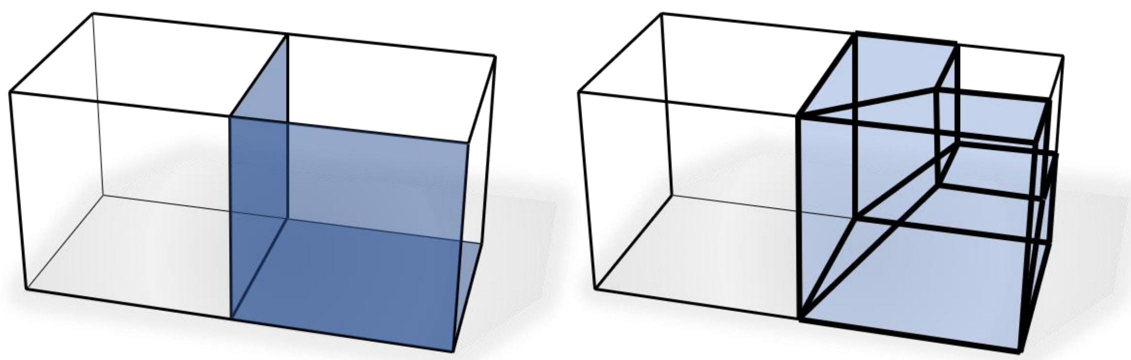
Takayama, EG 2019. Dual Sheet Meshing:
An Interactive Approach to Robust Hexahedralization

Turns / Singularities

- **Hinge (edge) turn**
  - Create pair of singularities
  - Or split singularity



- **Corner (vertex) turn**
  - Create
  - Split



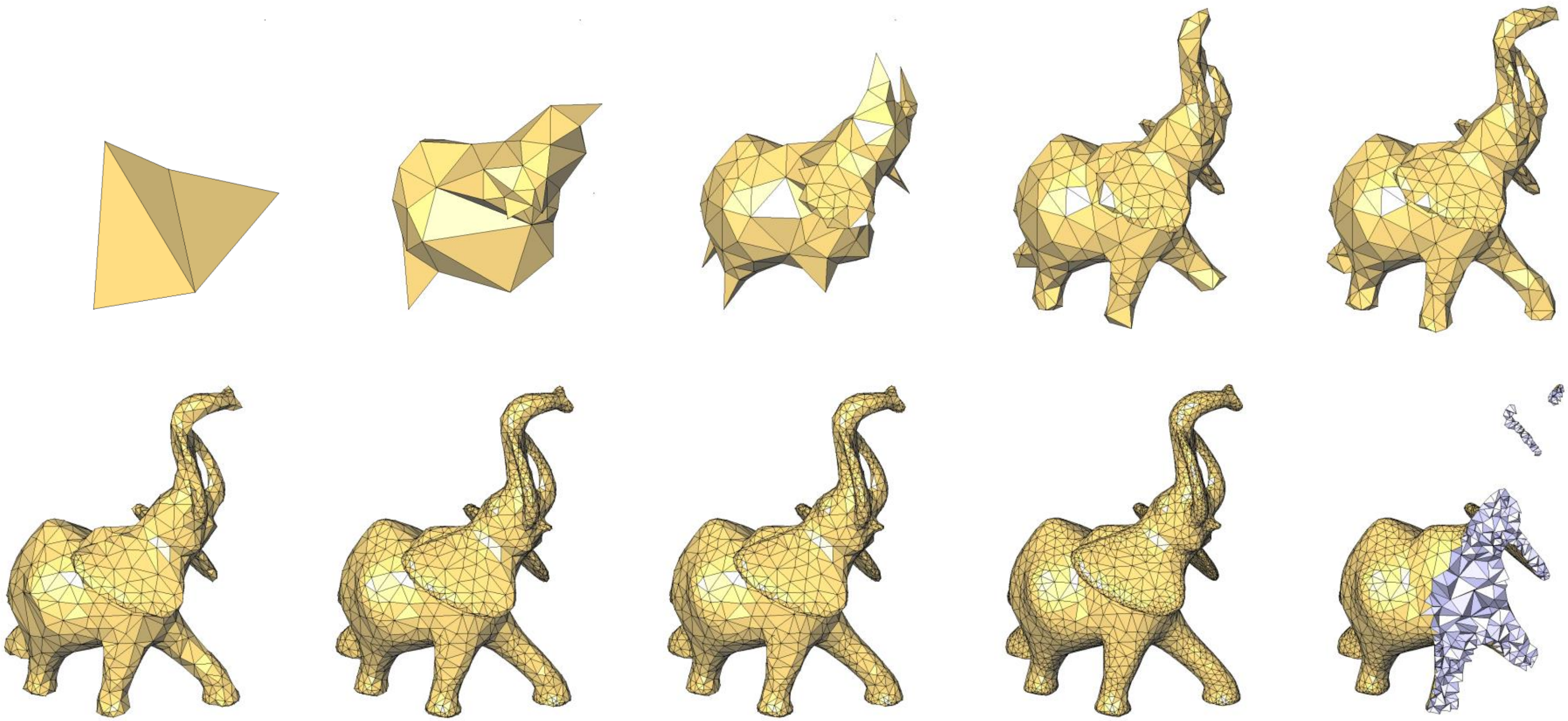
Weaknesses

- Requires polycube as input
- Not all domains can be polycubed
- Limited class of hex mesh topologies
  - Singularities only on boundary
- Padding & embedding non trivial
- Distortion: hard to predict & control

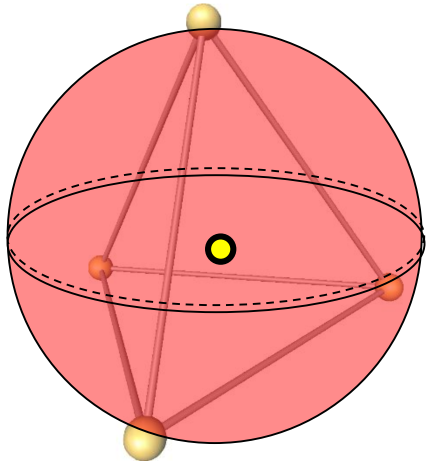
TOWARD PURE-HEX MESHING

Proposal

Parallel Delaunay refinement, with non-local operators



Delaunay Refinement



initialize

repeat

{

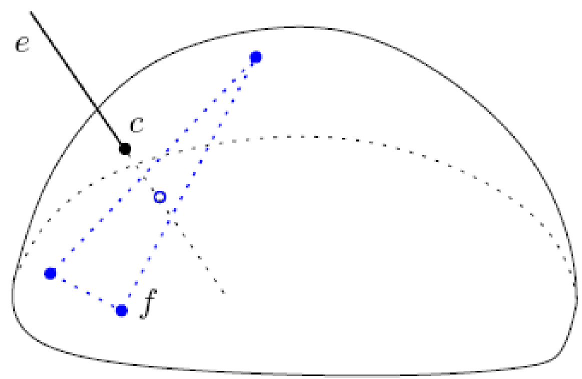
pick bad element (size, shape, ...)

refine element

update Delaunay filtering

}

until all elements are good



Delaunay Refinement

initialize

repeat

{

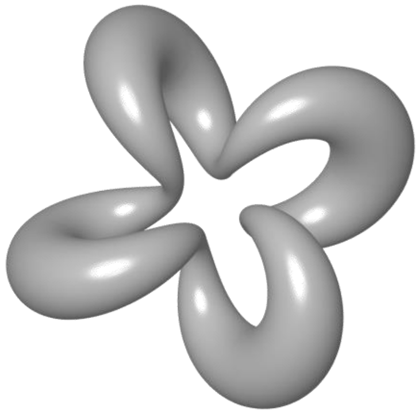
  pick bad element

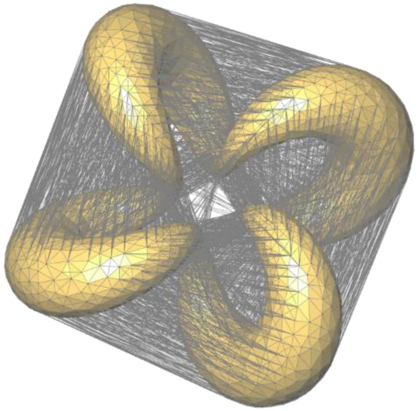
  refine element

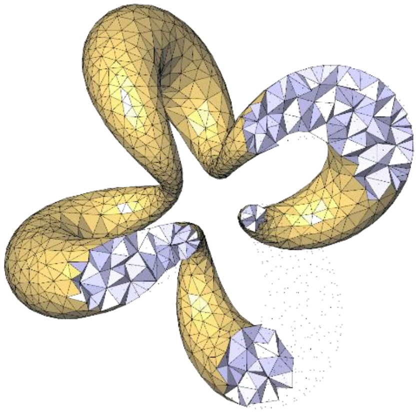
  update **restricted Delaunay**

}

until all elements are good





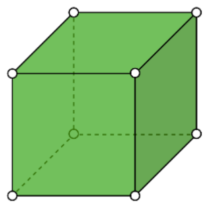


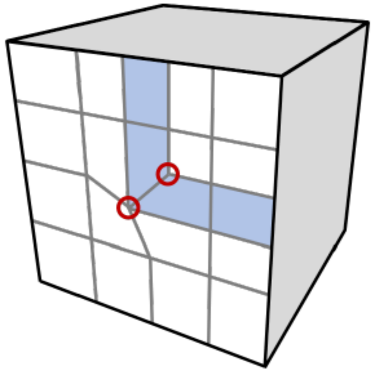
Proposal

Initialize

Trivial element/domain

Compute mapping





Repeat

{

Analyze bad elements (size, shape, orientation)

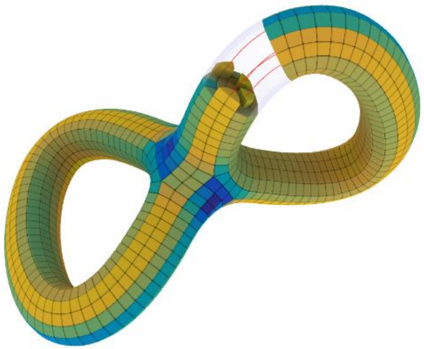
Global solve & insert sheets

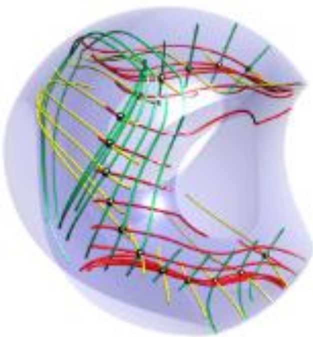
Update mapping

Update restricted elements

}

until all elements are good





3D cross frame-field [Huang]

Required Components

Repeat

{

Analyze bad elements

Scale-space frame field

Global solve & insert sheets

Predict/learn parameters/constraints

Update mapping

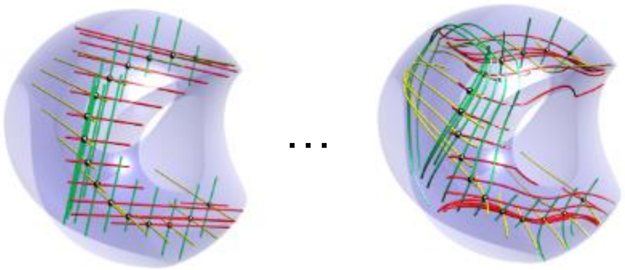
Minimize/bound distortion

Interpolant / higher-order

Guarantee valid embedding

Update restricted elements

}



Variance-Minimizing Transport Plans for Inter-surface Mapping
Mandad, Cohen-Steiner, Kobbelt, A, Desbrun. SIGGRAPH 2017.

Required Components

Repeat

{

Analyze bad elements

Scale-space frame field

Global solve & insert sheets

Predict/learn parameters/constraints

Update mapping

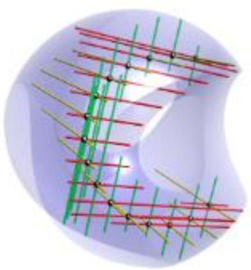
Minimize/bound distortion

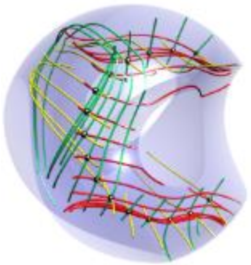
Interpolant / higher-order

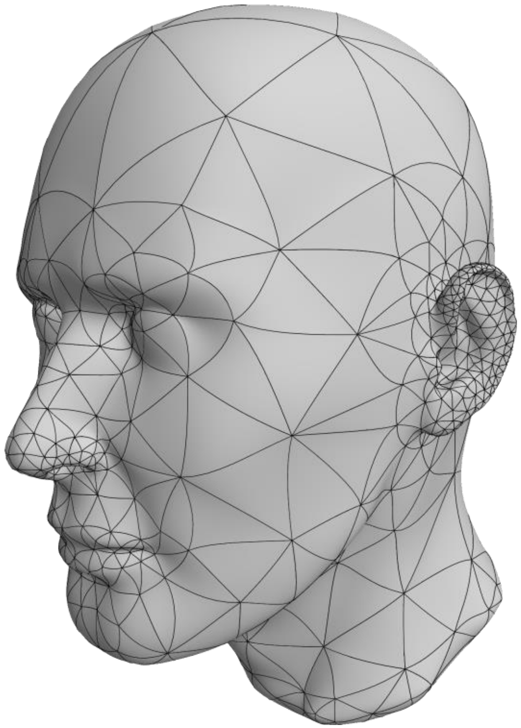
Guarantee valid embedding

Update restricted elements

}







Feng, A., Busé, Delingette, Desbrun. Curved Optimal Delaunay Triangulation. SIGGRAPH 2018.

Required Components

Repeat

{

Analyze bad elements

Scale-space frame field

Global solve & insert sheets

Predict/learn parameters/const

Update mapping

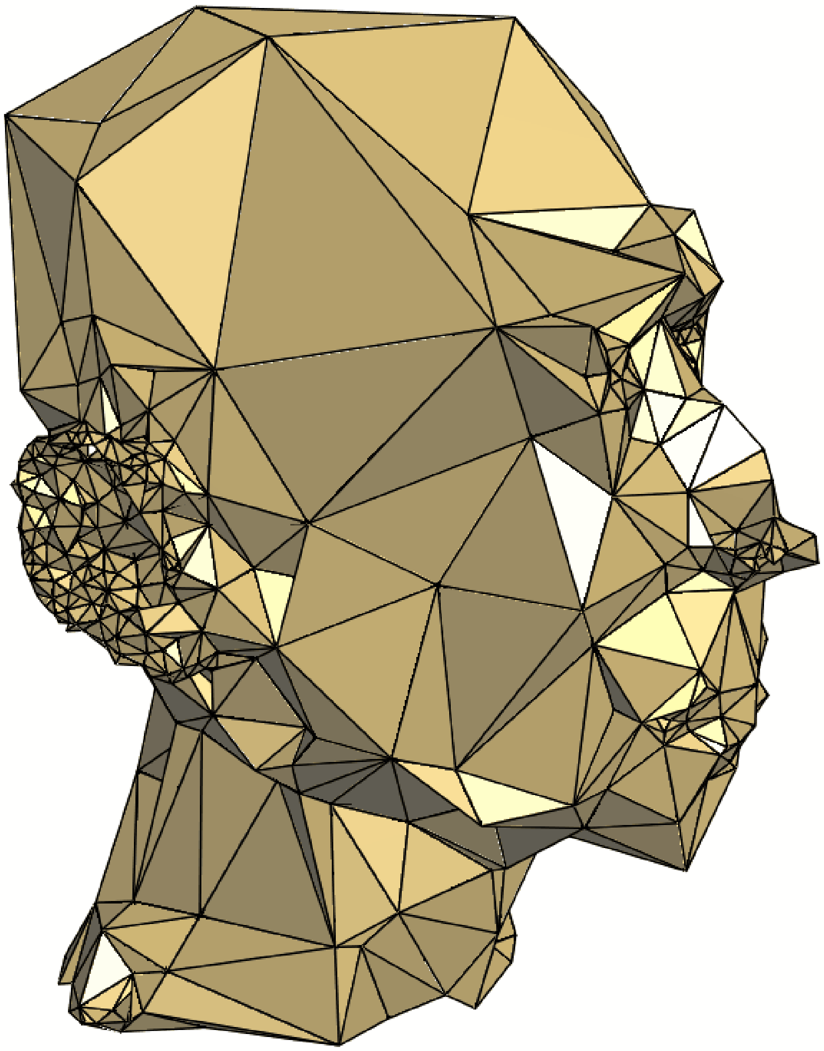
Minimize/bound distortion

Interpolant / higher-order

Guarantee valid embedding

Update restricted elements

}



Required Components

Analyze bad elements

- Scale-space frame field

Global solve & insert sheets

- Predict/learn parameters/constraints

Update mapping

- Minimize/bound distortion
- Interpolant / higher-order (h/p dilemma)
- Guarantee valid embedding
- Symmetry/structure preserving

Update restricted elements

Optimize

Shi, A., Desbrun, Bao, Huang.
Symmetry and Orbit Detection via Lie-Algebra Voting.
EUROGRAPHICS Symposium of Geometry Processing, 2016

Tournois, Wormser, A., Desbrun.
Interleaving Delaunay refinement and optimization for practical isotropic tetrahedron mesh generation.
SIGGRAPH 2009.

SUMMARY & OUTLOOK

Summary & Outlook

- Selective padding
- Warm-up with sheet-based meshing
- Objectives
  - Pure-hex meshing
  - Non-local refinement & optimization
  - Level-of-detail meshing
  - Optimize / given simulation operator
  - Mesh order/refinement dilemma
  - Seek for ~~good excellent~~ jet fighter student or post-doc

Thank you.

Pierre Alliez

Inria Sophia Antipolis – Méditerranée

Titane project-team: https://team.inria.fr/titane/

pierre.alliez@inria.fr