Scaffolding skeletons using spherical Voronoi diagrams: feasibility, regularity and symmetry.

A.J. Fuentes Suárez
E. Hubert

Inria Sophia Antipolis Mediterraneé
Université Côte d'Azur

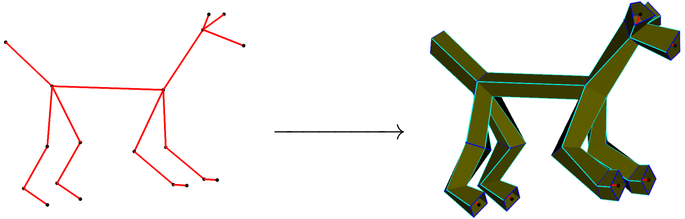
HAL: https://hal.inria.fr/hal-01774909v1

Outline

1. Introduction
2. Scaffolds
3. Models & Existence
4. Algorithms

Outline

Premises



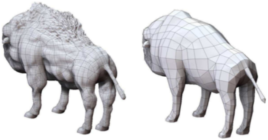
Skeleton Scaffold

Skeleton: finite set of spatial line segments that do not intersect except at endpoints.

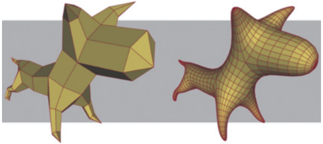
The skeleton S naturally defines a graph $G_S = (\mathcal{E}_S, \mathcal{V}_S)$ embedded in $\mathbb{R}^3$.

Scaffold: coarse quad mesh that *tightly follows* the structure of the skeleton. (informal definition)

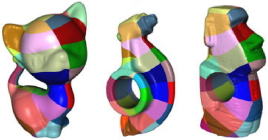
Motivation: an intermediate step in many applications



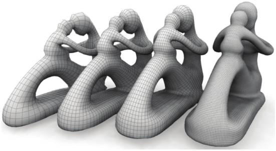
Sculpting [JLW10]



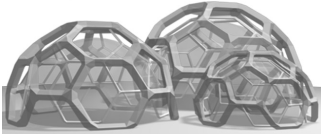
Subdivision surface [BMW12]



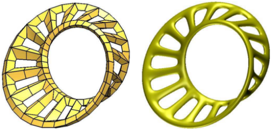
Compatible quadrangulation [YCJL09]



Semi-regular quad meshing [ULP$^+$15]

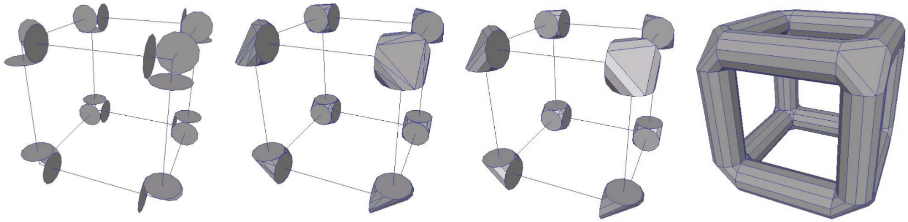


Architecture [SMA05]



Bi-quartic surfaces [KP16]

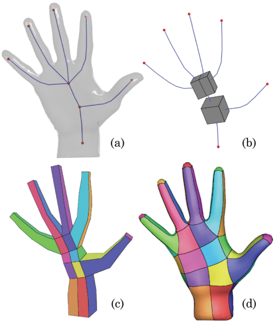
Previous work

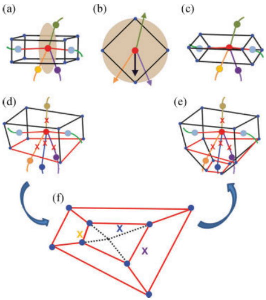


Triangular faces

1. Construct a pipe with a polygonal cross profile, then "stitch" the pipes at the joints [SMA05, JLW10].
2. Partition a cube at joints, then extrude quadrilateral "tubes" connecting the extremities of each edge [YCJL09, ULP+15].
3. Partition a sphere at joints into cells, then construct a tubular structure connecting the two cells of each edge [BMW12].

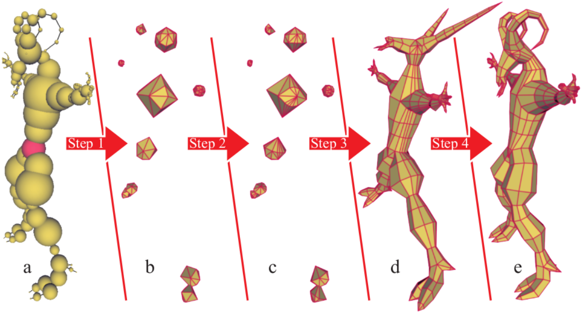
Previous work





1. Construct a pipe with a polygonal cross profile, then "stitch" the pipes at the joints [SMA05, JLW10].
2. Partition a cube at joints, then extrude quadrilateral "tubes" connecting the extremities of each edge [YCJL09, ULP+15].
3. Partition a sphere at joints into cells, then construct a tubular structure connecting the two cells of each edge [BMW12].

Previous work

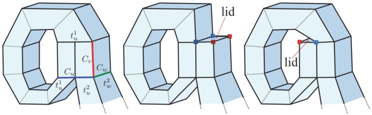


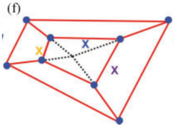
1. Construct a pipe with a polygonal cross profile, then "stitch" the pipes at the joints [SMA05, JLW10].
2. Partition a cube at joints, then extrude quadrilateral "tubes" connecting the extremities of each edge [YCJL09, ULP+15].
3. Partition a sphere at joints into cells, then construct a tubular structure connecting the two cells of each edge [BMW12].

Difficulties in previous methods

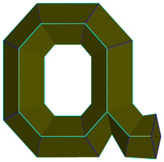
- Cycles.
- Symmetries.
- Optimality.

Usai *et al.* [ULP^{+}15] & Yao *et al.* [YCJL09]:
"lids", spurious quads around joints





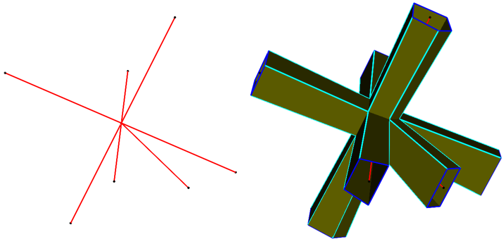
Problem: "lid" position, extra quads



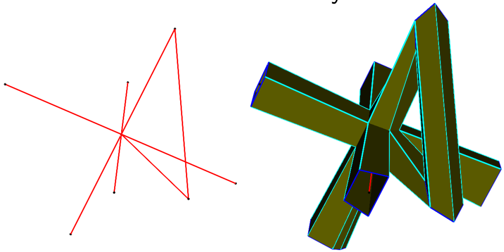
Our solution

Difficulties in previous methods

Skeleton without cycles



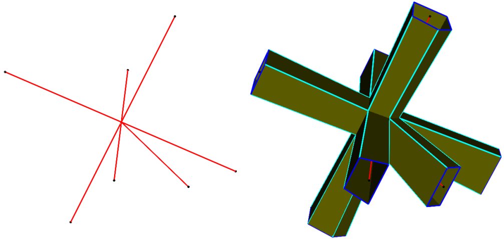
Skeleton with a cycle



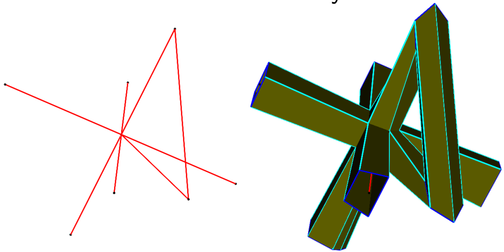
- Cycles.
- Symmetries.
- Optimality.

Difficulties in previous methods

Skeleton without cycles



Skeleton with a cycle

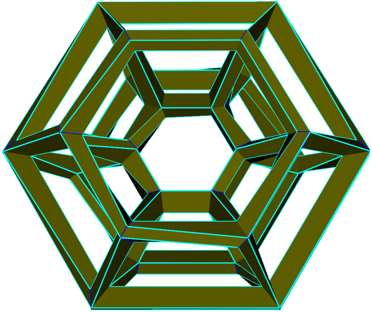


- Cycles.
- Symmetries.
- Optimality.

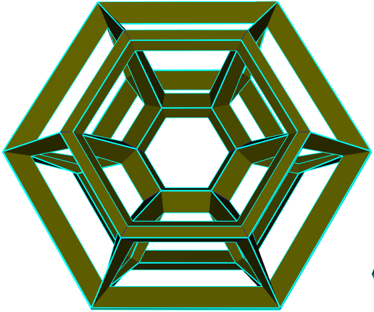
Contributions

Why symmetry?

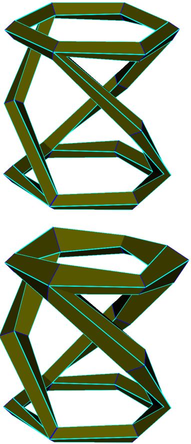
Scaffold
(no symmetries)



Symmetric scaffold
(many symmetries)



Rotation symmetry



∗Similar scaffolds were used in [KP16] for the construction of bi-quartic surfaces.

Scaffolding method: the outline

Skeleton to Quad dominant Mesh (SQM) method [BMW12].

Construction of a scaffold as a three-step process

1. Partition spheres at joints, one region per incident edge.
   - spherical Voronoi diagrams.
2. Discretize regions into cells (points on the boundary).
   - subdivide the boundary of a Voronoi region into a polyline.
3. Link points on the cells relative to the same edge.
   - pair points minimizing length.

Constraint

The two cells of every edge must have the same number of points.

Scaffolding method: the outline

Skeleton to Quad dominant Mesh (SQM) method [BMW12].

Construction of a scaffold as a three-step process

1. Partition spheres at joints, one region per incident edge.
   - spherical Voronoi diagrams.
2. Discretize regions into cells (points on the boundary).
   - subdivide the boundary of a Voronoi region into a polyline.
3. Link points on the cells relative to the same edge.
   - pair points minimizing length.

Constraint

The two cells of every edge must have the same number of points.

Scaffolding method: the outline

Skeleton to Quad dominant Mesh (SQM) method [BMW12].

Construction of a scaffold as a three-step process

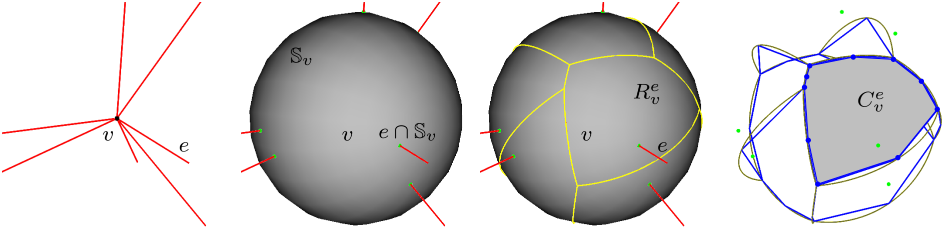
1. Partition spheres at joints, one region per incident edge.
   - spherical Voronoi diagrams.
2. Discretize regions into cells (points on the boundary).
   - subdivide the boundary of a Voronoi region into a polyline.
3. Link points on the cells relative to the same edge.
   - pair points minimizing length.

Constraint

The two cells of every edge must have the same number of points.

Outline

Scaffold cells



- $S \rightsquigarrow G_S(\mathcal{V}_S, \mathcal{E}_S)$
- $v \in \mathcal{V}_S \ e \in \mathcal{E}_S$
- e incident to v

- $\mathbb{S}_v$ sphere centered at v
- $\mathcal{A}_v = \{e \cap \mathbb{S}_v | e \multimap v\}$

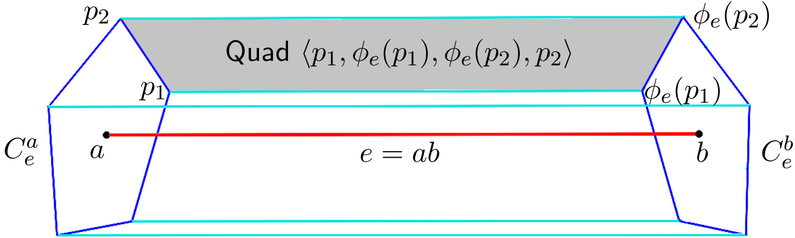
- $\mathrm{Vor}(\mathcal{A}_v)$ Voronoi diagram of $\mathcal{A}_v$
- R_v^e region around $e \cap \mathbb{S}_v$

- C_v^e cell discretization of R_v^e

Scaffold, formalization

Scaffold: is a pair (P_S, Φ_S), such that

1. $P_S = \{\mathcal{C}_v \mid v \in \mathcal{V}_S\}$, where each $\mathcal{C}_v = \{C_e^v \mid e \in \mathcal{E}_S, e \multimap v\}$ is a family of *cells* representing a partition of $\mathbb{S}_v$ according to $\mathrm{Vor}(\mathcal{A}_v)$.
2. $\Phi_S = \{\phi_e \mid e \in \mathcal{E}_S\}$ is a family of bijections ϕ_e between C_e^a and C_e^b for $e = ab$.



$C_e^a = \langle p_0, p_2, \ldots, p_n \rangle$, gives quads $\langle p_i, \phi_e(p_i), \phi_e(p_{i+1}), p_{i+1} \rangle$ $i = 1, 2 \ldots n$.

Regularity & Symmetry

Regular scaffold: all the cells have the same number of points.

Symmetric scaffold: respects all the symmetries $T \in \mathcal{T}_S$.

Skeleton symmetry: an isometry $T : \mathbb{R}^3 \to \mathbb{R}^3$ that maps elements of $G_S(\mathcal{V}_S, \mathcal{E}_S)$ to elements of G_S.

A scaffold respects the skeleton symmetry T if:

- Symmetric cells: $C_{T(e)}^{T(v)} = T(C_e^v)$ for all cells C_e^v.
- Symmetric links: $\phi_{T(e)} = T \circ \phi_e \circ T^{-1}$ for all edges e.

$\mathcal{T}_S$ is the group generated by some symmetries of S.

Regularity & Symmetry

Regular scaffold: all the cells have the same number of points.

Symmetric scaffold: respects all the symmetries $T \in \mathcal{T}_S$.

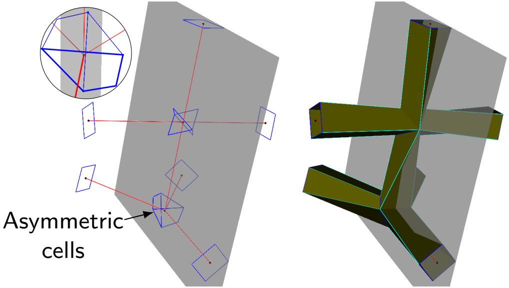
Skeleton symmetry: an isometry $T : \mathbb{R}^3 \to \mathbb{R}^3$ that maps elements of $G_S(\mathcal{V}_S, \mathcal{E}_S)$ to elements of G_S.

A scaffold respects the skeleton symmetry T if:

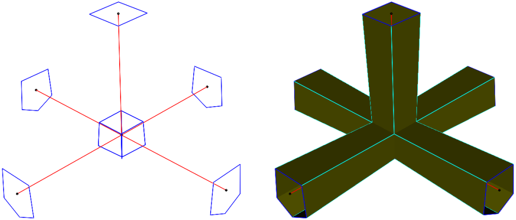
- Symmetric cells: $C_{T(e)}^{T(v)} = T(C_e^v)$ for all cells C_e^v.
- Symmetric links: $\phi_{T(e)} = T \circ \phi_e \circ T^{-1}$ for all edges e.

$\mathcal{T}_S$ is the group generated by some symmetries of S.

Regularity vs Symmetry



Regular asymmetric

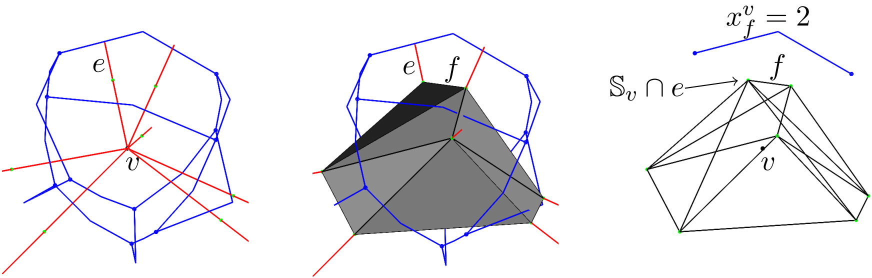


Symmetric irregular.

Outline

Spherical Voronoi diagram and arc subdivisions

- $\mathrm{Del}(\mathcal{A}_v)$: Delaunay triangulation of $\mathcal{A}_v$, dual of $\mathrm{Vor}(\mathcal{A}_v)$, equivalent to the convex hull of $\mathcal{A}_v$ [GM01].
- E_v edges of $\mathrm{Del}(\mathcal{A}_v) \equiv$ arcs on the boundaries of $\mathrm{Vor}(\mathcal{A}_v)$.
- x_f^v number of segments in the subdivision of the arc $f \in E_v$.
- $|C_e^v| = \sum_{\substack{f \in E_v \\ f \multimap (\mathbb{S}_v \cap e)}} x_f^v$ the number of points in the cell C_e^v.



Cells from $\mathrm{Vor}(\mathcal{A}_v)$. Convex hull of $\mathcal{A}_v$. $\mathrm{Del}(\mathcal{A}_v)$.

Compatibility constraints & Model

- $|C_e^a| = |C_e^b|$ for each skeleton edge $e = ab$ (Compatibility constraint).
- $x_f^v \geq 1$ each arc on the boundary of Voronoi regions must be represented by at least one segment.
- $\Lambda_i(x_f^v) \geq s_i$ extra constraints imposed on the minimal number of points in cells or arcs (linear forms on x_f^v with nonnegative coefficients, and constants $s_i > 0$).

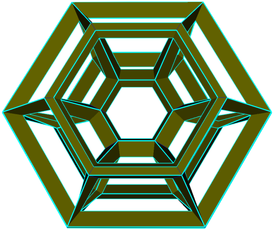
Model

$$\begin{cases} \sum_{\substack{h \in E_a \\ h \multimap (\mathbb{S}_a \cap e)}} x_h^a = \sum_{\substack{g \in E_b \\ g \multimap (\mathbb{S}_b \cap e)}} x_g^b & \forall e = ab \in \mathcal{E}_S. \\ x_f^v \in \mathbb{Z},\ x_f^v \geq 1 & \forall f \in E_v, v \in \mathcal{V}_S \\ \Lambda_i(x_f^v) \geq s_i & i = 1, 2, \ldots \end{cases}$$

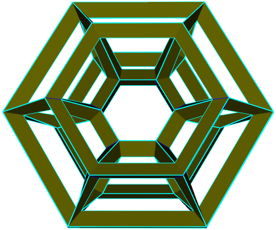
Some choices for Λ_i

- Minimal number of points on cells.

$|C_e^v| \geq 4$
$x_f^v \geq 1$

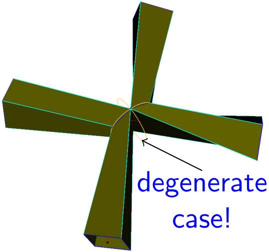


$|C_e^v| \geq 3$
$x_f^v \geq 1$

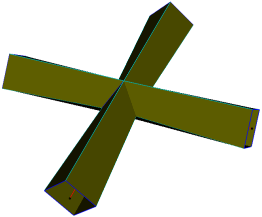


- Minimal number of segments on arcs.

$x_f^v \geq 1$
$|C_e^v| \geq 4$



$x_f^v \geq 2$
$|C_e^v| \geq 4$



Regular scaffold

- $|C_e^v| = q$ for all the skeleton edges e.
- q integer variable, number of points on each cell (cross profile: $q = 4$ quadrilateral, $q = 3$ triangular...)

Model (Regular)

$$\begin{cases} \displaystyle\sum_{\substack{h \in E_v \\ h \multimap (\mathbb{S}_v \cap e)}} x_h^v = q & \forall v \in \mathcal{V}_S, e \in \mathcal{E}_S, e \multimap v. \\ x_f^v, q \in \mathbb{Z},\ x_f^v \geq 1 & \forall f \in E_v, v \in \mathcal{V}_S \\ \Lambda_i(x_f^v) \geq s_i & i = 1, 2, ... \end{cases}$$

Symmetric model

Voronoi diagram commutes with symmetry:

$$\begin{array}{ccc} \mathcal{A}_v & \xrightarrow{\quad T \quad} & \mathcal{A}_{T(v)} \\ \Big\downarrow \text{Vor} & & \Big\downarrow \text{Vor} \\ \text{Vor}(\mathcal{A}_v) & \xrightarrow{T} & T(\text{Vor}(\mathcal{A}_v)) = \text{Vor}(\mathcal{A}_{T(v)}) \end{array}$$

In a symmetric scaffold:

$$x_f^v = x_{T(f)}^{T(v)} \quad \forall T \in \mathcal{T}_S.$$

This condition is also sufficient if the arcs are discretized into equal-length segments.

Existence of scaffolds

Theorem (Existence of regular symmetric scaffolds)

Given a skeleton S admitting the set of symmetries $\mathcal{T}_S$, there exist a solution $(\bar{x}^v_f, \bar{q})$ to

$$\begin{cases} \sum\limits_{\substack{h \in E_v \\ h \multimap (\mathbb{S}_v \cap e)}} x^v_h = q & \forall v \in \mathcal{V}_S, e \in \mathcal{E}_S, e \multimap v. \\ x^v_f, q \in \mathbb{Z},\ x^v_f \geq 1 & \forall f \in E_v, v \in \mathcal{V}_S \\ \Lambda_i(x^v_f) \geq s_i & i = 1, 2, ... \end{cases}$$

Satisfying

$$x^v_f = x^{T(v)}_{T(f)} \qquad \forall\, T \in \mathcal{T}_S, v \in \mathcal{V}_S, f \in E_v.$$

Existence of scaffolds (preliminary results)

Lemma (Locally uniform discretization)

For $v \in \mathcal{V}_S$, the local system

$$\sum_{\substack{f \in E_v \\ f \multimap (\mathbb{S}_v \cap e)}} x_f^v = \lambda_v \quad \forall e \multimap v, e \in \mathcal{E}_S$$

has a solution $(\tilde{x}_f^v, \tilde{\lambda}_v)$ with positive integer entries.

Proof.

- $\mathrm{Del}(\mathcal{A}_v)$ is equivalent to the convex hull of $\mathcal{A}_v$, which is an inscribed polyhedron.
- Positive real solution with $\lambda_v = 1$ is guaranteed by a numerical characterization of graphs of inscribable type due to Rivin [Riv96].
- A homogeneous linear system with integer coefficients has a positive integer solution whenever it has a positive real solution. □

Existence of scaffolds (proof)

Proof of theorem (Existence of scaffolds).

- For each node $v \in \mathcal{V}_S$ take the local solution $(\tilde{x}^v_f, \tilde{\lambda}_v)$ guaranteed by the locally uniform discretization lemma.
- Multiply each local solution by a (different) positive integer such that all the cells have the same number of points and the Λ_i constraints are satisfied.
  - $\hat{x}^v_f = s\frac{\hat{\lambda}}{\tilde{\lambda}_v}\tilde{x}^v_f$, where $\hat{\lambda} = \prod_{u \in \mathcal{V}_S} \tilde{\lambda}_u$ and $s = \max_i s_i$.
  - $|\hat{C}^v_e| = s\hat{\lambda}$ holds, the factors $s\hat{\lambda}/\tilde{\lambda}_v$ guarantee the Λ_i constraints.
- Once we have a regular solution we can symmetrize it by summing over the orbit of symmetries.
  - $\bar{x}^v_f = \sum_{T \in \mathcal{T}_S} \hat{x}^{T(v)}_{T(f)}$, then

$$\bar{x}^{T(v)}_{T(f)} = \sum_{R \in \mathcal{T}_S} \hat{x}^{RT(v)}_{RT(f)} = \sum_{R \in \mathcal{T}_S} \hat{x}^{R(v)}_{R(f)} = \bar{x}^v_f.$$

  - $|\bar{C}^v_e| = \bar{q} = s\hat{\lambda}|\mathcal{T}_S|$.

□

Existence of scaffolds (proof)

Proof of theorem (Existence of scaffolds).

- For each node $v \in \mathcal{V}_S$ take the local solution $(\tilde{x}_f^v, \tilde{\lambda}_v)$ guaranteed by the locally uniform discretization lemma.
- Multiply each local solution by a (different) positive integer such that all the cells have the same number of points and the Λ_i constraints are satisfied.
  - $\hat{x}_f^v = s\frac{\hat{\lambda}}{\tilde{\lambda}_v}\tilde{x}_f^v$, where $\hat{\lambda} = \prod_{u\in\mathcal{V}_S}\tilde{\lambda}_u$ and $s = \max_i s_i$.
  - $|\hat{C}_e^v| = s\hat{\lambda}$ holds, the factors $s\hat{\lambda}/\tilde{\lambda}_v$ guarantee the Λ_i constraints.
- Once we have a regular solution we can symmetrize it by summing over the orbit of symmetries.
  - $\bar{x}_f^v = \sum_{T\in\mathcal{T}_S}\hat{x}_{T(f)}^{T(v)}$, then
  $$\bar{x}_{T(f)}^{T(v)} = \sum_{R\in\mathcal{T}_S}\hat{x}_{RT(f)}^{RT(v)} = \sum_{R\in\mathcal{T}_S}\hat{x}_{R(f)}^{R(v)} = \bar{x}_f^v.$$
  - $|\bar{C}_e^v| = \bar{q} = s\hat{\lambda}|\mathcal{T}_S|$.

□

Outline

Computing a scaffold using Integer Linear Programming

- Optimal solution minimizing the number of quads can be found using Integer Linear Programming with the objective function:

$$Q = \sum_{v \in (\mathcal{V}_S - L_S)} \sum_{f \in E_v} 2x_f^v + \sum_{v \in L_S} \sum_{f \in E_v} x_f^v,$$

L_S nodes of G_S with only one incident edge.

- Number of quads in the scaffold given by

$$\tfrac{1}{2}\left(Q + \Xi_S\right),$$

Ξ_S a constant that only depends on the skeleton S.

- Existence of Scaffolds guarantees a solution.

General algorithm for constructing a scaffold

Input: The set of nodes $\mathcal{V}_S$ and edges $\mathcal{E}_S$ of the skeleton.
Output: The quads that represent a scaffold.

1. For each node $v \in \mathcal{V}_S$ define:
   - $\mathcal{A}_v$ the intersection of $\mathbb{S}_v$ with edges incident to v.
   - $\mathcal{H}_v$ convex hull of $\mathcal{A}_v$.
   - E_v edges of $\mathcal{H}_v$.
2. Define and solve the linear program on x_f^v that gives compatible cells.
3. Compute points on each cell. (Subdivide arcs into equal-length cords)
4. Define bijections of linked cells. (Minimizing total length)
5. Output the quads of the scaffold:
   - For each edge $e = ab$, let $C_e^a = \langle p_0, p_2, \ldots, p_n \rangle$, for $i = 1, 2 \ldots n$: Output quad $\langle p_i, \phi_e(p_i), \phi_e(p_{i+1}), p_{i+1} \rangle$

Algorithm to compute subdivisions

Input: Nodes $\mathcal{V}_S$ and edges $\mathcal{E}_S$, along with E_v for each $v \in \mathcal{V}_S$.
Output: x_f^v representing the subdivisions for each arc.

1. Initialize the linear program IP.
2. For each node $v \in \mathcal{V}_S$, and edge $f \in E_v$:
   - Add integer variable x_f^v to IP with restriction $x_f^v \geq 1$.
   - If the arc associated to x_f^v has length $\geq \frac{5\pi}{6}$: Add restriction $x_f^v \geq 2$.
3. For each cell C_e^v:
   - Add: $\sum_{\substack{f \in E_v \\ f \multimap (e \cap \mathbb{S}_v)}} x_f^v \geq 4$. (At least quadrangular cross profile)
4. For each edge $e \in \mathcal{E}_S$:
   - Add: $\sum_{\substack{g \in E_a \\ g \multimap (e \cap \mathbb{S}_v)}} x_g^a = \sum_{\substack{h \in E_b \\ h \multimap (e \cap \mathbb{S}_v)}} x_h^b$. (For regularity two eq. with RHS q)
   - For symmetry there are other extra restrictions.
5. Solve IP minimizing: $\sum_{v \in (\mathcal{V}_S - L_S)} \sum_{f \in E_v} 2x_f^v + \sum_{v \in L_S} \sum_{f \in E_v} x_f^v$.

Thank you

ARCADES

Marie Skłodowska-Curie Actions

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 675789.

References

[BMW12] J.A. Bærentzen, M.K. Misztal, and K. Wełnicka, *Converting skeletal structures to quad dominant meshes*, Computers & Graphics **36** (2012), no. 5, 555–561, Shape Modeling International (SMI) Conference 2012.

[GM01] Clara I. Grima and Alberto Márquez, *Computational geometry on surfaces*, Springer Netherlands, Dordrecht, 2001.

[JLW10] Zhongping Ji, Ligang Liu, and Yigang Wang, *B-mesh: A modeling system for base meshes of 3d articulated shapes*, Computer Graphics Forum **29** (2010), no. 7, 2169–2177.

[KP16] K. Karčiauskas and J. Peters, *Curvature continuous bi-4 constructions for scaffold- and sphere-like surfaces*, CAD Computer Aided Design **78** (2016), 48–59.

[PRZ17] Julian Panetta, Abtin Rahimian, and Denis Zorin, *Worst-case stress relief for microstructures*, ACM Transactions on Graphics **36** (2017), no. 4, 1–16.

[PZM+15] Julian Panetta, Qingnan Zhou, Luigi Malomo, Nico Pietroni, Paolo Cignoni, and Denis Zorin, *Elastic textures for additive fabrication*, ACM Transactions on Graphics **34** (2015), no. 4, 135:1–135:12.

[Riv96] Igor Rivin, *A characterization of ideal polyhedra in hyperbolic 3-space*, Annals of Mathematics **143** (1996), 51–70.

[SMA05] Vinod Srinivasan, Esan Mandal, and Ergun Akleman, *Solidifying wireframes*, Bridges: Mathematical Connections in Art, Music, and Science 2004 (Banf), 2005.

[ULP+15] Francesco Usai, Marco Livesu, Enrico Puppo, Marco Tarini, and Riccardo Scateni,

Implementation

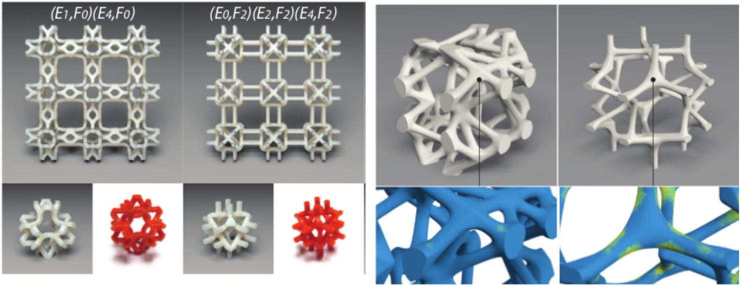
Libraries

Skelton: C++, (LGPL)
`https://gitlab.inria.fr/afuentes/skelton`
PySkelton: Python, (AGPL)
`https://gitlab.inria.fr/afuentes/pyskelton`

General workflow:

- Create or load a skeleton graph.
- Define properties of the scaffolder (min cell quads, max arc angle, symmetries).
- Compute scaffold.
- Show/Save output.

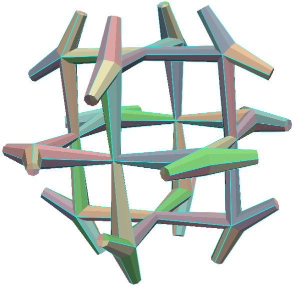
Material design

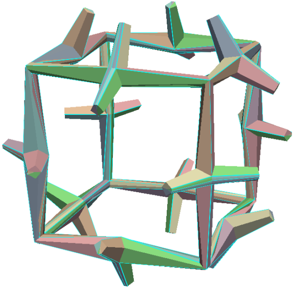


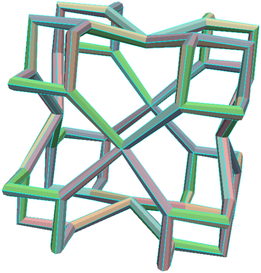
(From [PZM$^+$15, PRZ17])

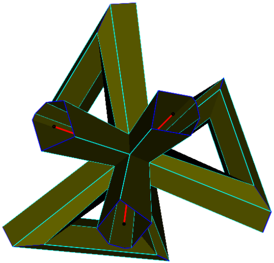
- Highly symmetric shape.
- Surface mesh.
- Volumetric mesh.

Material design

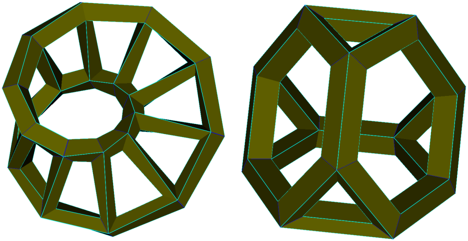




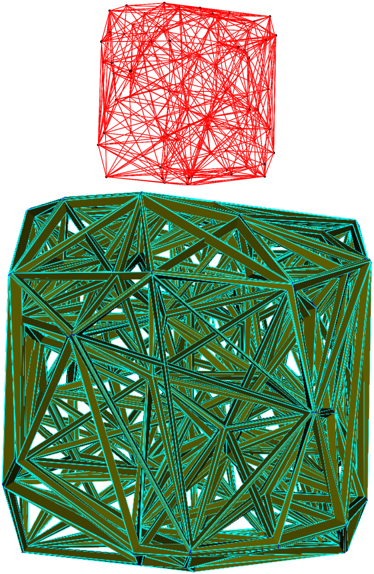




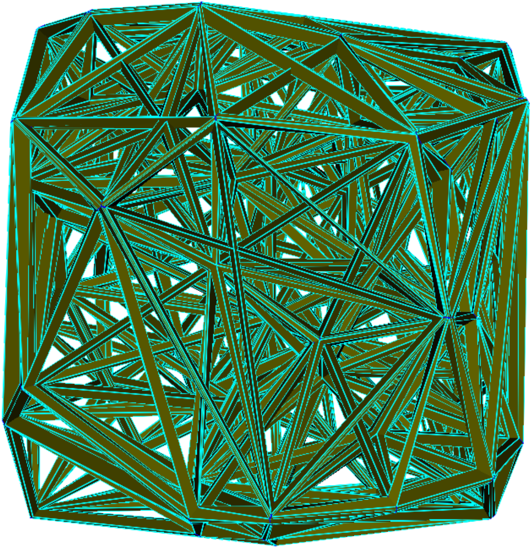
Rotation symmetry



Other skeletons with cycles



Random skeleton



Scaffold

Timings	
Total	1398 ms
LP Solver	37 ms
Convex Hulls	209 ms
Other	1152 ms
LP Solver (reg.)	62 ms

Scaffold	
Nodes	100
Edges	605
Quads	3327
Quads (reg.)	13310
Cross prof. (reg.)	22 sides

Linear Programs	
Variables	2932
Equations	605
Variables (reg.)	2933
Equations (reg.)	1210