CGTA 2025
CONFERENCE ON GEOMETRY: THEORY AND APPLICATIONS

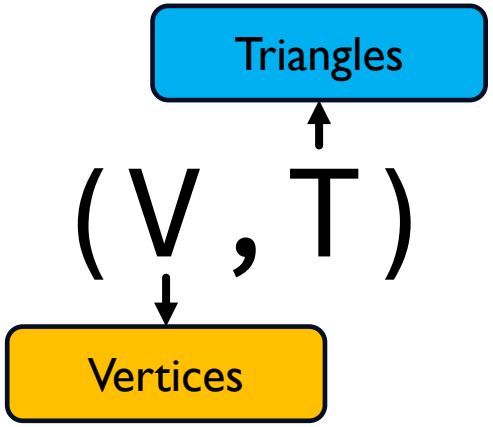
ELTE | IK
INFORMATIKAI KAR

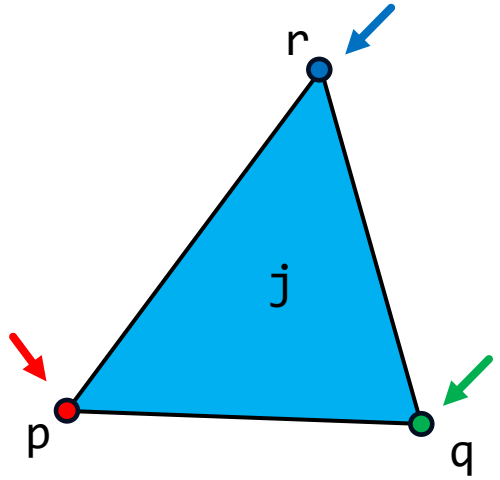
EKÖP
Egyetemi Kutatói Ösztöndíj Program

ENHANCED MESH SLICING

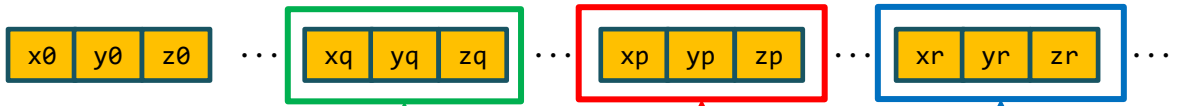
GÁBOR FÁBIÁN @ DEPARTMENT OF NUMERICAL ANALYSIS
ELTE EÖTVÖS LORÁND UNIVERSITY, BUDAPEST, HUNGARY

FACE-VERTEX MESHES

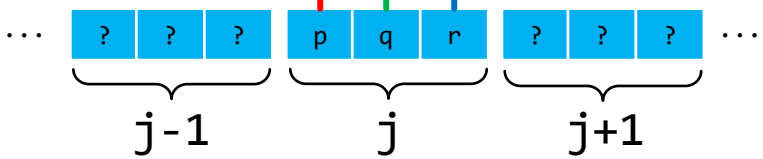




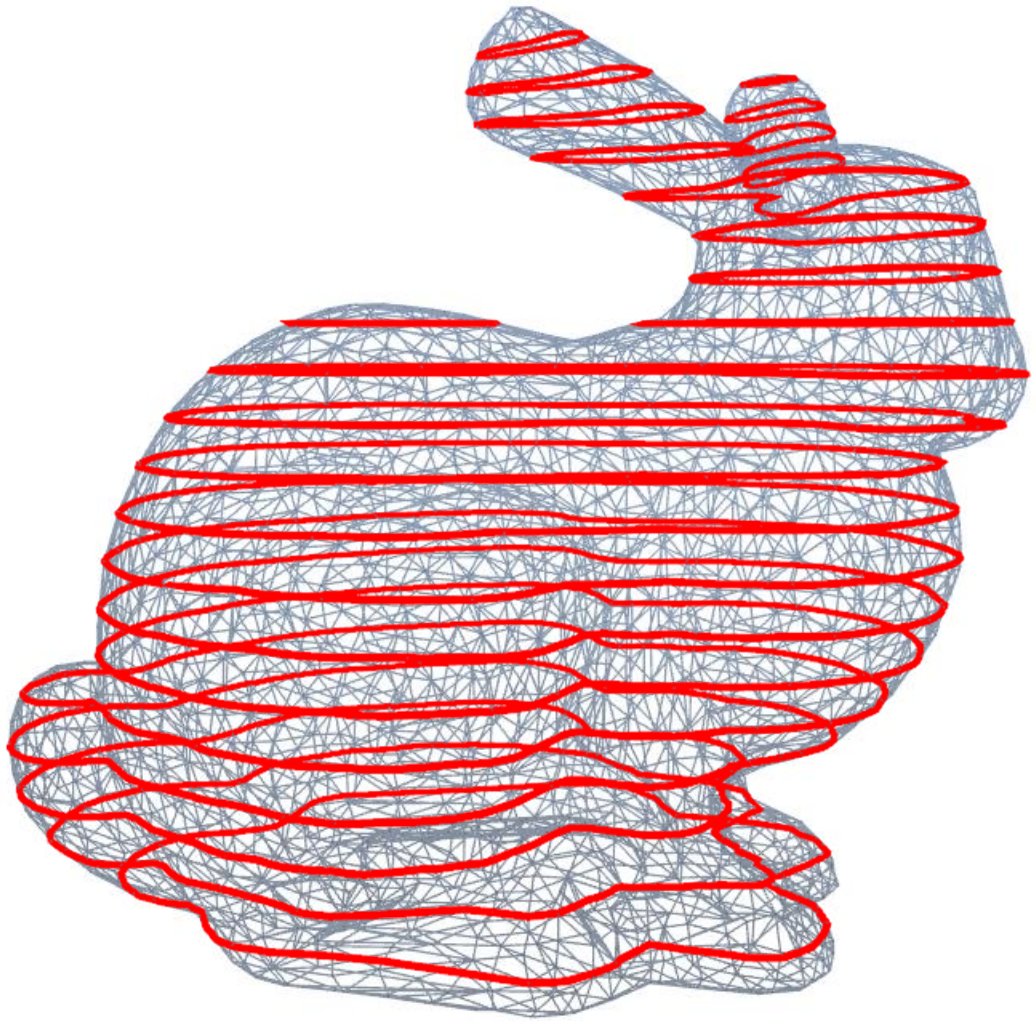
VertexArray



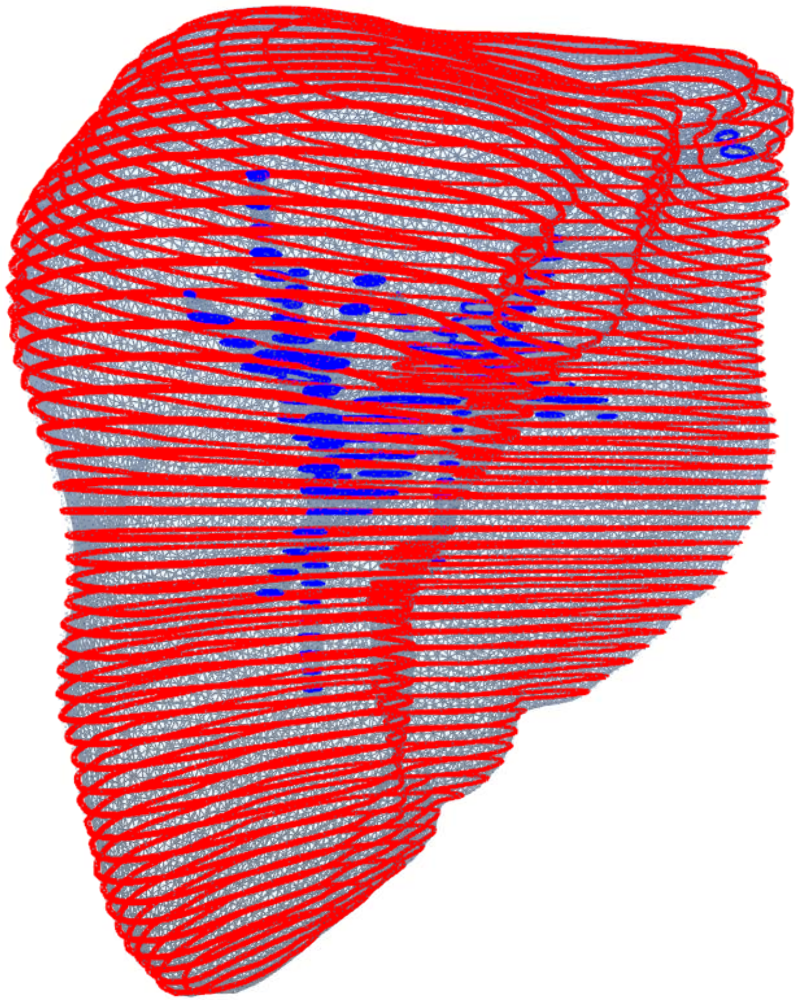
IndexArray



MESH SLICING: BUNNY



MESH SLICING: LIVER



UNIFORM MESH SLICING PROBLEM

- Mesh

$$(V, T)$$

$$n = |T|$$

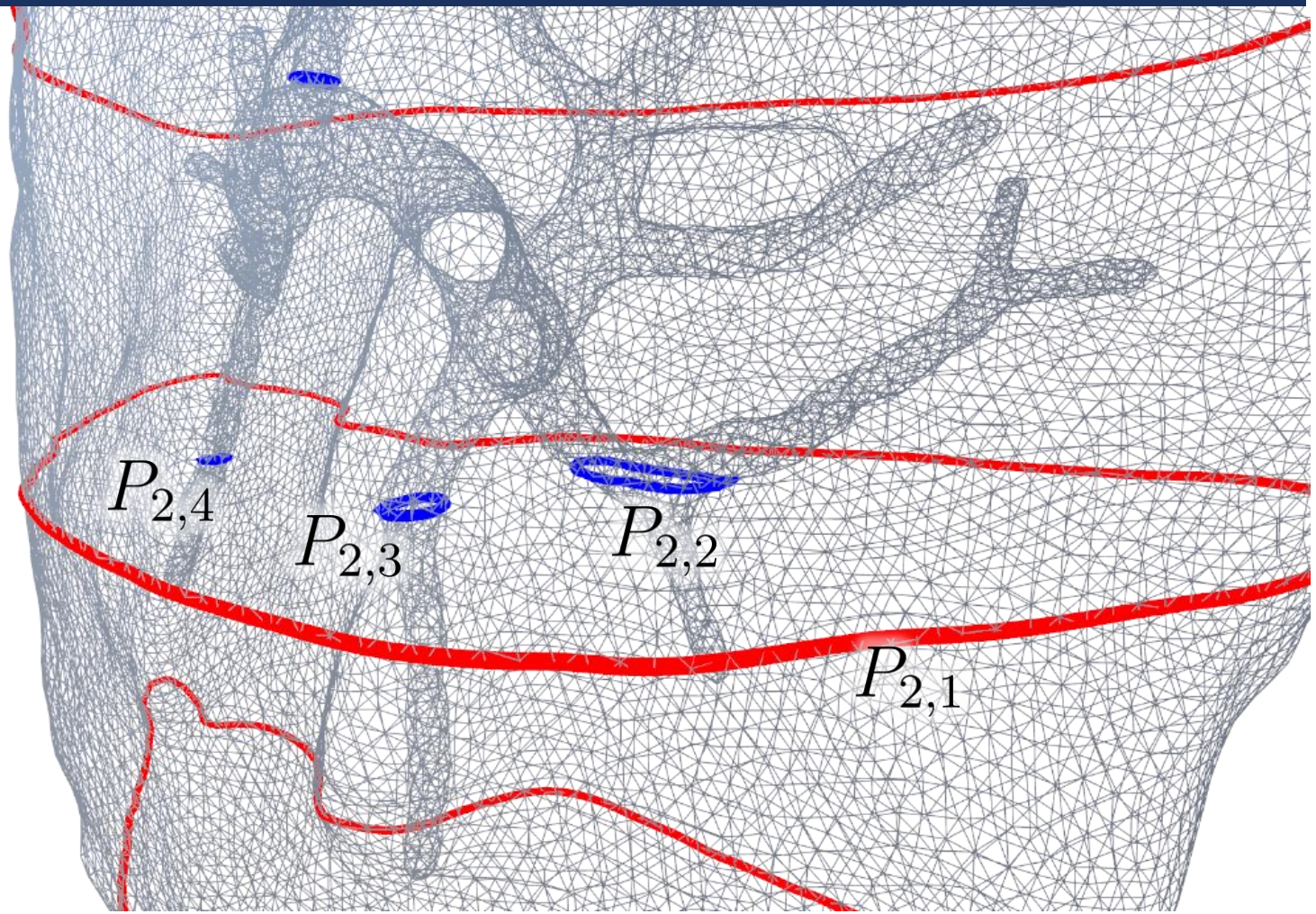
- Parallel planes

$$y = y_0 + ih \quad (i = 1, \ldots, k)$$

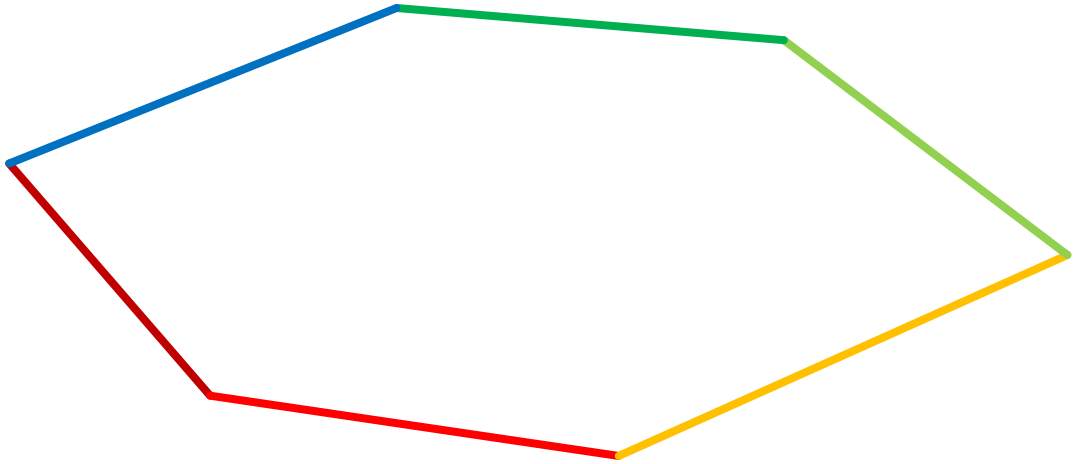
- Closed, **oriented** contours

$$P_{i,j} \in \mathbb{R}^{3 \cdot m_{i,j}}$$

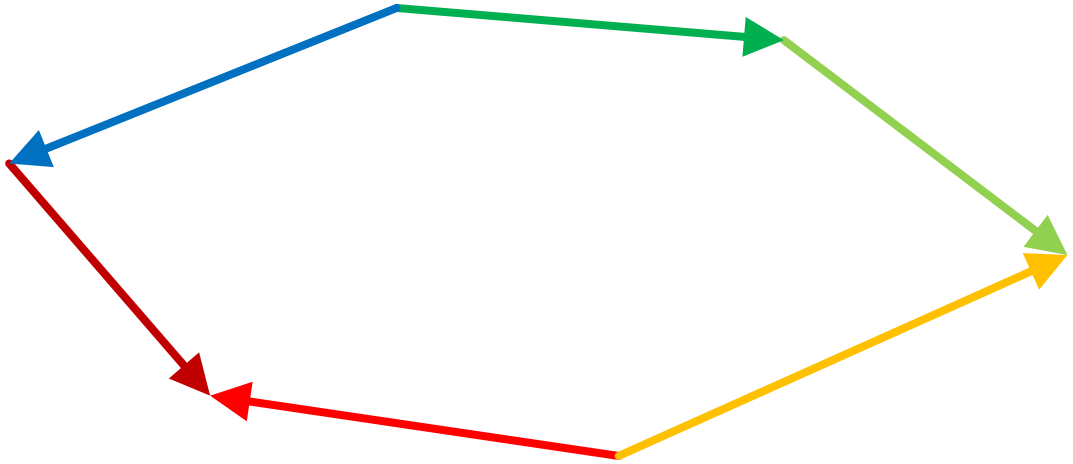
$$m = \sum_{i,j} m_{i,j}$$



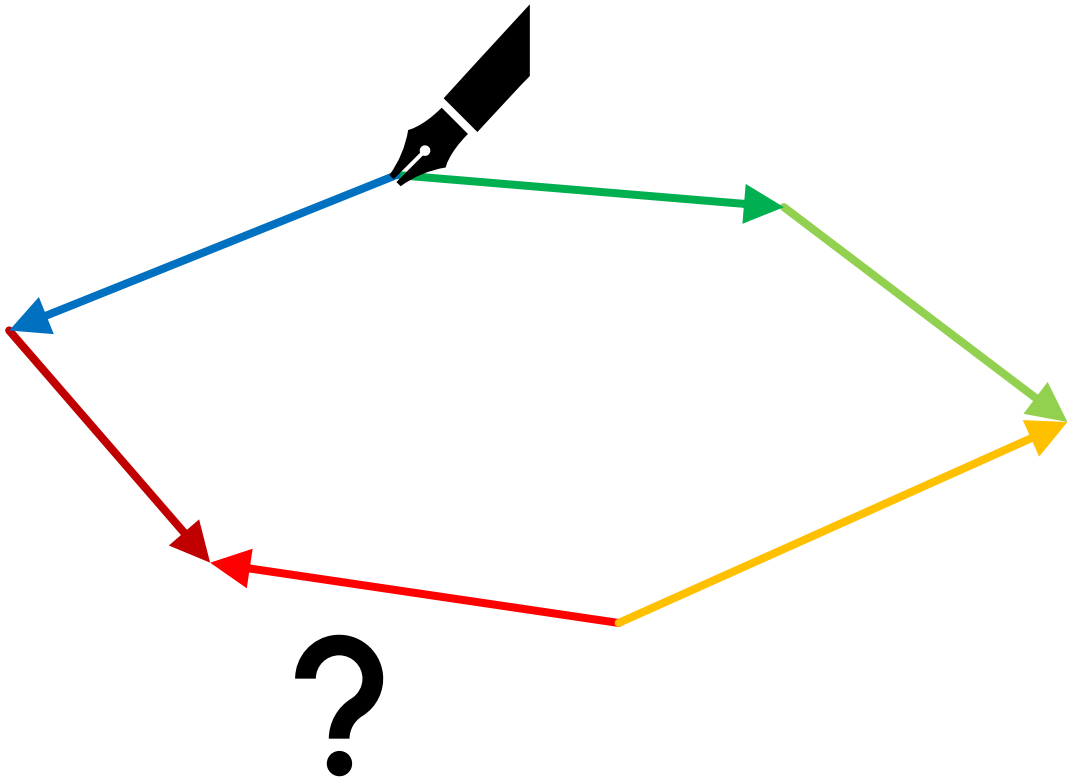
CONTOUR CONSTRUCTION



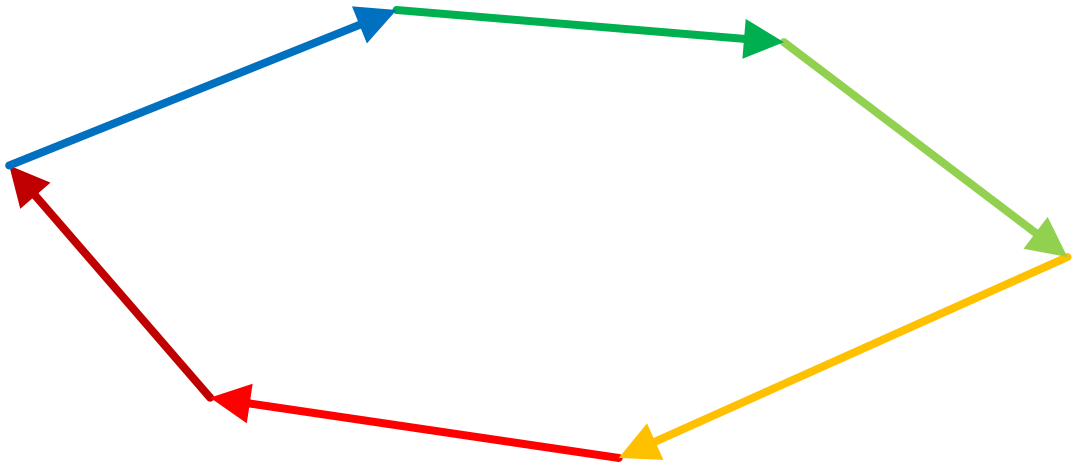
CONTOUR CONSTRUCTION



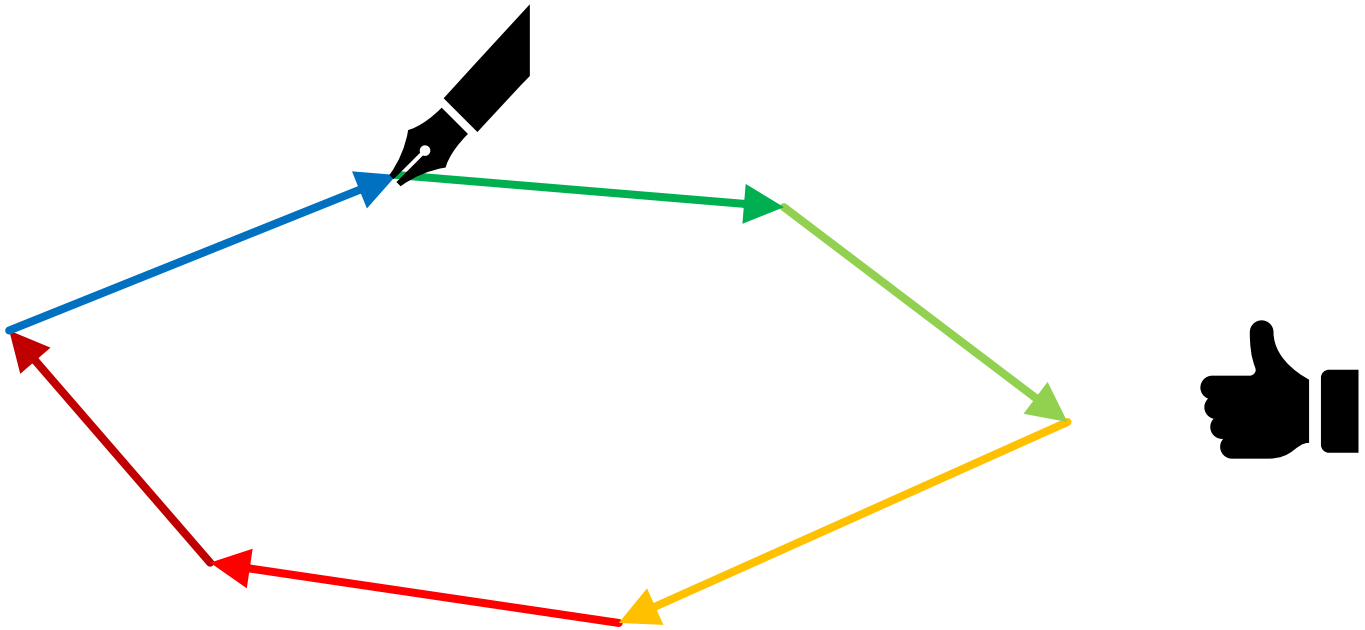
CONTOUR CONSTRUCTION



CONTOUR CONSTRUCTION



CONTOUR CONSTRUCTION



MINETTO'S ALGORITHM FOR UNIFORM MESH SLICING

Minetto, R. et al.: An optimal algorithm for 3D triangle mesh slicing, *Computer-Aided Design*, 92, pp 1—10, 2017.

- 1. Build triangle lists → $\mathcal{O}(n+k)$
- 2. Plane sweep
  - Insert triangles to the active set
  - Compute plane-triangle intersections → $\mathcal{O}(m)$
  - Add segments to the segment list
  - Remove triangles from the active set
- 3. Contour construction → $\mathcal{O}(m)$
- 4. Set orientation of contours → $\mathcal{O}(m)$

$$\mathcal{O}(n+m+k)$$

MINETTO'S ASSUMPTIONS

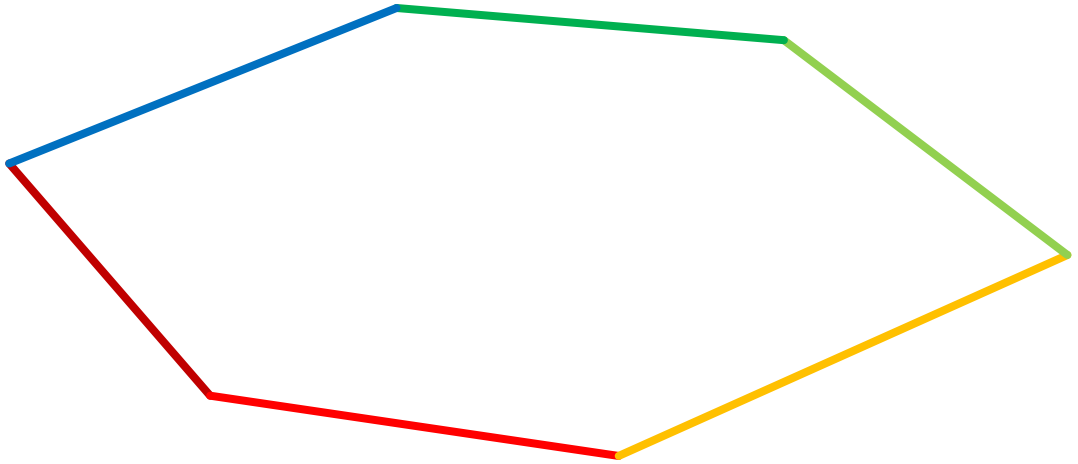
- The mesh is a closed orientable 2-manifold without self-intersections.
- Planes don't hit vertices of the mesh.

- Plane-triangle intersections are **segments**.
- Plane-mesh intersections are closed loops: **contours**.

CONTOUR CONSTRUCTION – MINETTO

Hashing

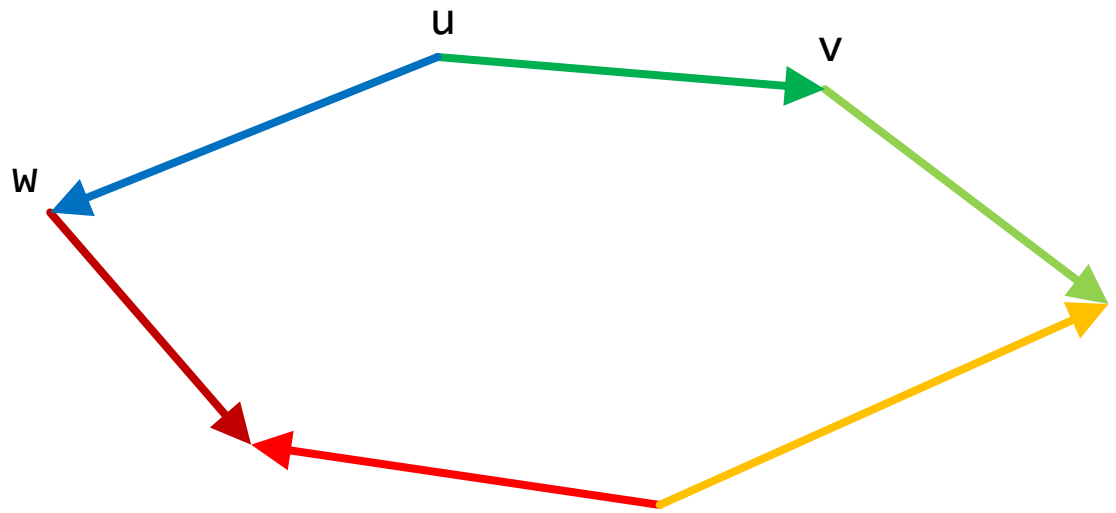
- keys: vertices
- values: adjacent vertices



CONTOUR CONSTRUCTION – MINETTO

Hashing

- keys: vertices
- values: adjacent vertices

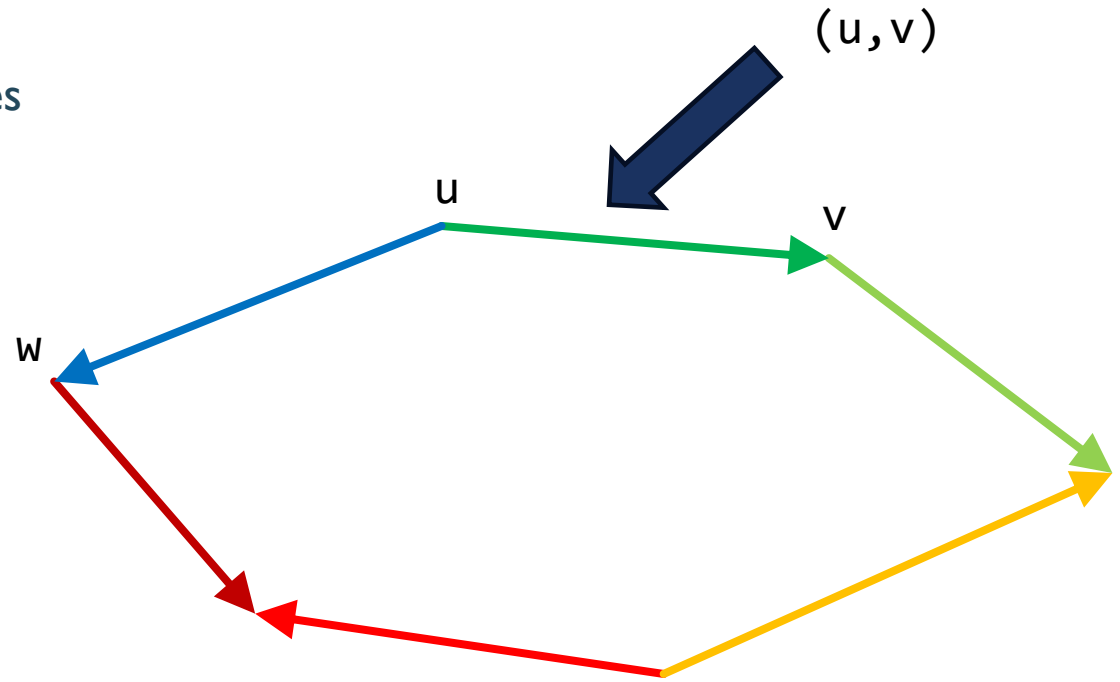



```
H[u] = ( , )
H[v] = ( , )
H[w] = ( , )
```

CONTOUR CONSTRUCTION – MINETTO

Hashing

- keys: vertices
- values: adjacent vertices

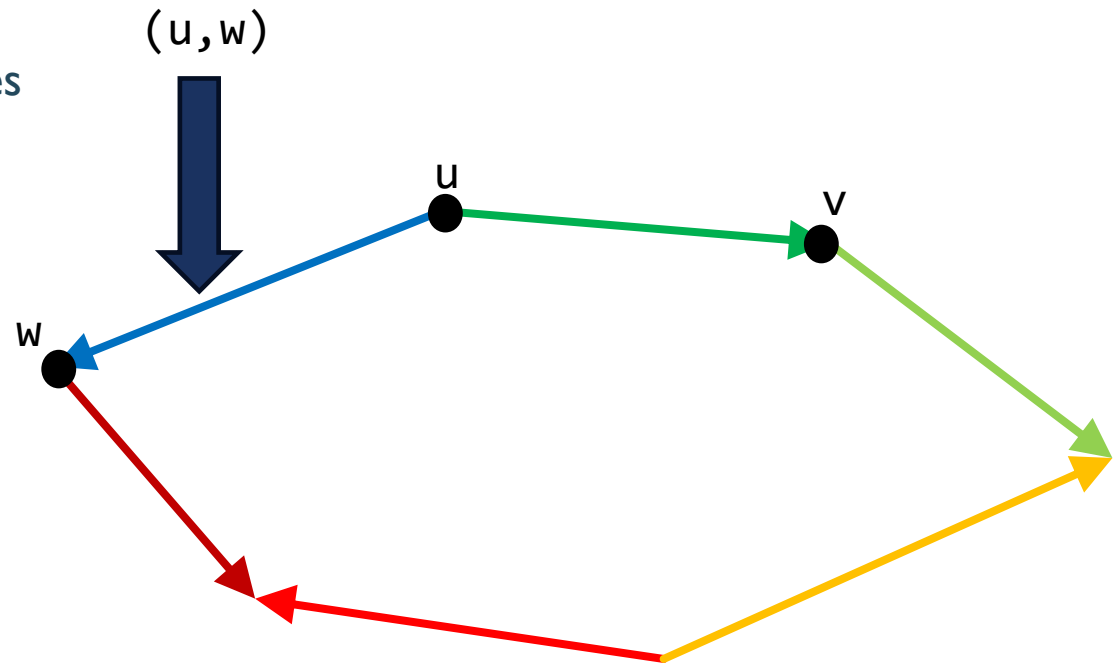



```
H[u] = (v, )
H[v] = (u, )
H[w] = ( , )
```

CONTOUR CONSTRUCTION – MINETTO

Hashing

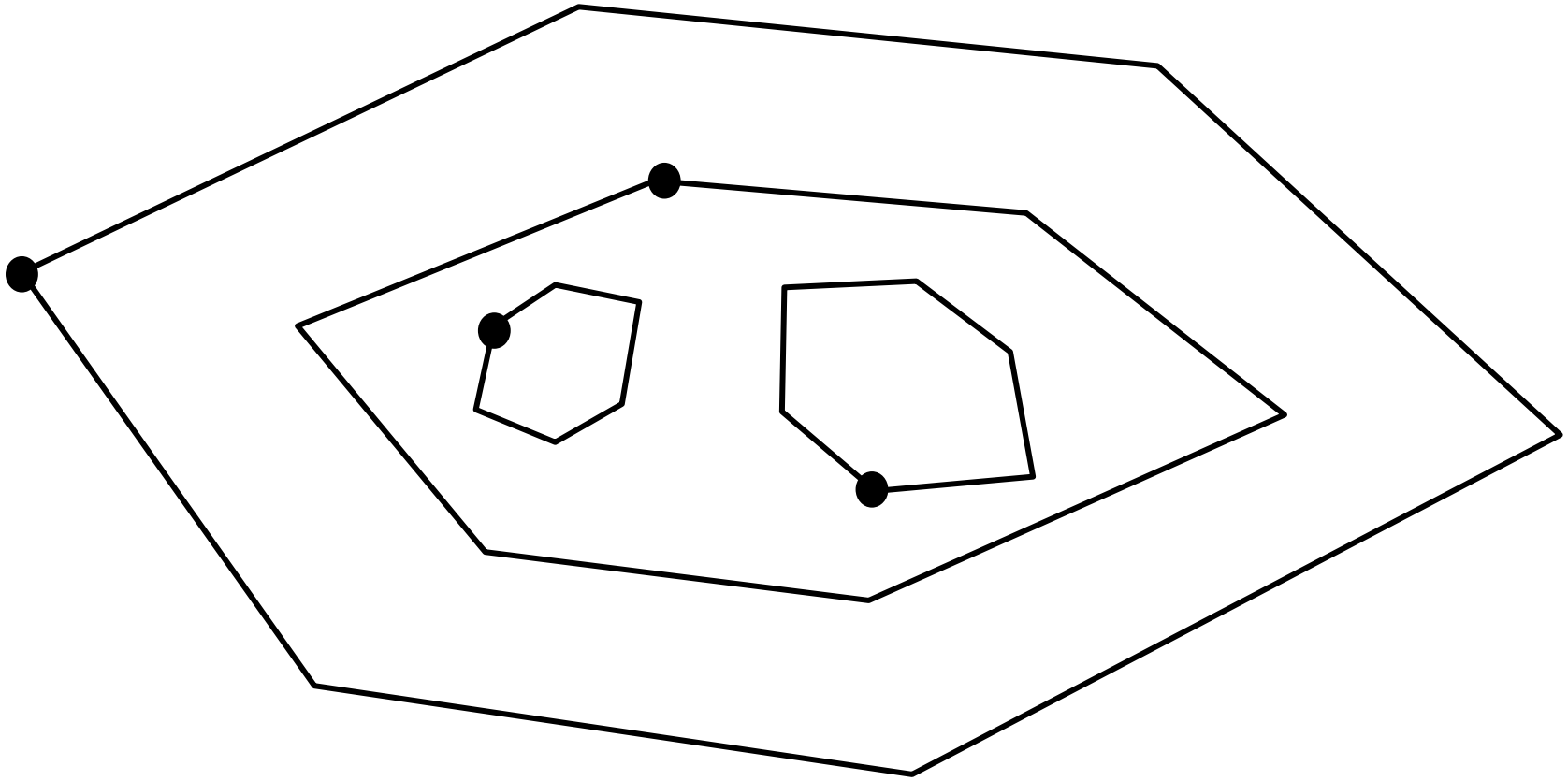
- keys: vertices
- values: adjacent vertices




```
H[u] = (v,w)
H[v] = (u, )
H[w] = (u, )
```

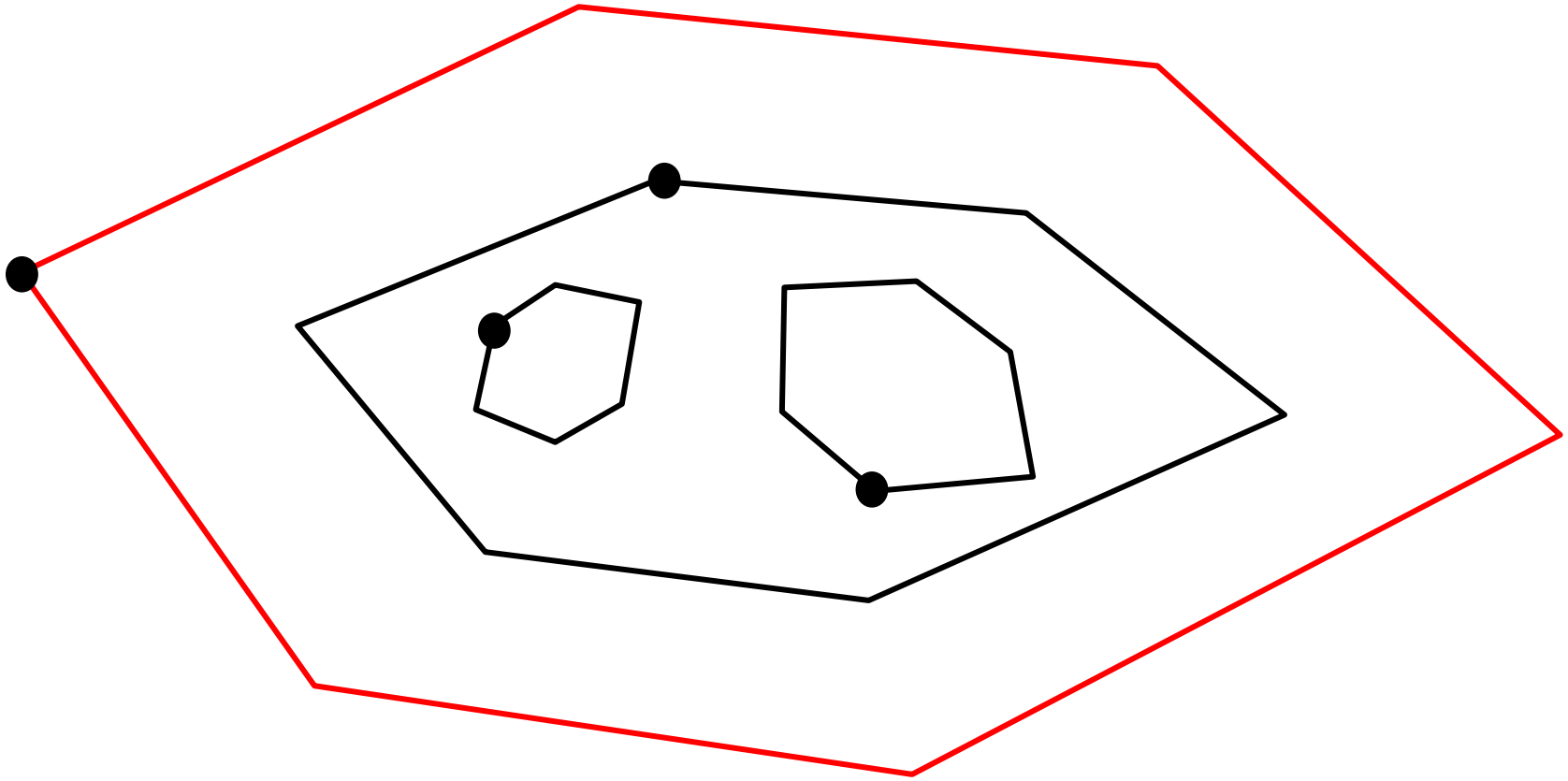
ORIENTATION SETTING – MINETTO

Point-in-polygon algorithms…



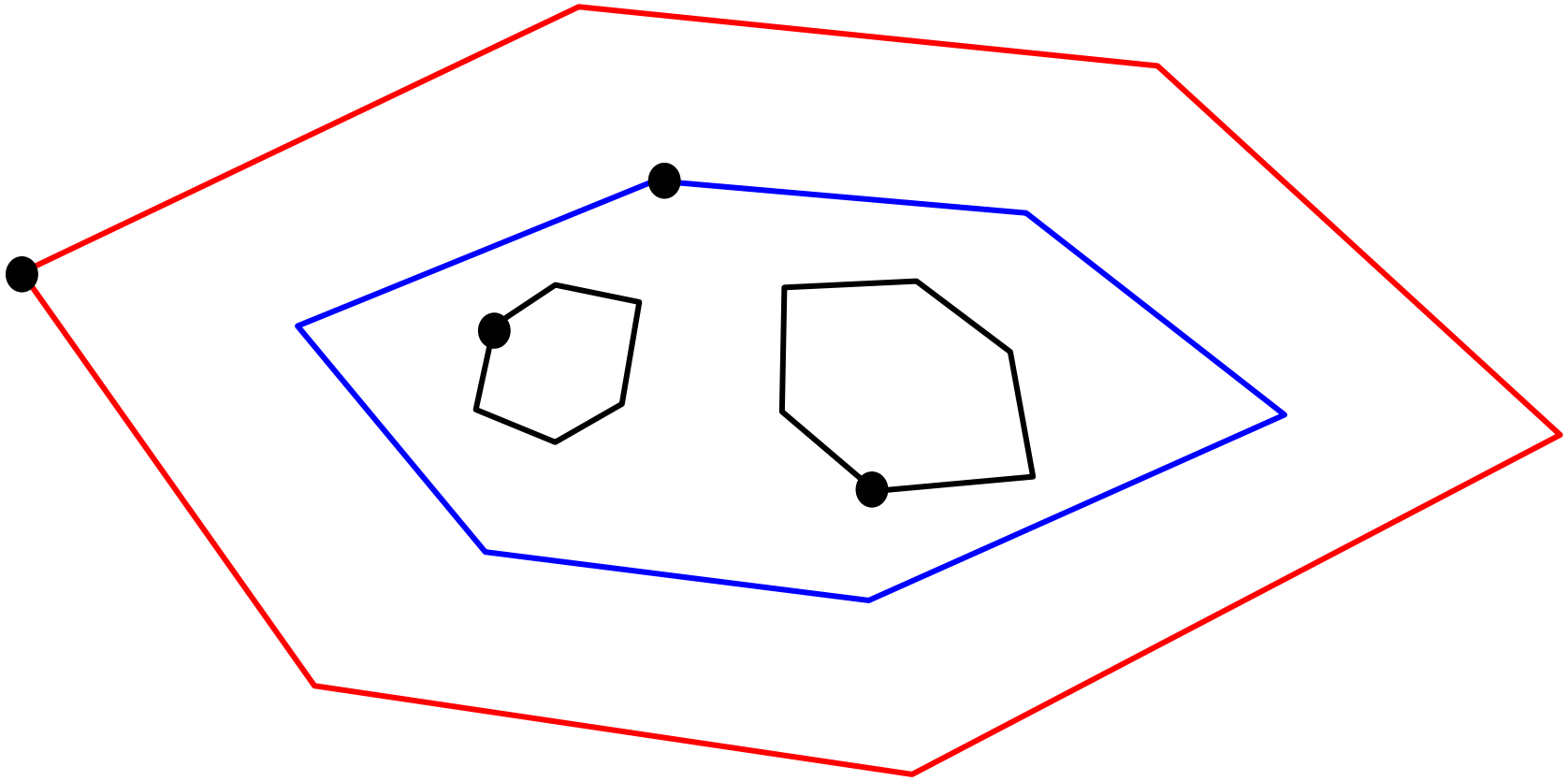
ORIENTATION SETTING – MINETTO

Point-in-polygon algorithms…



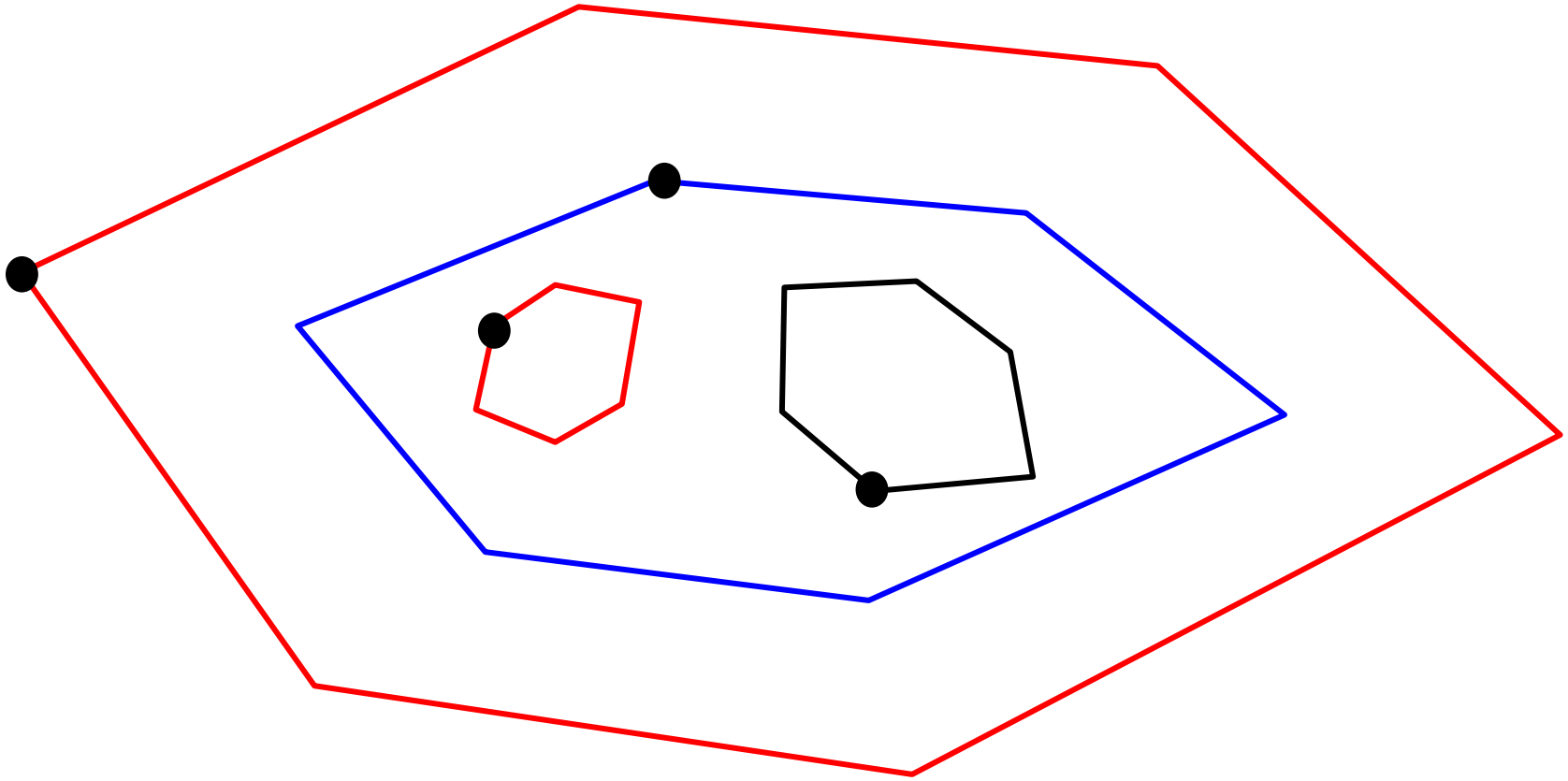
ORIENTATION SETTING – MINETTO

Point-in-polygon algorithms…



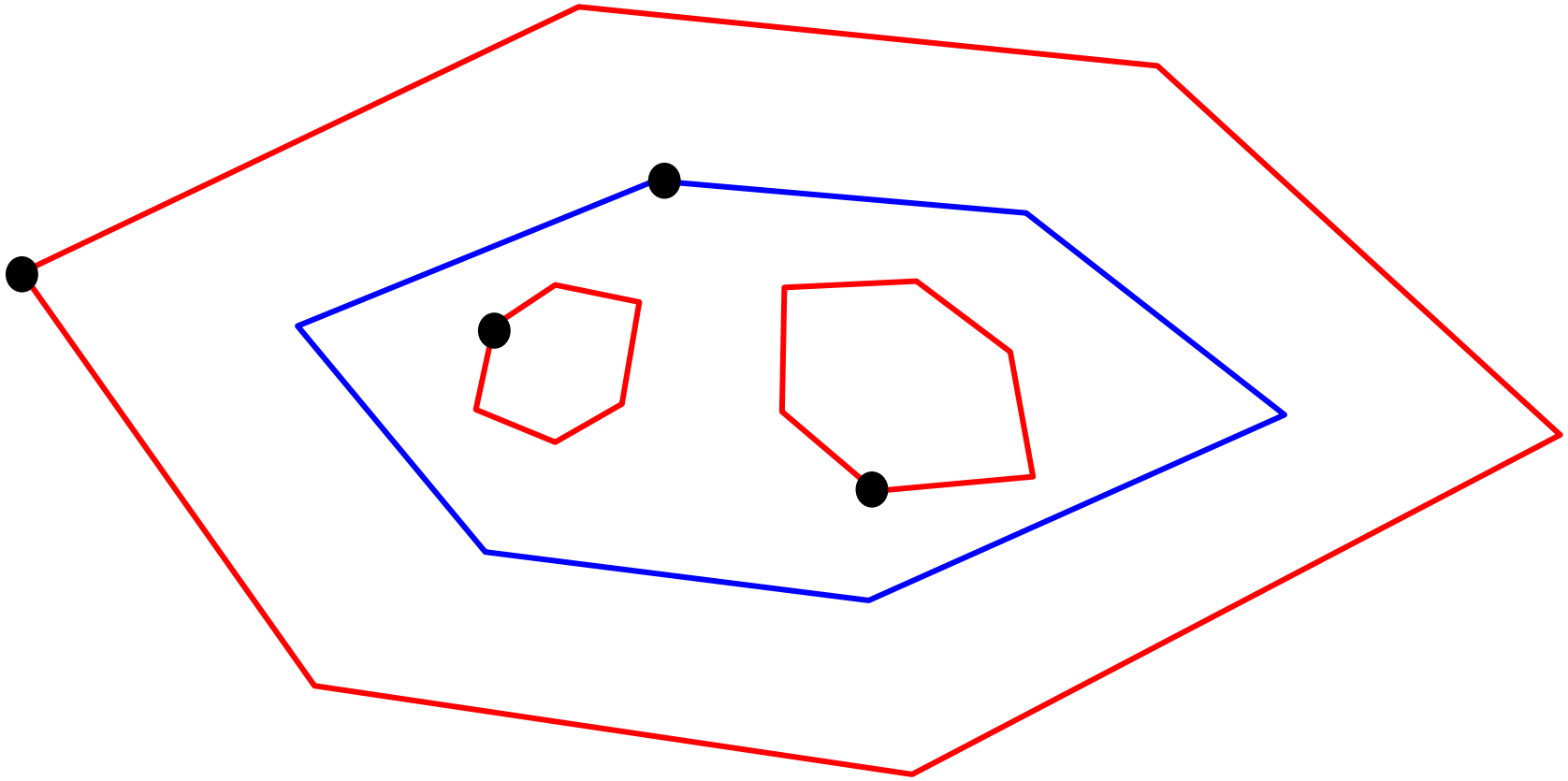
ORIENTATION SETTING – MINETTO

Point-in-polygon algorithms…



ORIENTATION SETTING – MINETTO

Point-in-polygon algorithms…

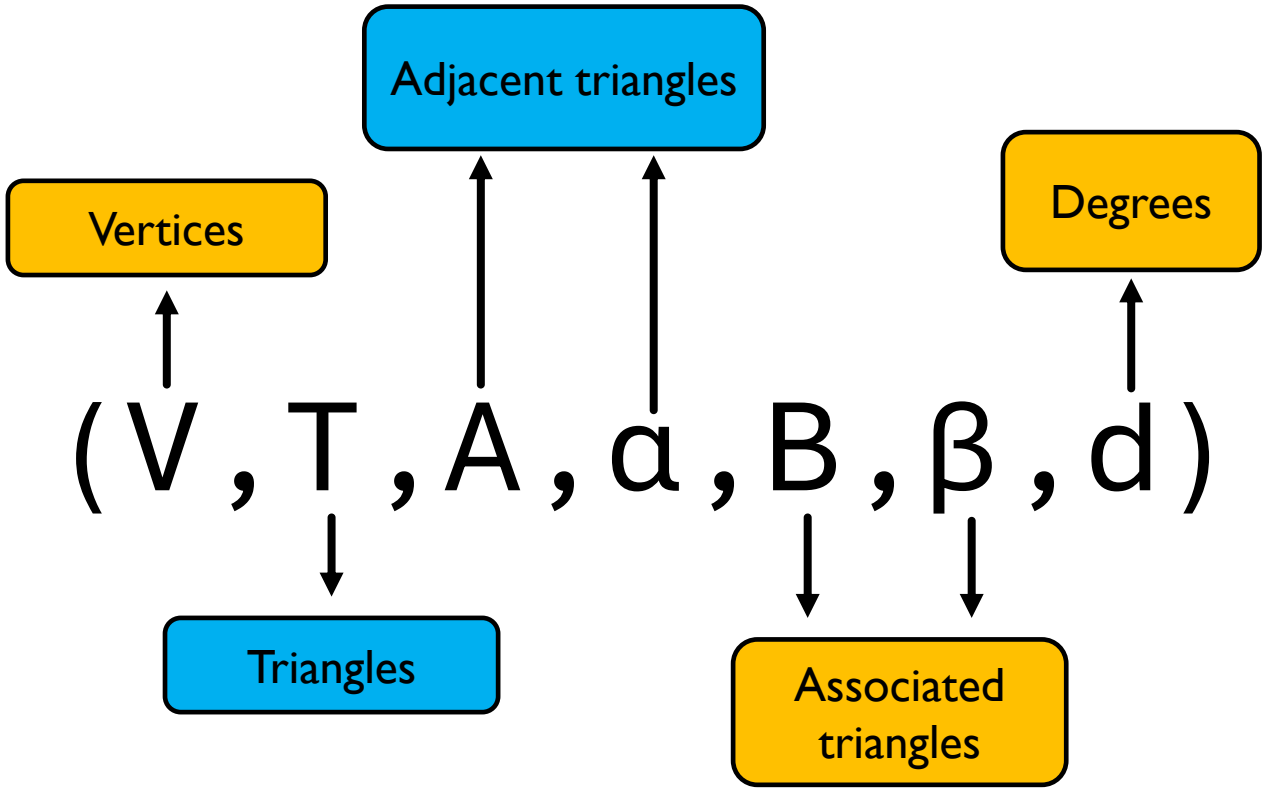


PROBLEMS & SOLUTION

- Hashing is „fragile”
- Hash collisions are frequent in case of detailed geometries
- Works for closed, orientable 2-manifolds
- It also works, if the normals are not consistently oriented
- Wouldn’t it be cheaper to recover normal consistency in advance than solving a bunch of PIP problems?

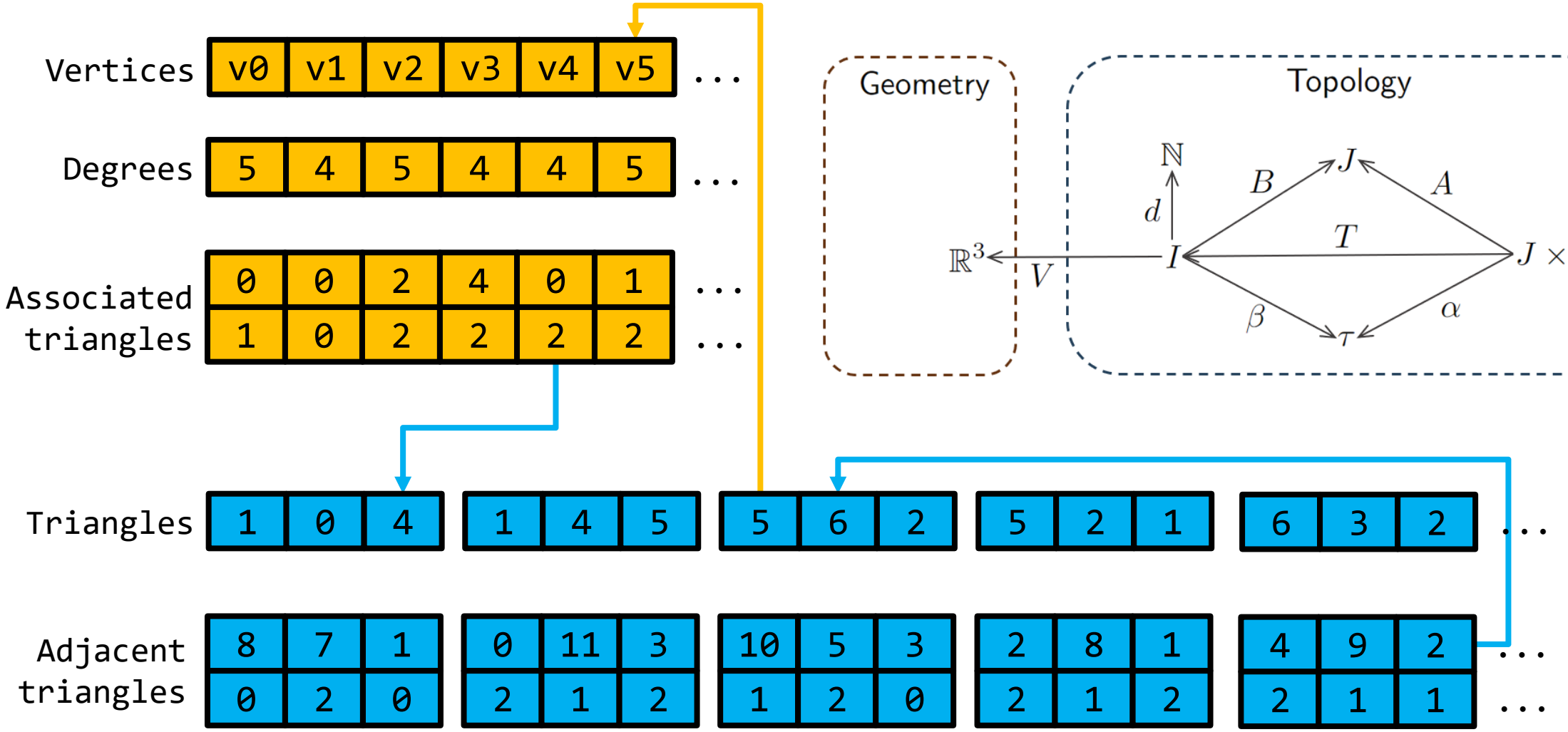
- Our idea
  - Convert mesh into a **topological data structure** before slicing
  - **Do not use hashing** by contour construction
  - Data structure should store infomation about **orientation** of triangles

Fábián, G.: SolidMesh – Data Structure for Global Mesh Operations. Computer Aided Design & Applications 22(4), 2025.



SOLIDMESH DATA STRUCTURE (2024)

SOLIDMESH EXAMPLE



MINETTO'S ALGORITHM

- 1. Build triangle lists → $\mathcal{O}(n+k)$
- 2. Plane sweep
  - Insert triangles to the active set
  - Compute plane-triangle intersections → $\mathcal{O}(m)$
  - Add segments to the segment list
  - Remove triangles from the active set
- 3. Contour construction → $\mathcal{O}(m)$
- 4. Set orientation of contours → $\mathcal{O}(m)$

$$\mathcal{O}(n+m+k)$$

OUR ALGORITHM

- 0. Convert mesh to SolidMesh
- 1. Build triangle lists → $\mathcal{O}(n+k)$
- 2. Plane sweep
  - Insert triangles to the active set
  - Compute plane-triangle intersections → $\mathcal{O}(m)$
  - Add segments to the segment list
  - Remove triangles from the active set
- 3. Oriented contour construction → $\mathcal{O}(m)$
- ~~4. Set orientation of contours~~

$$\mathcal{O}(n+m+k)$$

MINETTO'S ASSUMPTIONS

- The mesh is a closed orientable 2-manifold without self-intersections.
- Planes don't hit vertices of the mesh.

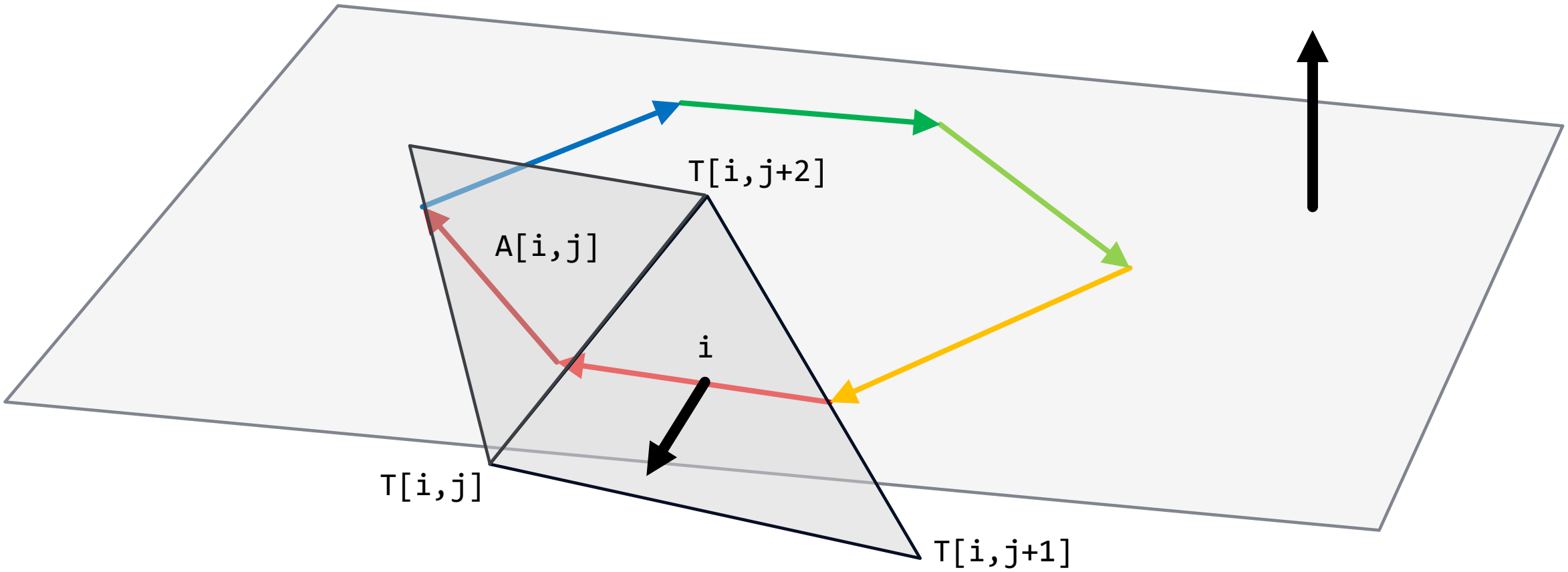
- Plane-triangle intersections are **segments**.
- Plane-mesh intersections are closed loops: **contours**.

OUR ASSUMPTIONS

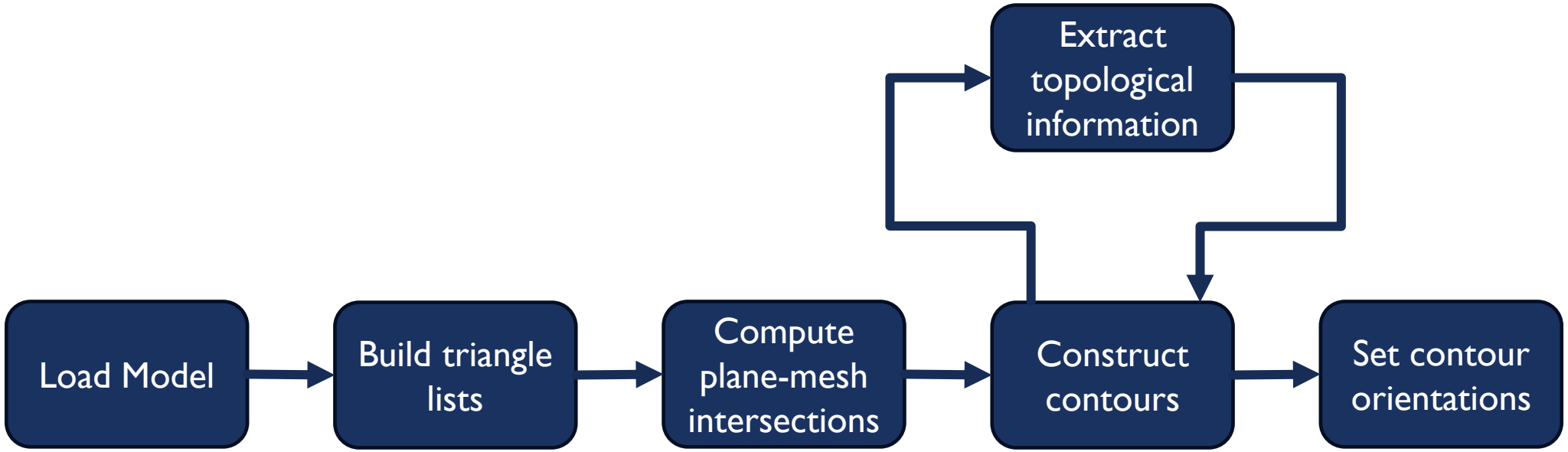
- The mesh is a closed orientable 2-manifold ~~without self-intersections~~ with consistently oriented normals.
- Planes don't hit vertices of the mesh.

- Plane-triangle intersections are **oriented segments**.
- Plane-mesh intersections are oriented closed loops: **oriented contours.**

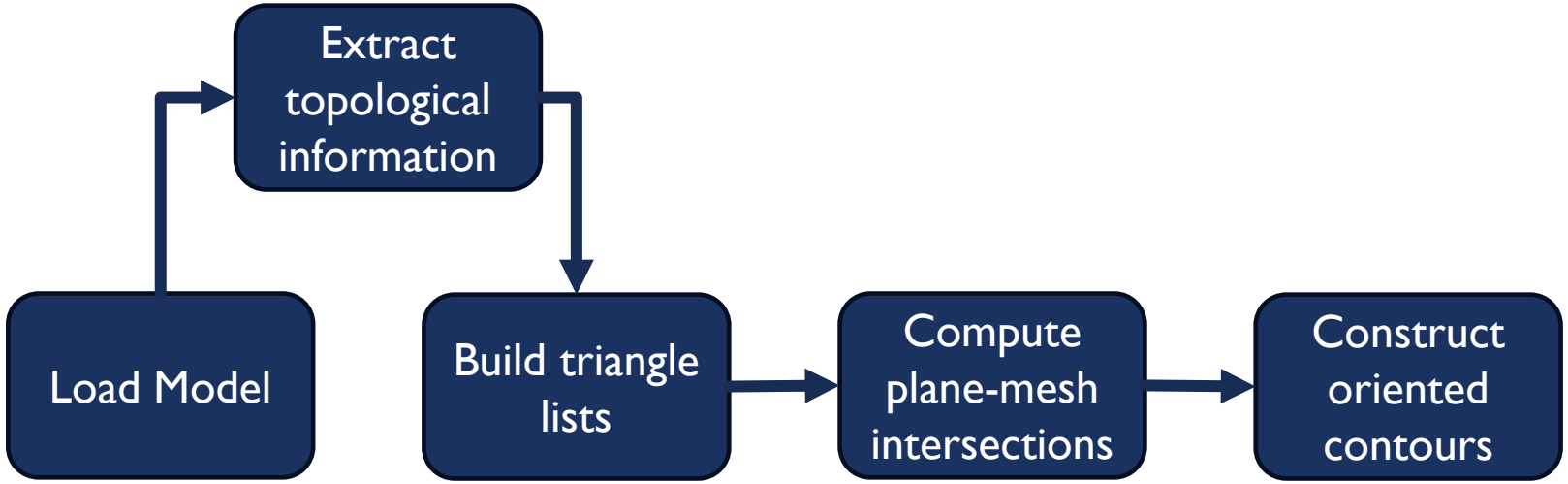
ORIENTED CONTOUR CONSTRUCTION – OURS



MINETTO'S MESH SLICING



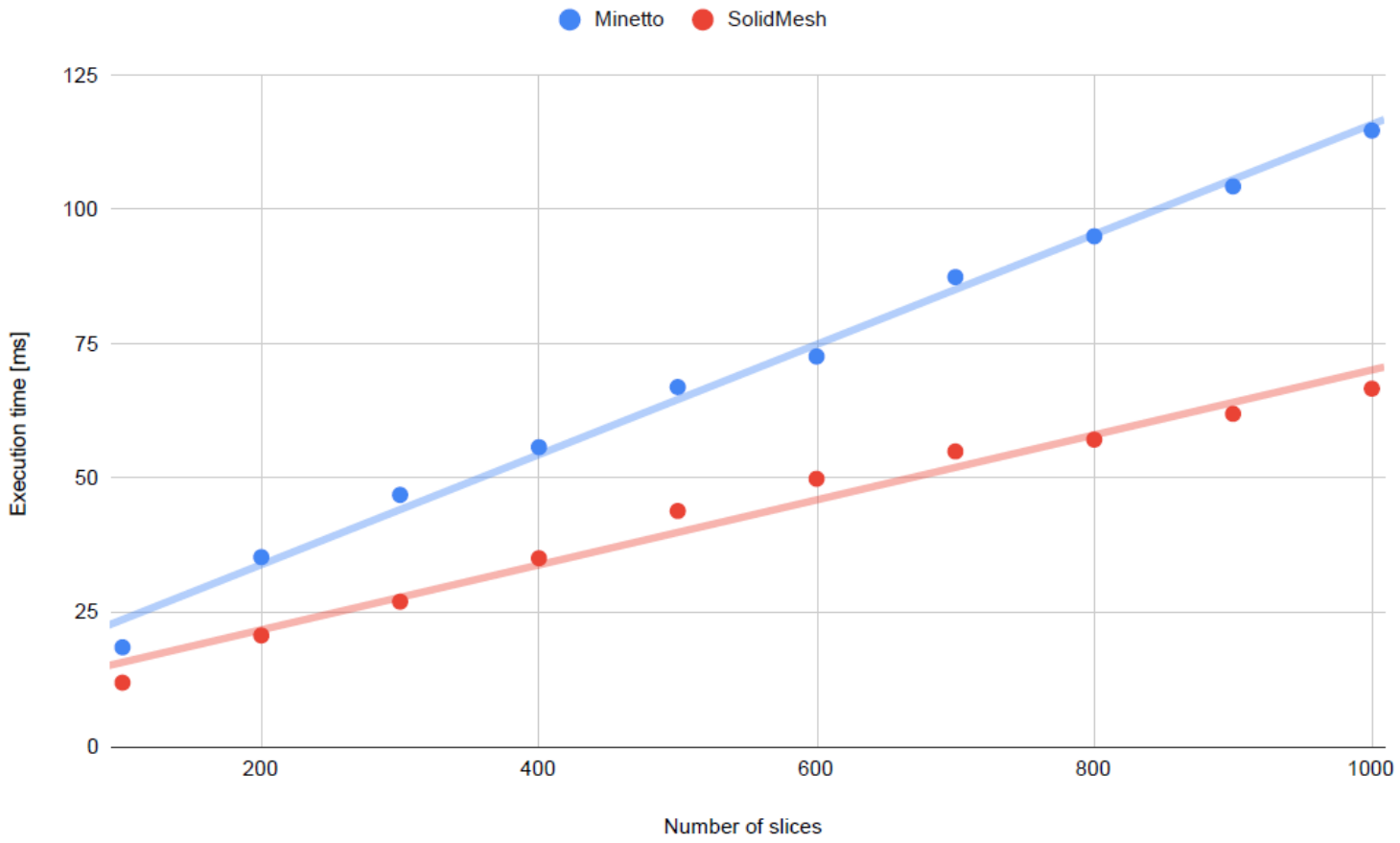
OUR MESH SLICING



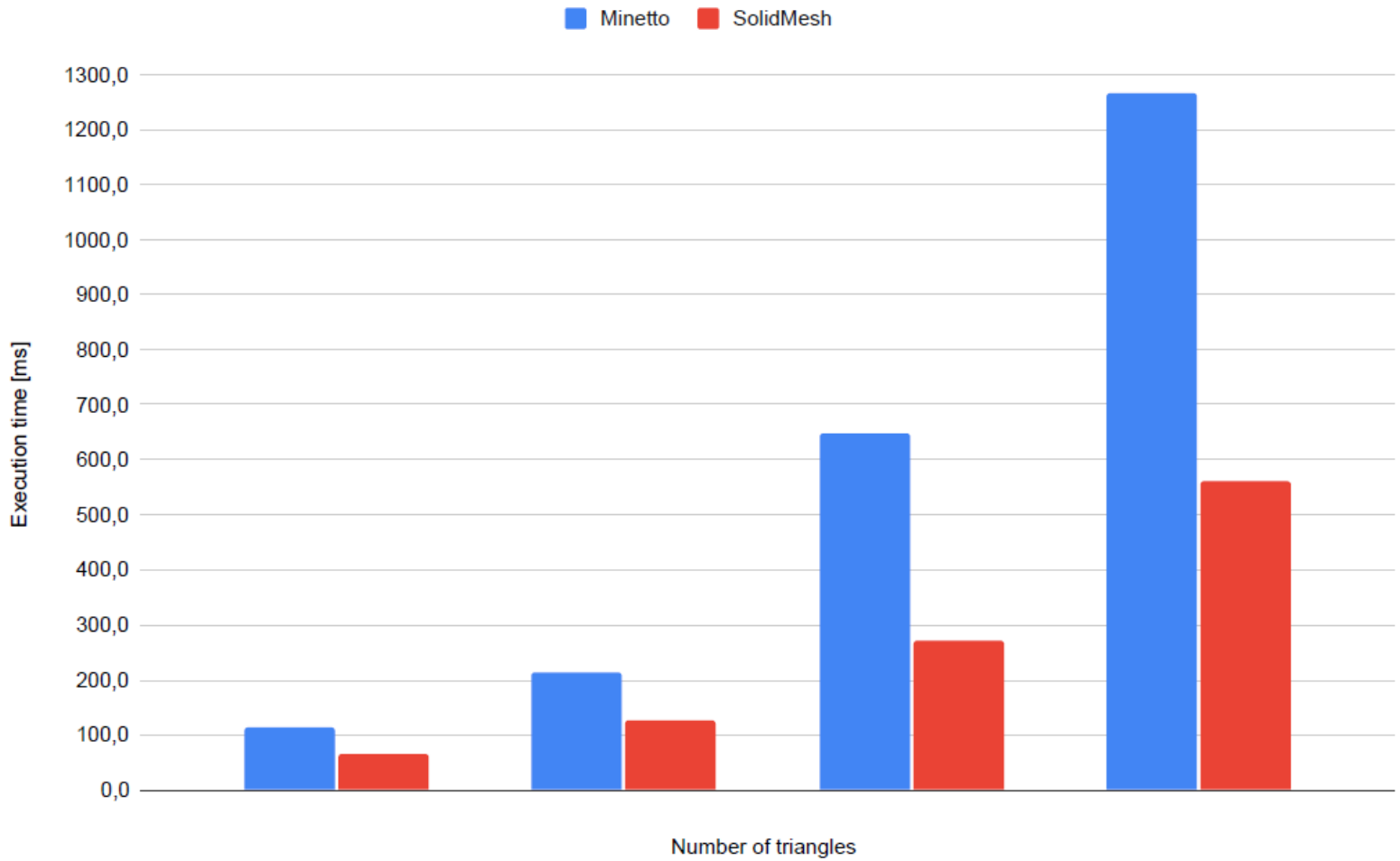
unity

C#

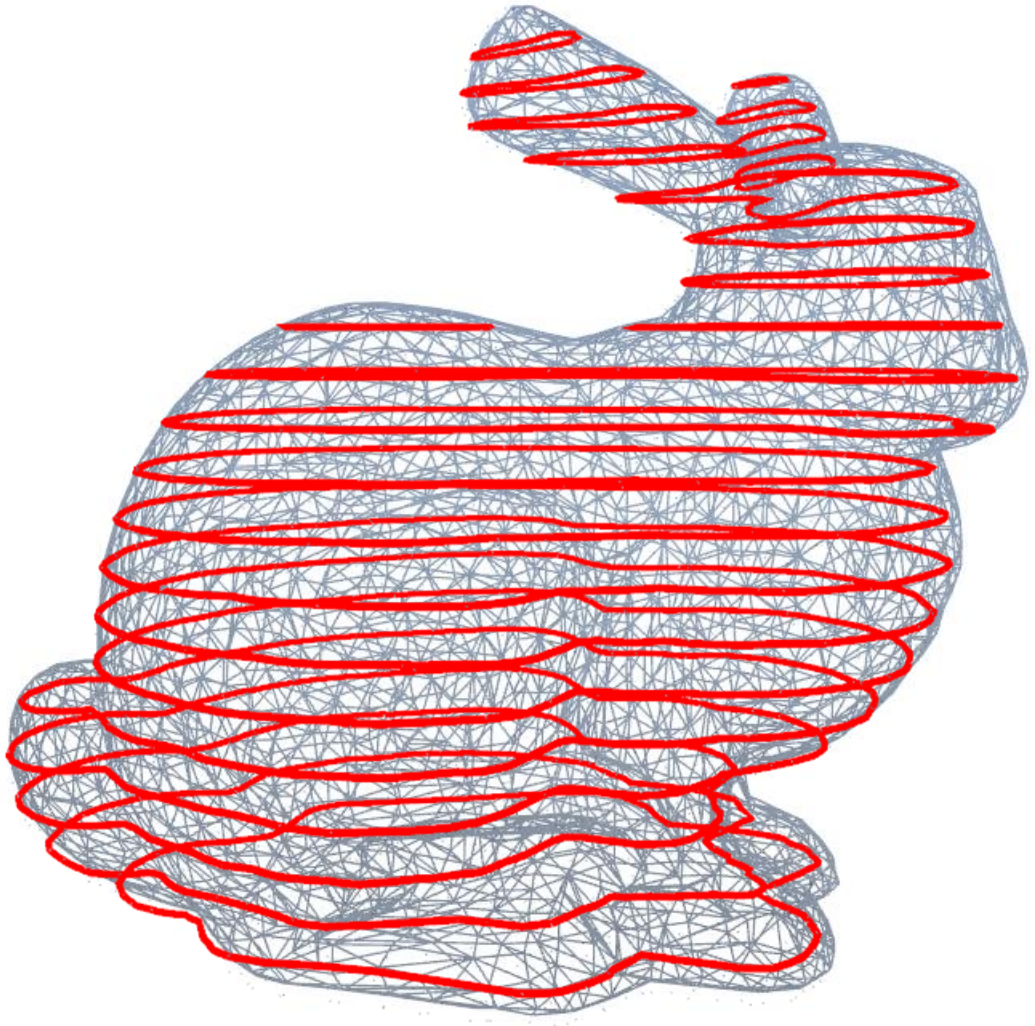
RESULTS – NUMBER OF SLICES



RESULTS – NUMBER OF TRIANGLES



RESULTS – TIME / SEGMENT

Model	Triangles	Slices	Loops	Segments	Minetto		SolidMesh	
					Time [ms]	Time/Seg [ns]	Time [ms]	Time/Seg [ns]
Bunny	5002	1000	1282	129955	114.8	0.88	67.1	0.52
	5002	2000	2564	259936	220.5	0.85	125.7	0.48
	5002	3000	3845	389790	327.4	0.84	186.3	0.48
	5002	4000	5125	519536	622.1	1.20	245.6	0.47
	5002	5000	6405	649486	733.7	1.13	304.7	0.47
Torusknot	1760	1000	4310	136662	134.9	0.99	73.9	0.54
	1760	2000	8617	273134	250.8	0.92	137.7	0.50
	1760	3000	12924	409650	374.7	0.91	198.0	0.48
	1760	4000	17230	546144	691.6	1.27	259.3	0.47
	1760	5000	21538	682656	819.6	1.20	326.2	0.48
Liver	38142	1000	2532	375582	333.7	0.89	196.2	0.52
	38142	2000	5059	750772	839.8	1.12	369.8	0.49
	38142	3000	7582	1124799	1347.0	1.20	542.8	0.48
	38142	4000	10107	1499955	1857.2	1.24	719.2	0.48
	38142	5000	12625	1873866	2265.0	1.21	1087.1	0.58

THANK YOU FOR YOUR ATTENTION!