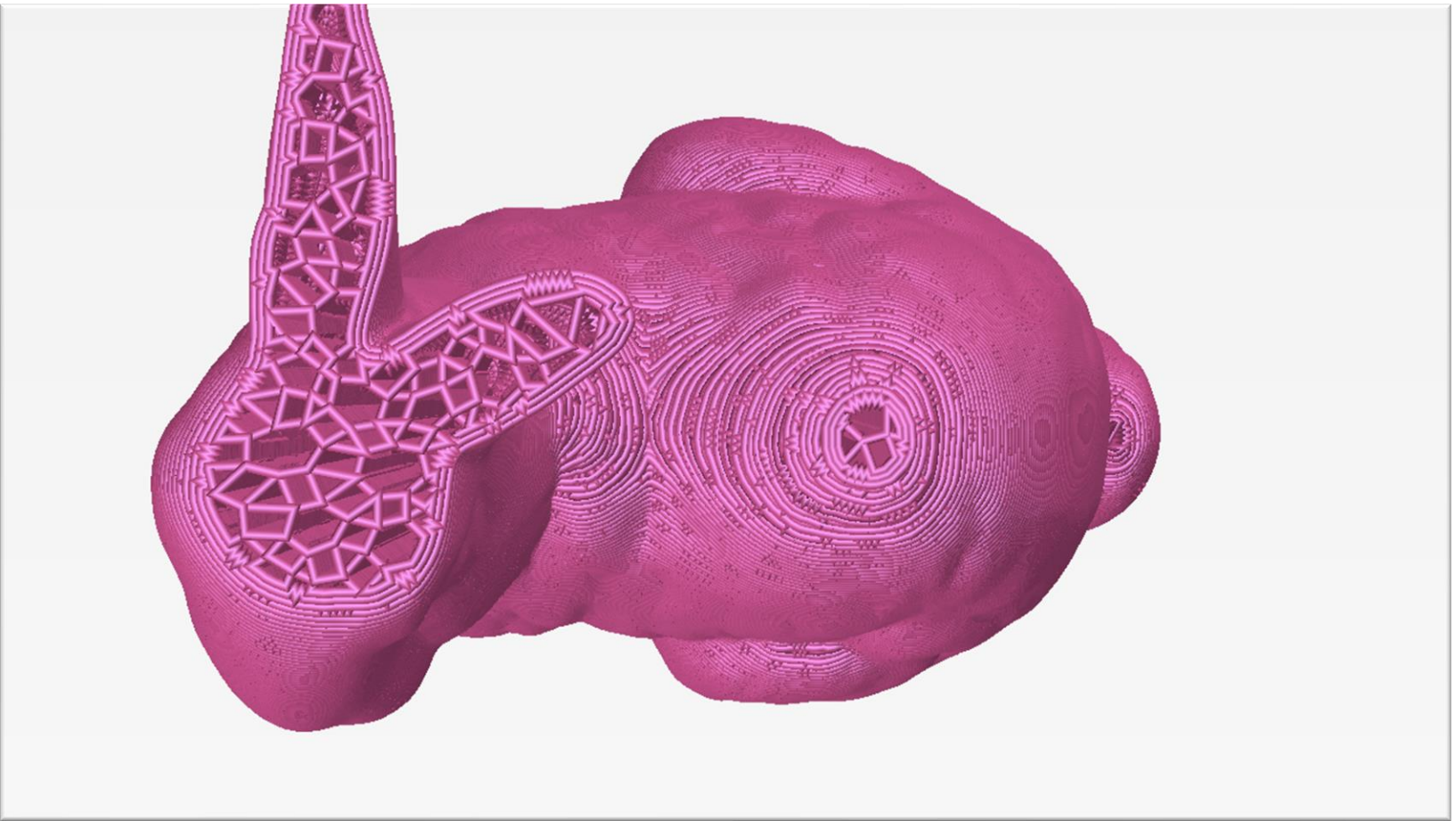
Continuous Toolpath Planning in a Graphical Framework for Sparse Infill Additive Manufacturing

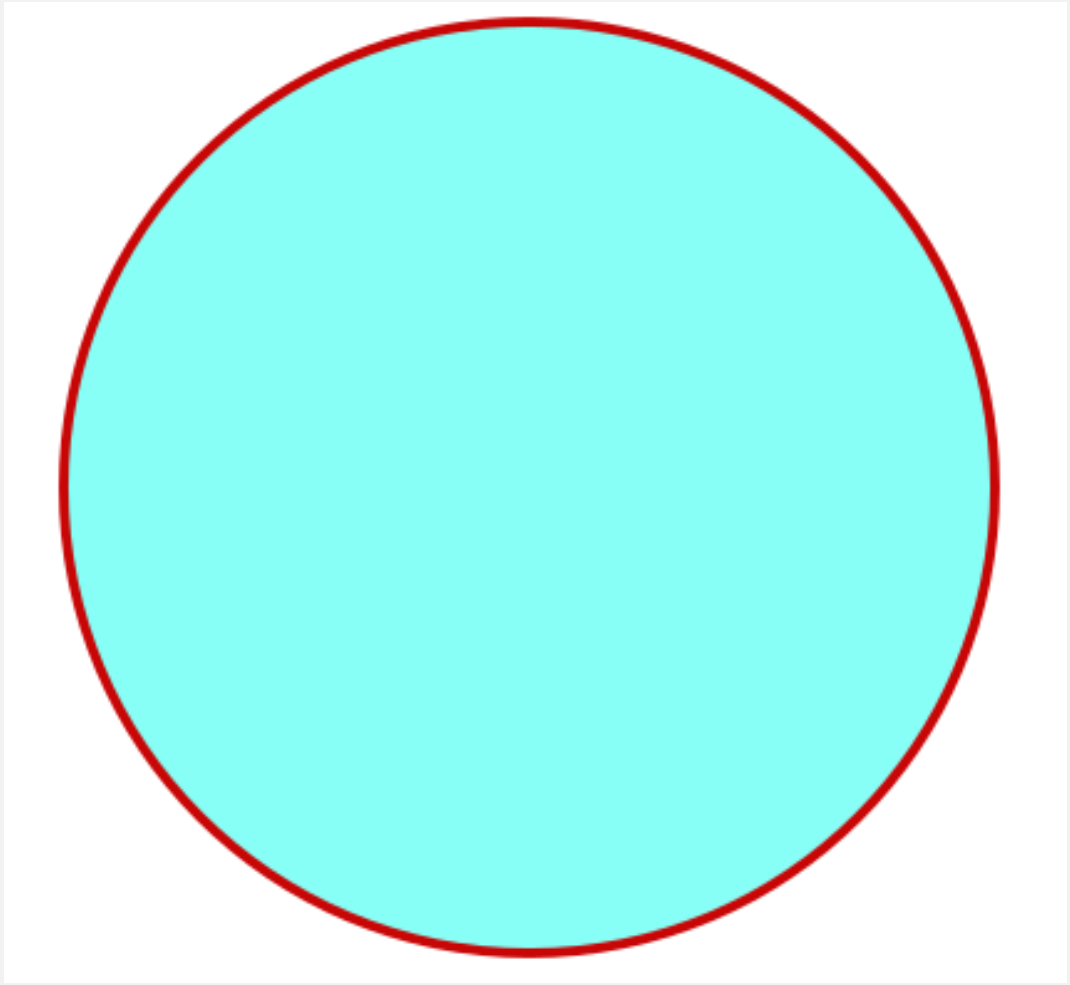


Prashant Gupta*
Bala Krishnamoorthy*
Gregory Dreifus**

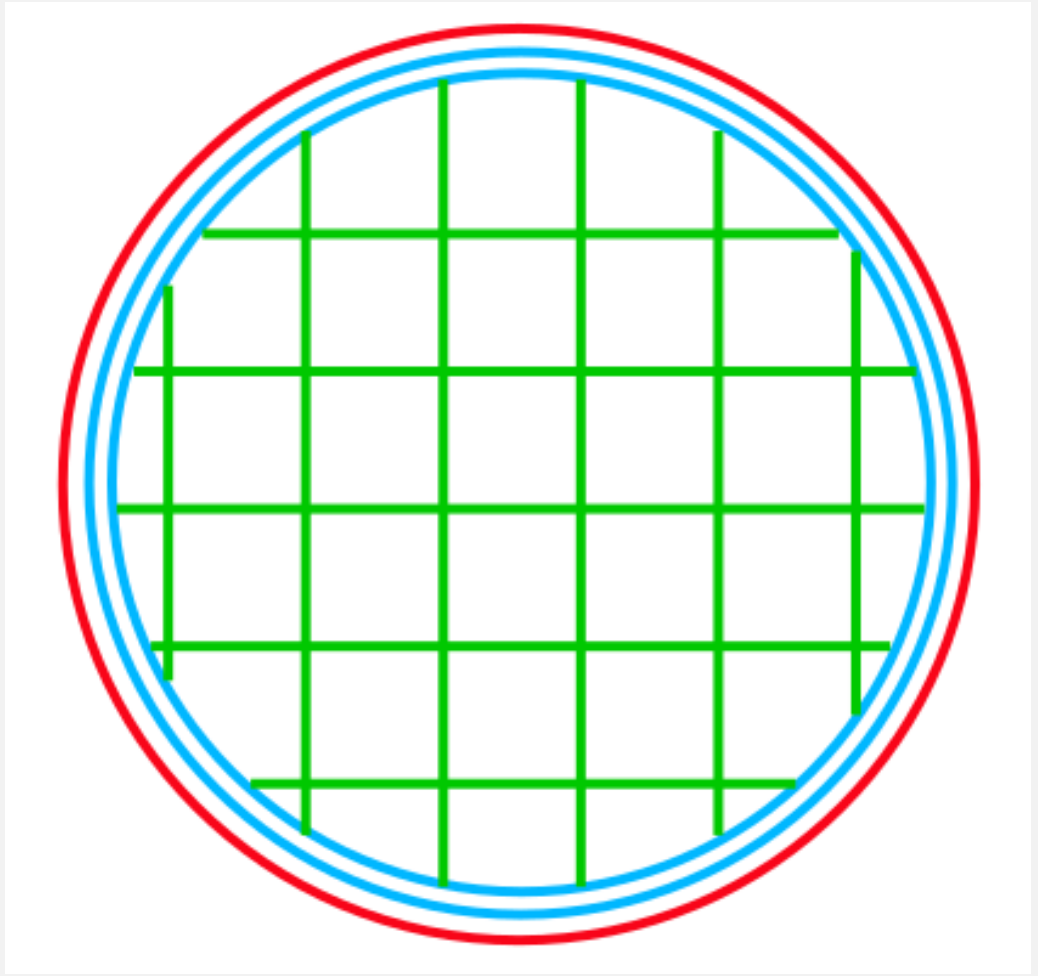
*Washington State University
**MIT

NSF
OAK RIDGE National Laboratory

Sparse infill in 3D printing



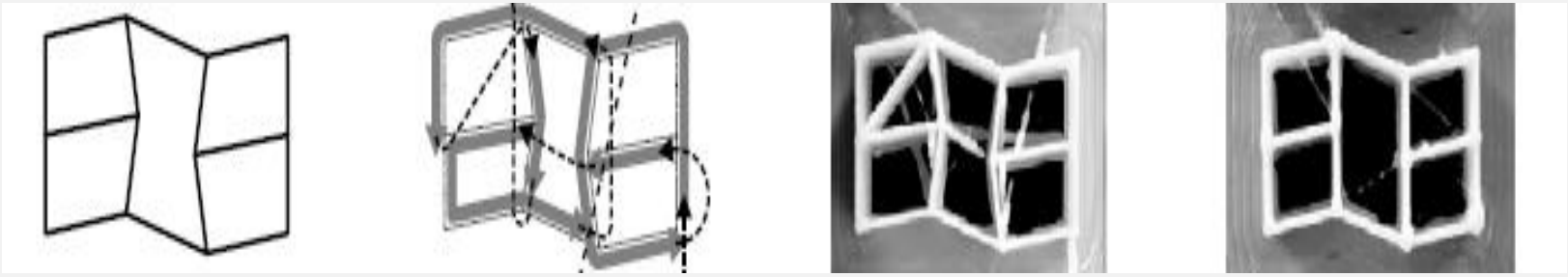
Infill Region(Blue)



Sparse Infill mesh

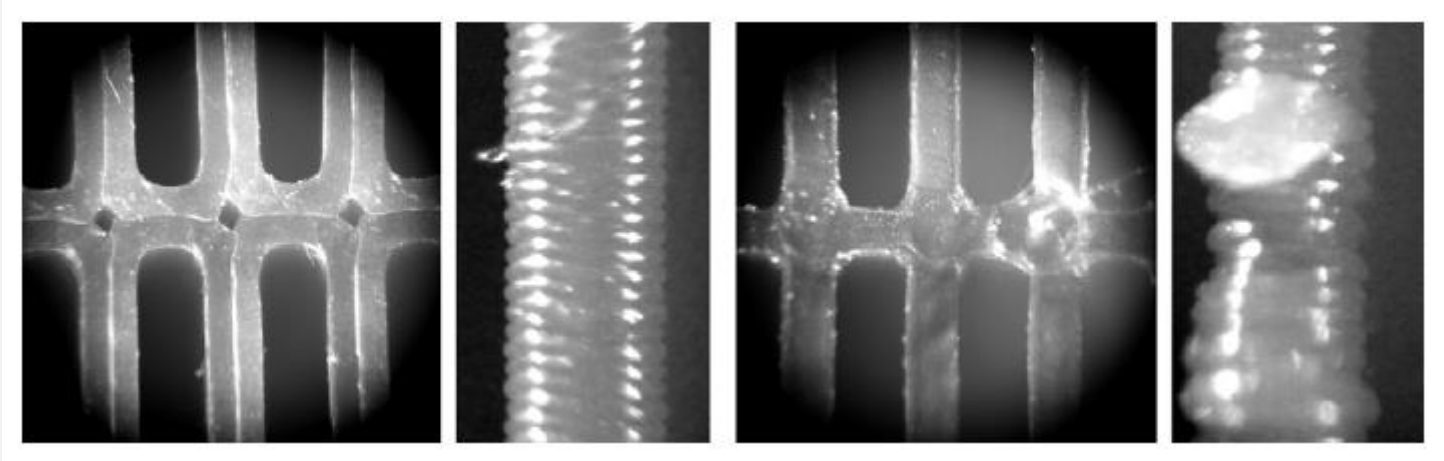
Challenges

- Discontinuous extrusion [Kuipers et al., 2019]



Input Order Leakage Bulging

- Crossover of extrusion paths [Kuipers et al., 2019]

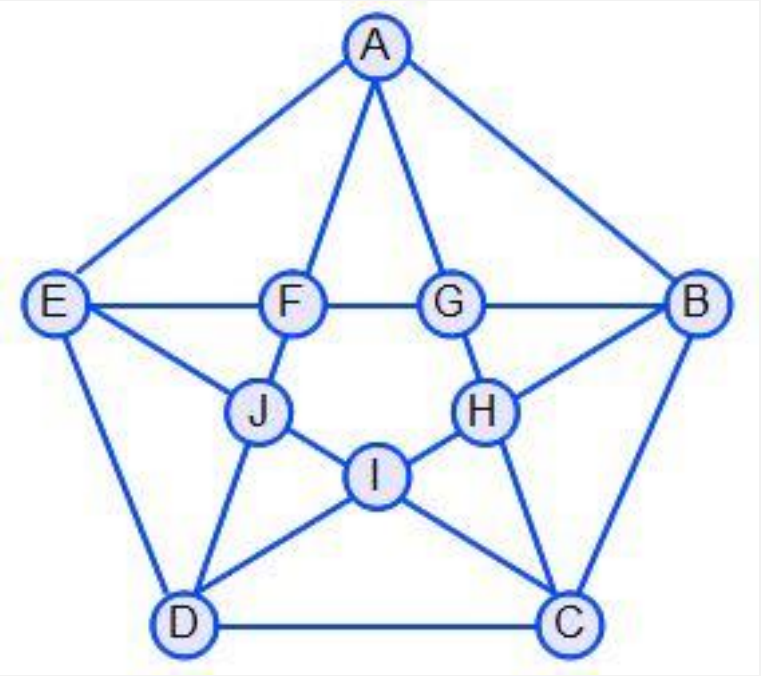


No Crossover Crossover

Challenges

Q: Can we create a framework to print continuously without crossover?

- We can use **Euler mesh**, where each vertex is connected to even number of edges.
- But, **Euler mesh** for arbitrary domain with identical types of elements is challenging
  - ***NP-complete*** for triangulations with even parity (Aichholzer et al., 2014)
- Crossover avoidance in mesh is still open

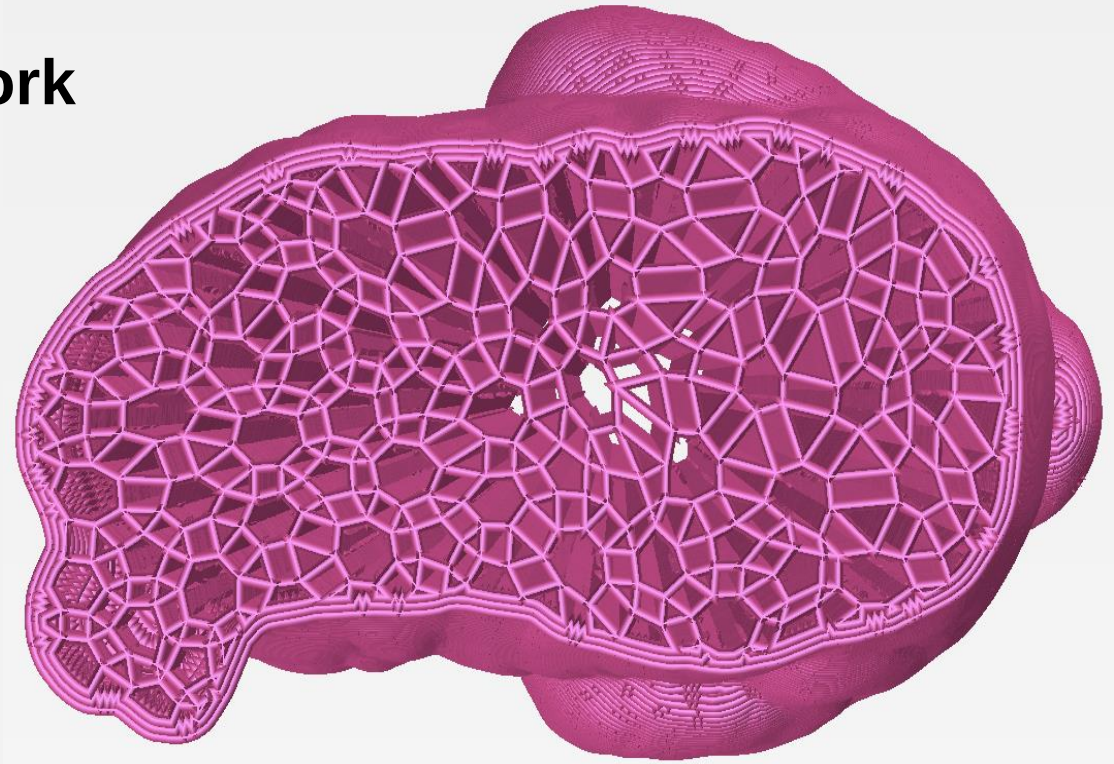


Euler Mesh

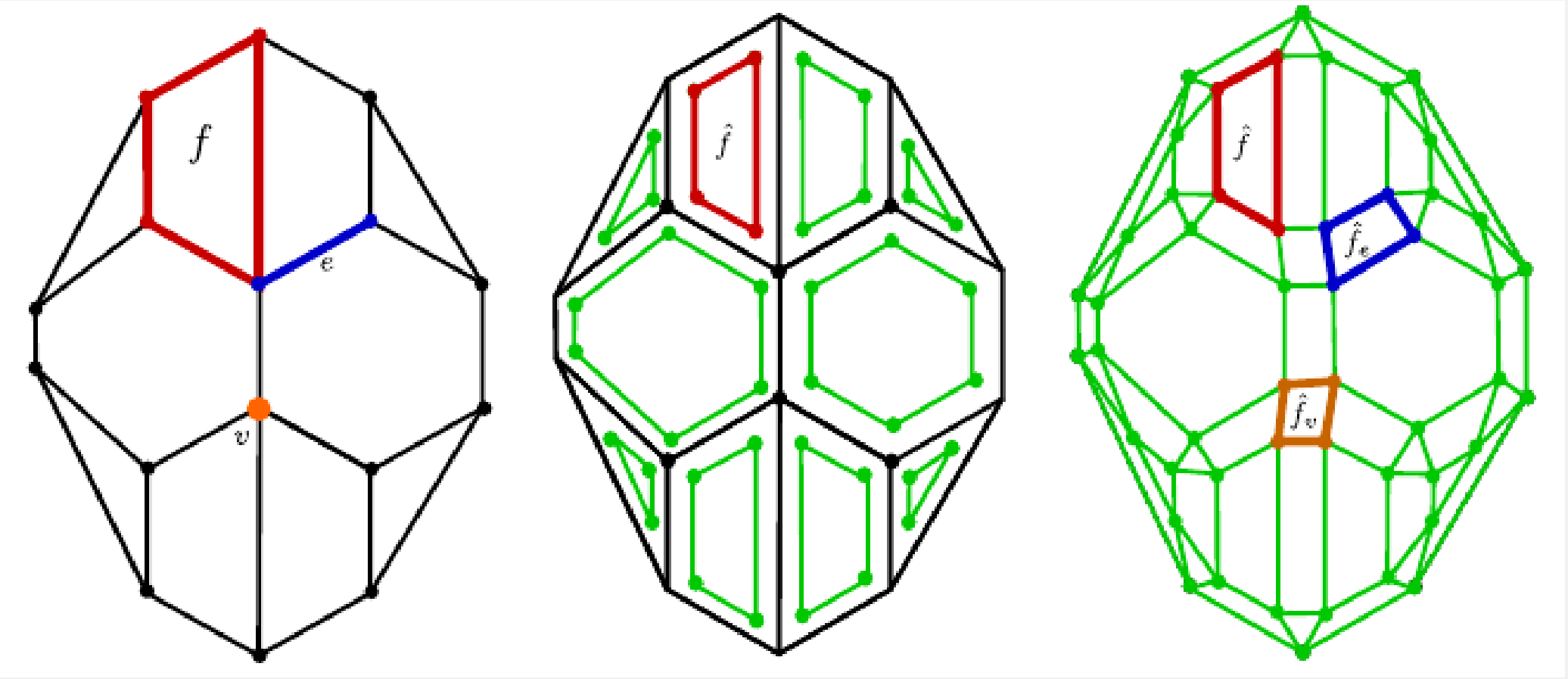
Our Contributions

- **Euler Transformation** for any **polygonal region with/out holes** to generate Euler mesh
- Generalized **Euler Transformation** that allows **combinatorial** and **topological changes**

- Efficient **continuous path planning framework** for 3D printing using Euler Transformation.
  - ✓ any geometry
  - ✓ holes
- Algorithm for tool path with **no crossover**



Euler Transformation*



a) Input Mesh b) Mitered Offset c) Joining

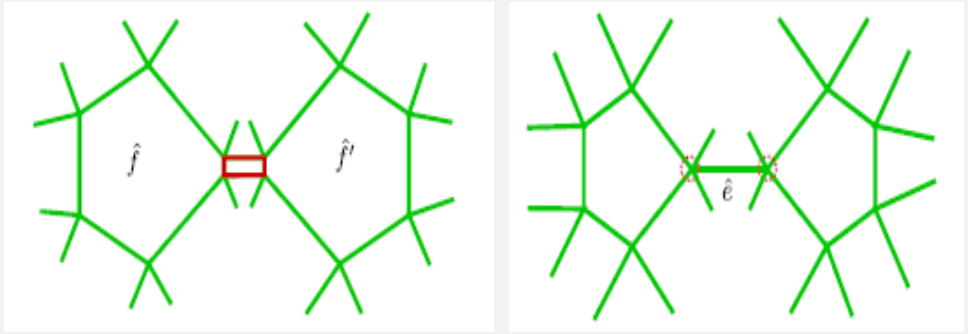
- Can generate Euler mesh in polynomial Time for any input mesh.

* G and Krishnamoorthy. Euler transformation of polyhedral complexes. 2018, arXiv:1812.02412

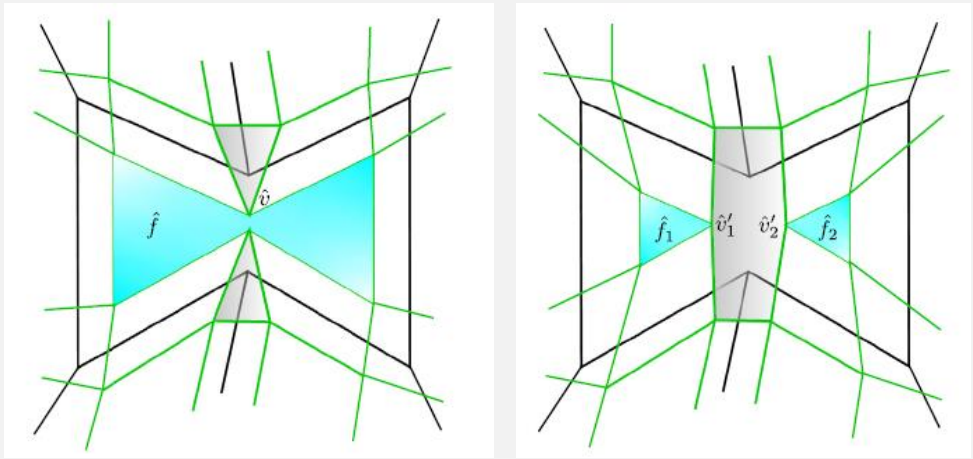
Properties of Euler Transformation (ET)

$\mathrm{ET}(K) = \widehat{K}$

- Every vertex in $\widehat{K}$ connected to 4 other vertices.
- $\left|\widehat{V}\right| = 2|E|$, $|\widehat{E}| = 4|E|$, $\left|\widehat{F}\right| = |V| + |E| + |F|$
  V,E,F: sets of vertices, edges, faces (polygons) in K
- Total Euclidean length of the edges in $\widehat{E}$ roughly doubles.
- Generalized ET with combinatorial and topological changes.

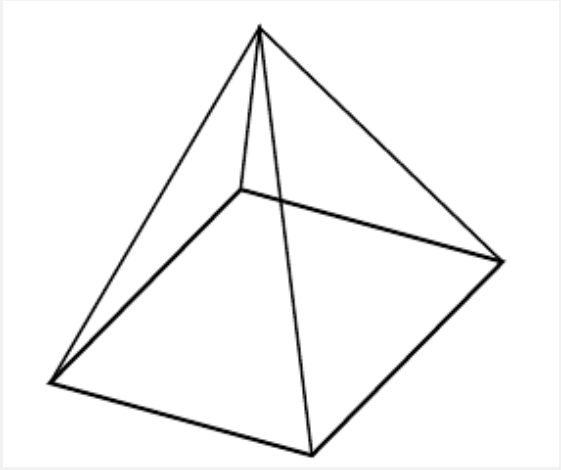


Combinatorial Change

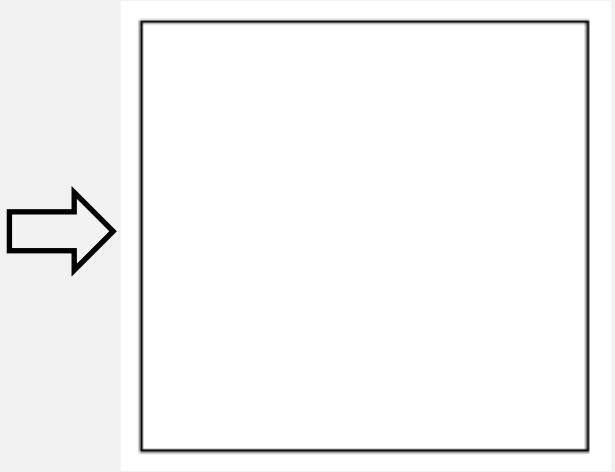


Topological change

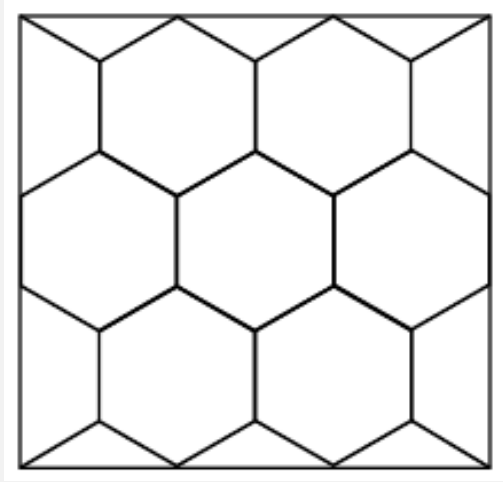
Outline of Our Framework



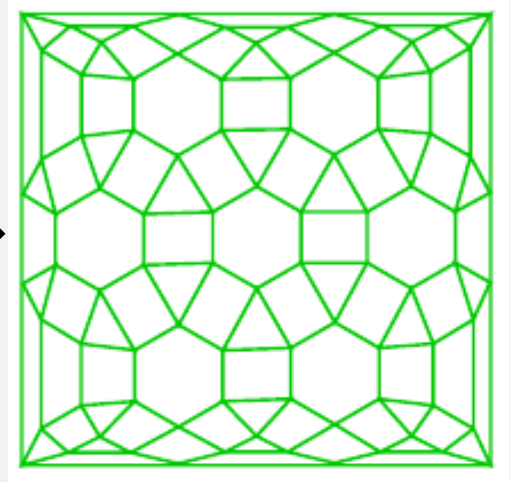
Input STL file



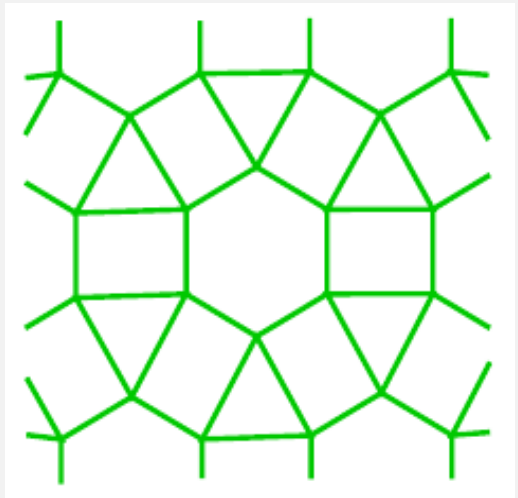
Union of projected polygons in each layer



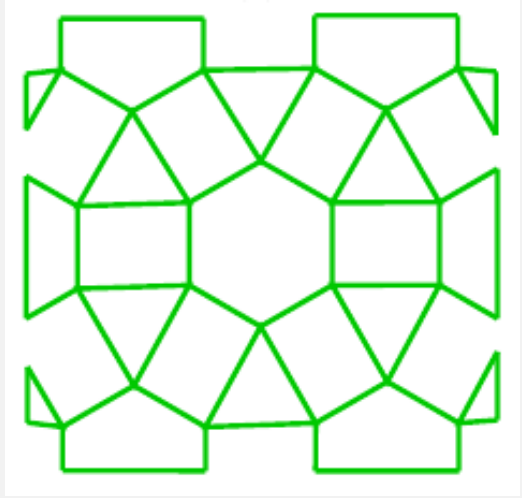
Input mesh



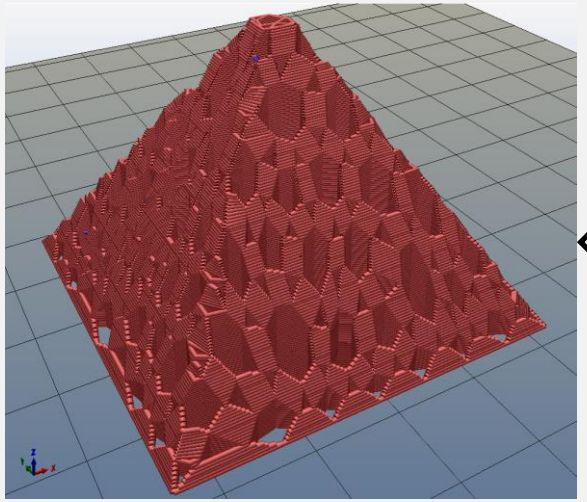
Transformed Mesh



Clipped Mesh



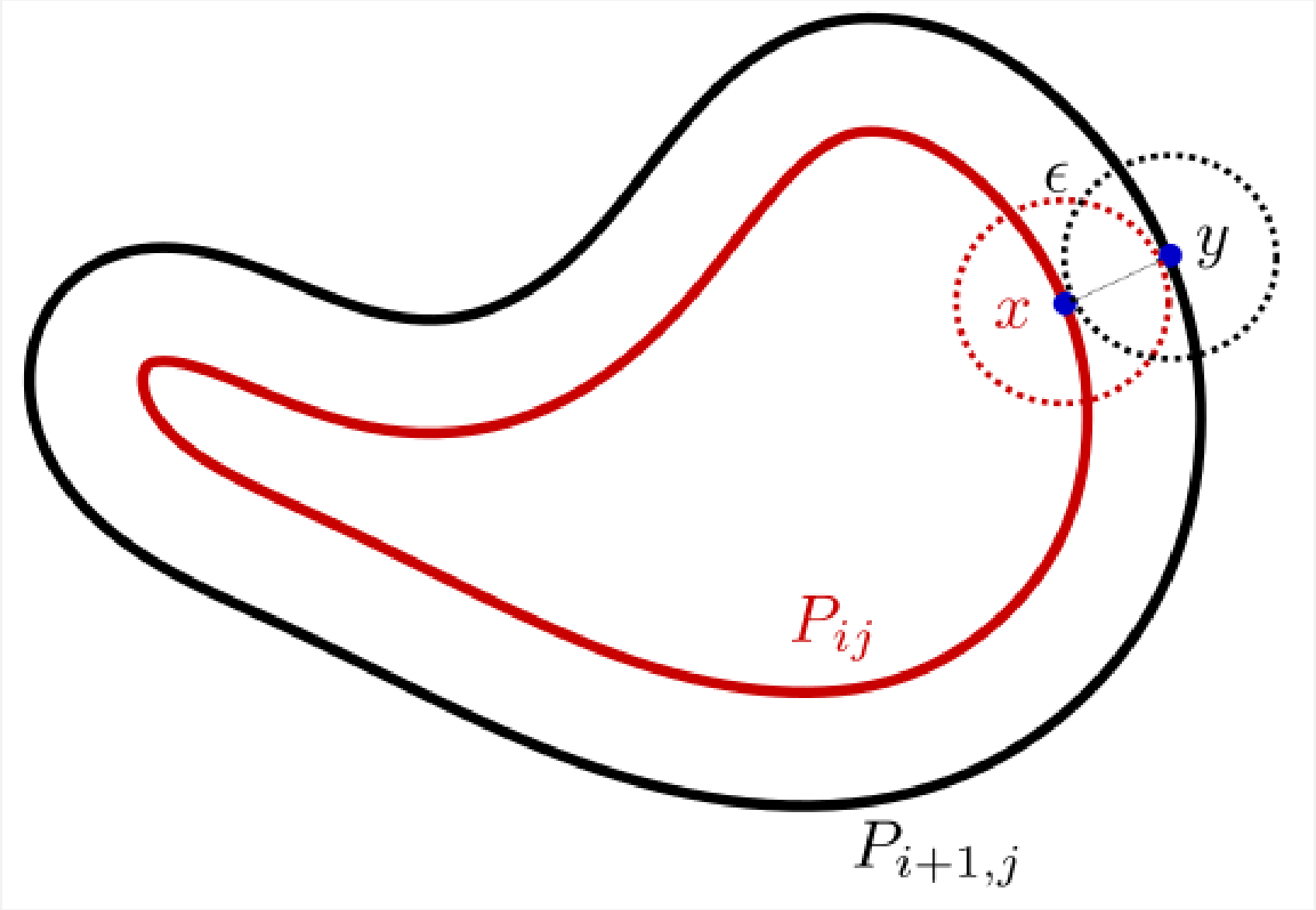
Patched Mesh



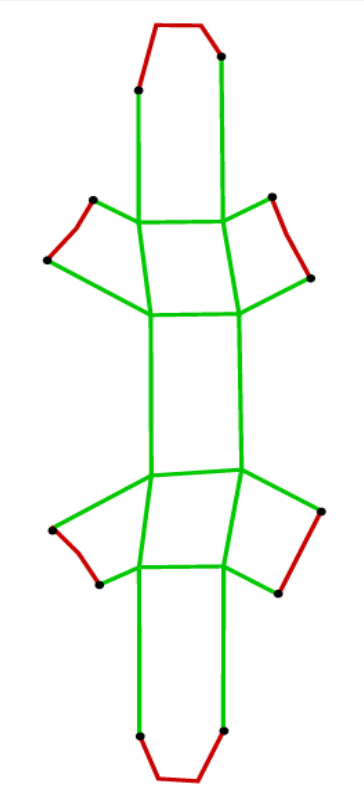
Preprint Pyramid

Printed Pyramid (Layers)

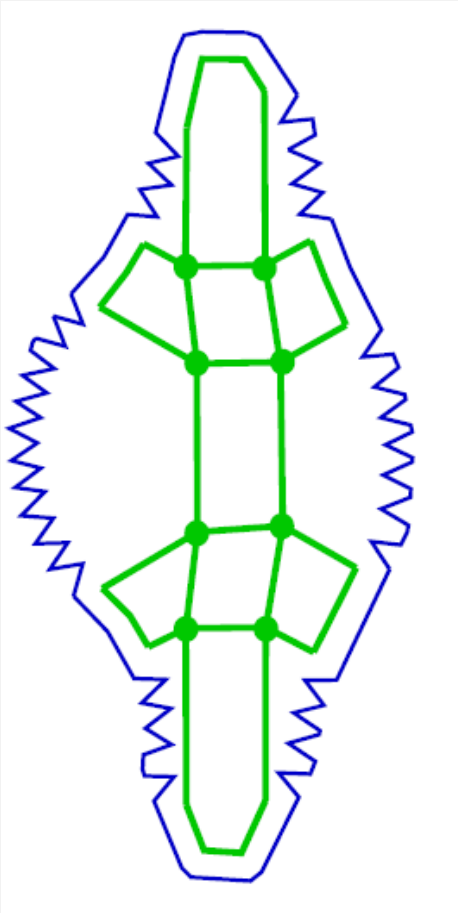
ϵ-Continuous Layers



Support Perimeter



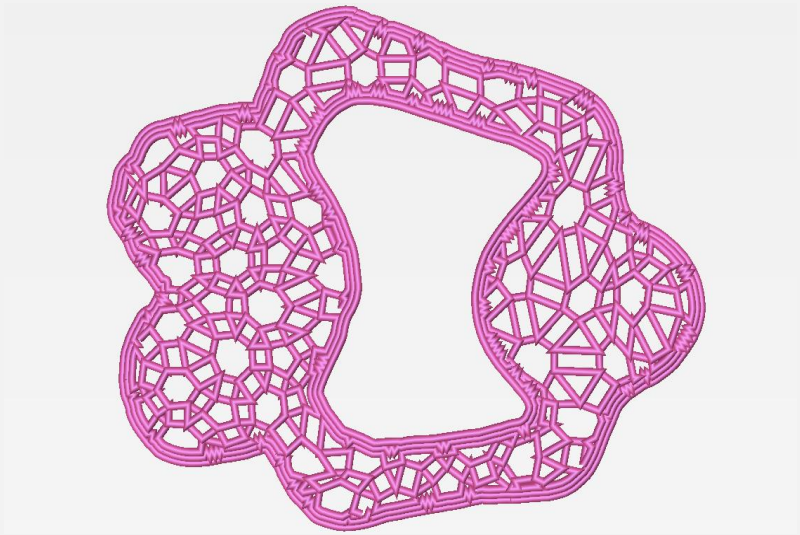
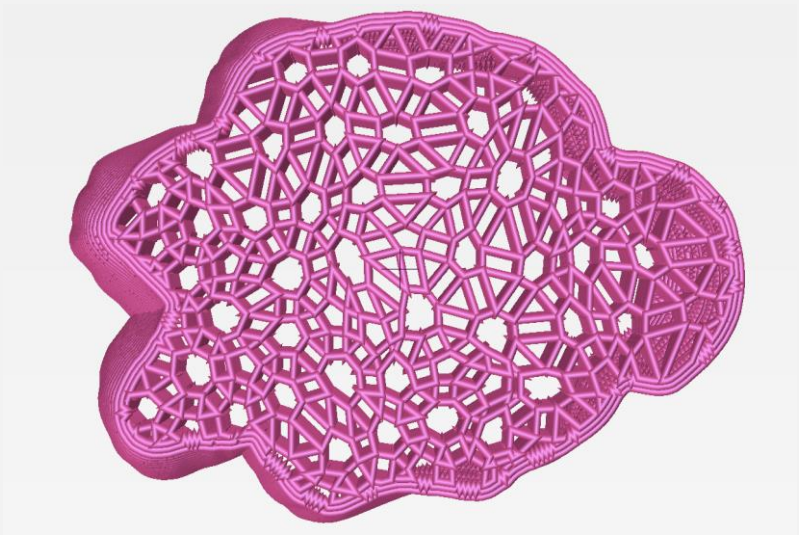
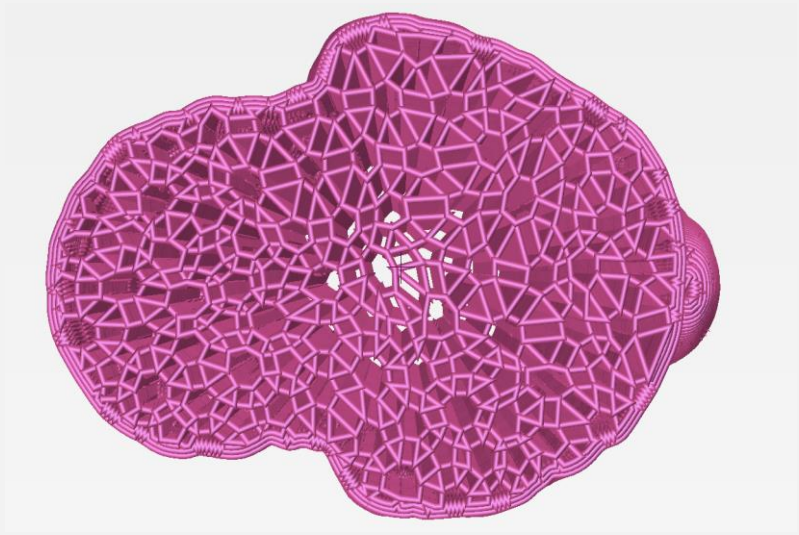
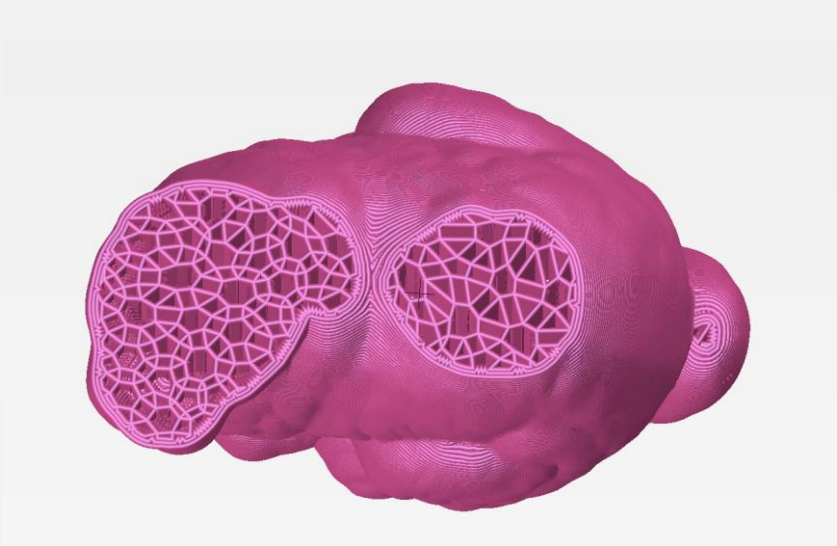
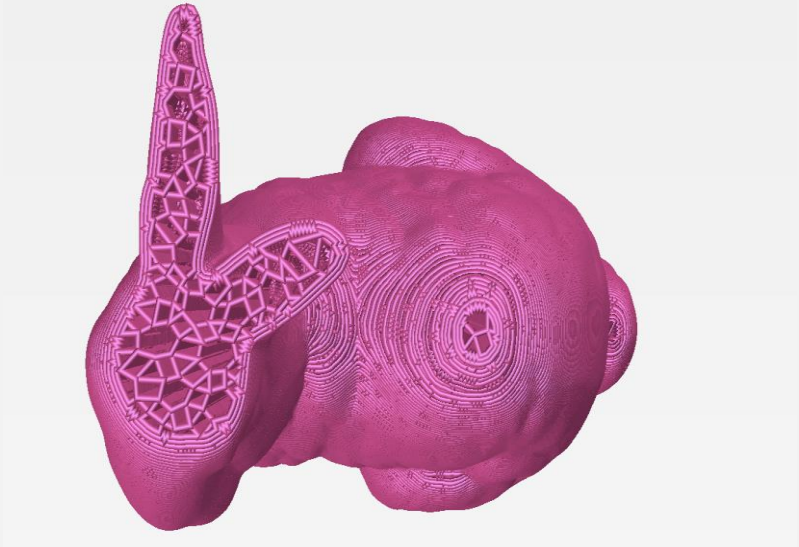
a) Patched Mesh



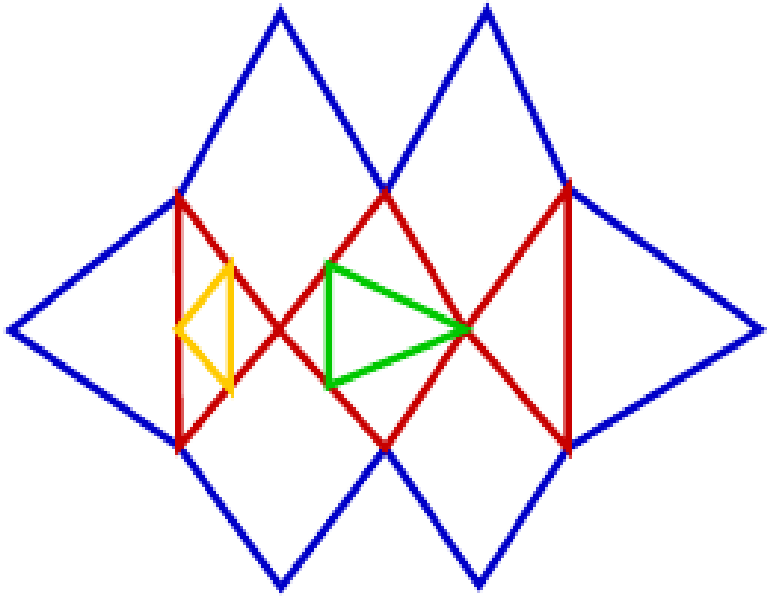
b) Support Perimeter (Blue)

Add regular perimeter around support perimeter to improve surface quality.

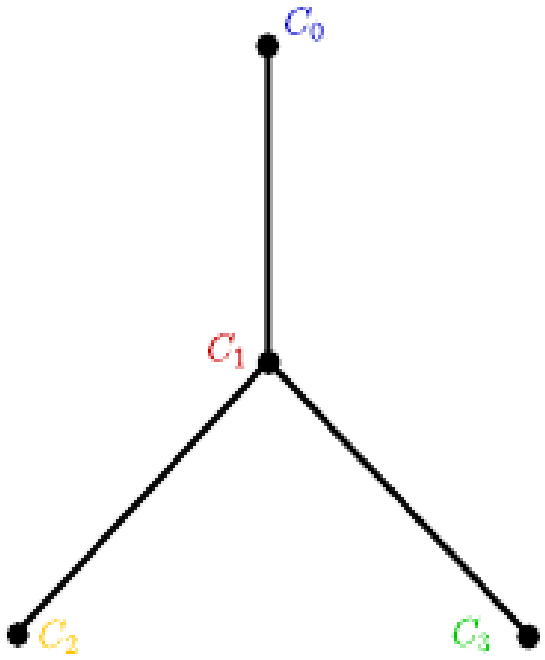
Example: Stanford Bunny

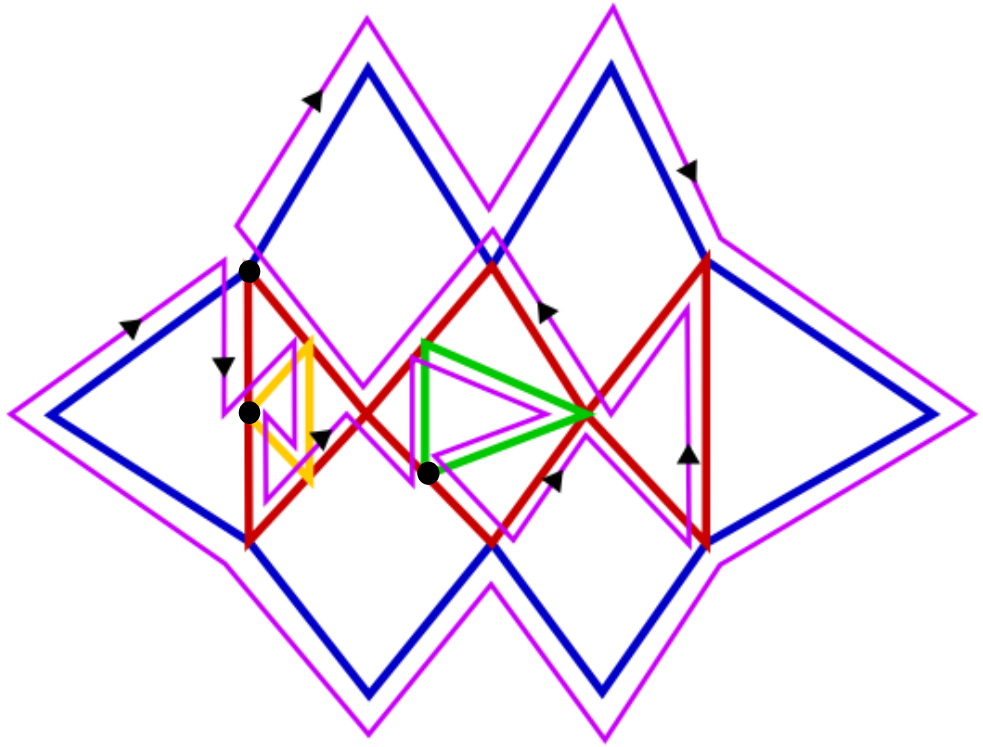

Algorithm for Toolpath with no crossover



a) Euler Mesh



b) Circuit Tree



c) Euler tour (pink), Restrictions (Black dots)

Open Problems

- “Optimized” toolpath algorithm with no crossovers?
- Mechanical properties of infill lattice under ET?
- Continuous toolpath framework for surface mesh?
- Non-planar 3D printing?

Thank You!