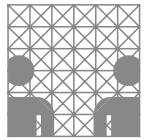


Conformal Surface Printing

on a 5-axis 3D Printing System

Paul Bartel

University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences
Department of Informatics

Technical Aspects of Multimodal Systems

25. June 2024

Outline

5-axis 3D printing

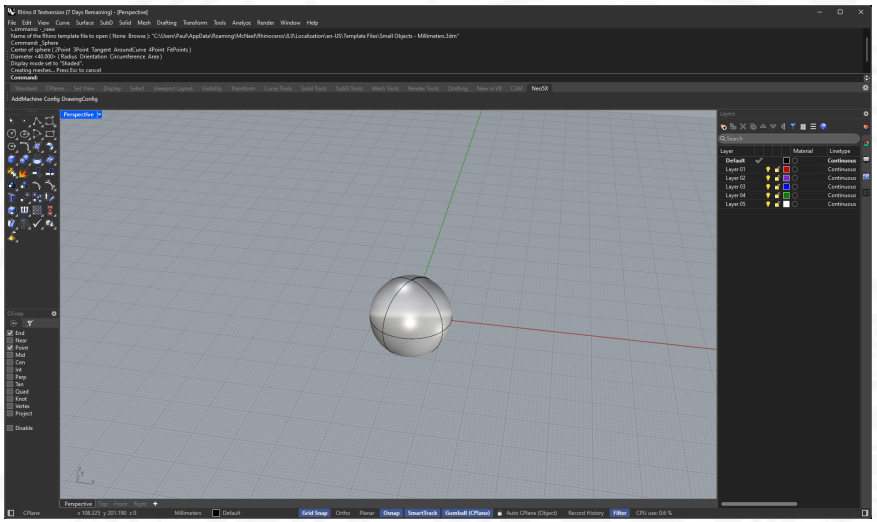
- Adding two additional axes allows for greater freedom in extruder orientation
- Extruder can conform to the object's surface, unlike conventional 3-axis systems
- Potential:
  - Supportless printing [1]
  - Better mechanial properties [2]
  - Better surface quality [3]

Problem

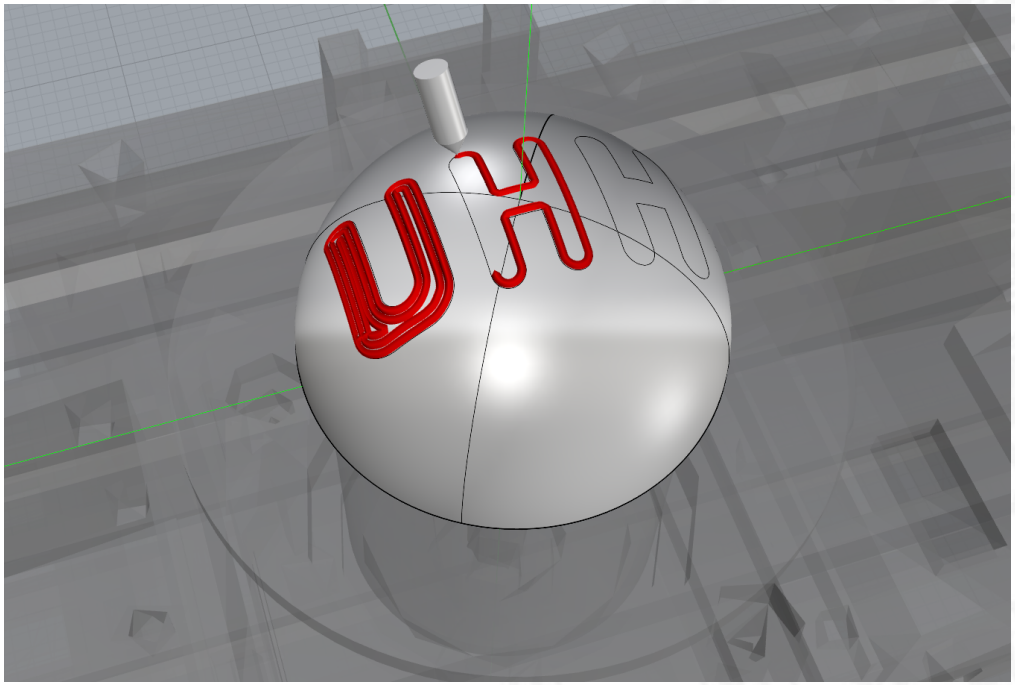
- Additional axes increase complexity in toolpath calculation
- 5-axis is a niche in the AM space
  - Unlike 3-axis, almost no software solutions available
  - Most being scripts, closed source/paid
- Open5x[4] exists, but far from general purpose slicer
- Uses visual programming environment inside Rhino to run
  $\rightarrow$ usage unintuitive for average user

Rhino

- Rhino is an extensive 3D CAD software, can be used for various applications
- Uses freeform NURBS modelling for precise representation of surfaces
- Functionality can be extended with downloadable or self-written plugins

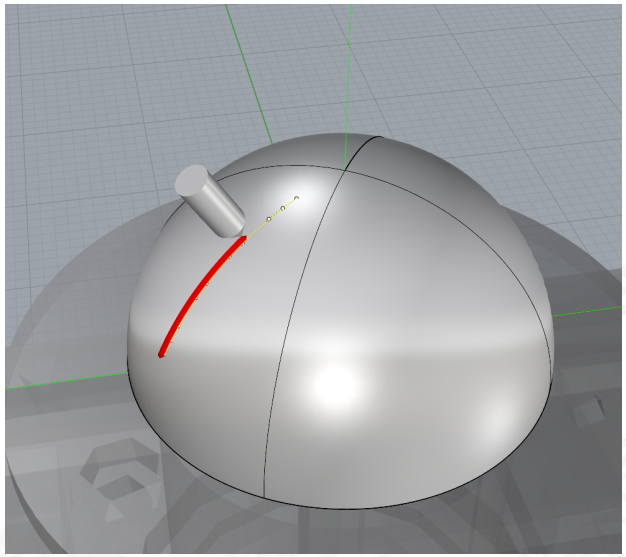


- Rhino plugin by Florens Wasserfall and Daniel Ahlers (in early development)
- Extends Rhinos functionality with 5-axis slicing
- Supports:
  - Loading machine models
  - Tool and material configuration
  - Generating G-code from points and surface normals
  - Toolpath visualization



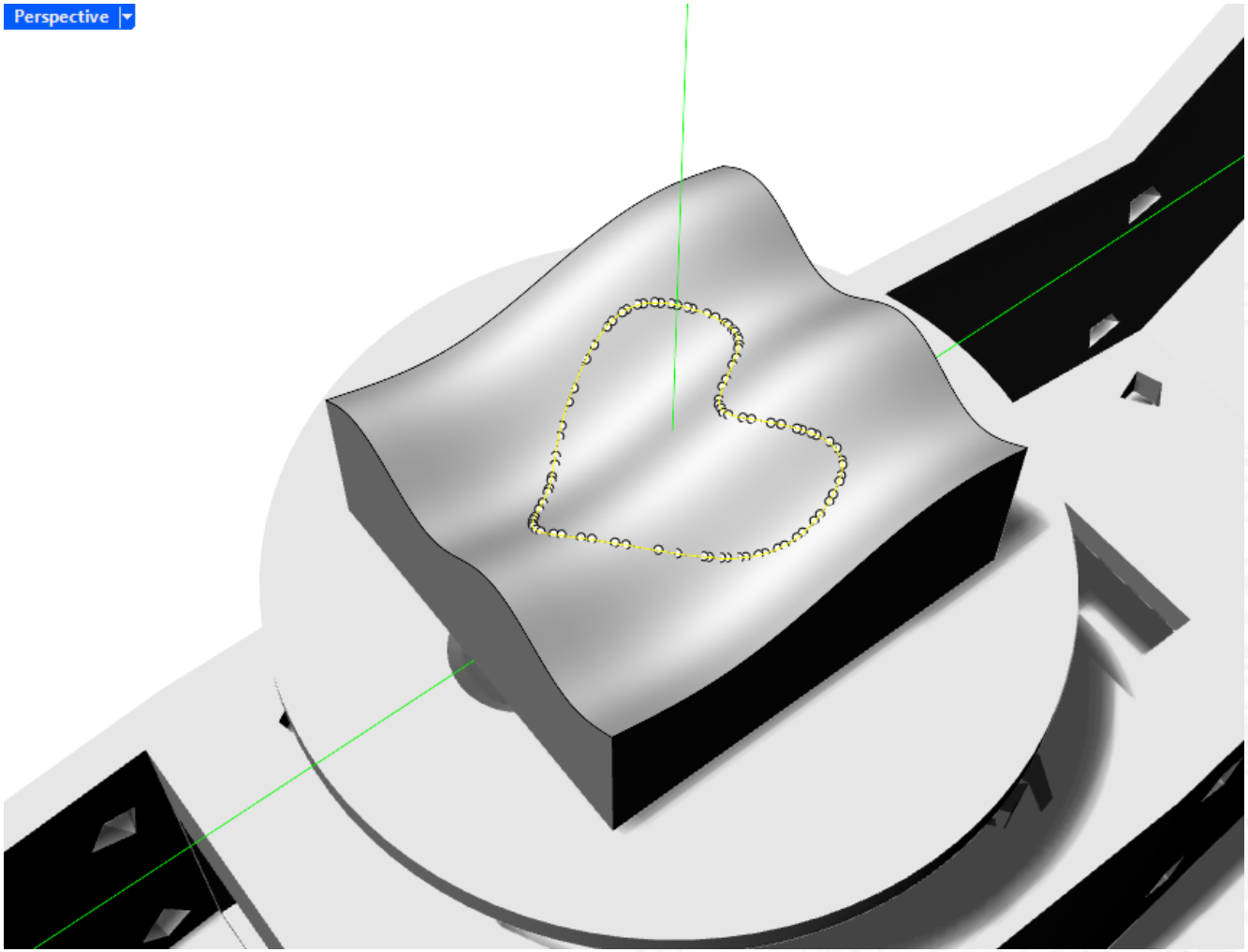
Goal

- Slicer only supports the generation of contour line toolpaths on an already printed object

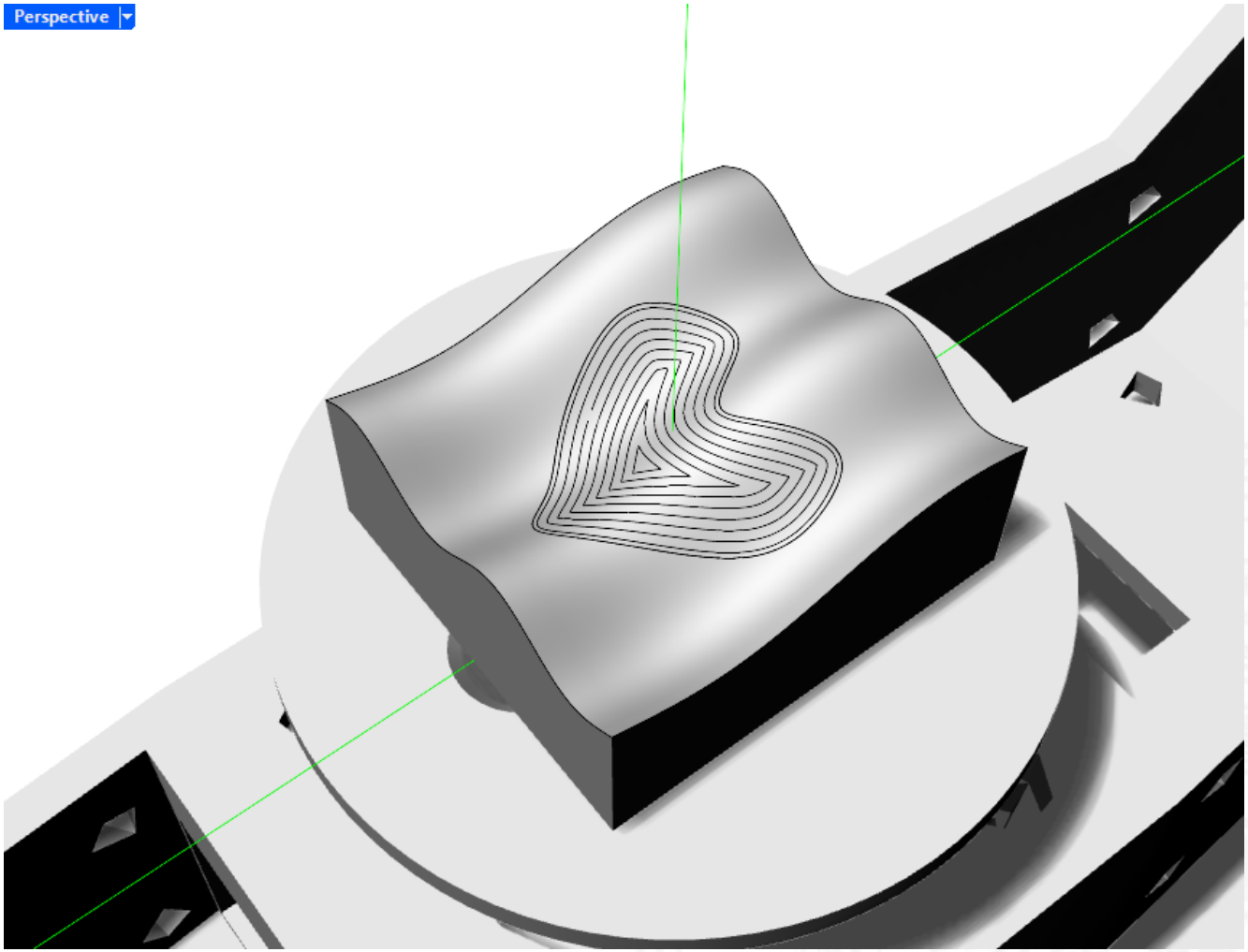


Goal: Extend Neo5x to support filling a bounded area on a surface.

1. Allow user to pick a surface and a closed curve
2. Create space-filling pattern within bounded area
3. Utilize the created pattern to generate a toolpath using the provided Neo5x interface.

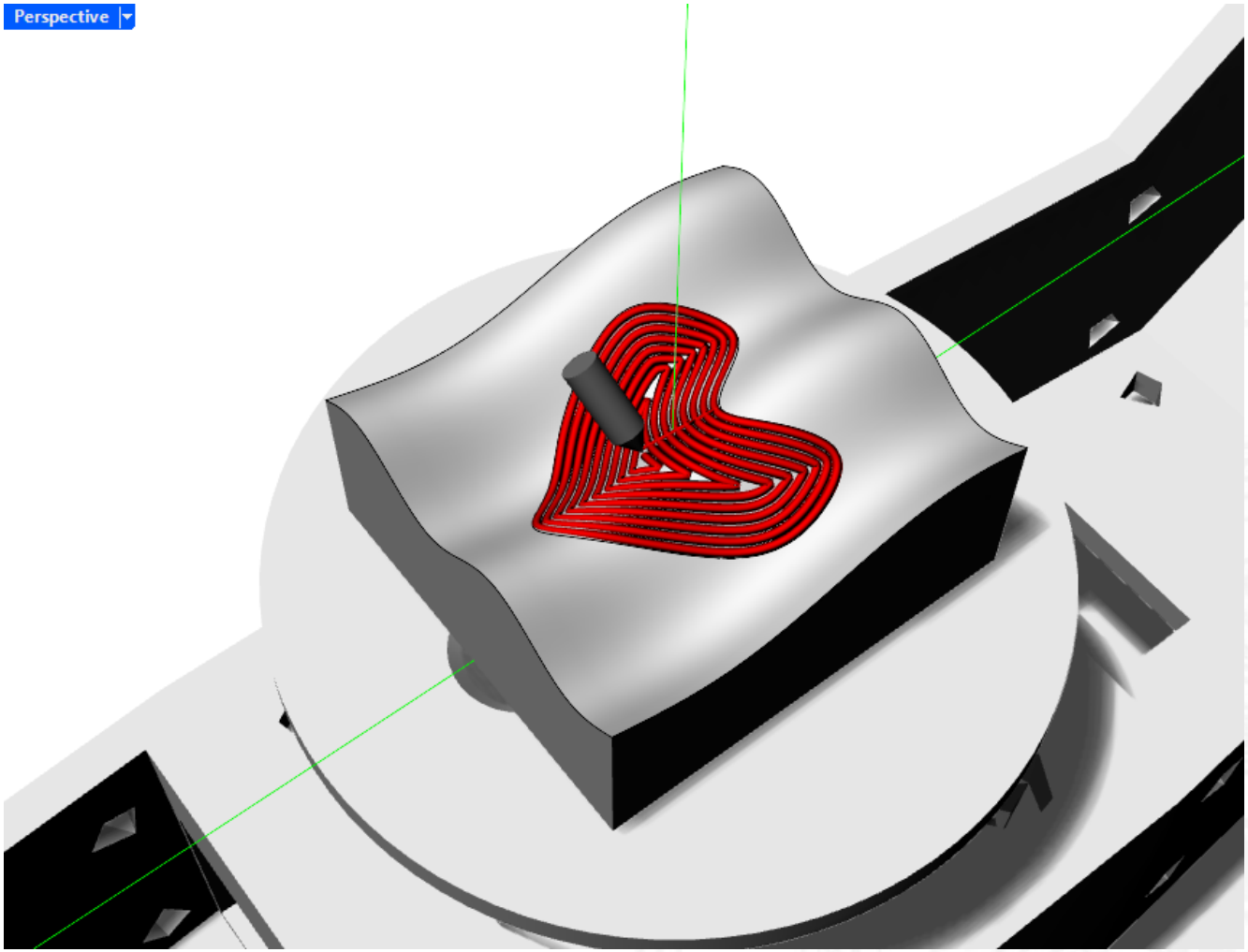


Step 1: Select surface and closed curve



Step 2: Create fill pattern within bounded area

Goal

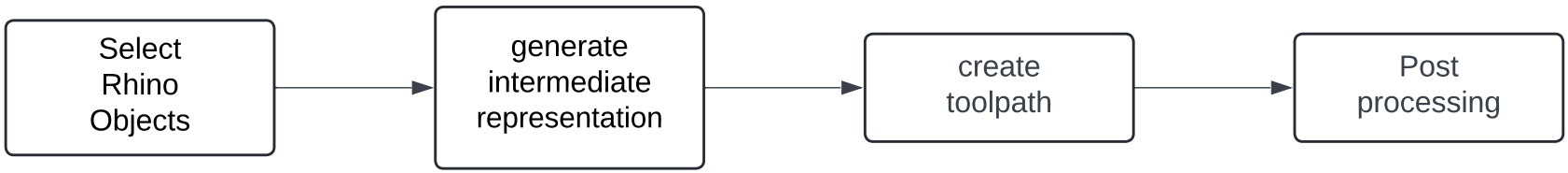


Step 3: Generate space-filling toolpath

Outline

Neo5x Interface

- User interacts with slicer through Camblocks
- A Camblock represents a workstep in a design e.g.
  - creating a contour line on an object
  - PnP operation (not implemented)
  - etc.
- Each Camblock holds config options and a collection of toolpath objects

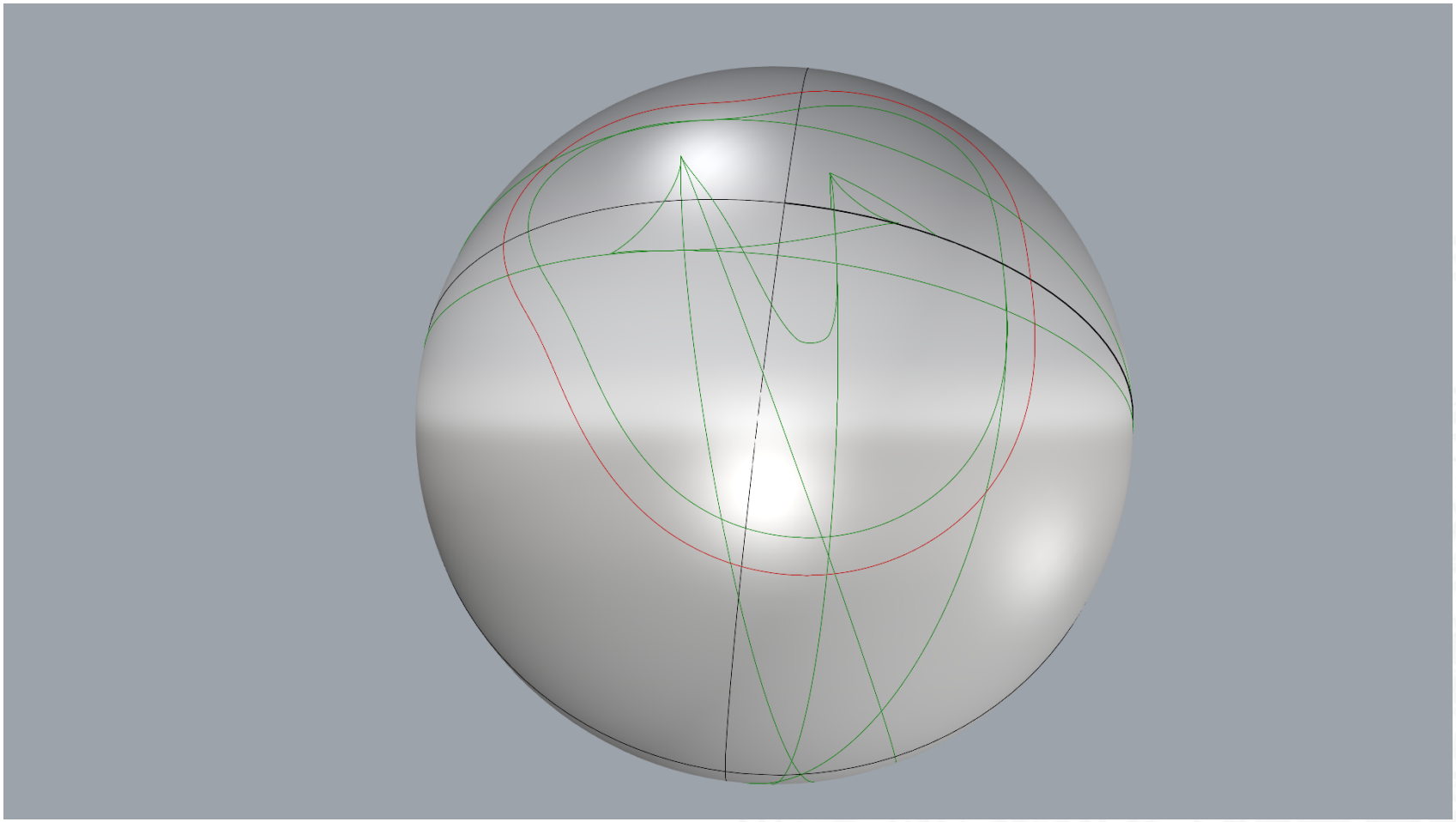


- Most important part: Space-filling curves
- Multiple ways to get such a pattern on the desired surface
- Looking at two different Methods
  1. Directly generating on surface
  2. Projecting

Direct approch

- Fortunatly Rhino does support offsetting curves directly on surfaces
- Offsetting can be used to generate simple space-filling patterns

Direct approch

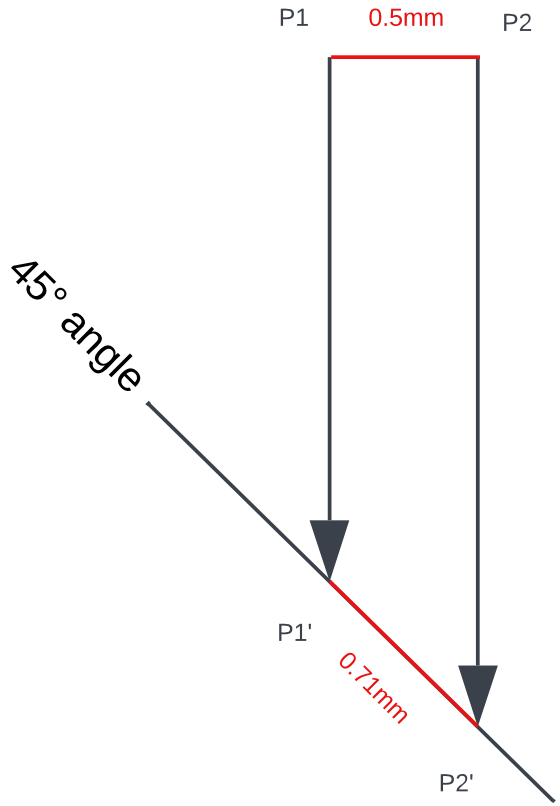


Red: initial curve Green: offset curve

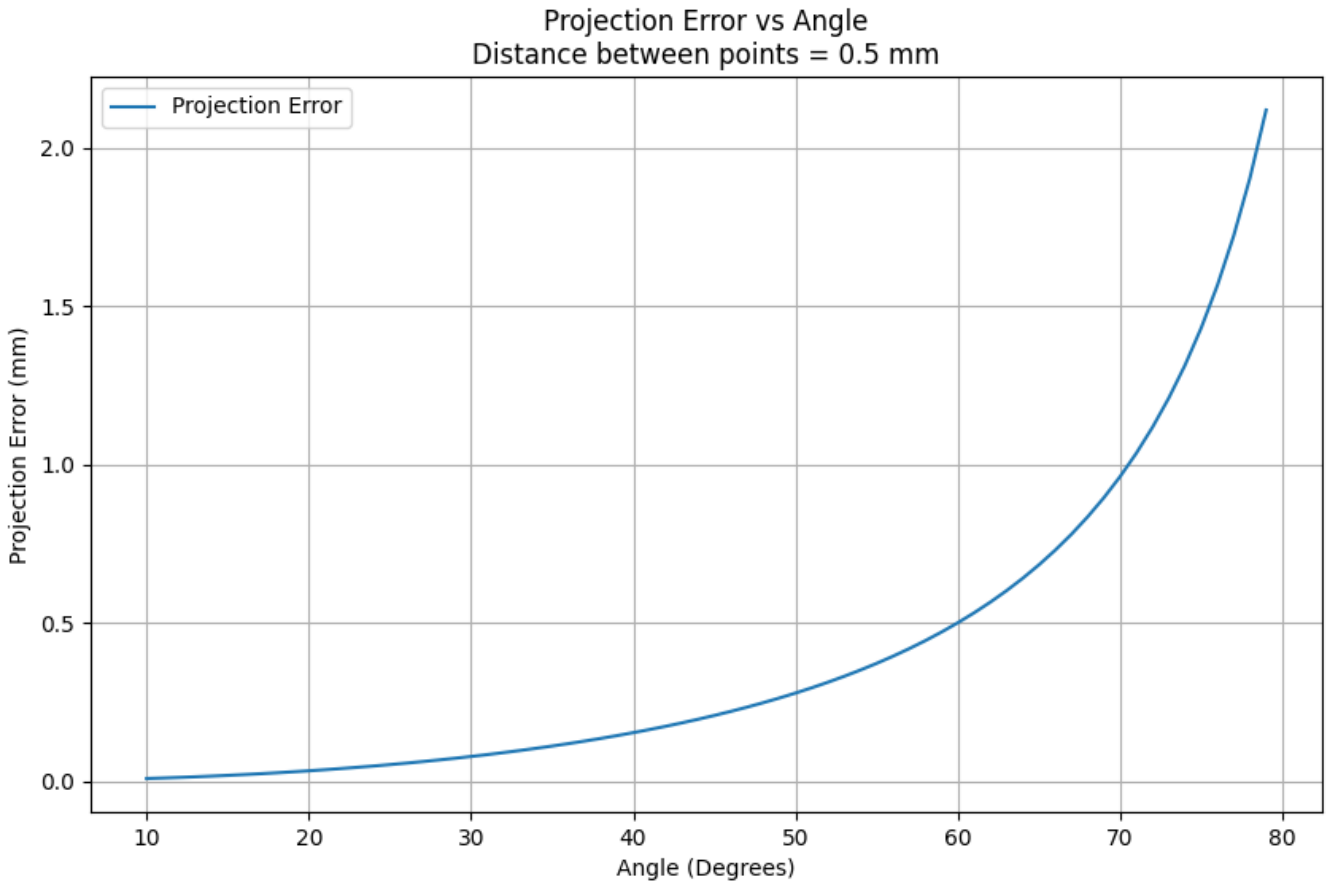
Projection Approch

- Generate space-filling pattern in 2D plane and project them to the surface
- Positives: very simple, creating pattern in 2D is easier
- Negatives: Projection error based on the slope of the surface

Projection Approch



Projection Approch



- Projecting error can be mitigated to a certain degree

Outline

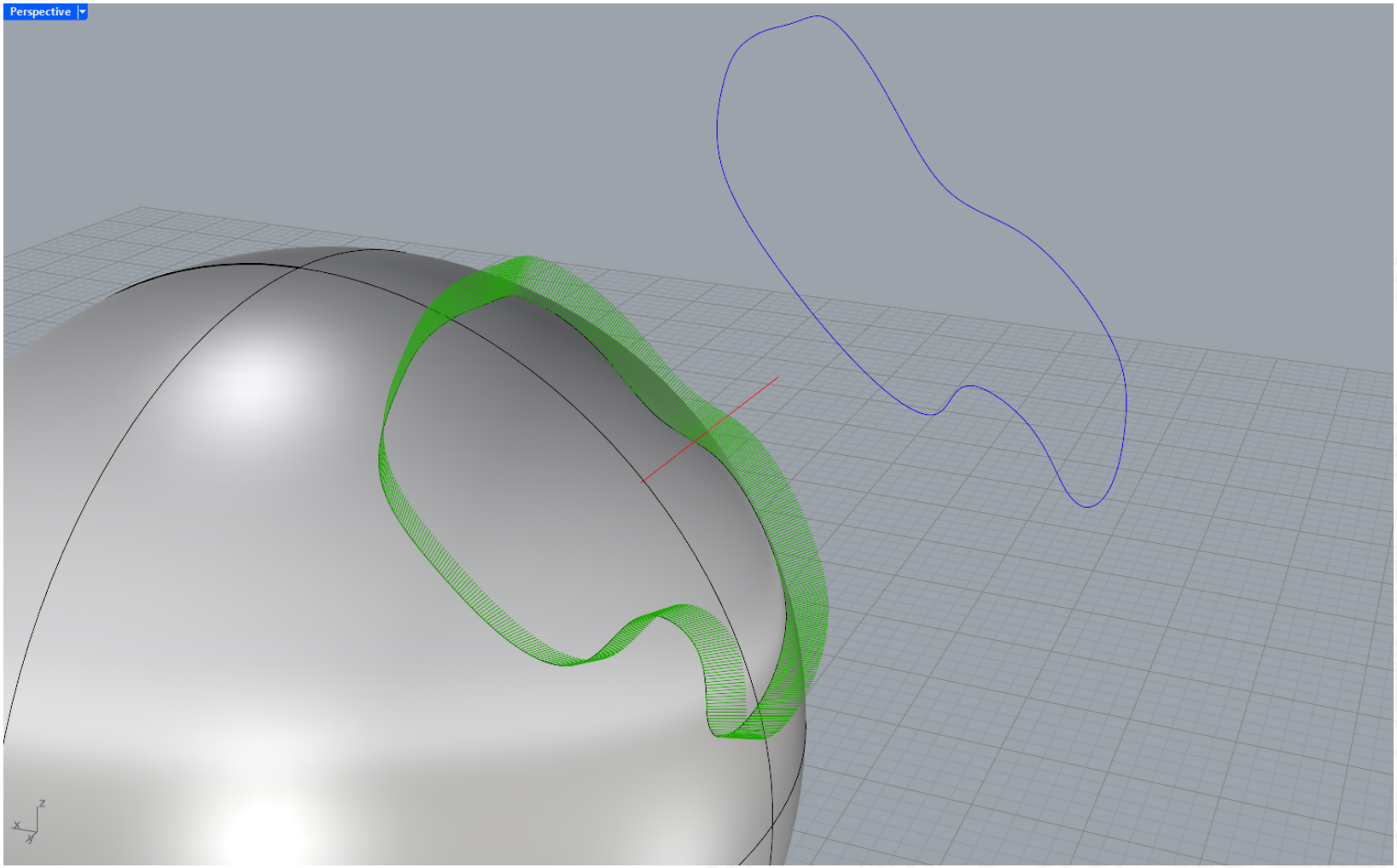
Project curve to plane

- Rhino supports the creation of planes and projecting curves onto them
- Plane needs an origin point and a normal vector

For best results plane should face in general direction of bounded surface area

1. Sample the curve points evenly and obtain surface normals
2. Average the obtained normals
3. Use the averaged normal vector to construct the plane
4. Project curve to constructed plane

Project curve to plane

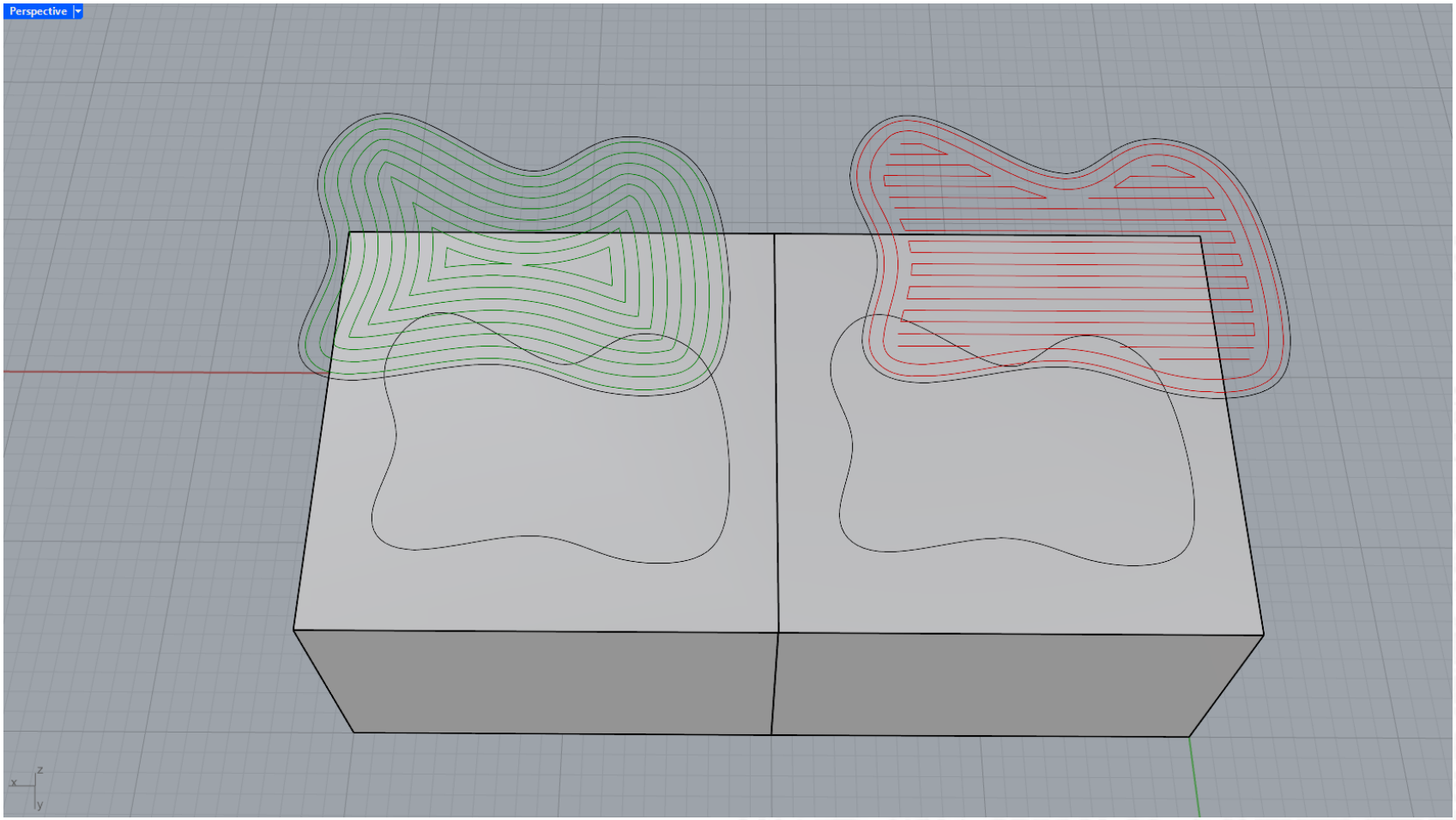


Green: sampled normals, Red: averaged normal, Blue: projected curve

Generating space-filling pattern

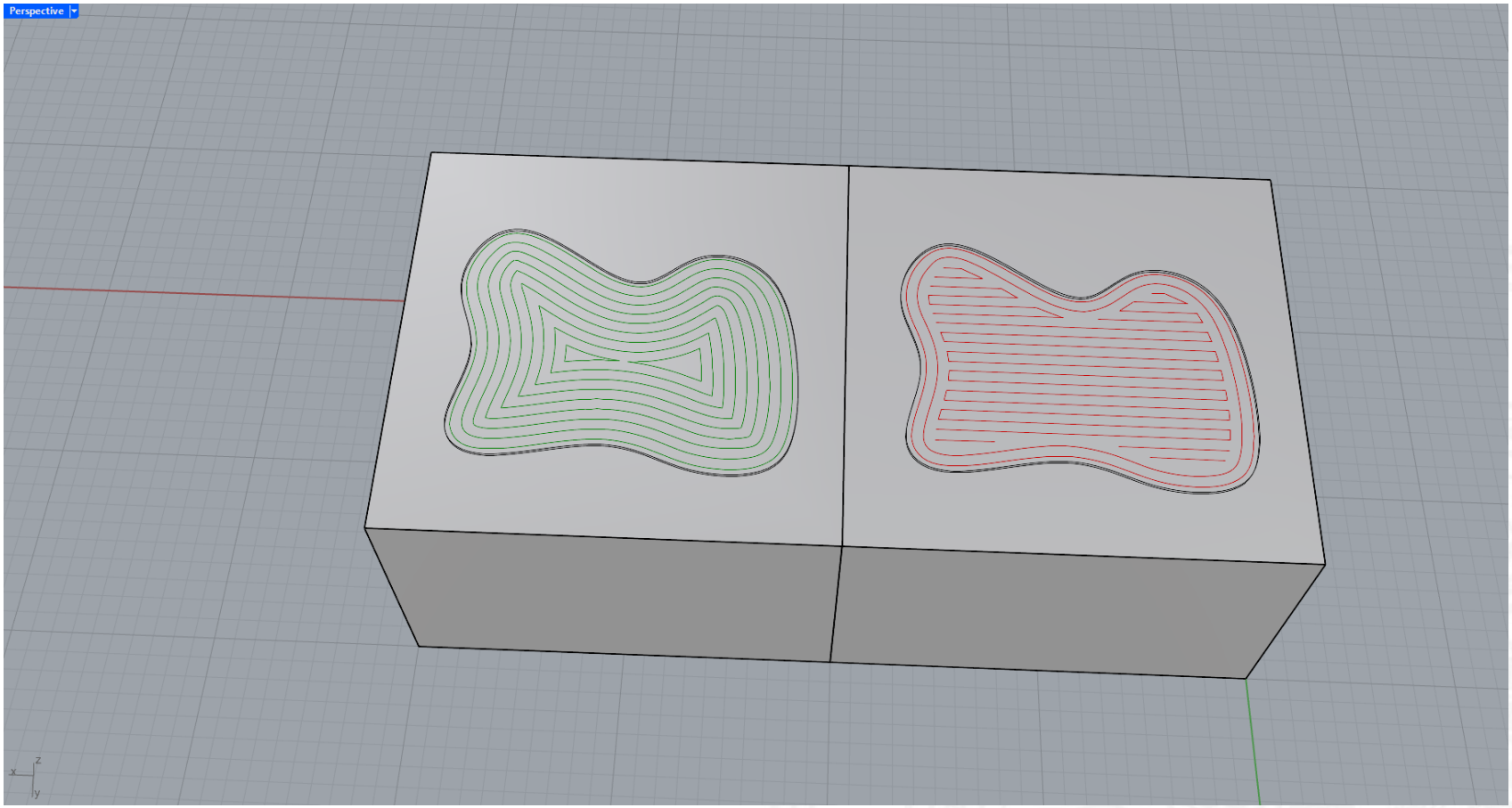
- Space-filling pattern are created in the plane
- Currently, two types of fill pattern supported:
  1. Contour-Parallel
  2. ZigZag
- Each line on the plane should be x_{mm} apart, x being the extrusion-width
- After being created, project them back to the surface

Generating space-filling pattern



Green: Contour-Parallel, Red: ZigZag

Project back to original surface



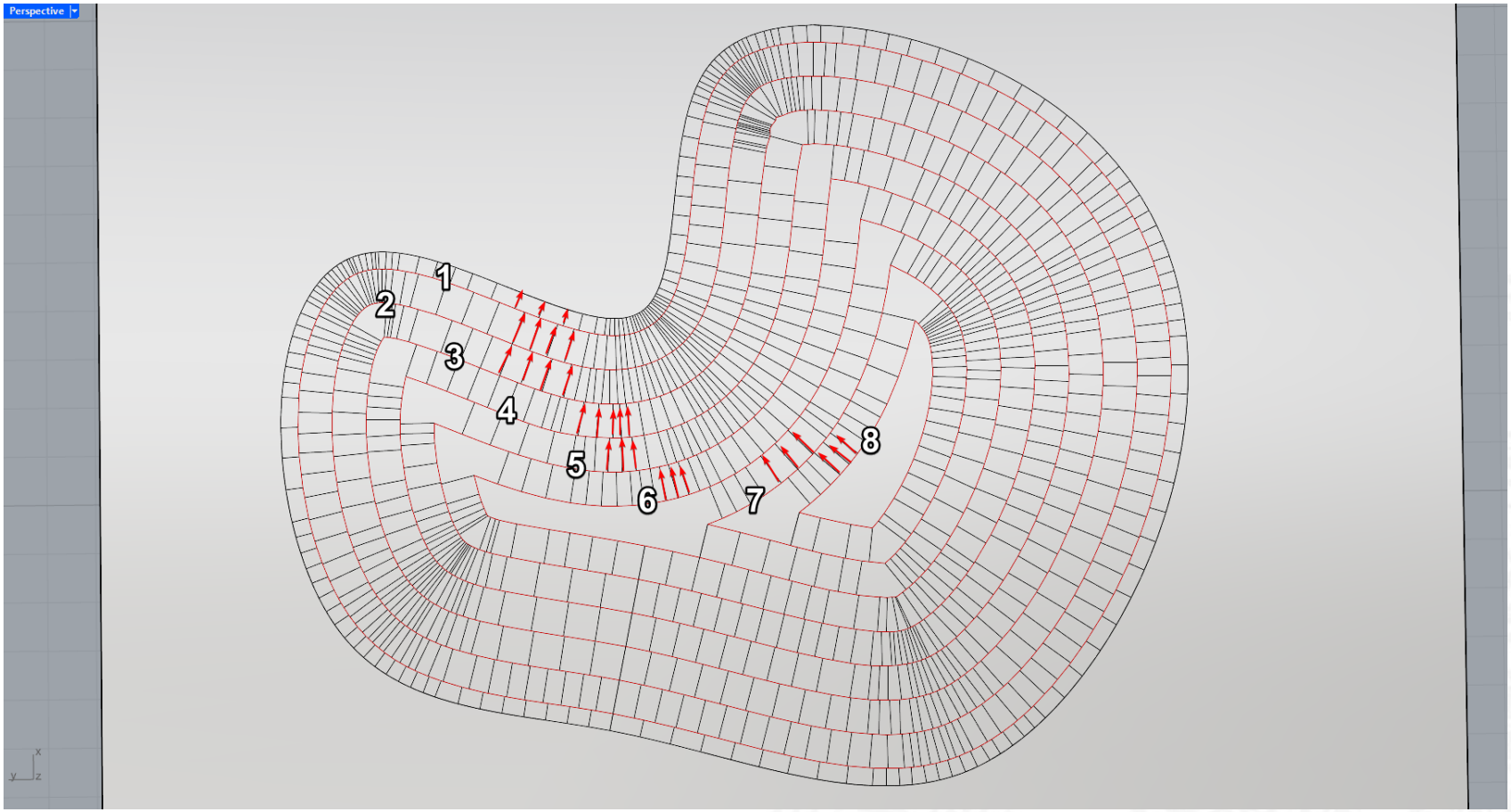
Green: Contour-Parallel, Red: ZigZag

Compensate projection error

- In some segments, lines are further apart than they should be
- Increase extrusion-width for this part
  - If you increase extrusion-width, you have to adjust it on the parallel line segment aswell

- Extrusion-width can only be adjusted to a certain degree
  - *max*: 2 · nozzle diameter
  - *min*: 0.75 · nozzle diameter
- anything beyond a certain steepness cannot be compensated with extrusion-width alone

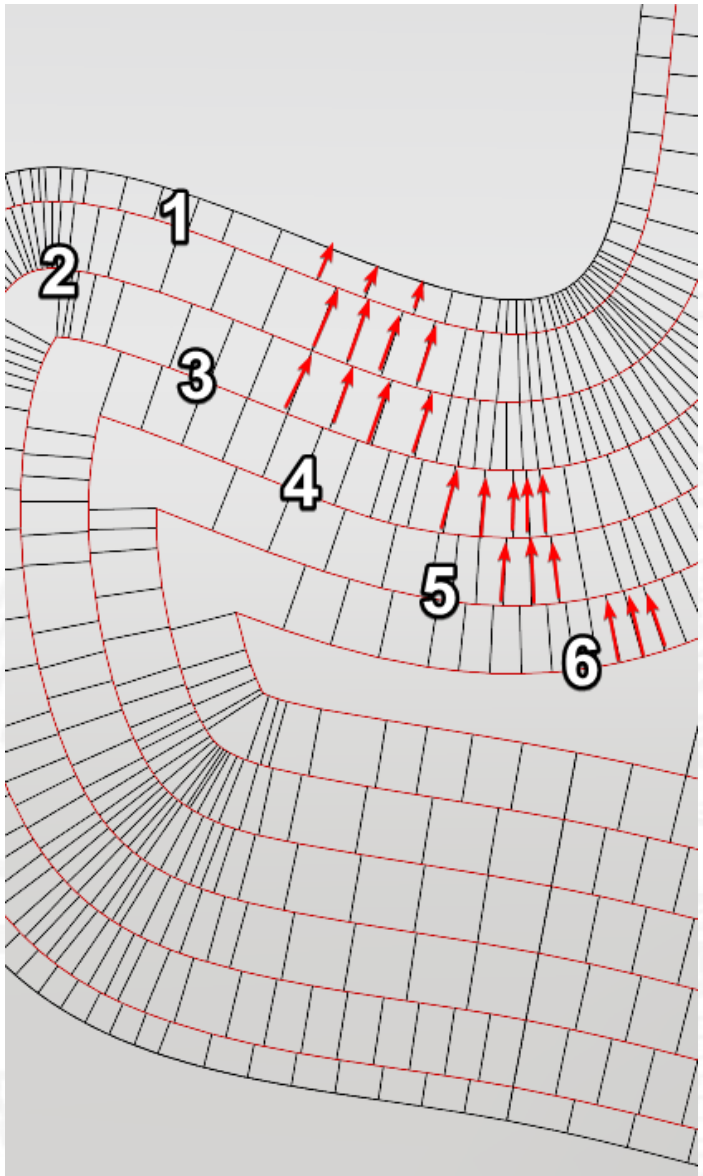
Compensate projection error



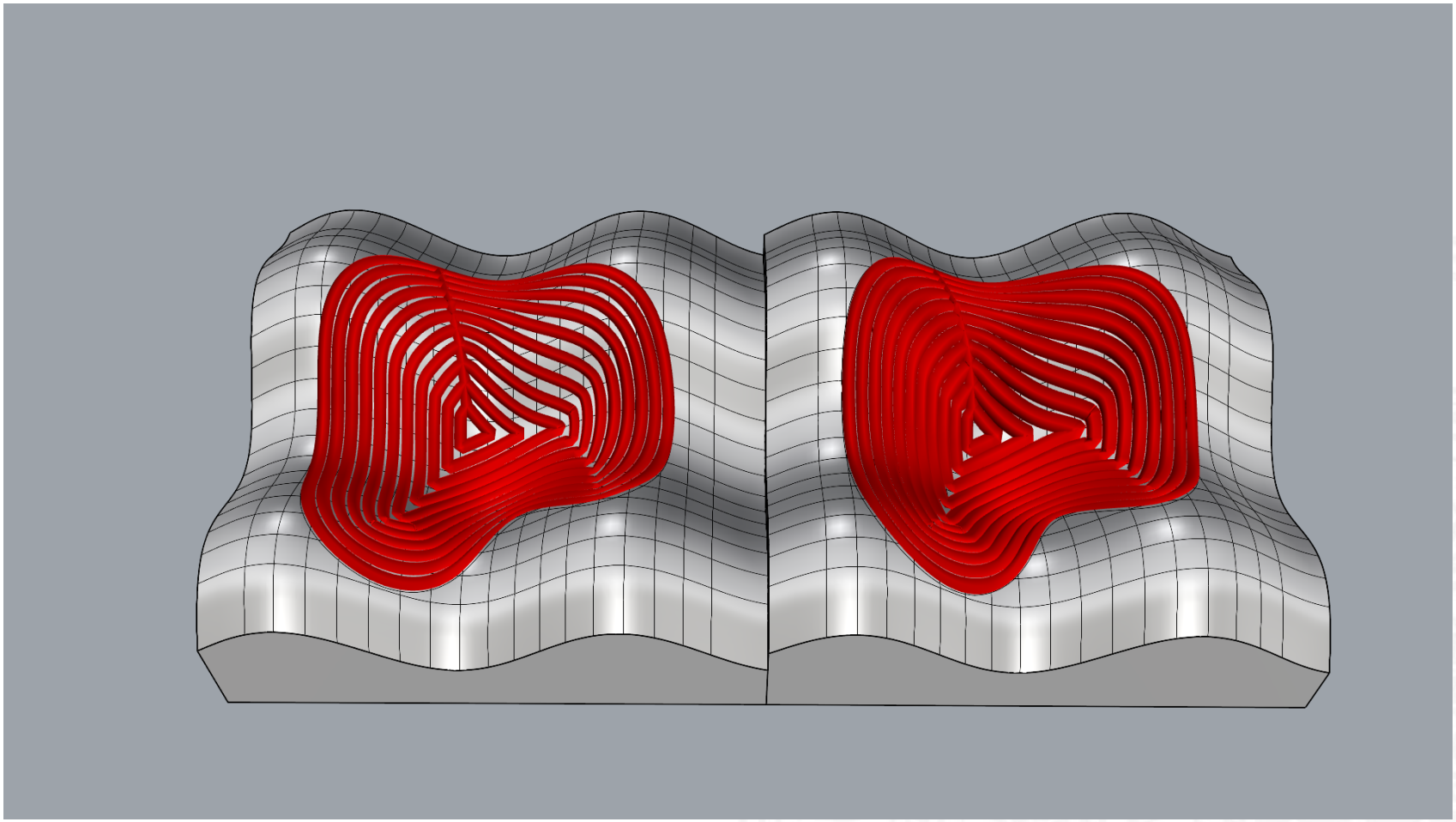
Every curve 1-8 will have its extrusion-width adjusted accordingly
Start with outer-most curve

Compensate projection error

- A section between two lines is an extrusion segment
- Calculate extrusion-width at point with $(distance - \frac{ex_{width}}{2}) \cdot 2$
  - distance = length of red line
  - width = extrusion width from neighbour-curve
- Every Curve needs to know who its neighbour-curve is
- Neighbourhood information is created while creating the pattern

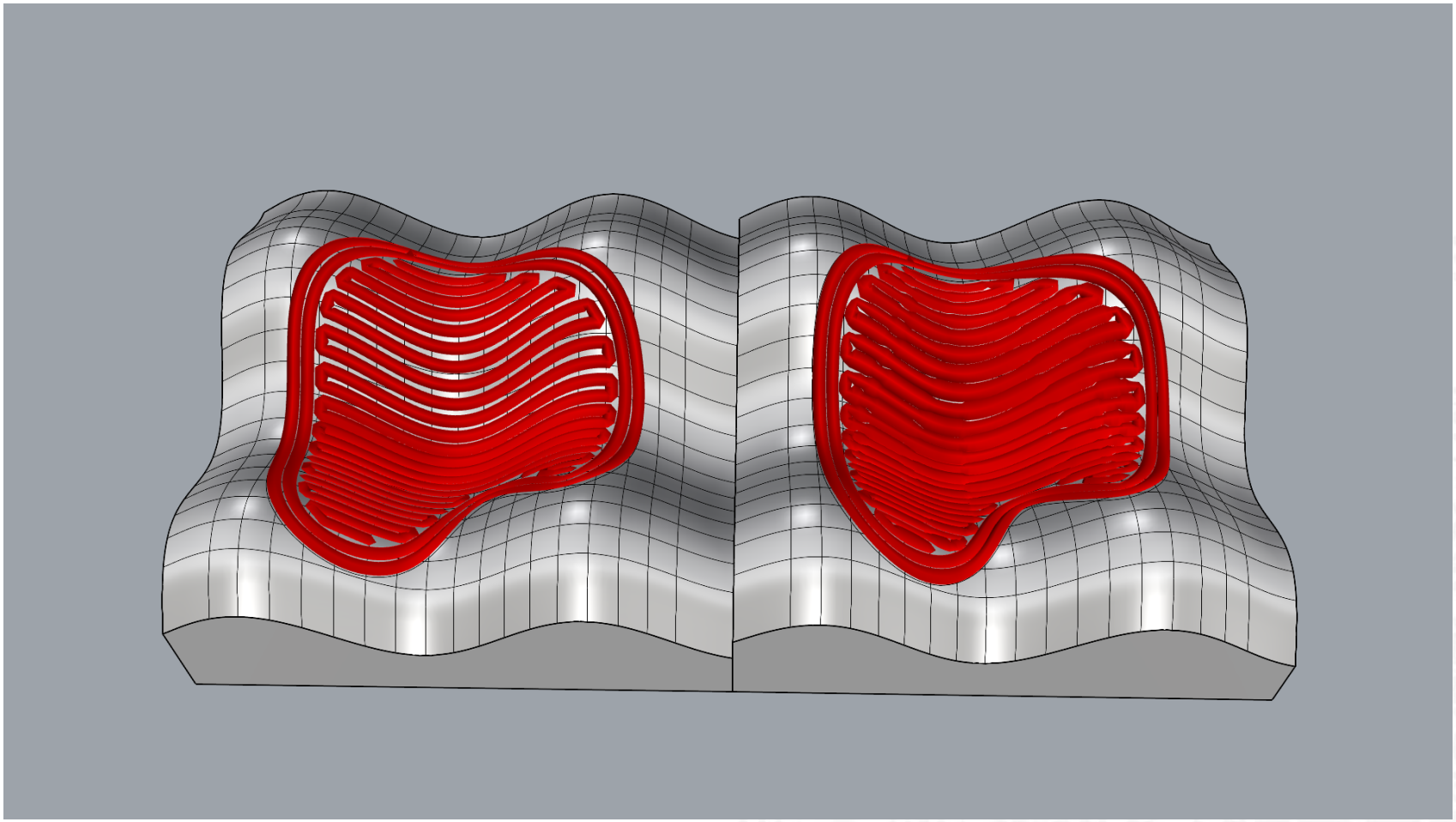


Compensate projection error



Left: without compensation, Right: with compensation

Compensate projection error

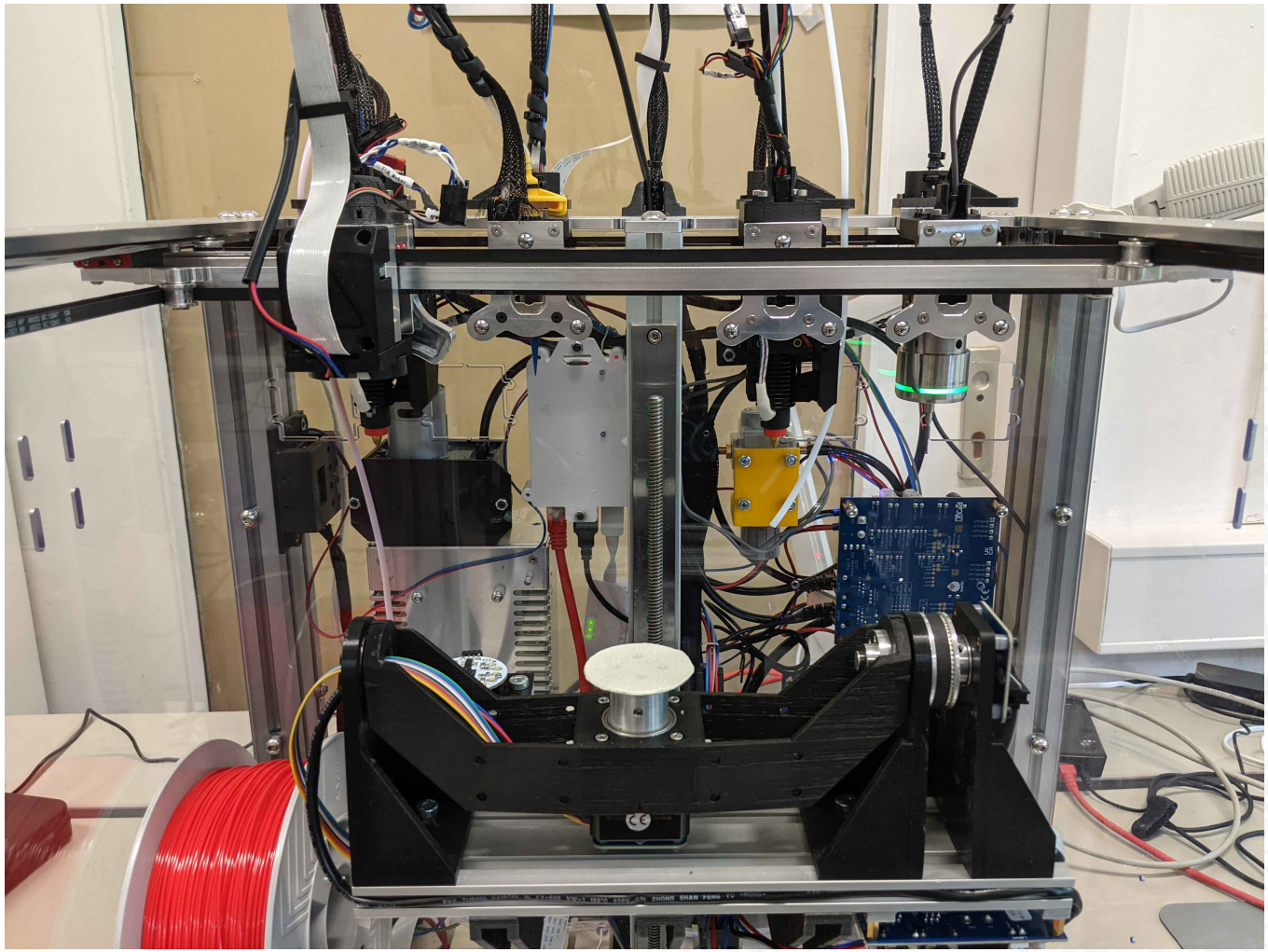


Left: without compensation, Right: with compensation

Outline

Hardware

- Printed on a modified E3D Toolchanger (Open5x)

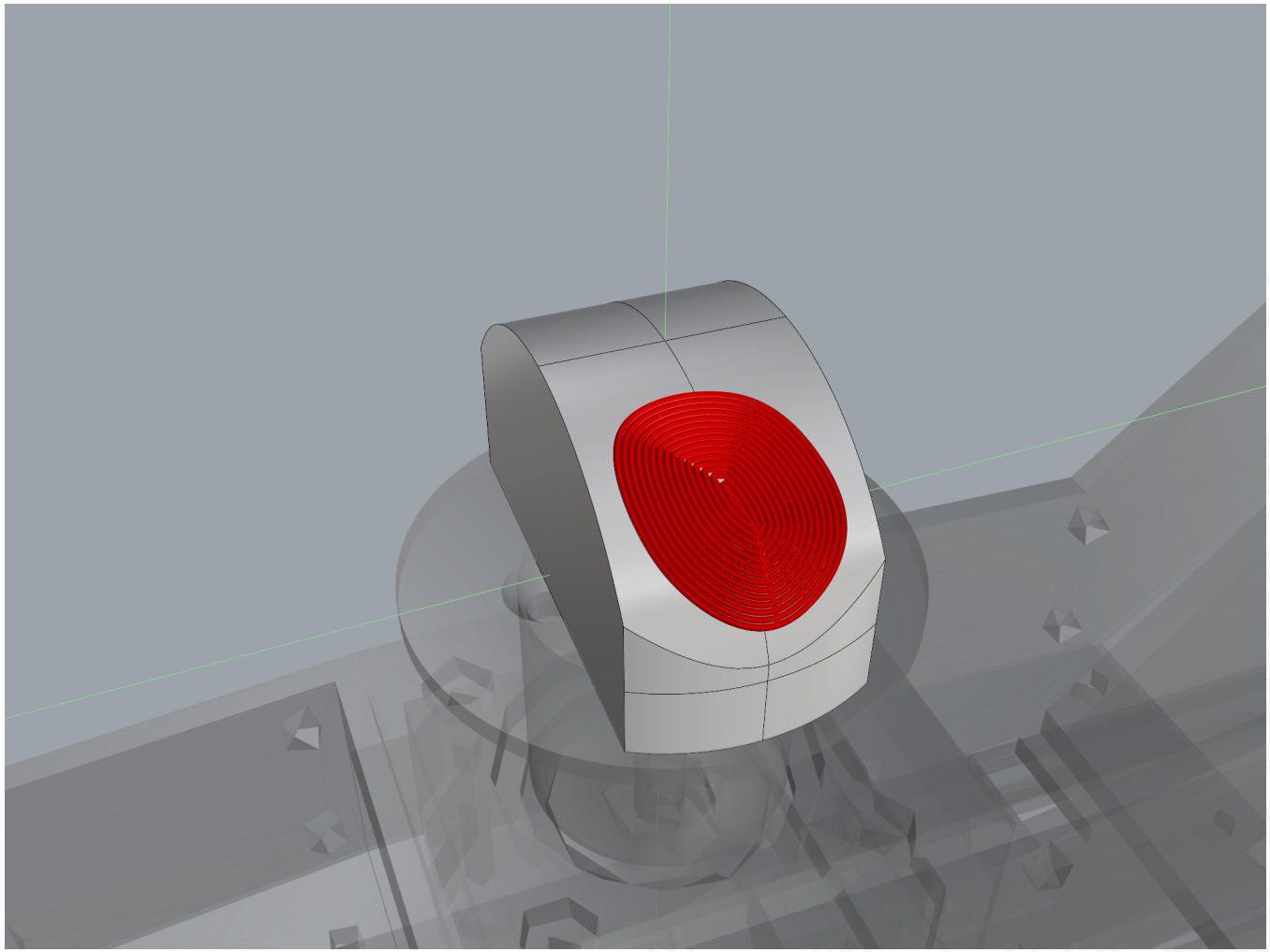


Test prints

Still time left, not all test are completed yet

- Neo5x doesn't support full 3D model slicing
  - Base object sliced with PrusaSlicer and printed with 3 axes
- Surface printing is done next as a separate job with all 5 axes

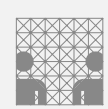
Small area: Contour-Parallel



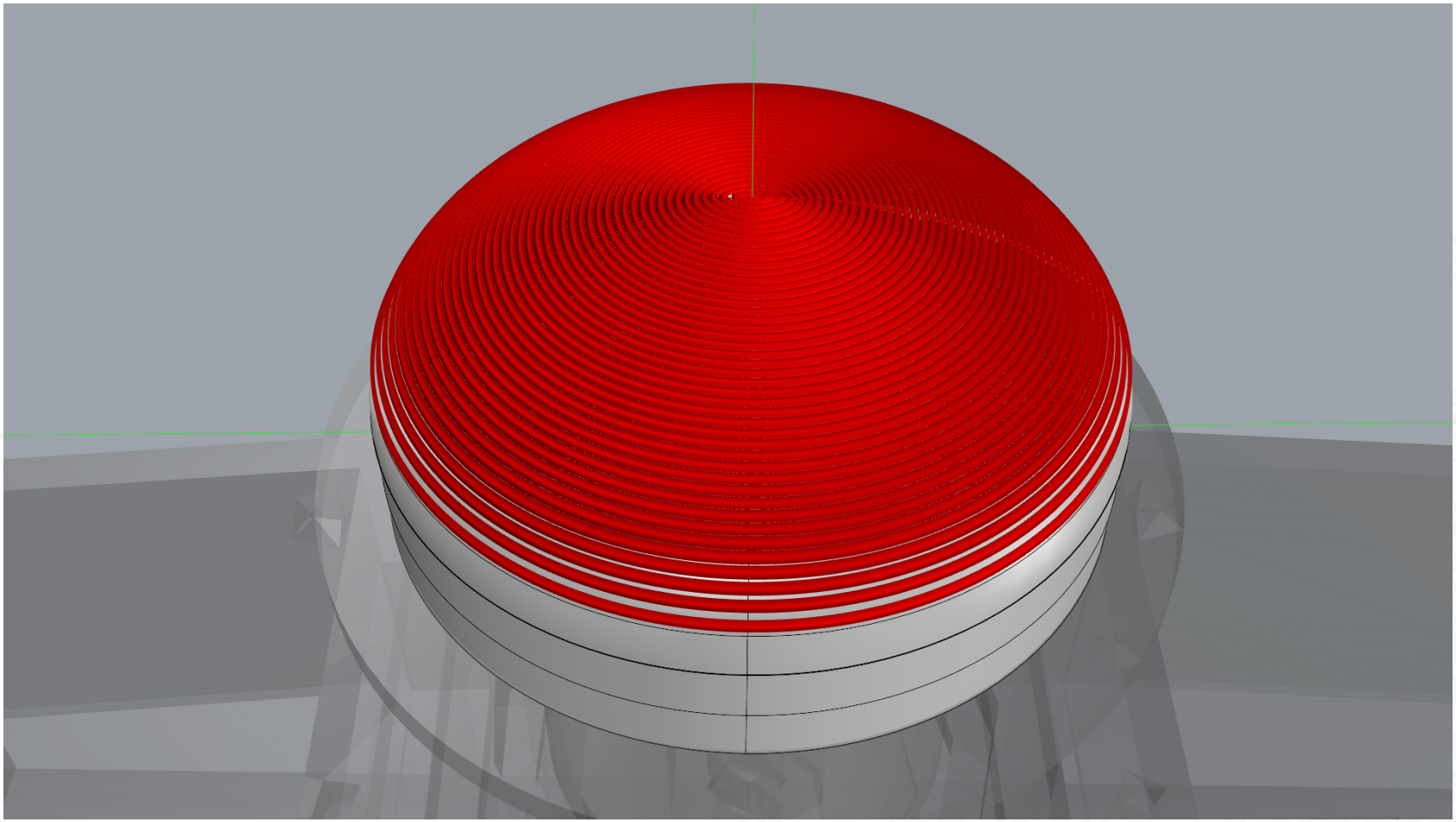
Rhino: Contour-Parallel

Small area: Contour-Parallel

Real print: Contour-Parallel



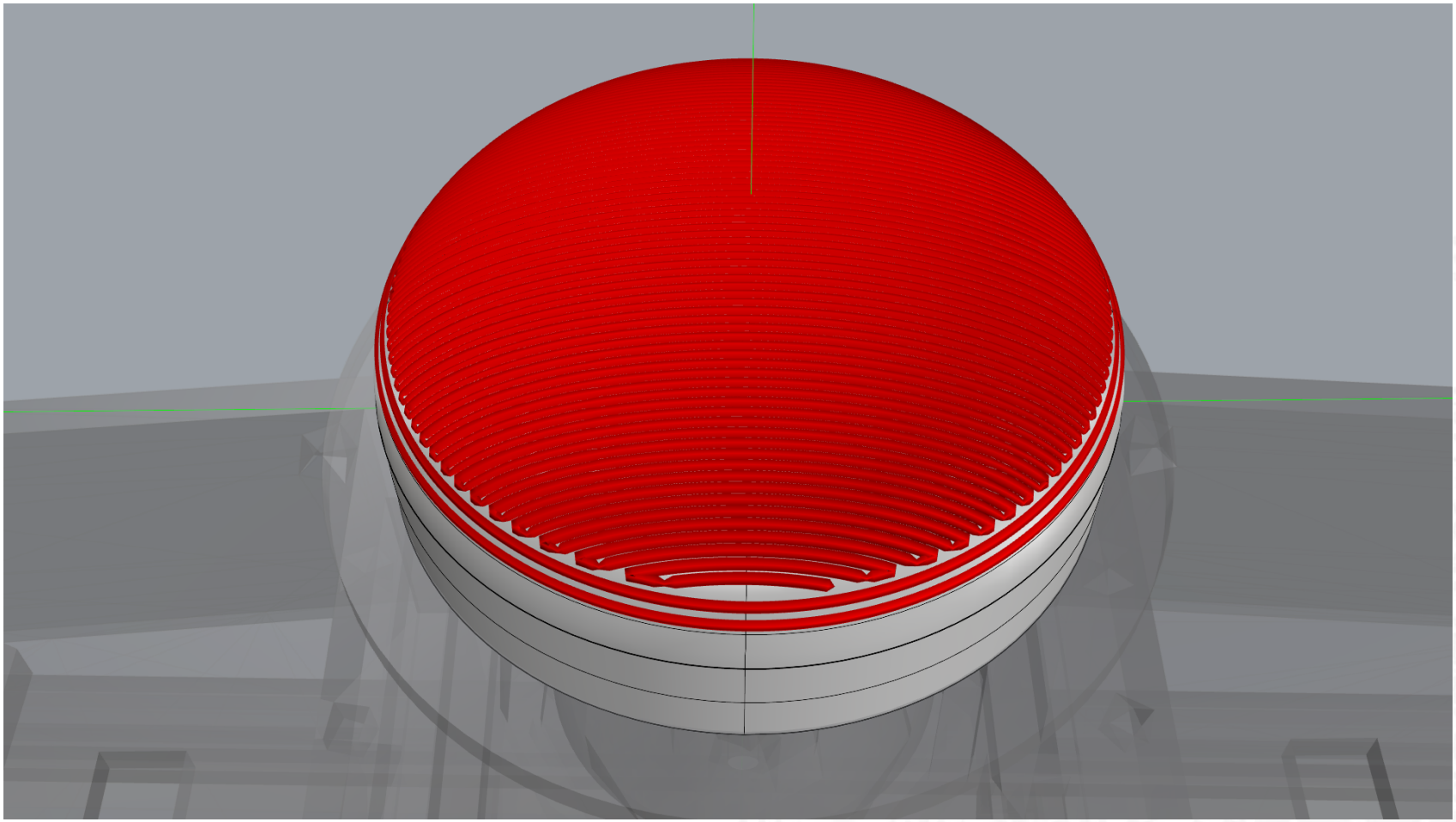
Big area: Contour-Parallel



Rhino: Contour-Parallel

Big area: Contour-Parallel

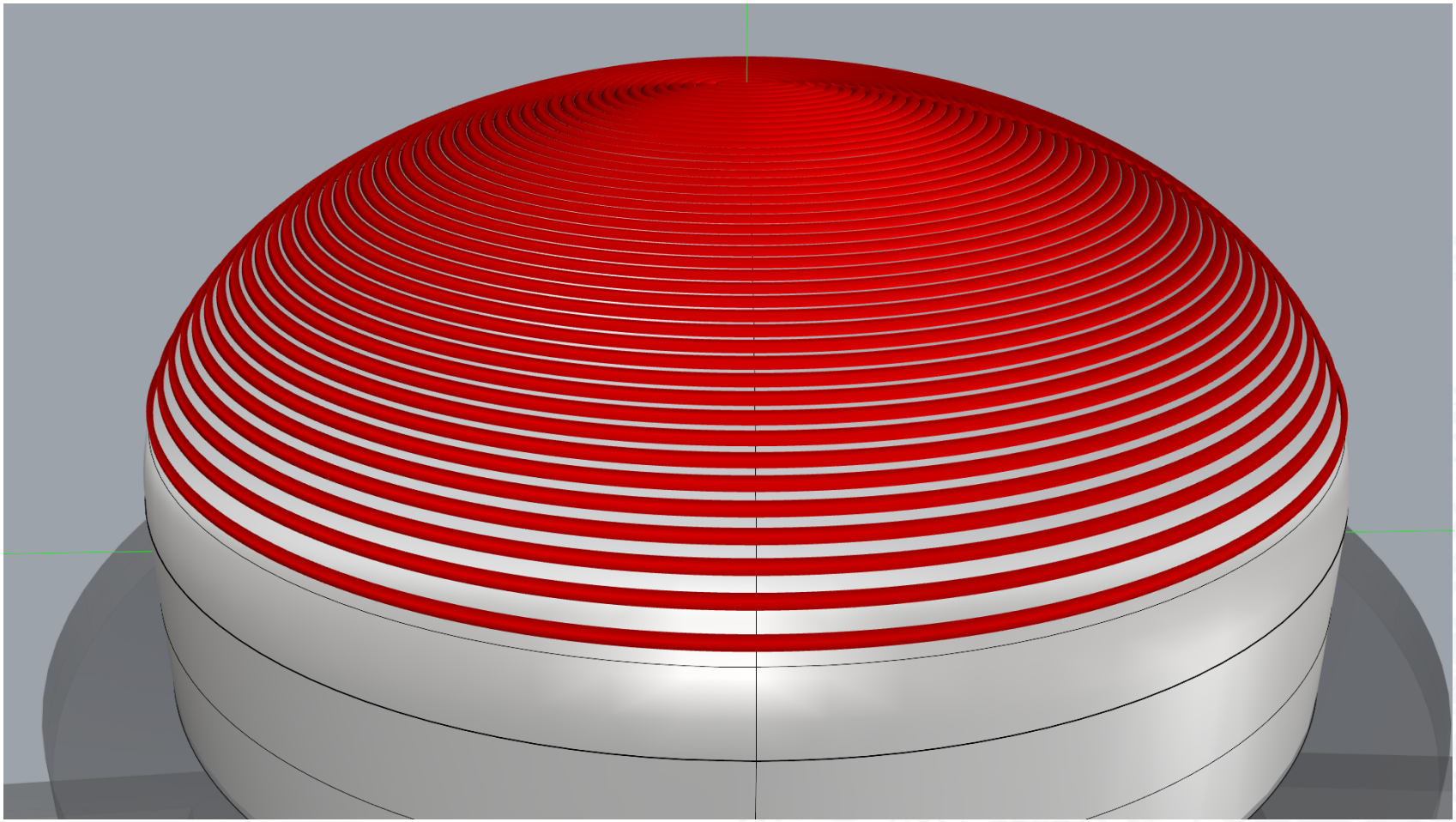
Real print: Contour-Parallel



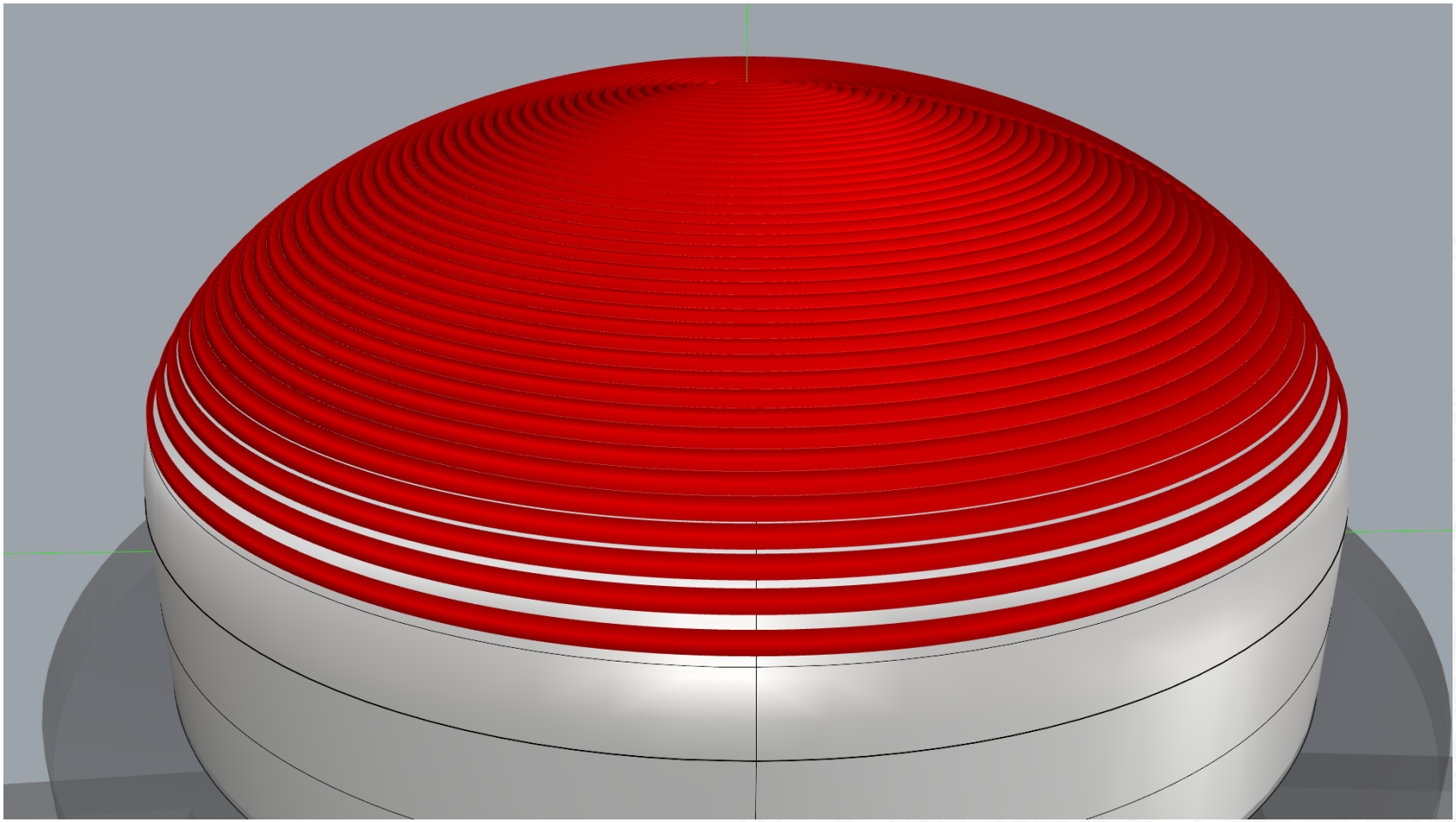
Rhino: ZigZag

Big area: ZigZag

Real print: ZigZag



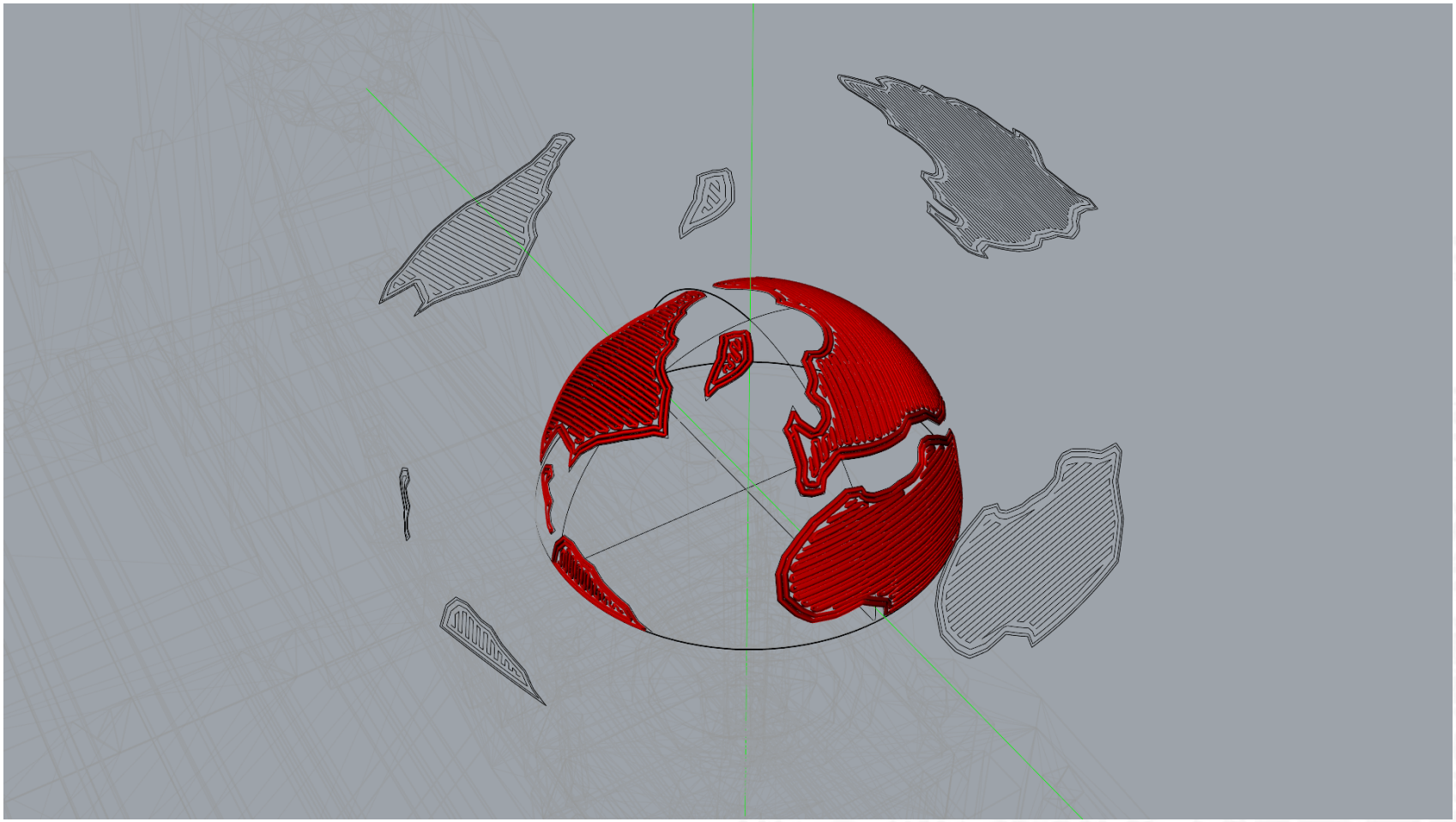
Variable extrusion off



Variable extrusion on

Variable Extrusion

Left: off, Right: on



Didn't finsih it in time, so only Rhino screenshot available

Evaluation

- Overall pretty happy with results
- Projection is not perfect: anything past 60° slope cannont realistically be compensated
  - Find those areas and create new curves
- Currently can only limit area with one curve → not usable for some applications

Outline

Conclusion

- Sucessfully extended functionality of Neo5x Plugin
- Projects space-filling curves created in a plane onto the surface
- Which are used to generate a valid toolpath
- Counteract projecting error by adjusting material extrusion

Future work

- Find a way to mitigate underfill in cases where projecting doesn't work well
- Allow for more freedom when trying to limit an area that should be filled
- Allow for printing multiple layers

Outline

References

[1] Freddie Hong et al.
Open5x: Accessible 5-axis 3d printing and conformal slicing.
In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pages 1–6, 2022.

[2] Nathaniel Kaill, R.I. Campbell, and Patrick Pradel.
A comparative study between 3-axis and 5-axis additively manufactured samples and their ability to resist compressive loading.
04 2022.

[3] B. Ramos et al.
Optimal 3d printing of complex objects in a 5-axis printer.
Optimization and Engineering, pages 1–32, 2022.

[4] Robert McNeel Associates.
McNeel Forum.
`https://discourse.mcneel.com/`, 2024.

References (cont.)

[5] Robert McNeel Associates.
Rhinoceros 3D.
`https://www.rhino3d.com/`, 2024.

[6] Consuelo Rodriguez-Padilla, Enrique Cuan-Urquizo, Armando Roman-Flores, José L. Gordillo, and Carlos Vázquez-Hurtado.
Algorithm for the conformal 3d printing on non-planar tessellated surfaces: Applicability in patterns and lattices.
Applied Sciences, 11(16):7509, 2021.