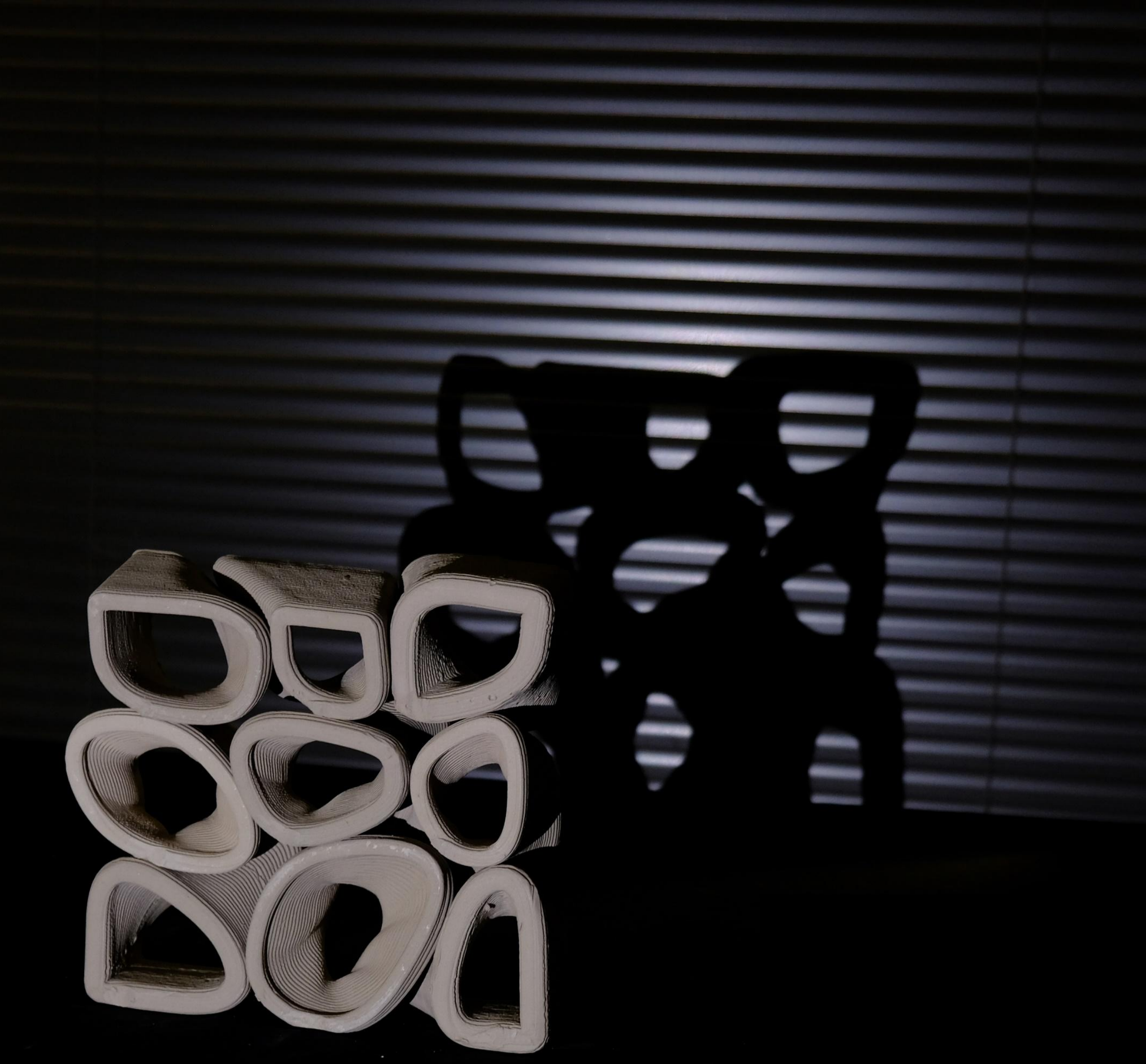
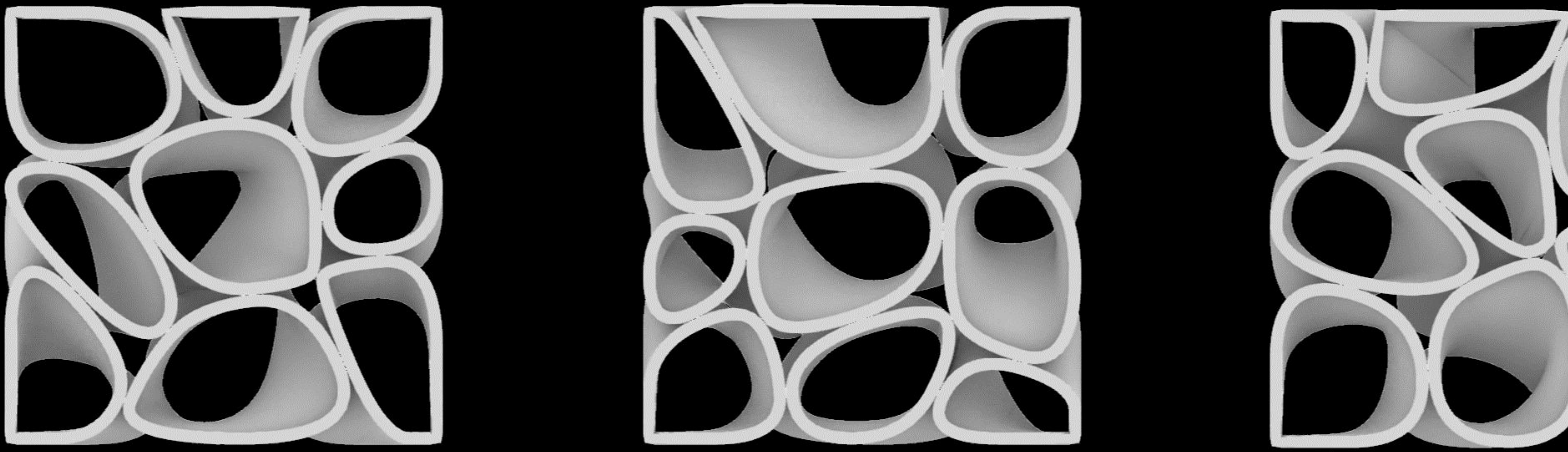
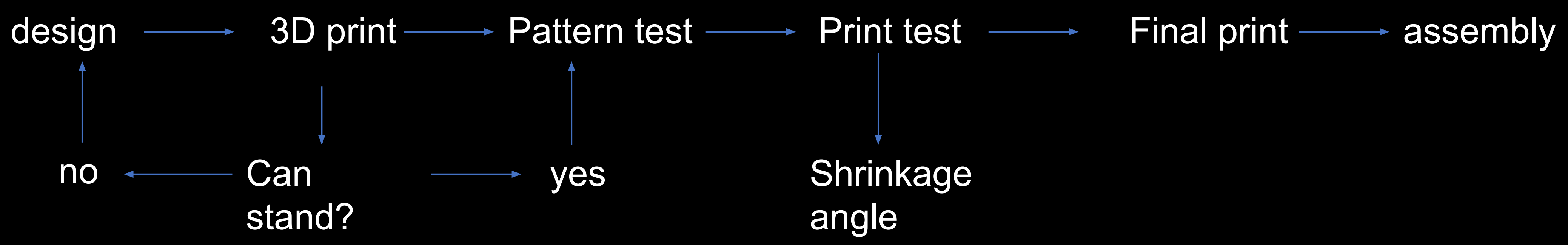
By Layer

Zhijuan Liu   Xiujin Liu   Zach Keller



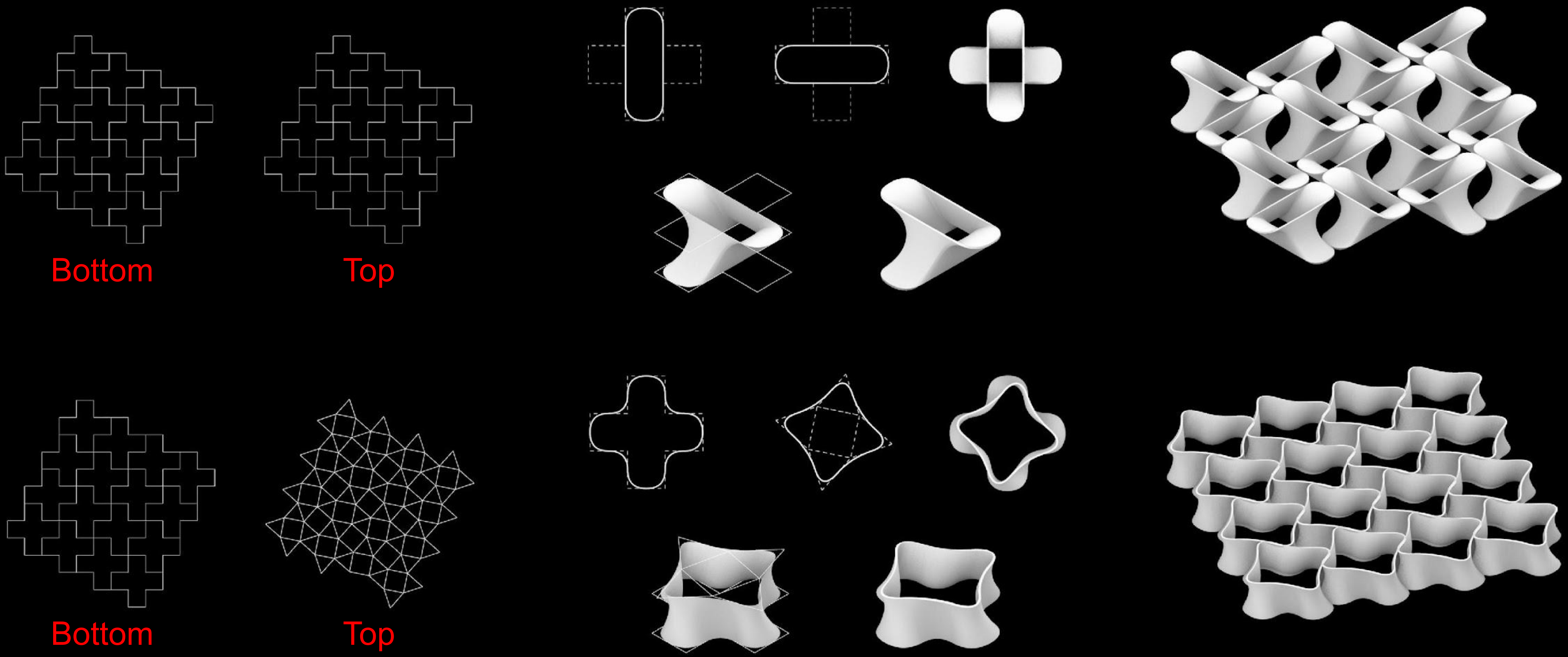


Workflow


design
3D print
Pattern test
Print test
Final print
assembly
no
Can stand?
yes
Shrinkage angle

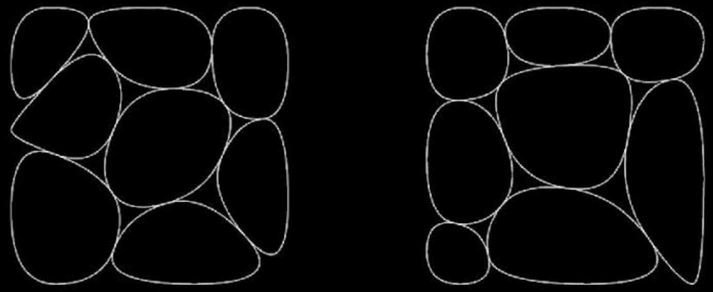
Concept Logic

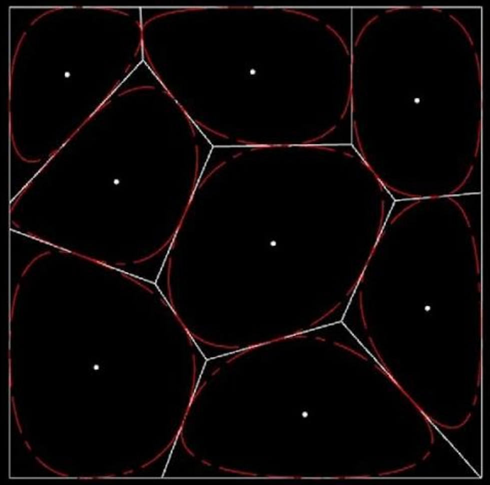
Using the top and bottom double mosaic graphics to keep the structure stable.



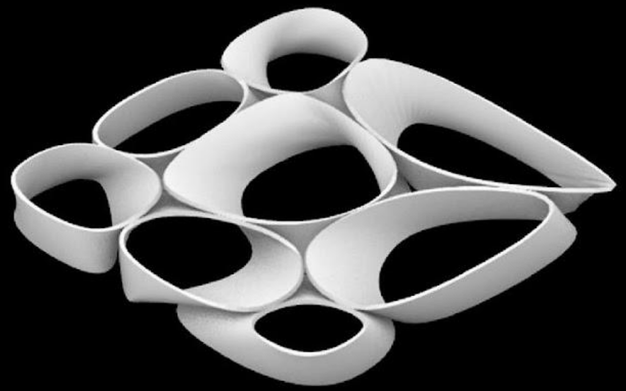
Concept Logic

Using the top and bottom double mosaic graphics to keep the structure stable.



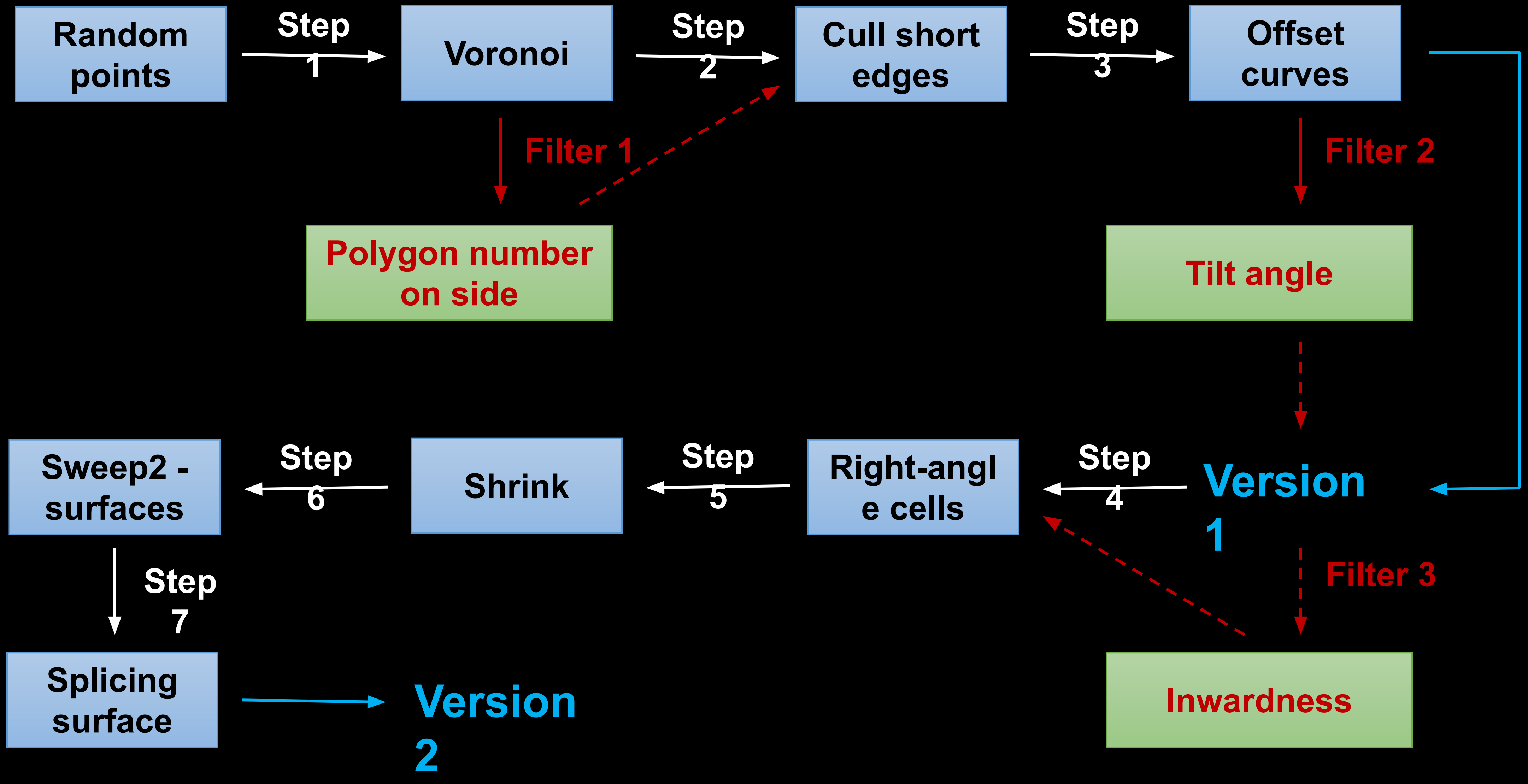


Create inlaid curves with Voronoi



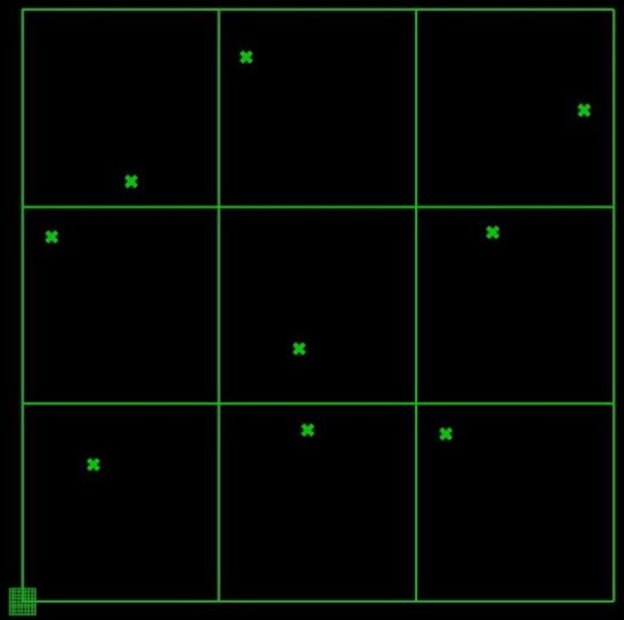
We gave up repeating the inlay pattern. Because we wanted to take advantage of clay printing. We didn't want to create repetitive pieces like bricks, but to create completely different and unique pieces.

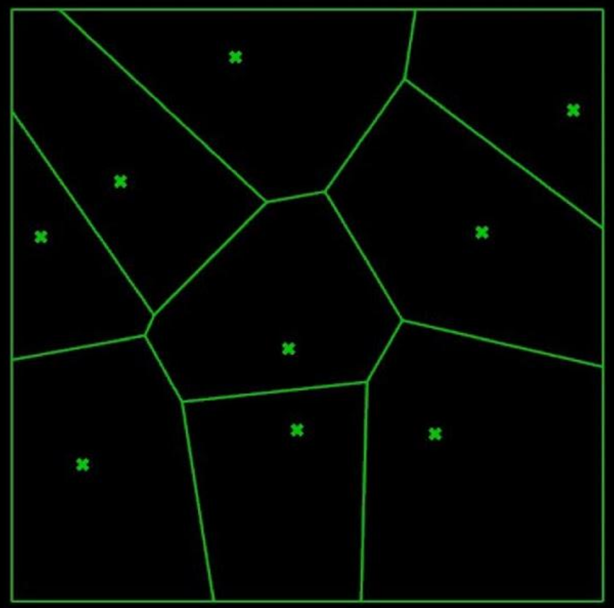
Parametric Logic of Design

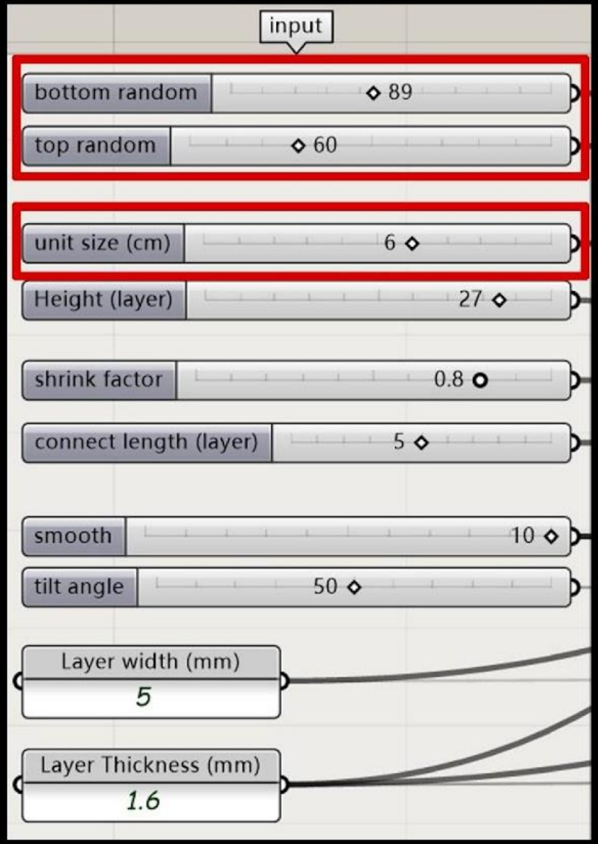


Step 1 **Random points**

Random points are generated in each grid of the nine-box grid. Then use these points to generate Voronoi. The nine-box grid ensures a roughly even distribution of points.



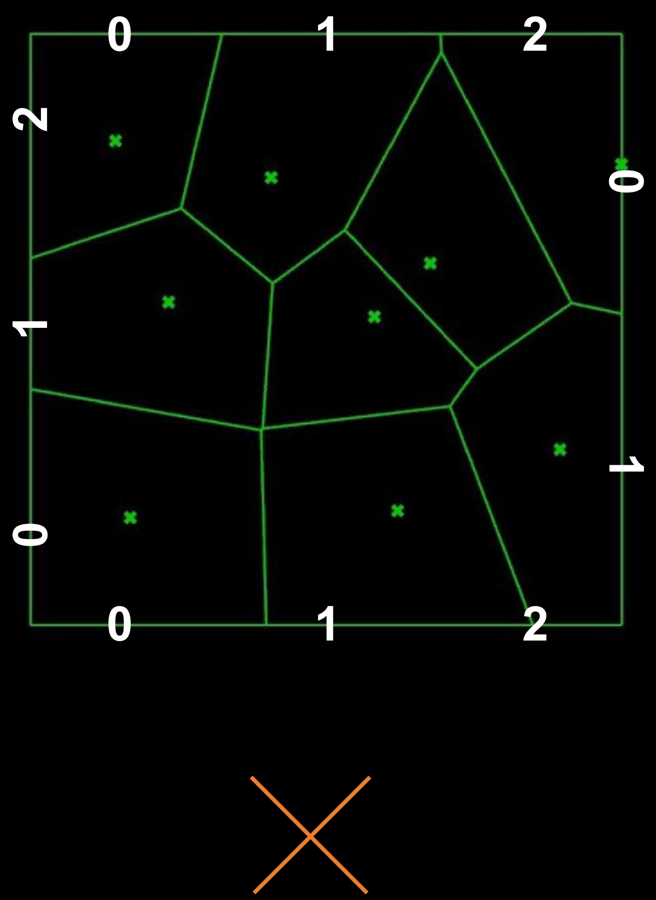


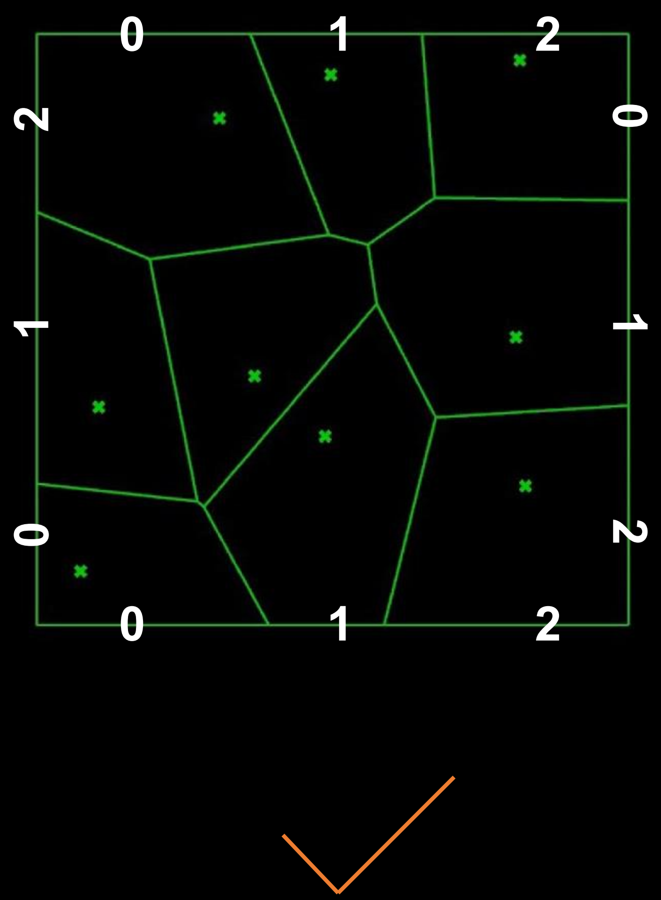


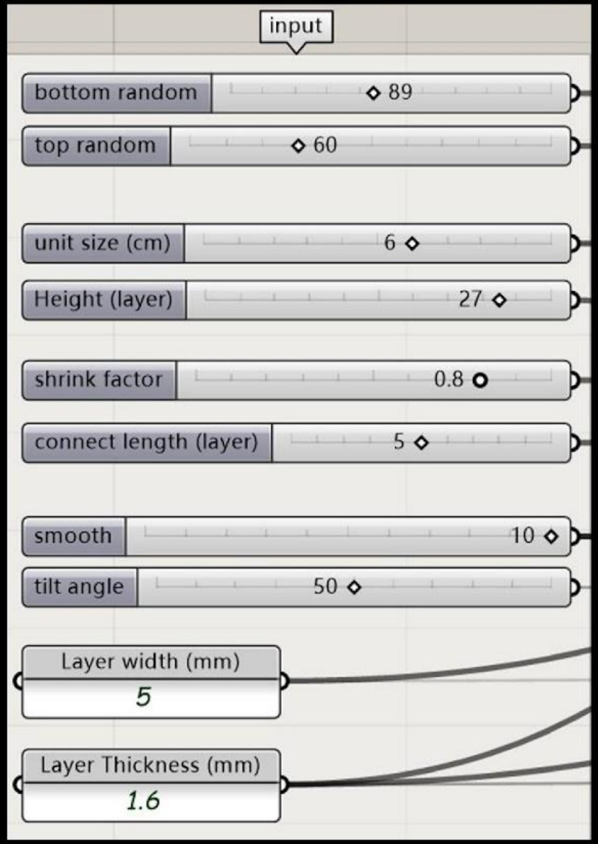
I want to produce a distribution of three curves on each side and one in the middle. This way, although the top and bottom layers have different shapes, the two curves at the top and bottom of each layer can have roughly relative positions.

Filter 1 **Polygon number on side**

Despite the limitations of the nine-box grid, Voronoi generation still leads to some uneven distribution.
The filtering is to eliminate this possibility.

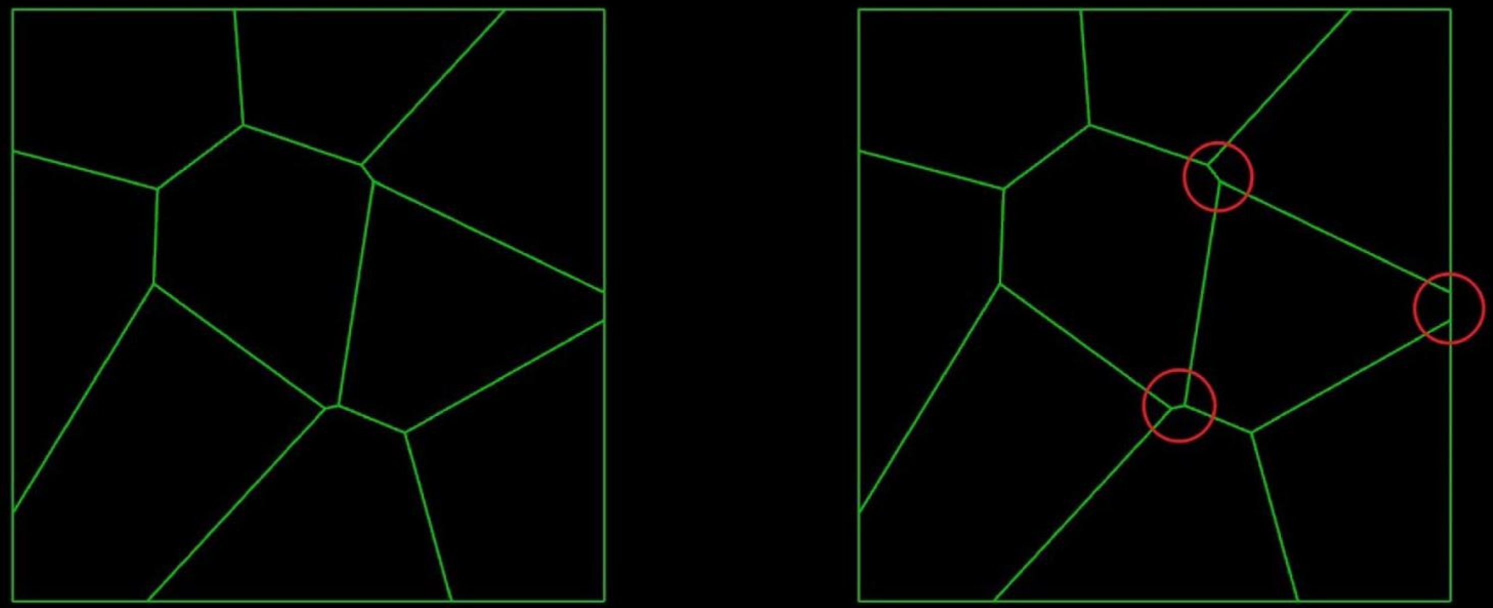


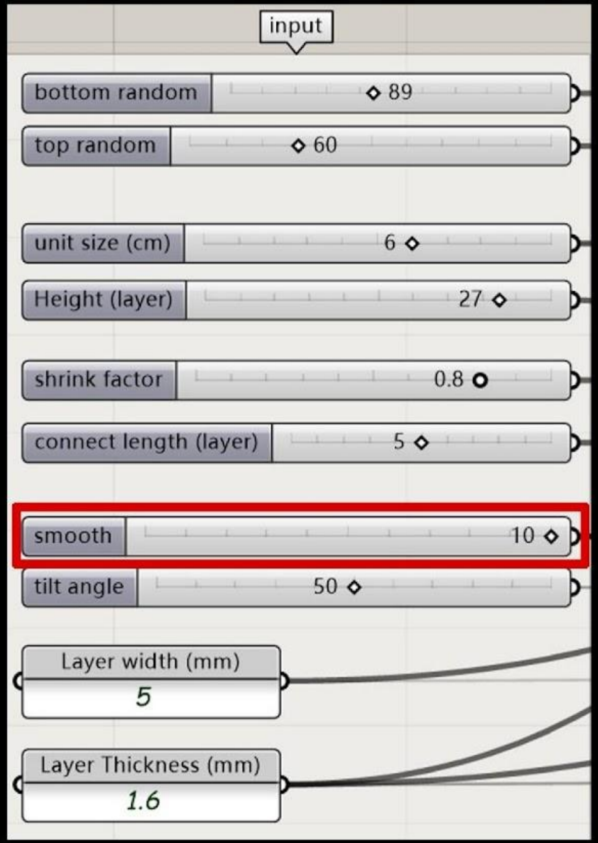




Step 2 **Cull short edges**

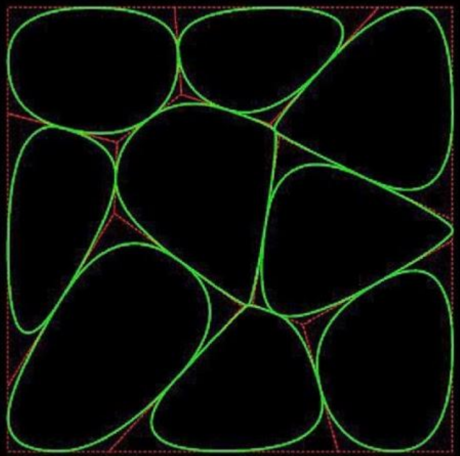
Polygons with short edges cause the curves generated by the polygon to have sharp corners. The “***smooth***" parameter in ***input*** represents the shortest length that can be allowed to exist.



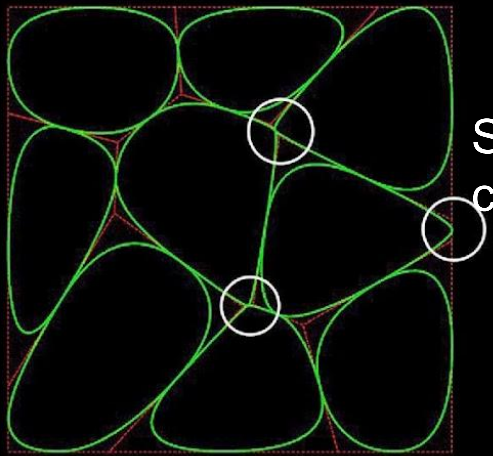


Step 2 **Cull short edges**

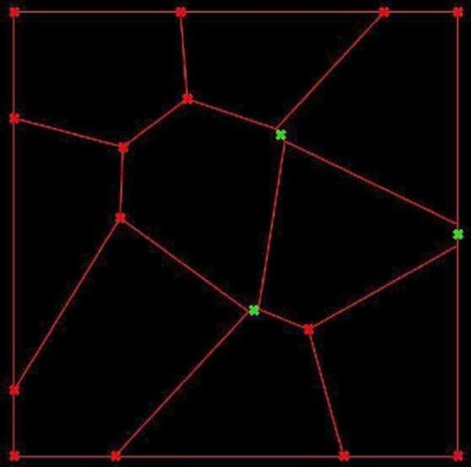
Polygons with short edges cause the curves generated by the polygon to have sharp corners. The "***smooth***" parameter in ***input*** represents the shortest length that can be allowed to exist.



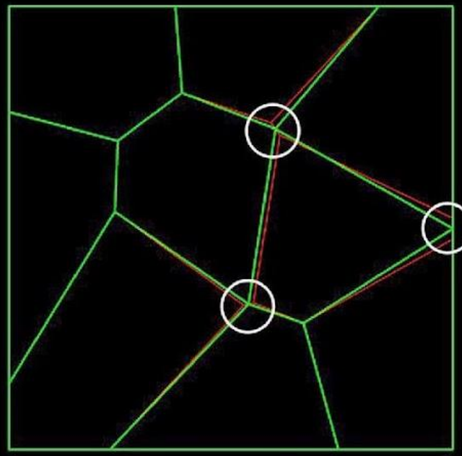
Curves generated by the polygon



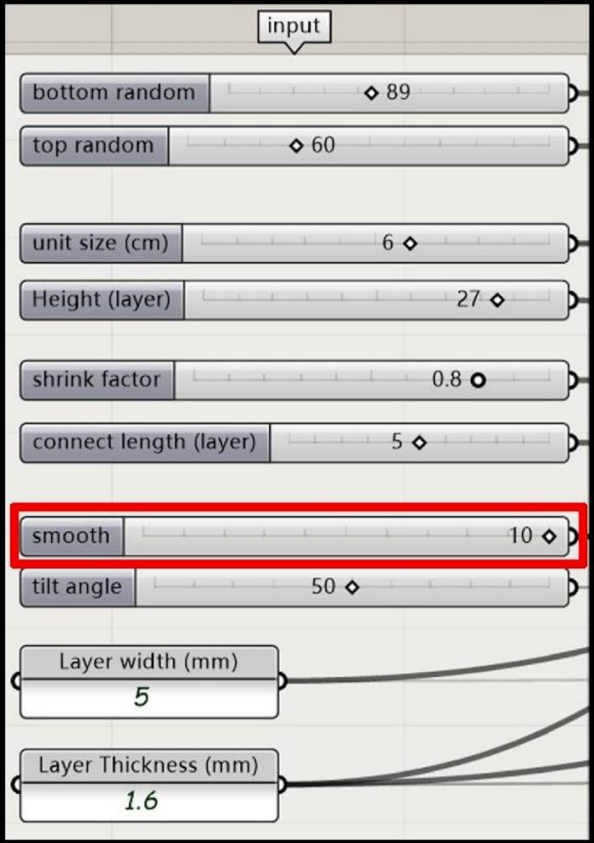
Sharp corners



Get middle point of the short curve

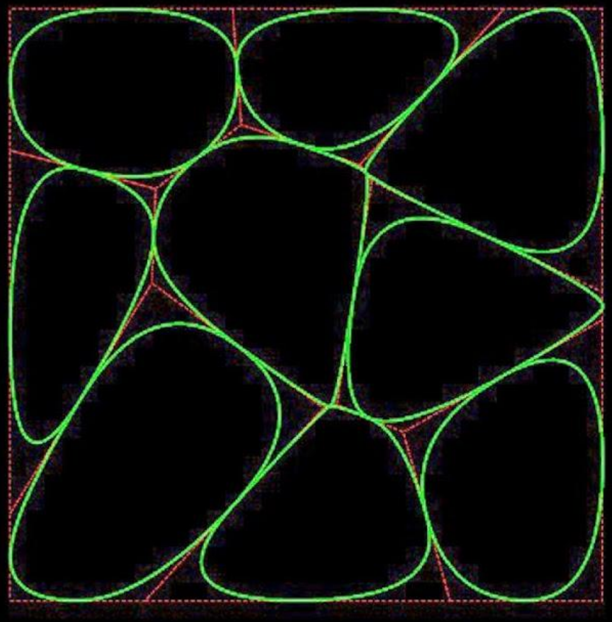


Reconnect the polygon with new points

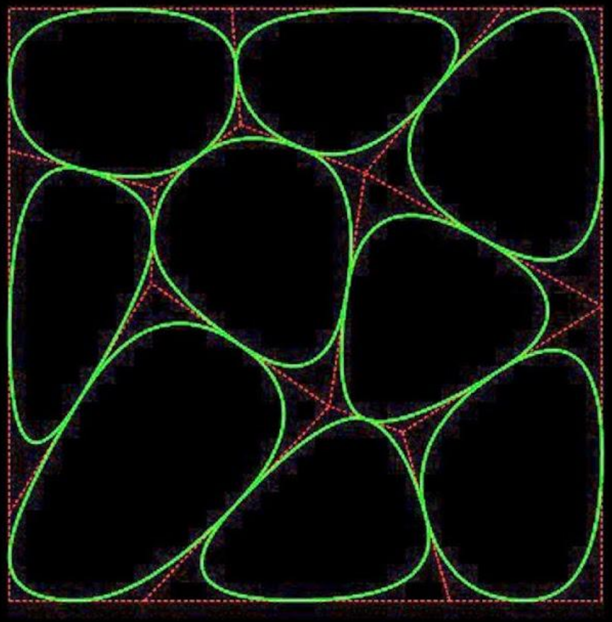


Step 2 **Cull short edges**

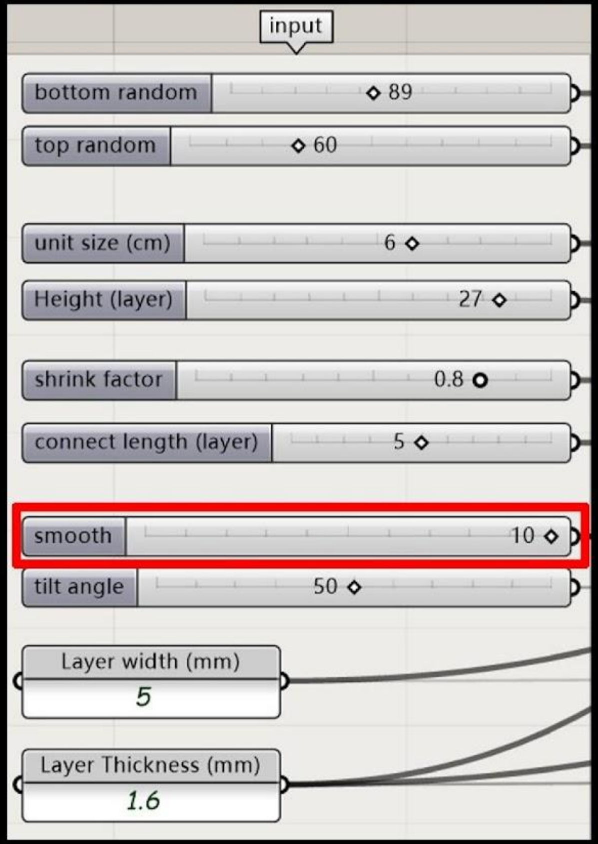
Polygons with short edges cause the curves generated by the polygon to have sharp corners. The "***smooth***" parameter in ***input*** represents the shortest length that can be allowed to exist.



Before correction

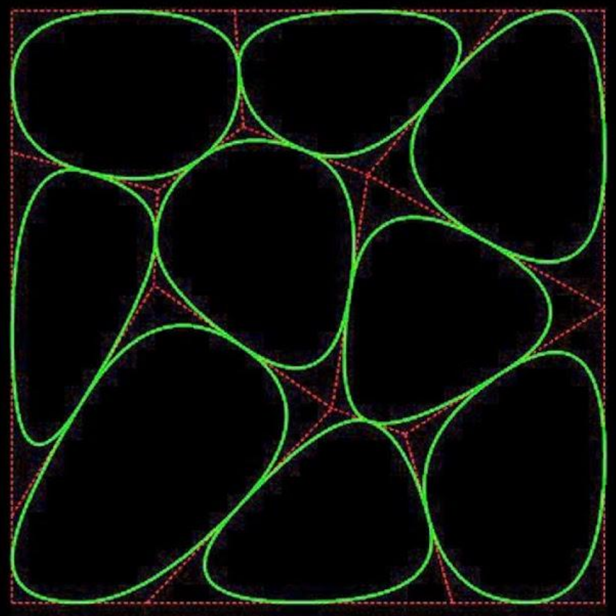


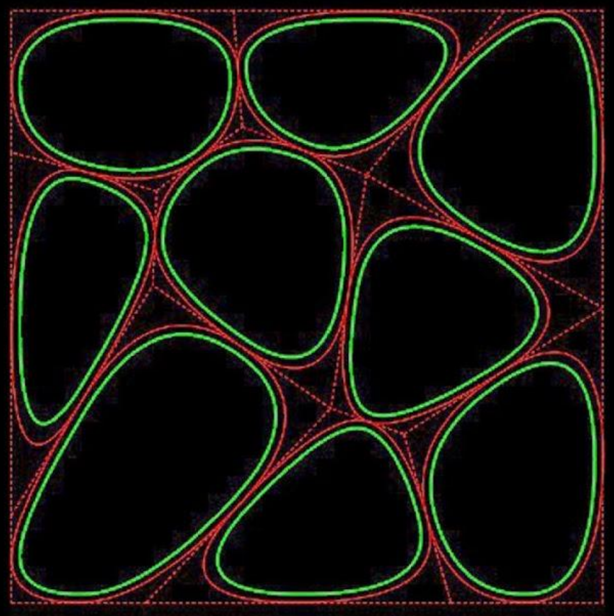
After correction

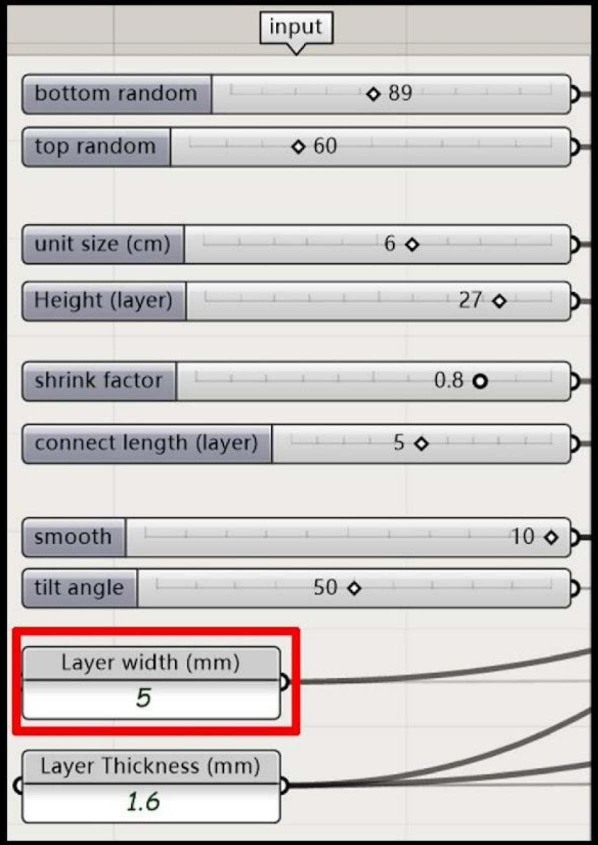


Step 3 **Offset curves**

Allow for the width of the clay when printing.

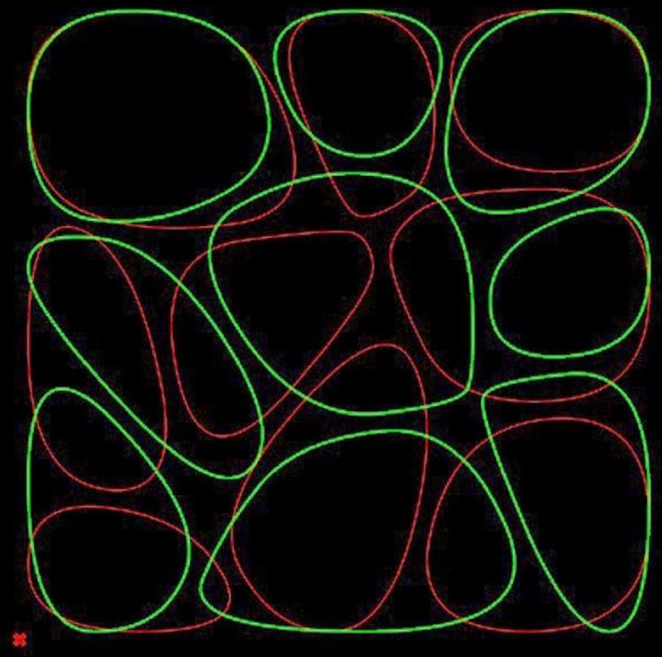


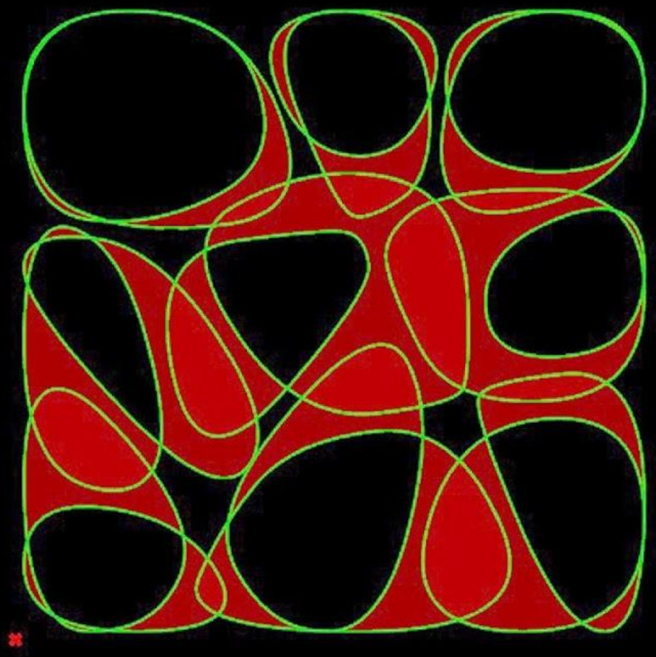


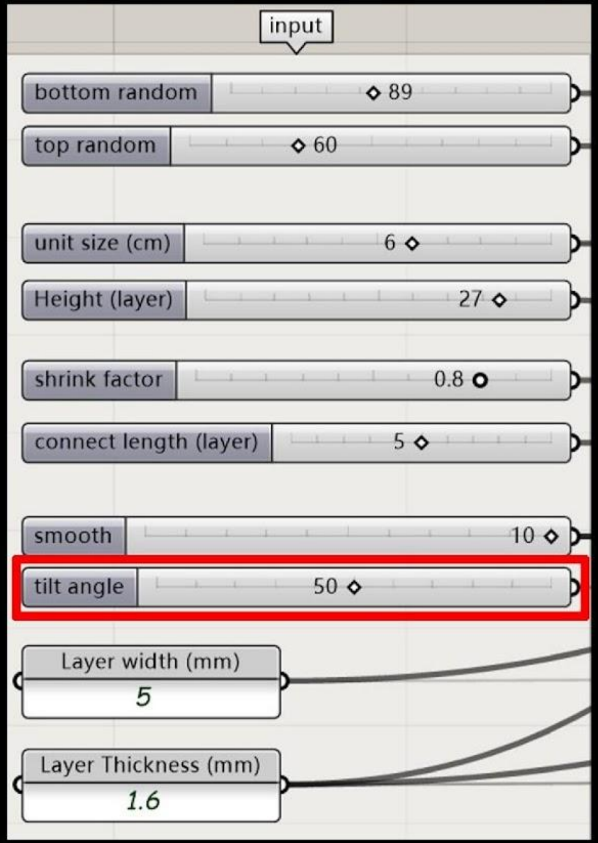


Filter 2 **Tilt angle**

Excessive tilt angles can lead to print failures. The “***tilt angle***" parameter in ***input*** represents the largest angle that can be allowed to exist.



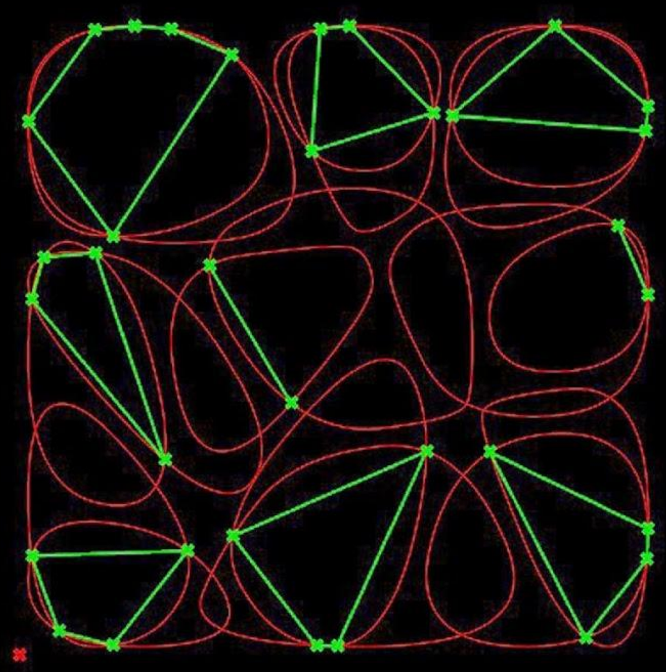




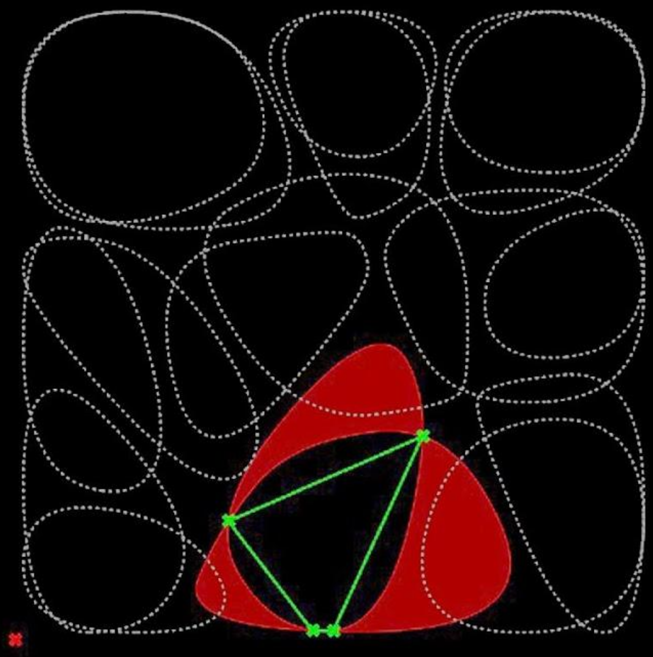
1. Get intersection part of the top layer and bottom layer for each grid.

Filter 2 **Tilt angle**

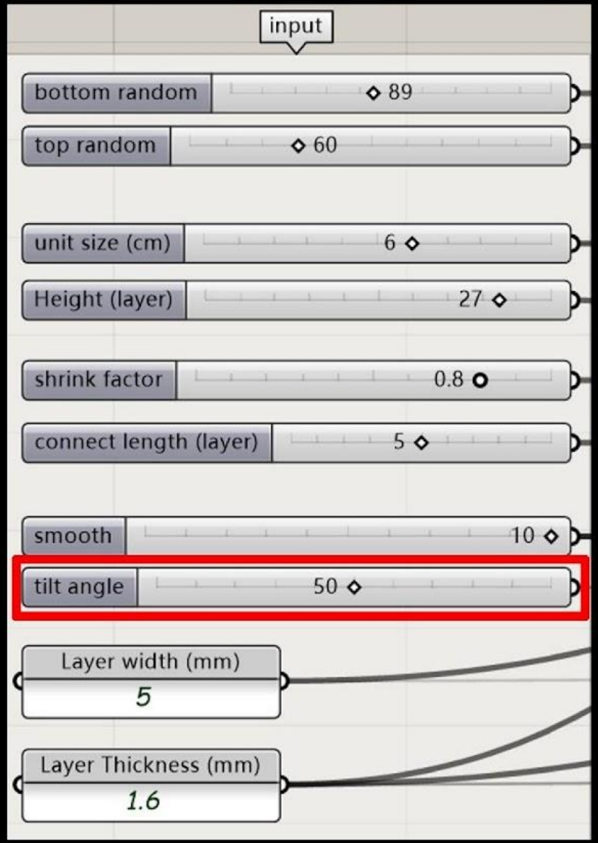
Excessive tilt angles can lead to print failures. The “***tilt angle***" parameter in ***input*** represents the largest angle that can be allowed to exist.



intersection point



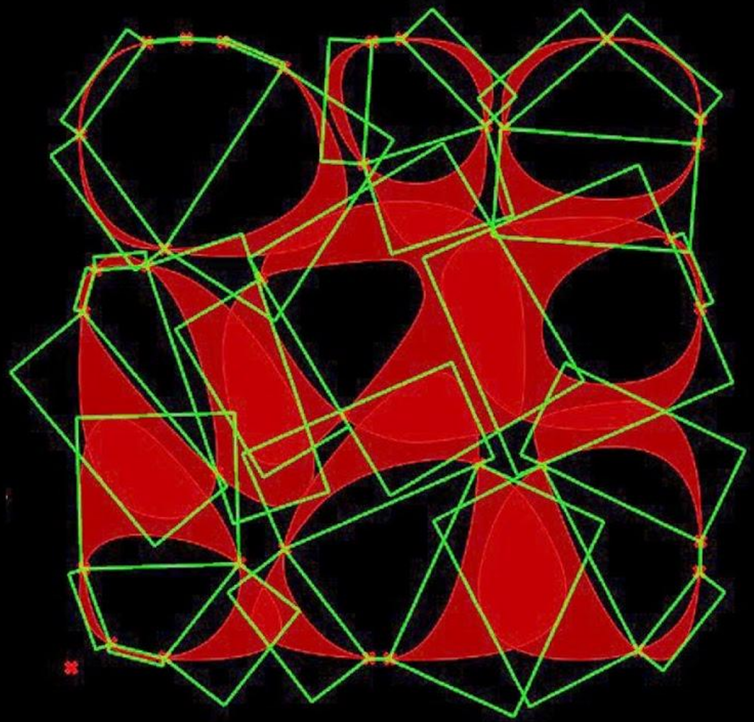
example



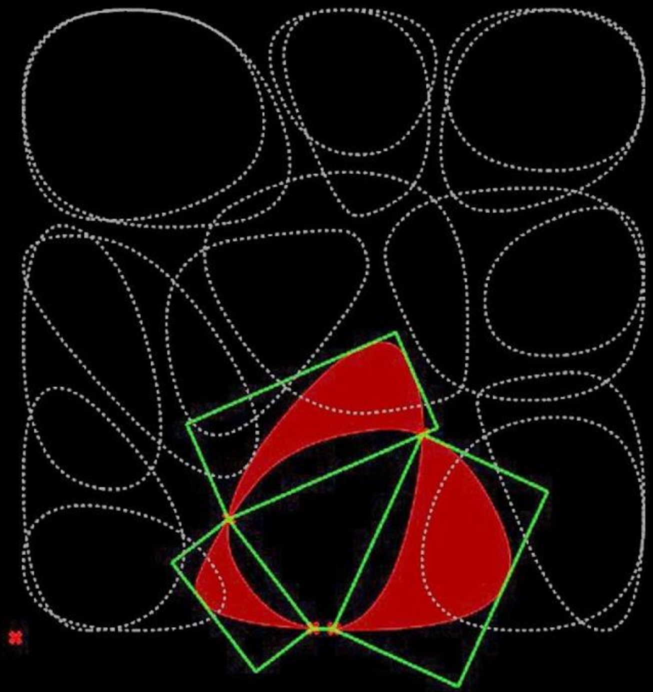
2. Get the intersection point of each intersecting curve.

Filter 2 **Tilt angle**

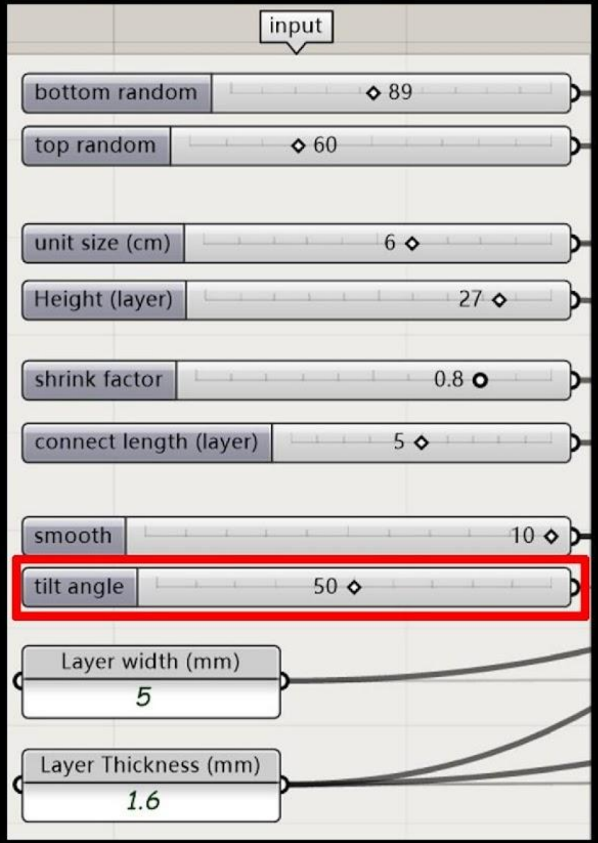
Excessive tilt angles can lead to print failures. The "***tilt angle***" parameter in ***input*** represents the largest angle that can be allowed to exist.



"boundingbox"



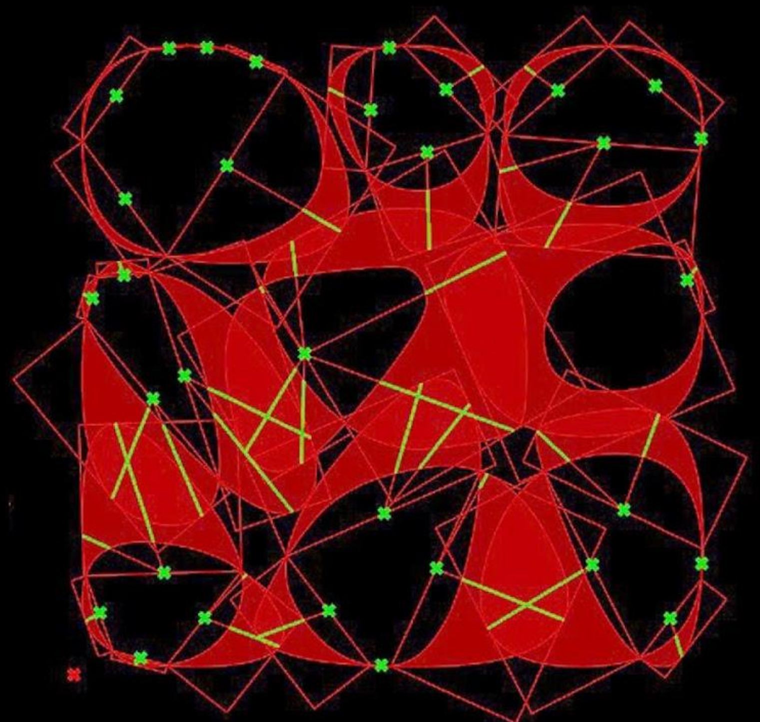
example



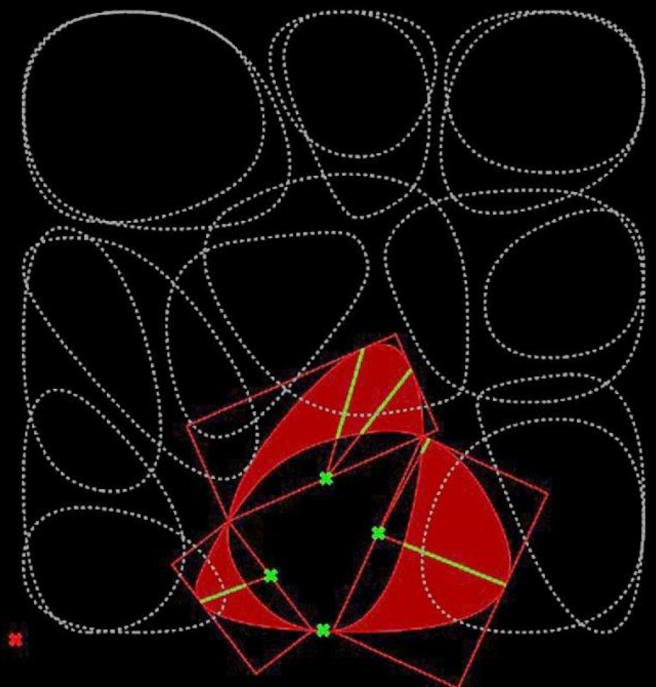
3. Create a "*boundingbox"* for intersecting curves.

Filter 2 **Tilt angle**

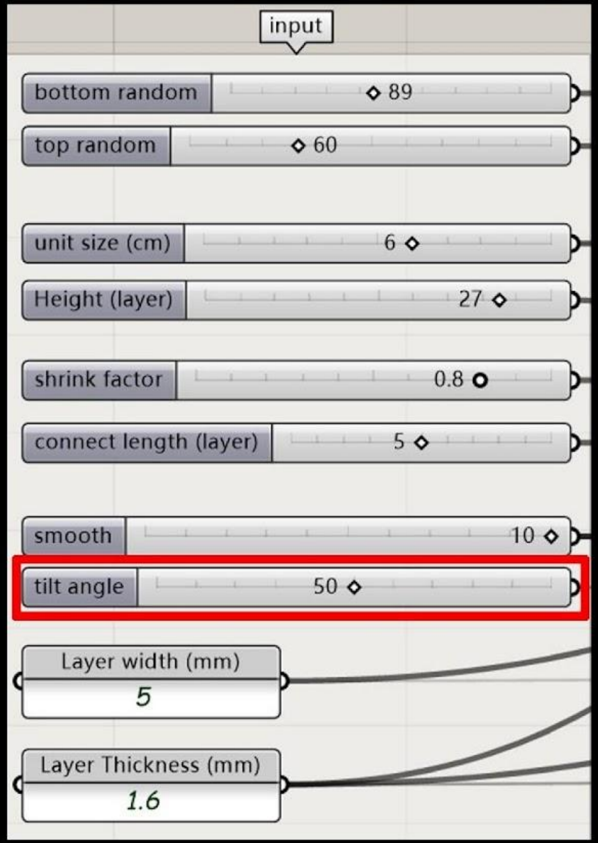
Excessive tilt angles can lead to print failures. The “***tilt angle***" parameter in ***input*** represents the largest angle that can be allowed to exist.



the widest part (line)



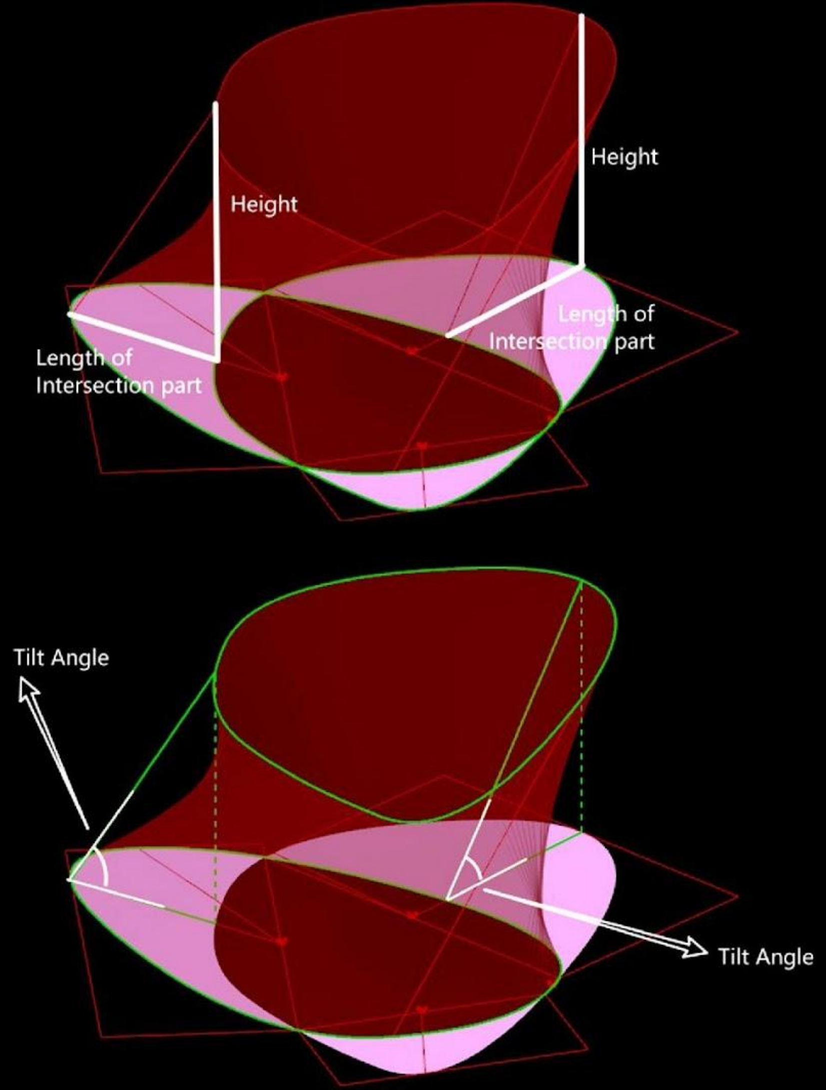
the widest part (line)



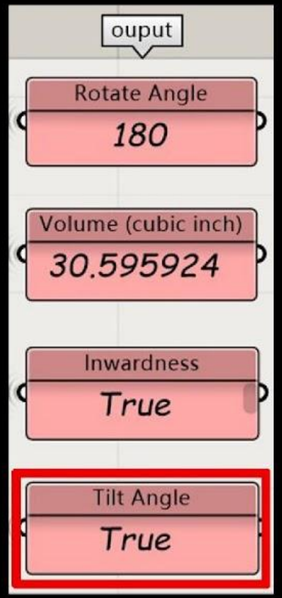
4. Calculate the widest part (line) of the intersection by the intersection of the curve and the "*boundingbox*".

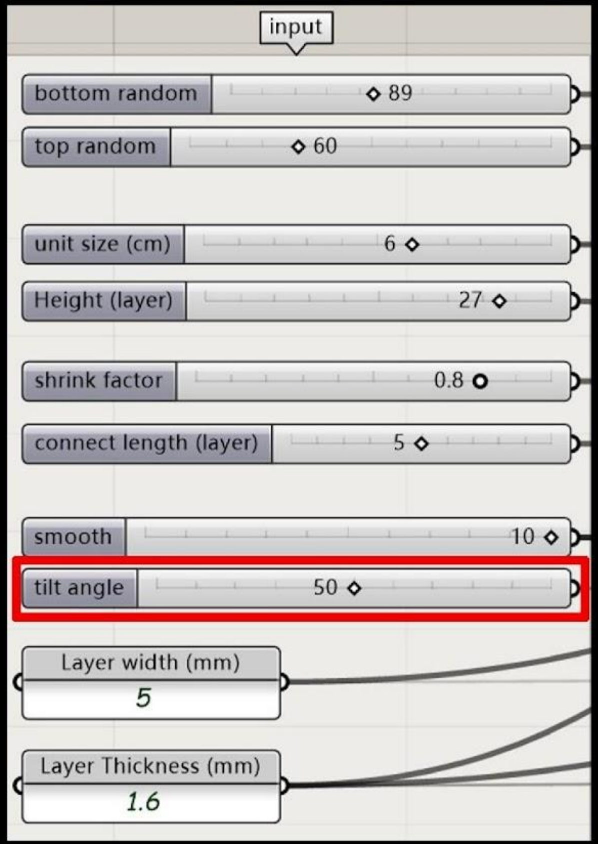
Filter 2 **Tilt angle**

Excessive tilt angles can lead to print failures. The "***tilt angle***" parameter in ***input*** represents the largest angle that can be allowed to exist.



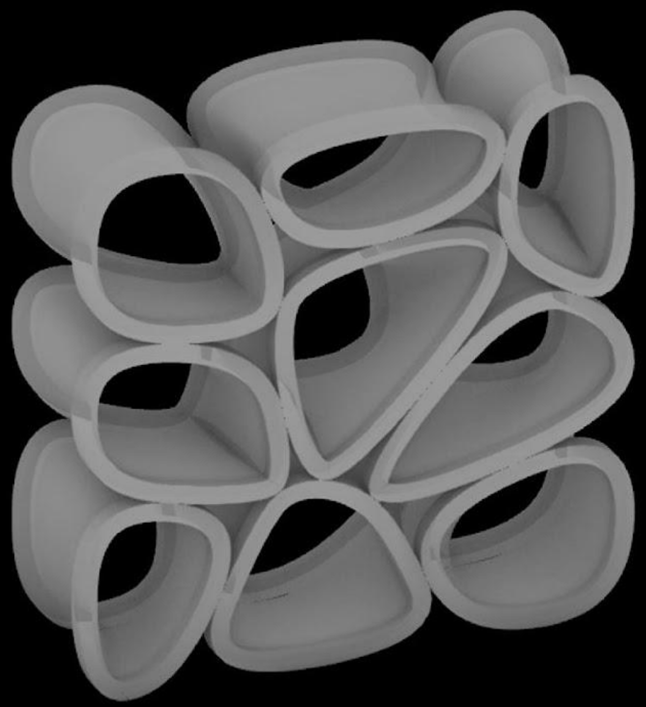
The ratio of the height and the widest length of the intersecting part is the tangent of the tilt angle.

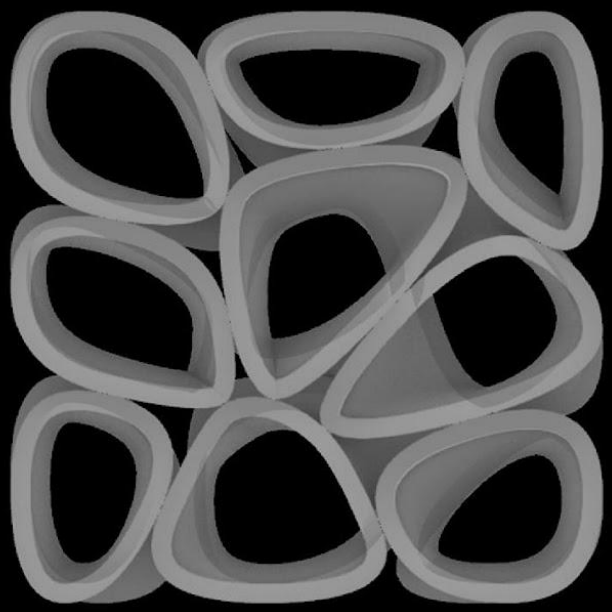




Version 1

The problem with the first version was that it was structurally unstable and could not stand up on its own.

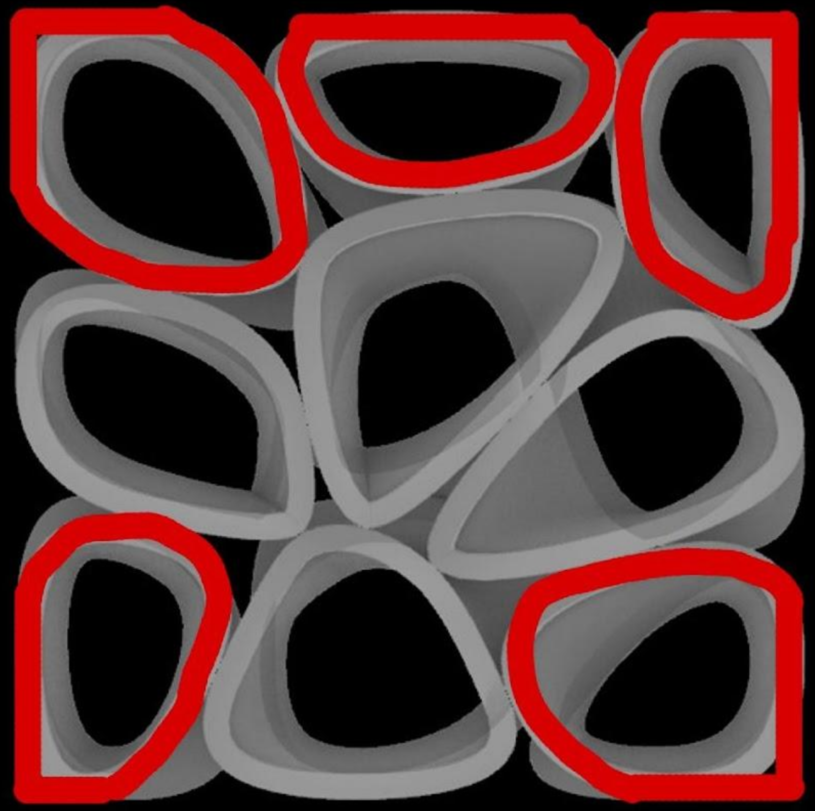




Version 1

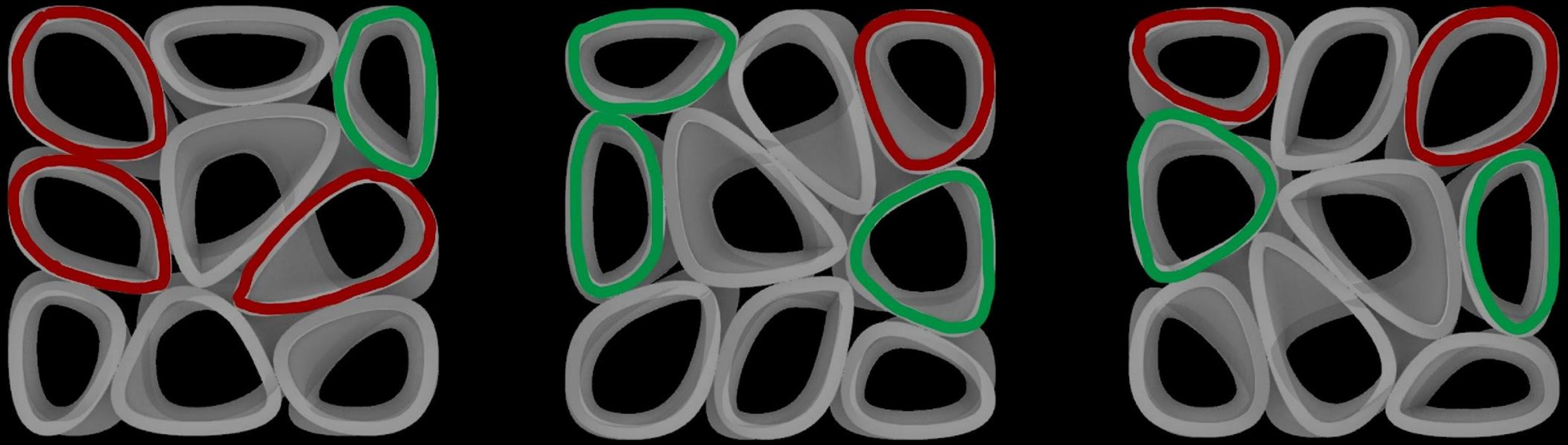
The problem with the first version was that it was structurally unstable and could not stand up on its own.

Fix 1: Change the arcs of the four vertices to right angles. The topmost part of the cell at the top is also changed to straight in order to connect with the next group.



Version 1

The problem with the first version was that it was structurally unstable and could not stand up on its own.



Tips facing inward

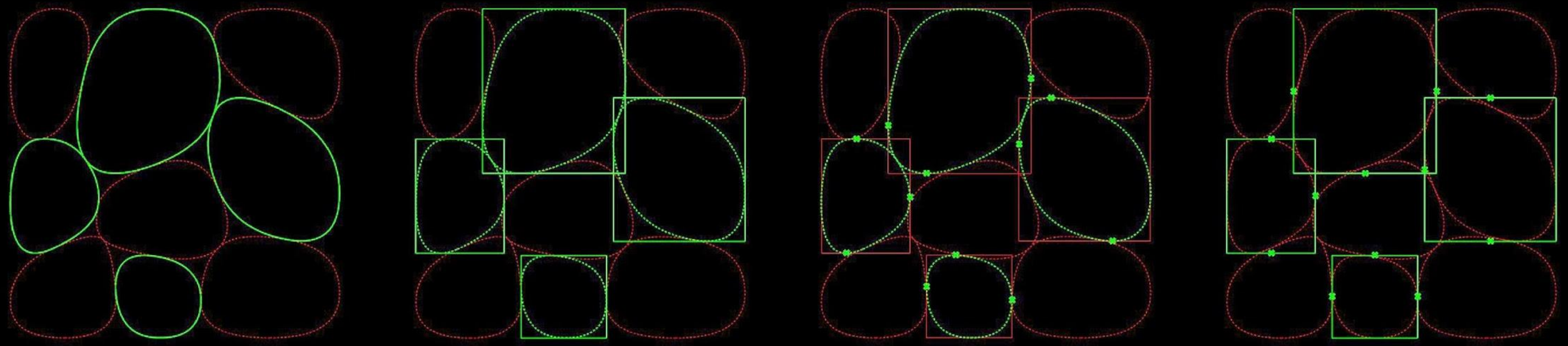
Tips NOT facing inward

Fix 2: The remaining units, except for the vertex unit and the middle unit, all need to have their tips facing inward. This way it can snap into its neighboring cells and ensure overall stability.

Filter 3 **Inwardness**

The remaining units, except for the vertex unit and the middle unit, all need to have their tips facing inward. This way it can snap into its neighboring cells and ensure overall stability. (Fix 2)

First determine whether the graph now has the possibility of structure inward. If the whole cluster is rotated by 0, 90, 180, 270 degrees, respectively, there are different combinations of the second row of outer cells. Calculate which angle of rotation will guarantee the inwardness of the structure's second layer.



Unit curve of layer

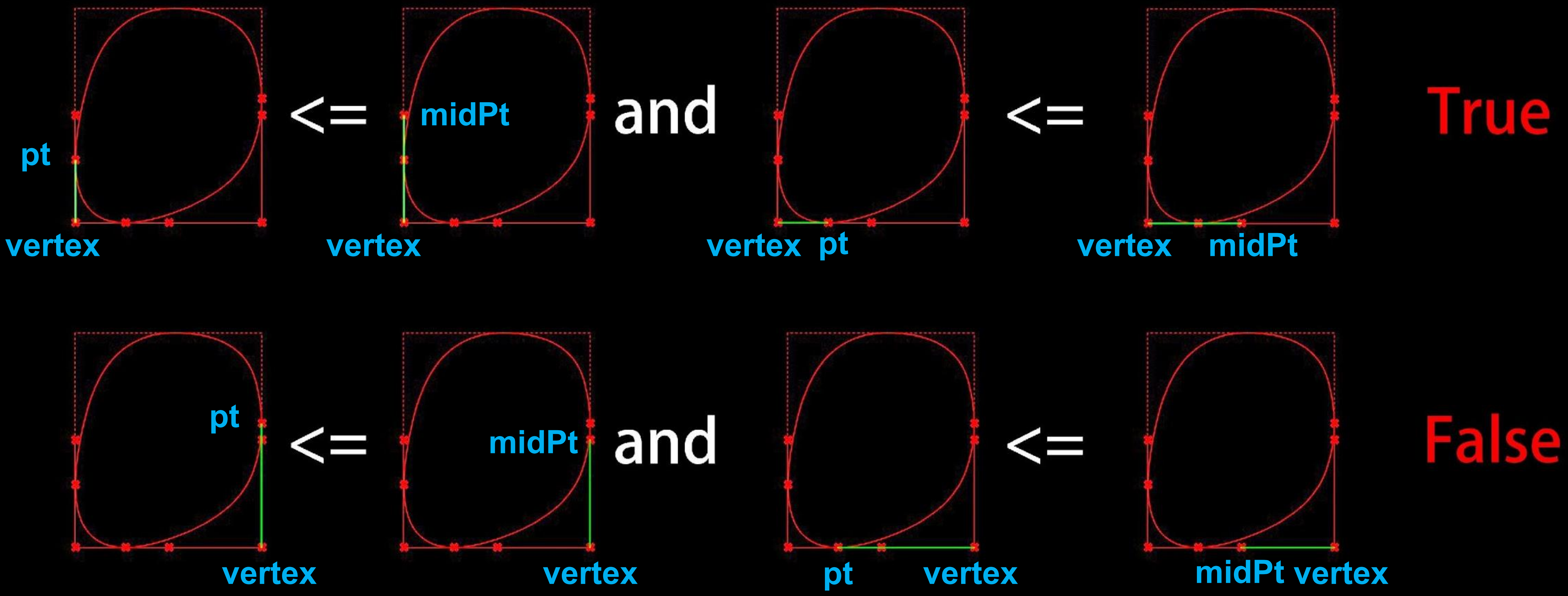
Get “*boundingbox*”

Intersection point of curves and “*boundingbox*”

“*boundingbox*” side midpoint

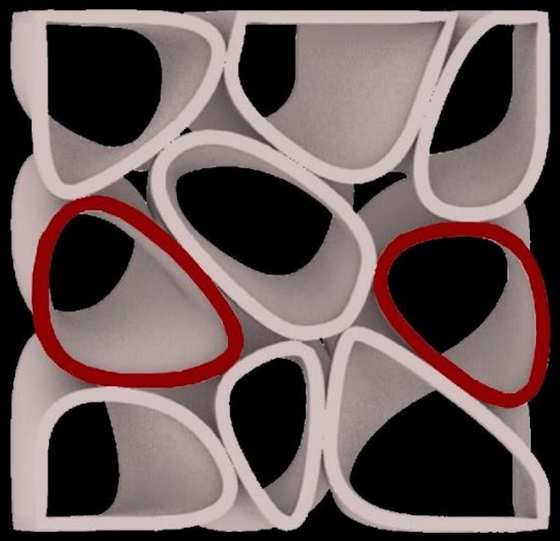
Filter 3 **Inwardness**

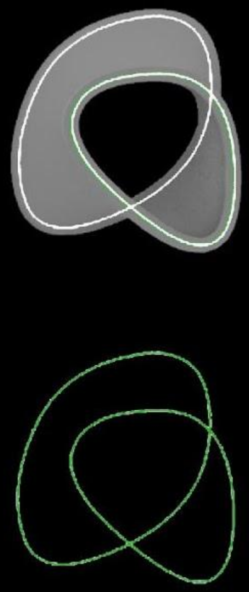
Ensuring that the curve is trending toward a certain **vertex** requires determining the relationship between the location of the **intersection point (pt)** and the **midpoint (midPt)**. If both intersection points are closer to the vertex than the midpoint is to the vertex, it indicates that the curve is trending toward the vertex.



Filter 3 **Inwardness**

For pairs of opposite cells, inwardness in the same direction is required. This will ensure that there is the same rotation angle. Any layer can meet this requirement, such as the cells of top layer & top layer, cells of bottom layer & top layer.
For the top layer cell and the bottom layer cell, it is sufficient that one or two cells has inwardness.

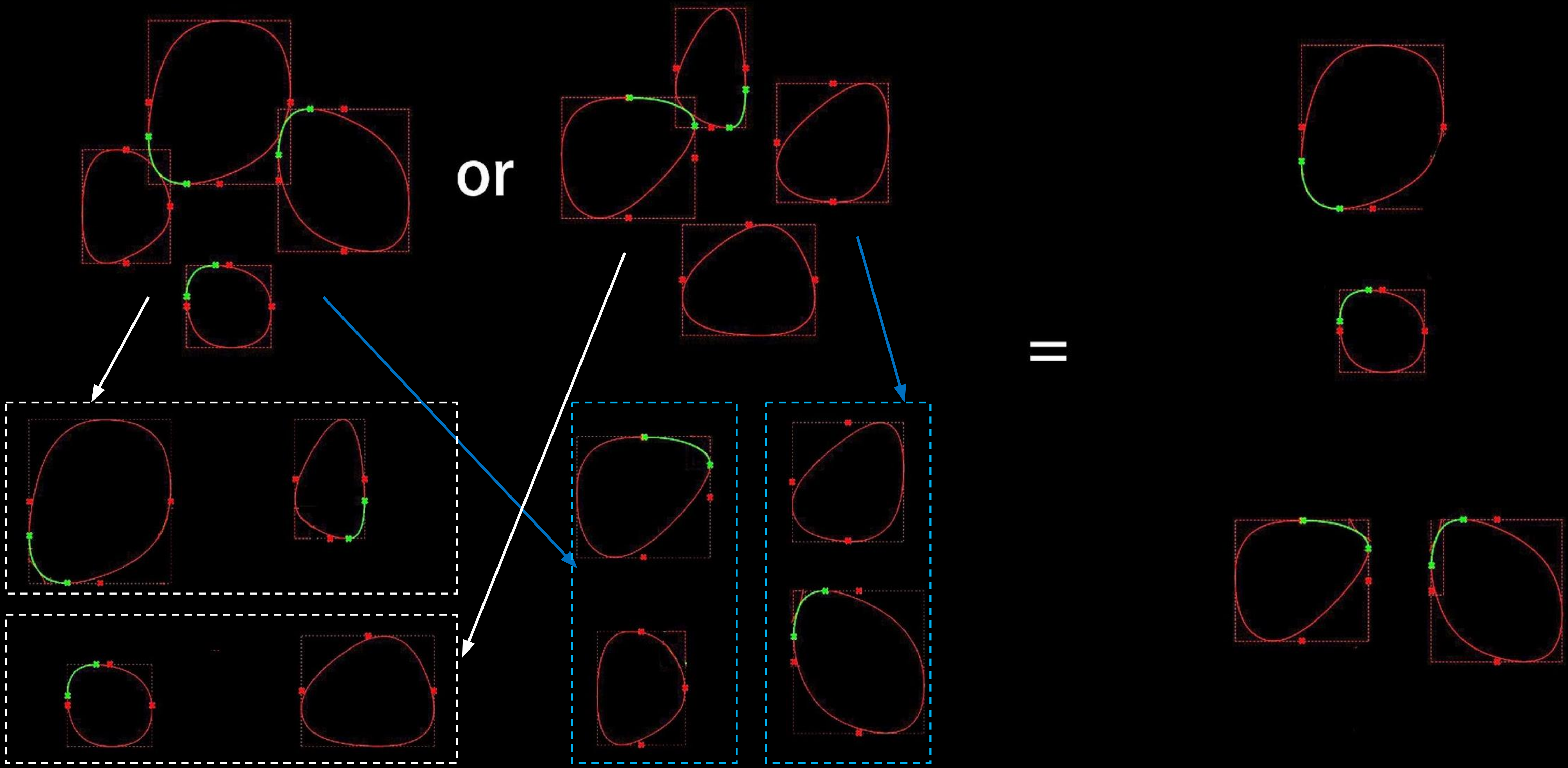




pairs of opposite cells:
inwardness in the same direction
(any layer)

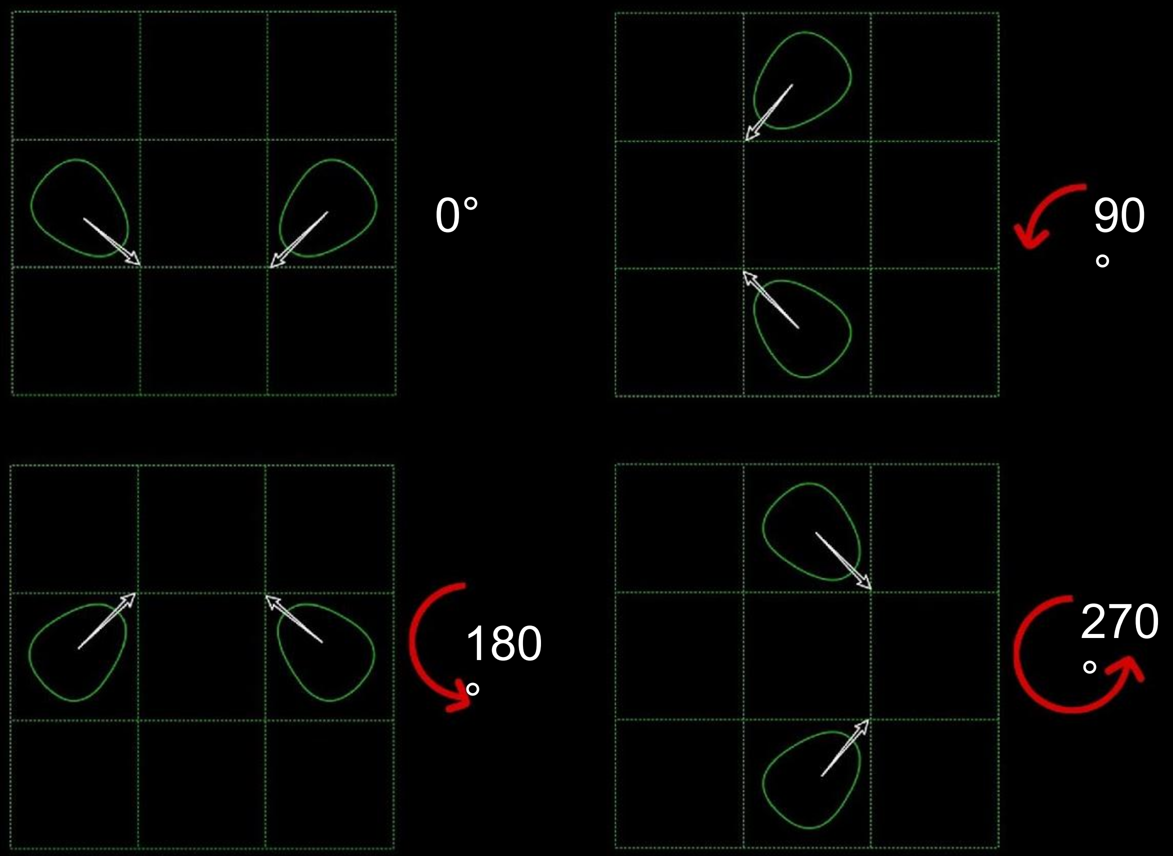
top and bottom layer cell:
More than one has inwardness

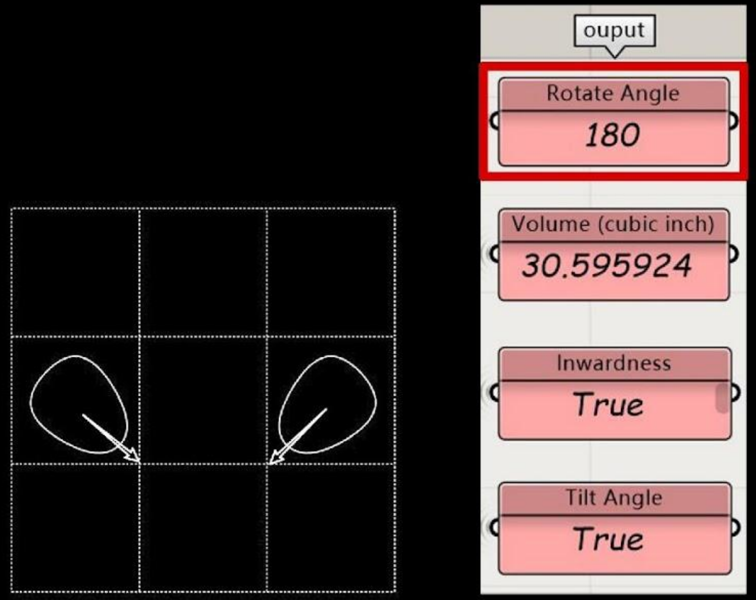
Filter 3 **Inwardness**



Filter 3 **Inwardness**

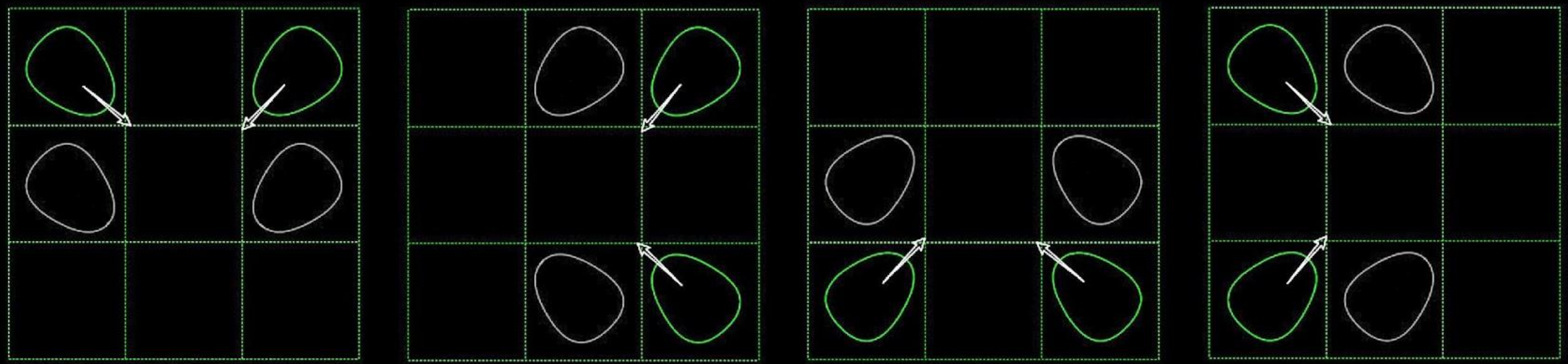
Different pairs of opposite cells can lead to different rotation angles of the group. Because this screening process is to get whether the second row has inwardness.

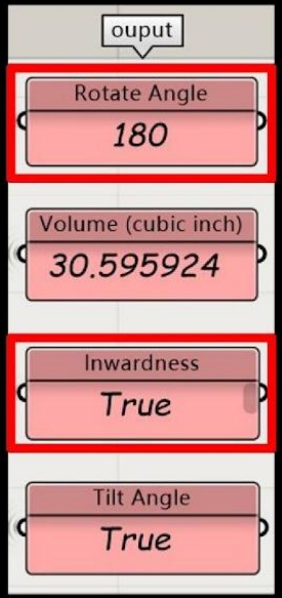




Filter 3 **Inwardness**

Once the rotation angle and the inwardness of the second row of cells are determined, the inwardness of the third row of cells can be examined and determined. The original position of the third row of cells is different for different rotation angles.

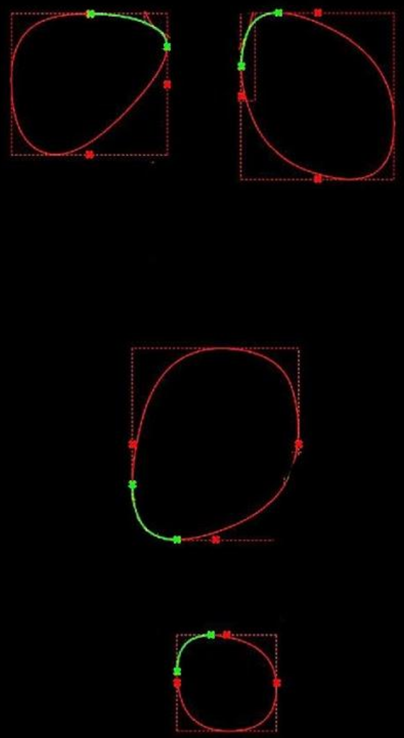


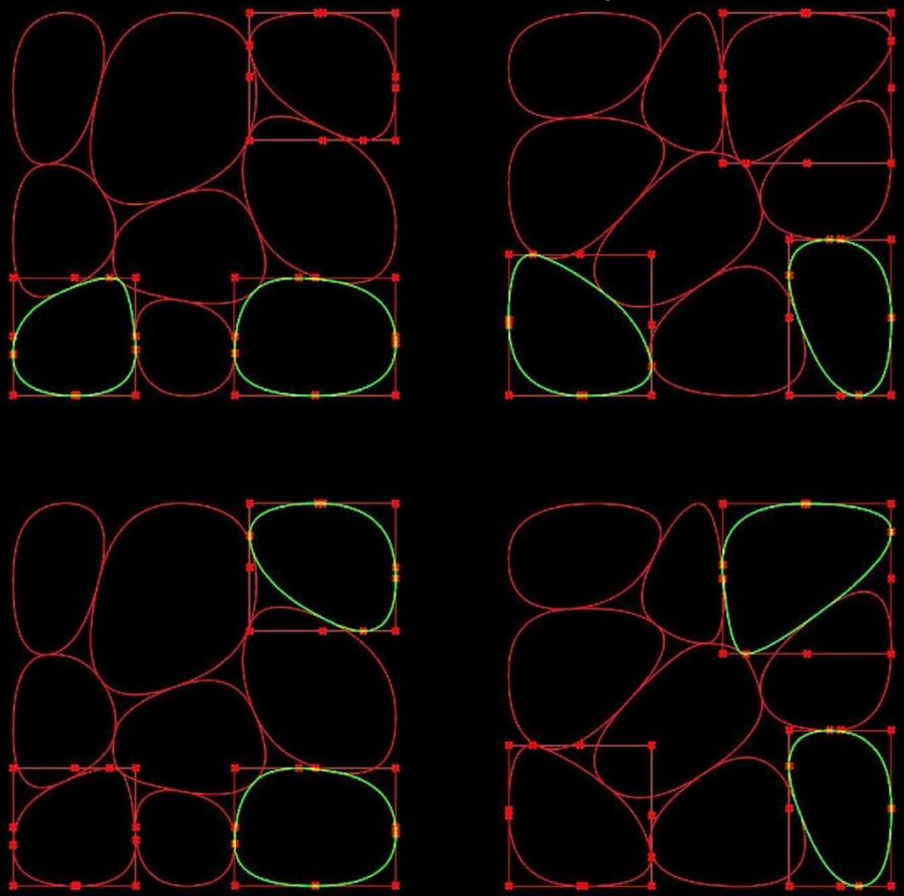


Filter 3 **Inwardness**

In this example, there are two possible angles for the inwardness of the second row. So they need to be tested separately to see if the third row also has inwardness to determine the final angle.

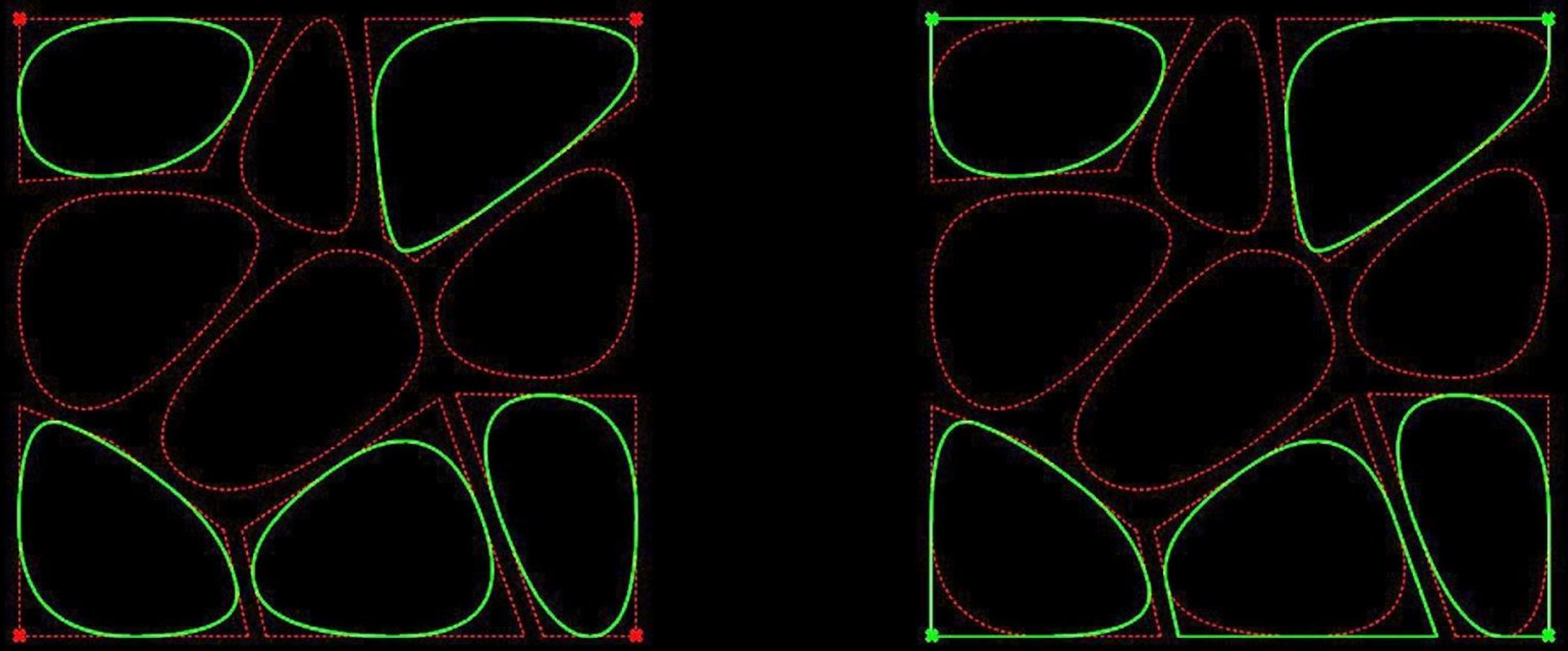
Top layer Bottom layer





Step 4 **Right-angle cells**

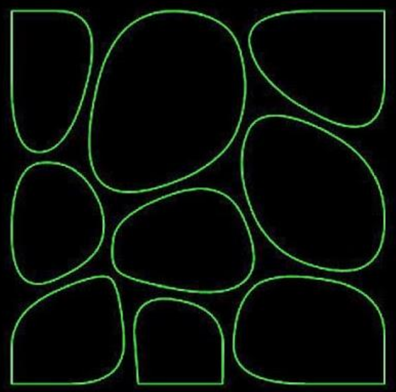
Change the arcs of the four vertices to right angles. The topmost part of the cell at the top is also changed to straight in order to connect with the next group. (Fix 1)



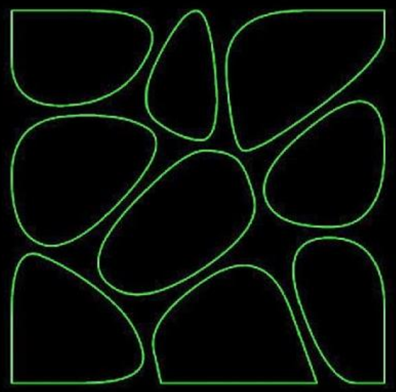
Original cells **Right-angle cells**

Step 5 **Shrink**

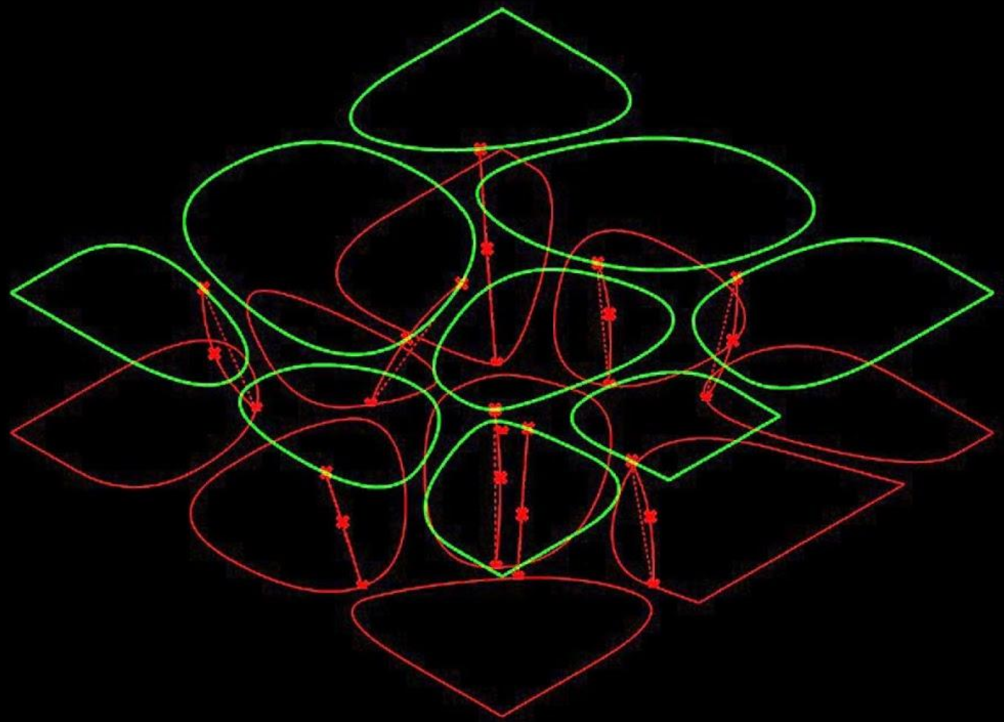
The “***shrink factor***" parameter in ***input*** represents the shrinkage coefficient of the surface formed by connecting the top and bottom curve cells.

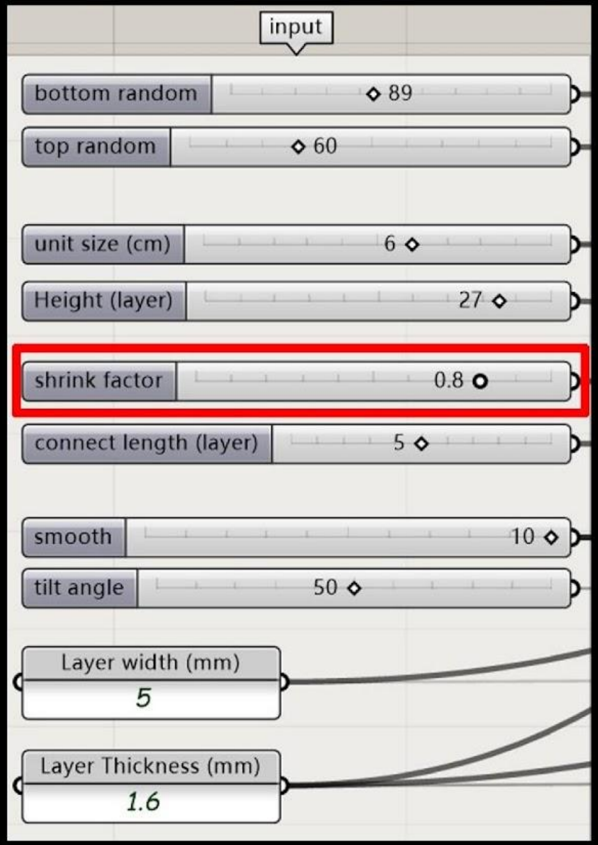


Top layer



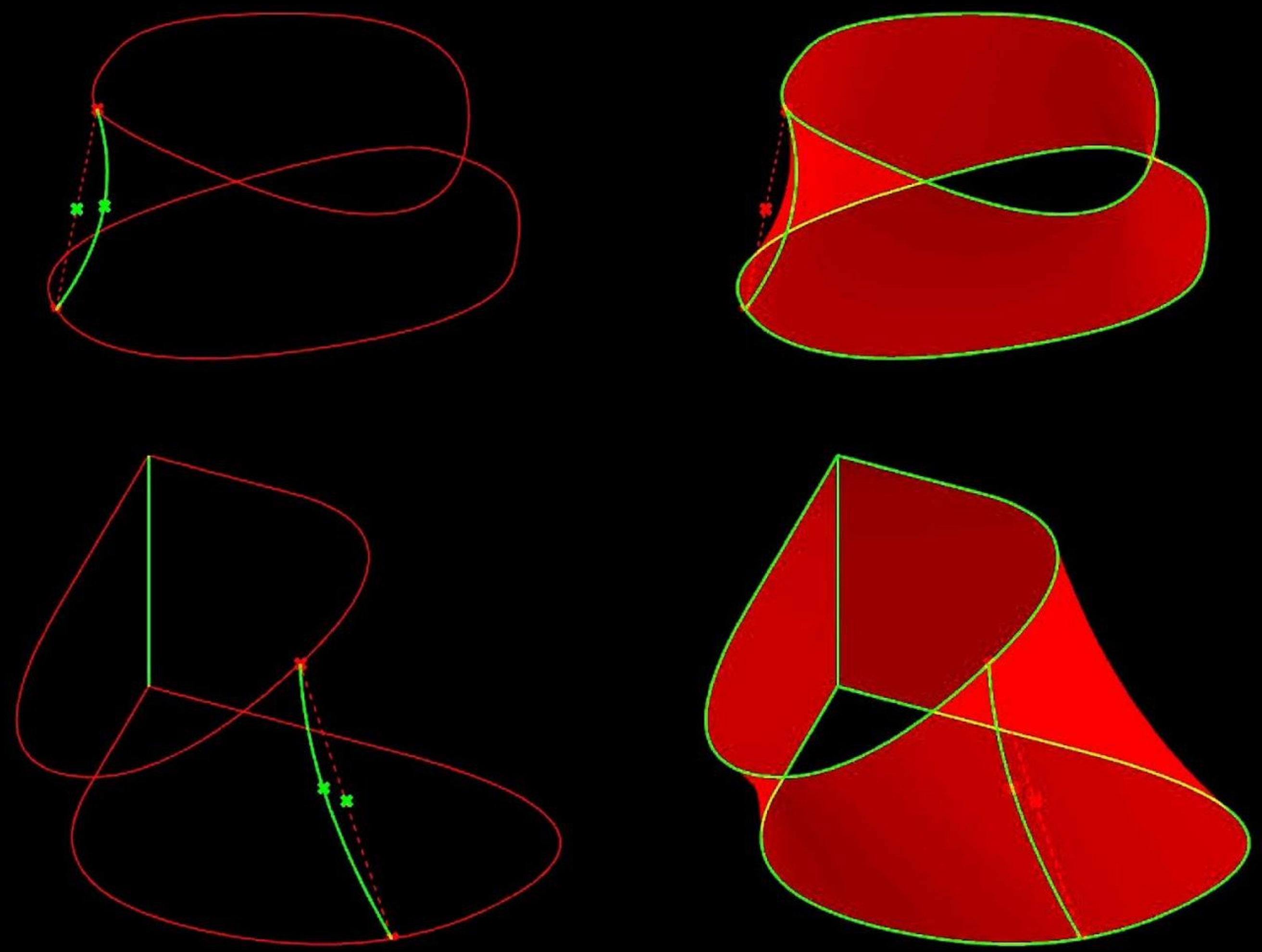
Bottom layer

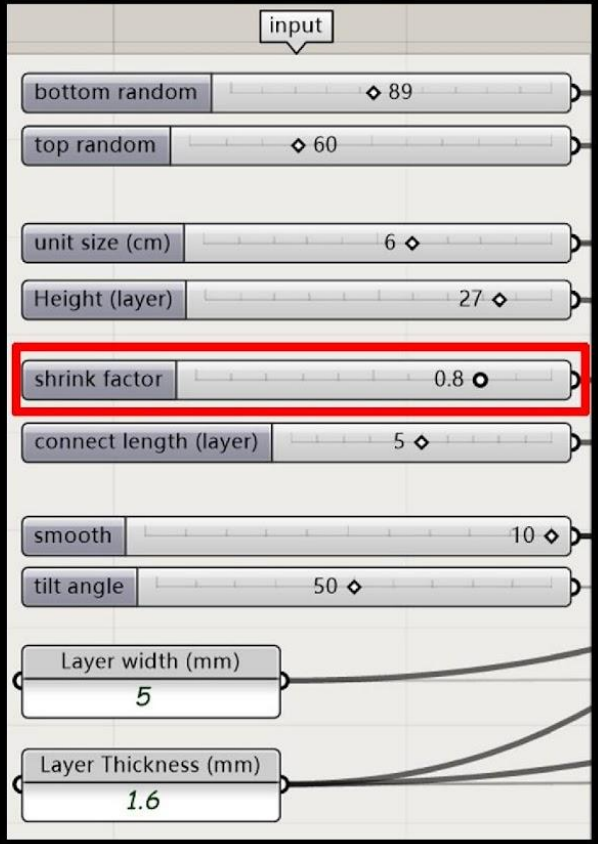




Step 6 ***Sweep2 - surfaces***

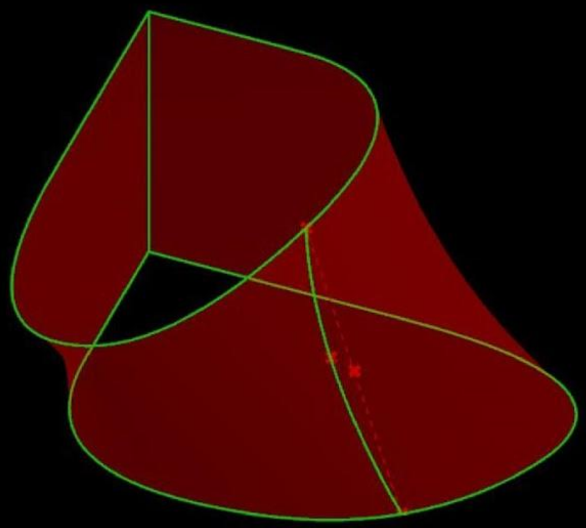
For cells with right angles or straight edges, the surface generation needs to be controlled with right angled edges and shrinkage lines together. For other cells, only shrink lines are used to control surface generation.

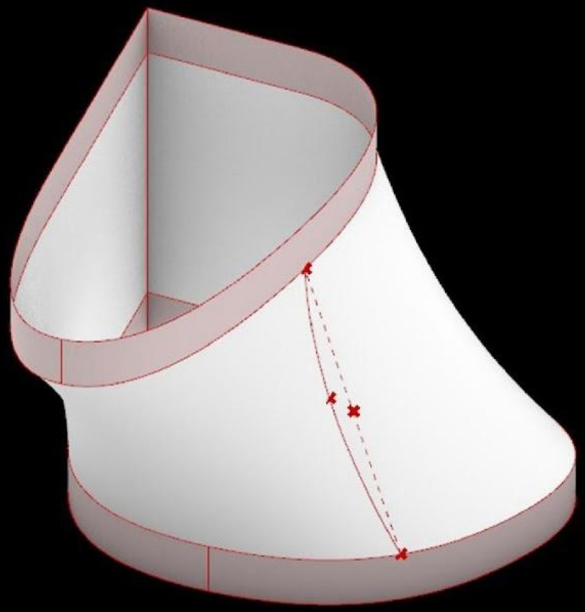


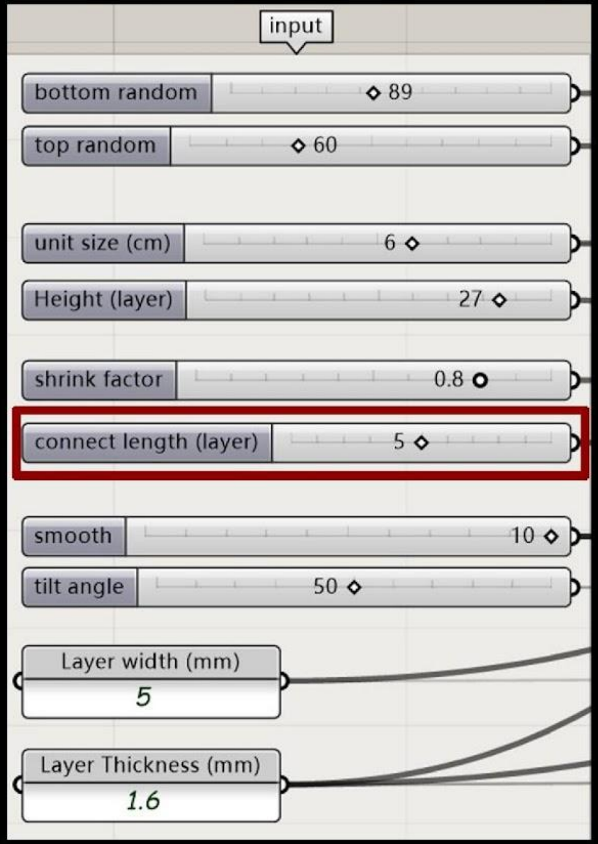


Step 7 **Splicing surface**

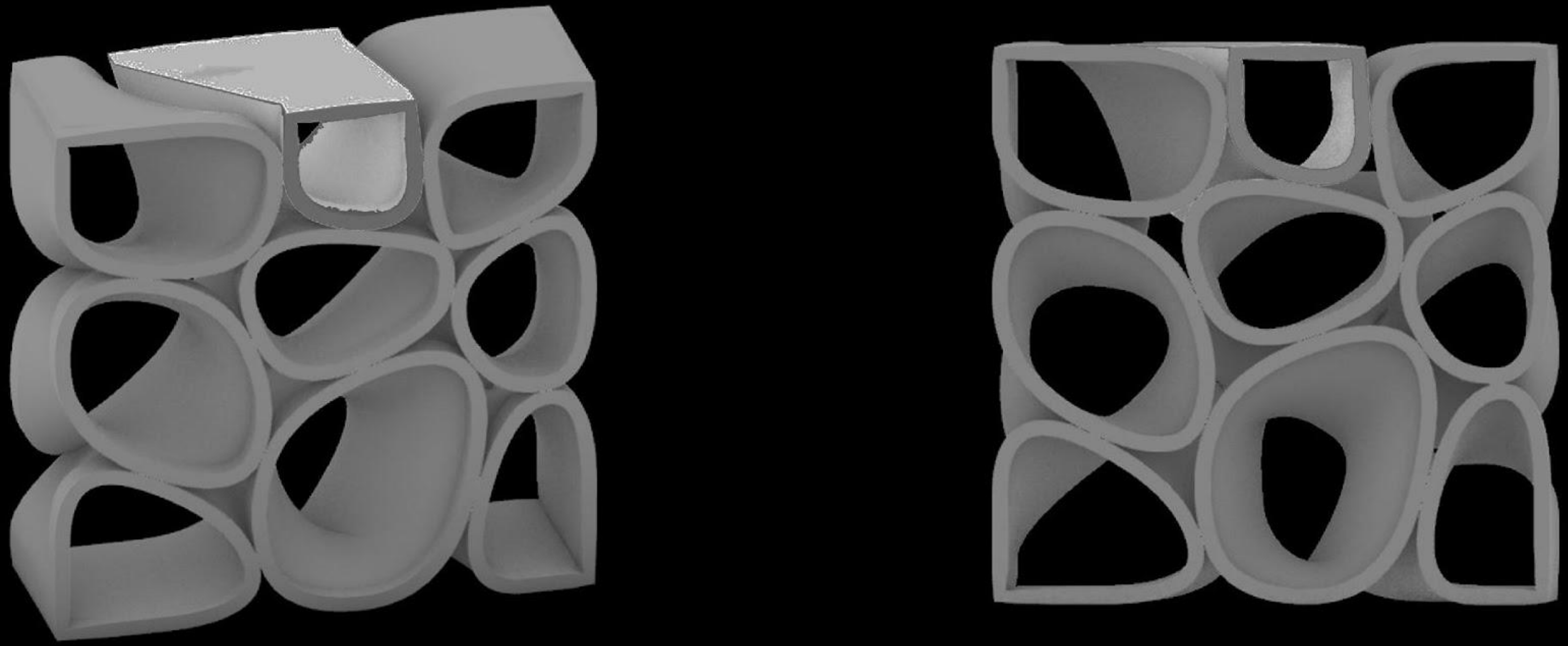
To make the connection more frictional, increase the connection surface in the top and bottom layers. The "***connect length***" parameter in ***input*** determines the height of the connecting surface.



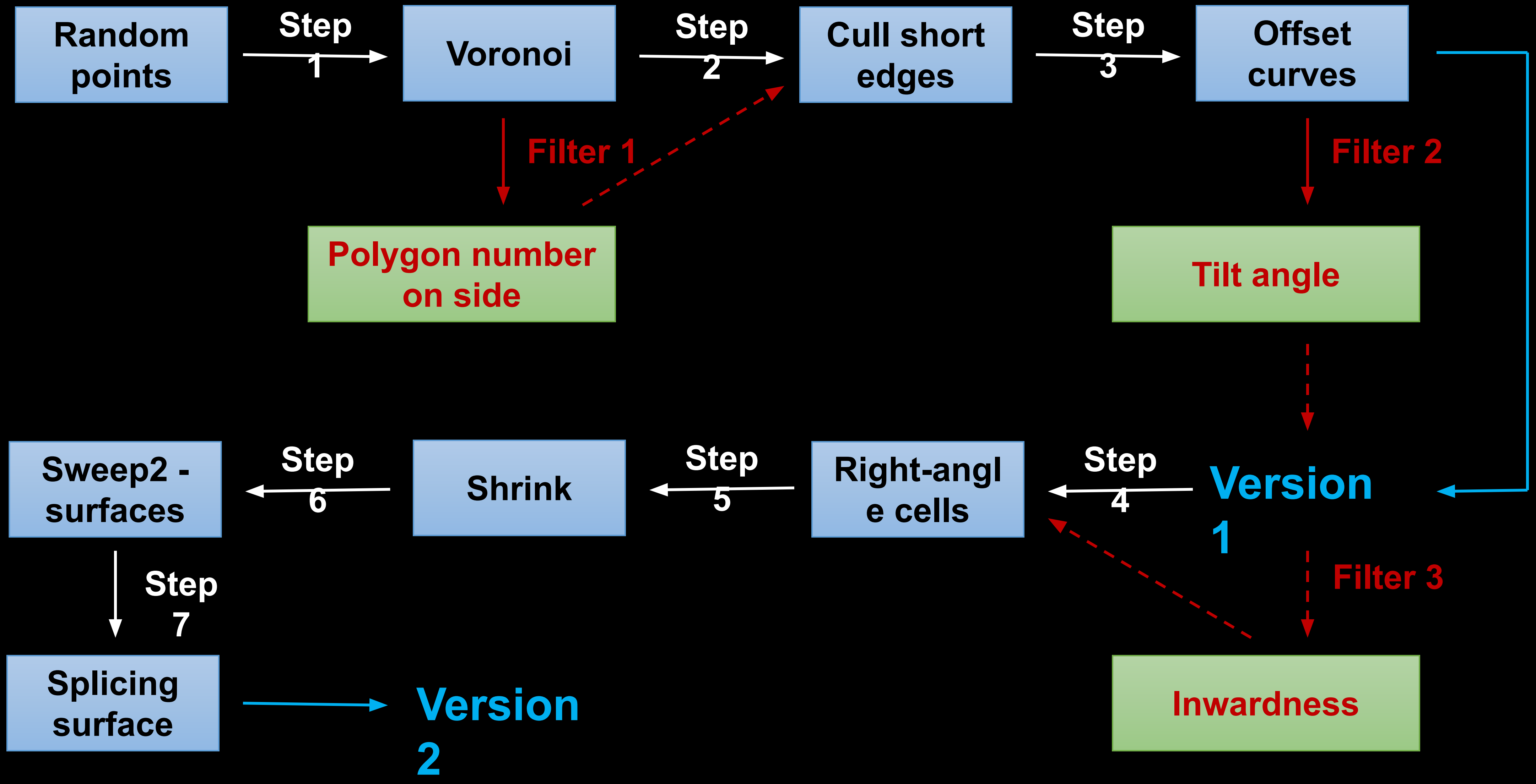


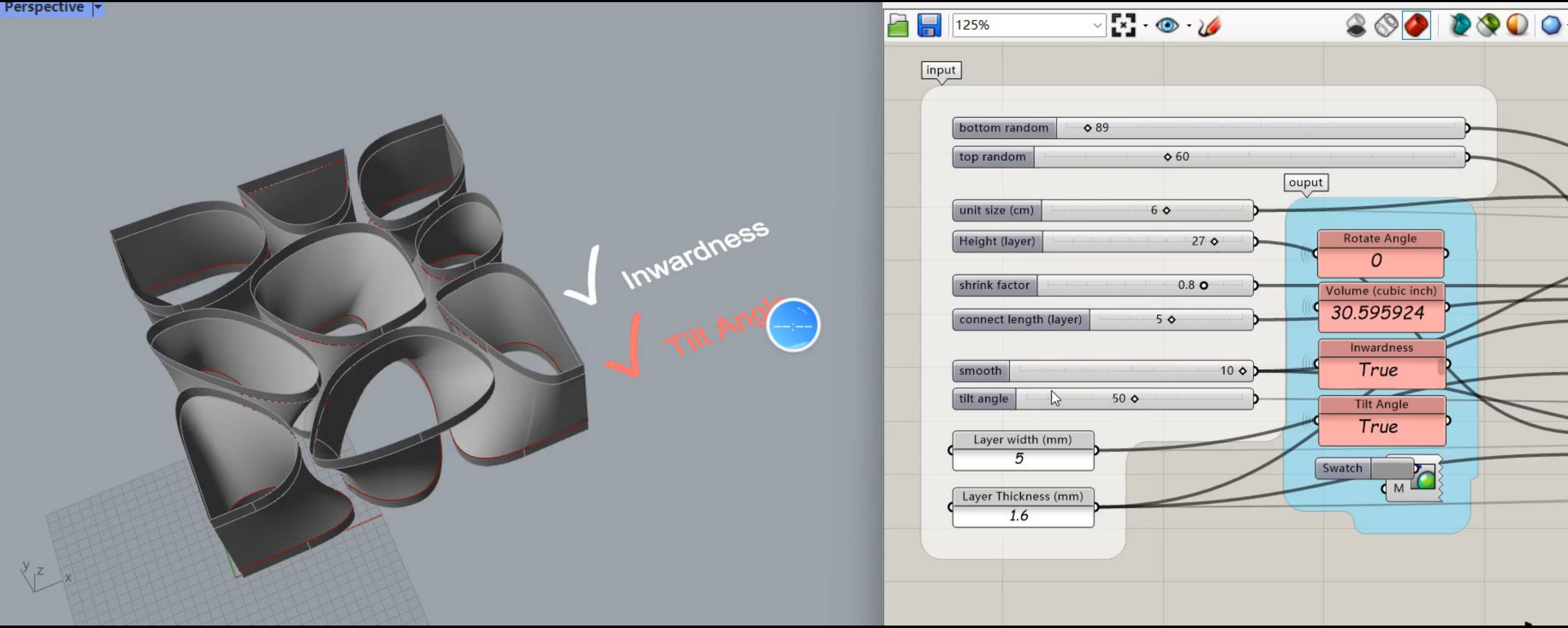


Version 2



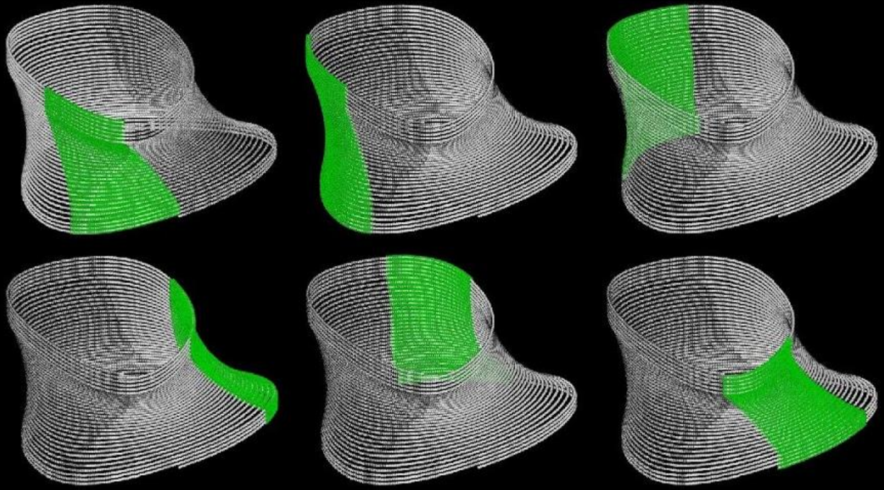
Parametric Logic of Design

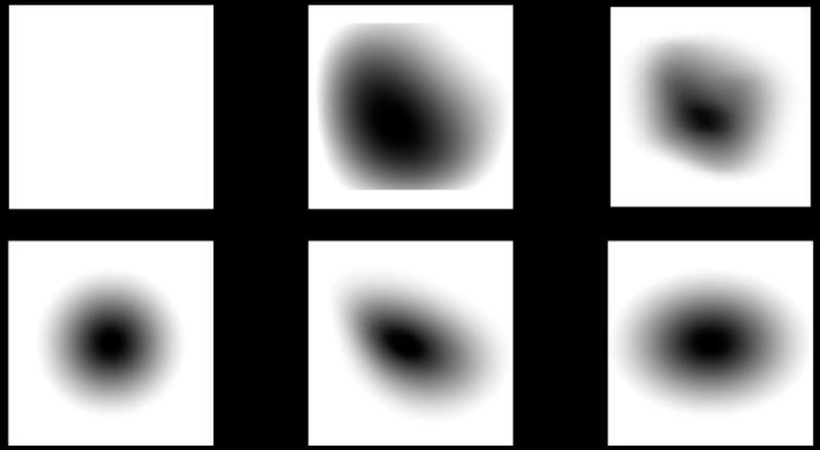


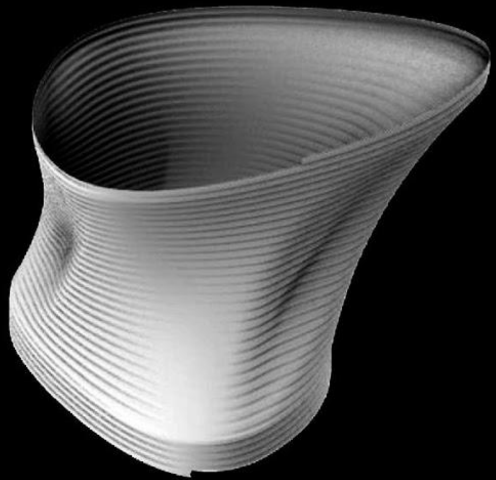

Perspective
Inwardness
Tilt Angle
125%
input
bottom random
89
top random
60
ouput
unit size (cm)
6
Height (layer)
27
shrink factor
0.8
connect length (layer)
5
smooth
10
tilt angle
50
Layer width (mm)
5
Layer Thickness (mm)
1.6
Rotate Angle
0
Volume (cubic inch)
30.595924
Inwardness
True
Tilt Angle
True
Swatch
M

Pattern 1

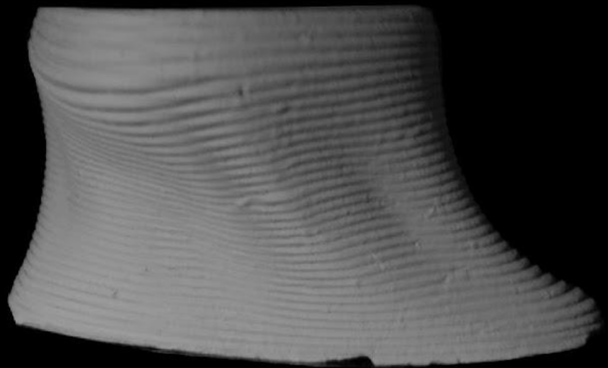
Vertical layering gives bitmap. Get different types of bulges on the surface to simulate the natural feel of a tree trunk.







GH Simulation



Clay Model

Pattern 2

Pattern 2

Pattern 2

Pattern 2