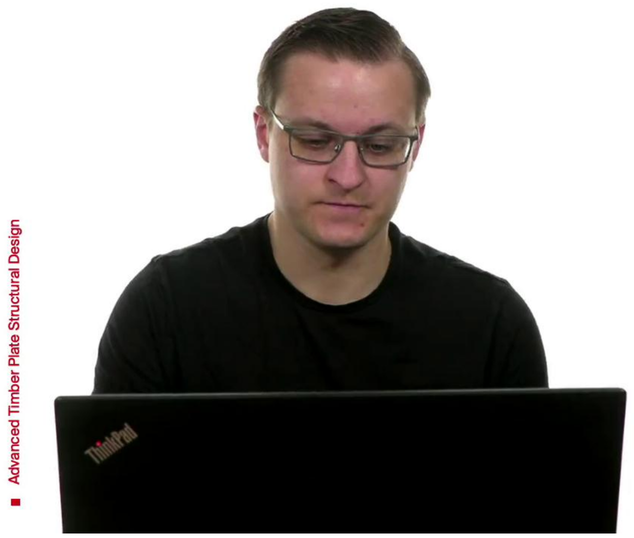
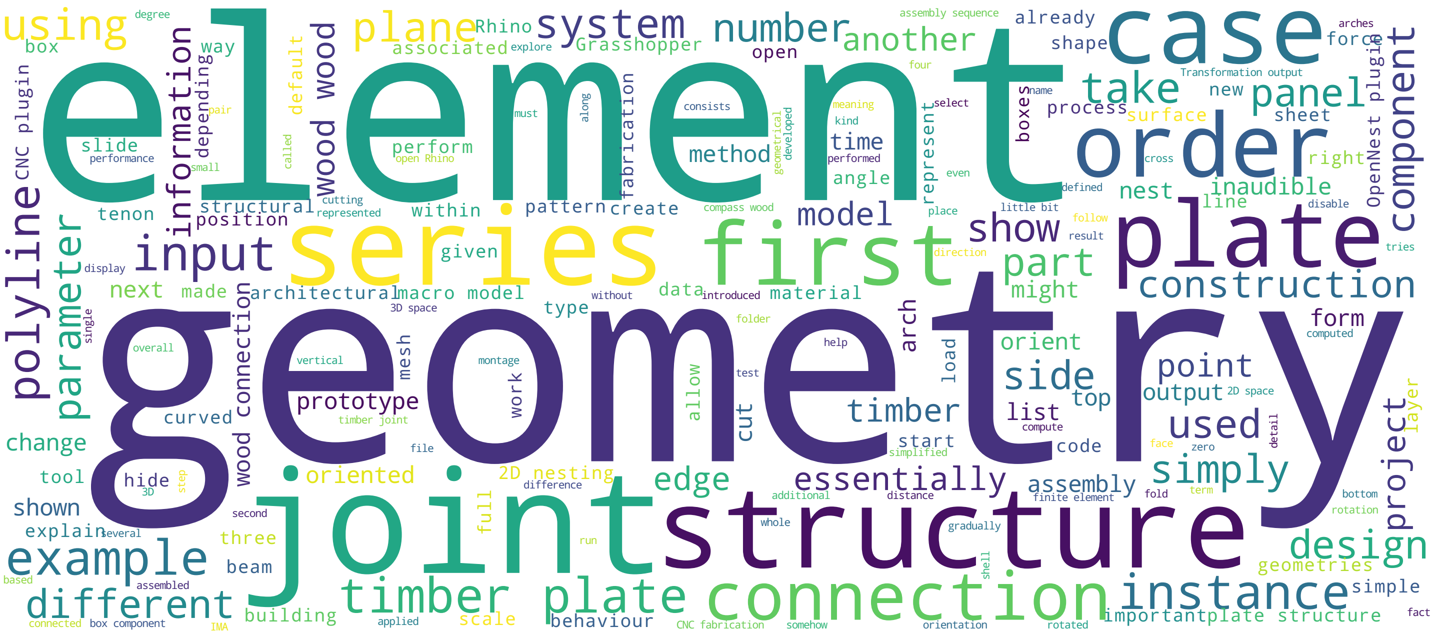
EPFL IBOIS




# Joinery Generation and Fabrication

Structure of the chapter:

- Introduction
  - Introduction to Joinery Solver
  - Quiz
- Geometry Generation
  - Selection of Study Cases
  - Joint Types
  - Joinery Generation for Timber Plates
- Conclusion and Fabrication
  - 2D Nesting
  - CNC Fabrication




[Search MOOC](#)

[Video](#)

EPFL

# Software Installation

Petras Vestartas

## Requirements:

- ☐ OpenNest
- ☐ Raccoon
- ☐ compas_wood (Rhinoceros3d Grasshopper versions)

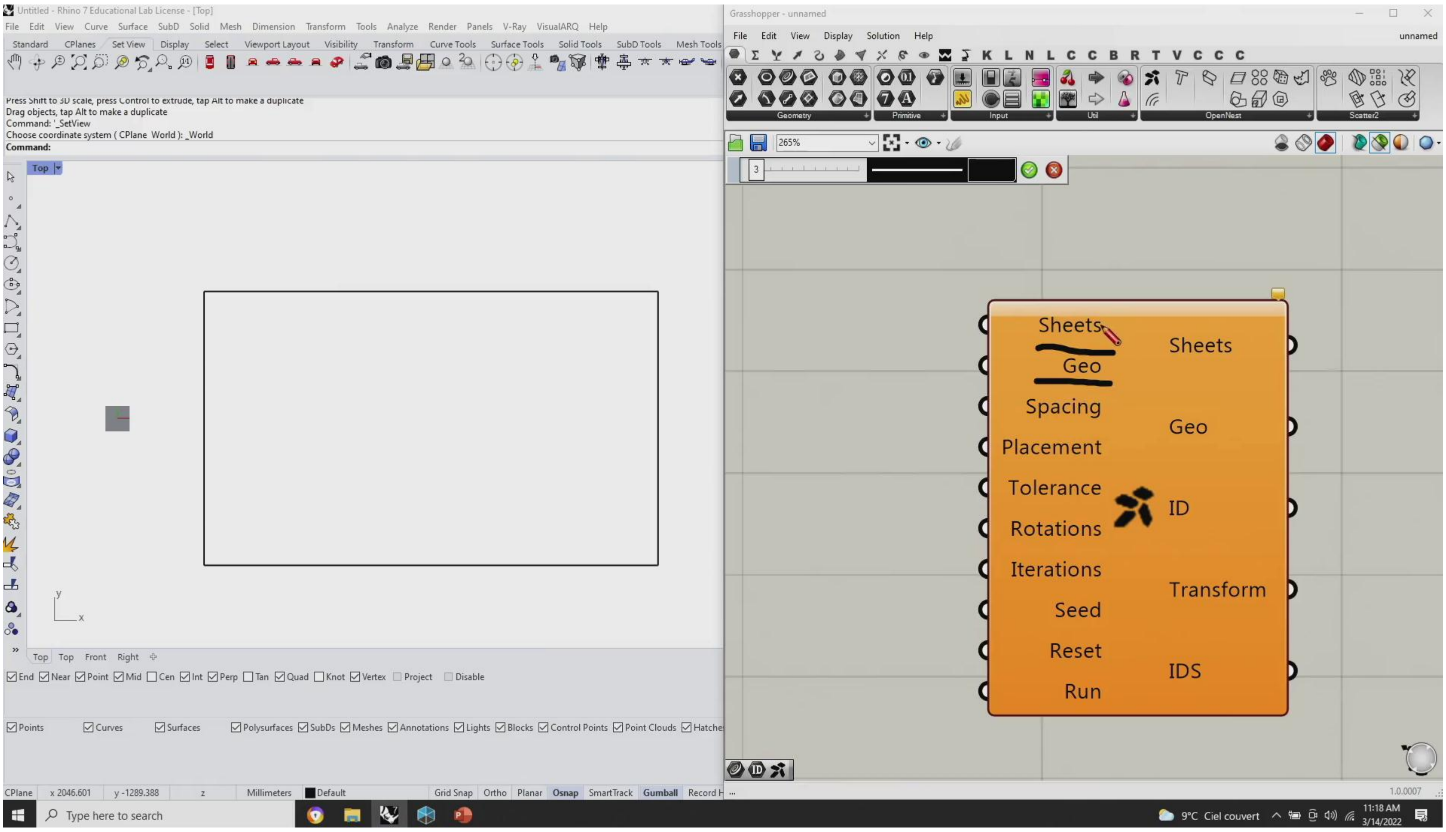


This is the third part for demonstration of 2D nesting using the OpenNest plugin and the CNC plugin, as the final practical example. We need following software applications. OpenNest for 2D nesting, then we need the CNC plugin. Then we also might use Compass Wood for demonstrating examples for fabrication and 2D nesting. Let's start. Let's first install the OpenNest plugin.

Notes

Summary

0m 04s

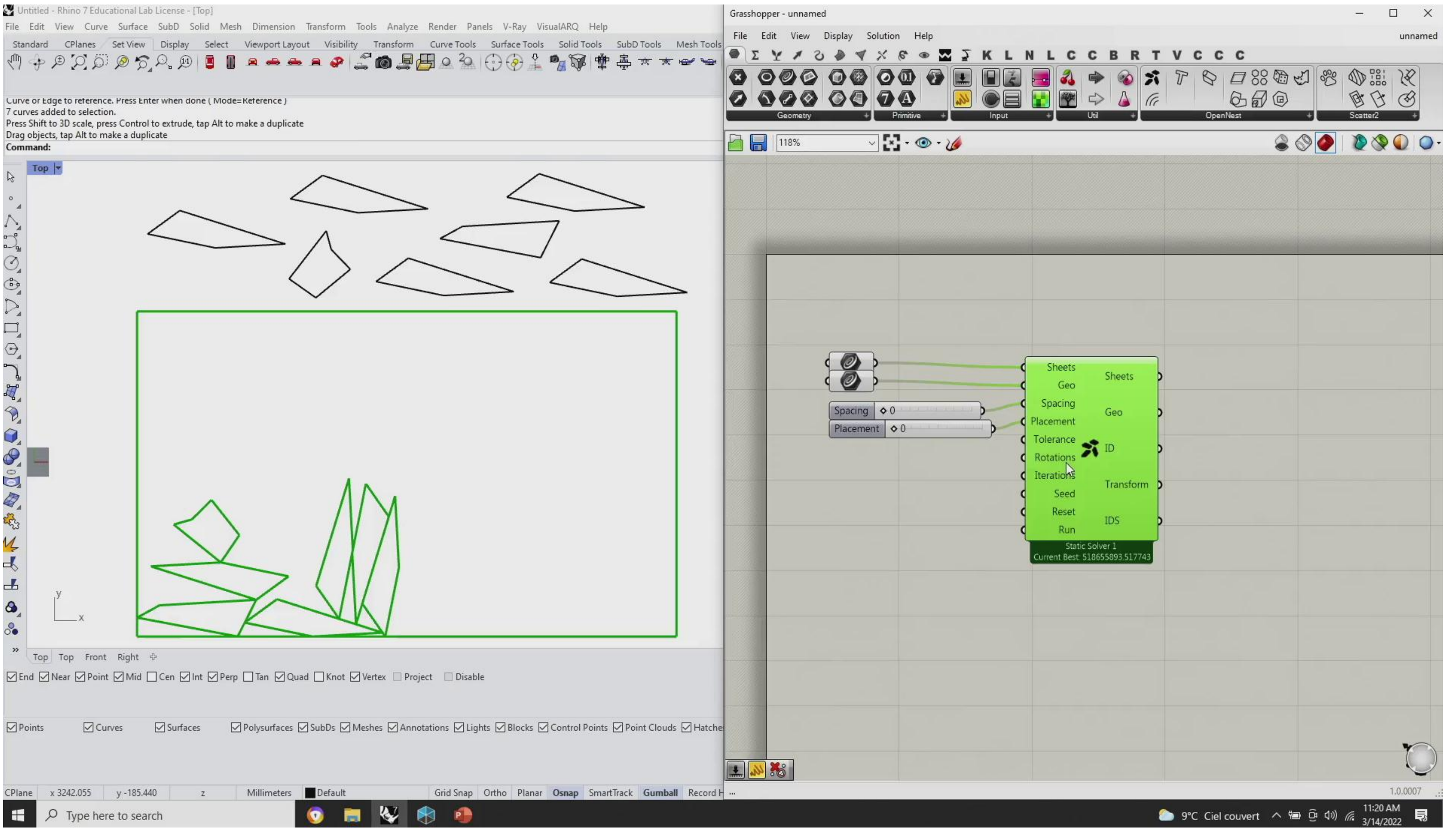



You can download OpenNest plugin from food4rhino. This is the website. In the search bar, you can type OpenNest and then you will find this plugin that was used for 2D nesting at IBOIS. It is essentially the plugin that helps to nest 2D elements of additional attributes as a Grasshopper and Rhino plugin, in order to optimise a series of cuts in order to save space for laser cutting or CNC fabrication. The installation is straightforward, you can click Install button that will open Rhino 7 or you can download manually for previous versions of Rhino. In case you're using the manual, the second method, you need them to follow these instructions which is a standard Grasshooper plugin installation procedure. Let's open Rhino and I'll show you a few examples for OpenNest. First, I'm going to explain the series of input parameters. This is the main component that you can find after installation, which is called OpeNest. You can simply click and open it in Grasshopper canvas. I'll explain a few parameters that are the principal ones. There are essentially two key parameters, the Sheets and the Geometry. The Sheets is representing the name of paper, cardboard or a plywood sheet and the Geometry is the geometry you'd like to nest.

Notes

Summary

0m 39s

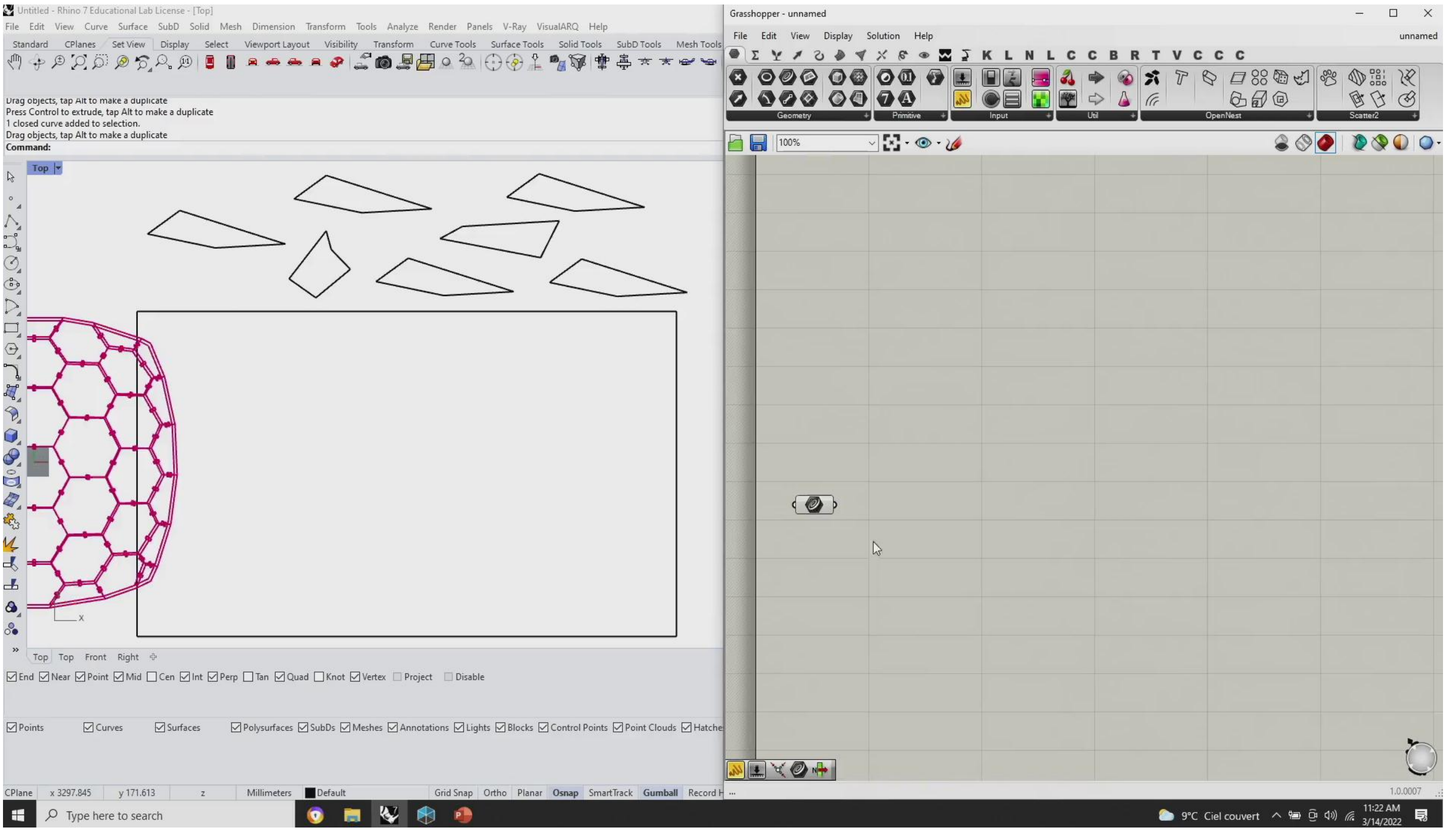



In this case, the sheet will be the rectangle drawn on your screen. For the sake of simplicity, draw a series of simple shapes that will be an abstract version for nesting. Then we go into a series of timber plates that I'm going to explain before. These are the elements that we're going to nest. Let's reference them to Rhino by right clicking and select Multiple Curves. If I connect right now these geometries, the geometry input, you see that something happened. The nesting operation has been already made. If I scale a little bit, you see that for simple shapes it works close to real time. For bigger shapes, for more elements, it takes more time. There are a few key parameters that most of the time you might use, the Spacing which is essentially the distance between elements. Then there are three parameters for Placement. It simply has an order of nesting, where one goes from one corner along one edge then another property goes from the left edge and the other one tries to nest elements diagonally. Let's leave by default the placement to zero, which is default one. Then the tolerance is how precise we can perform this operation. Rotations is an important parameter if you like to keep the default rotation.

Notes

Summary

2m 26s

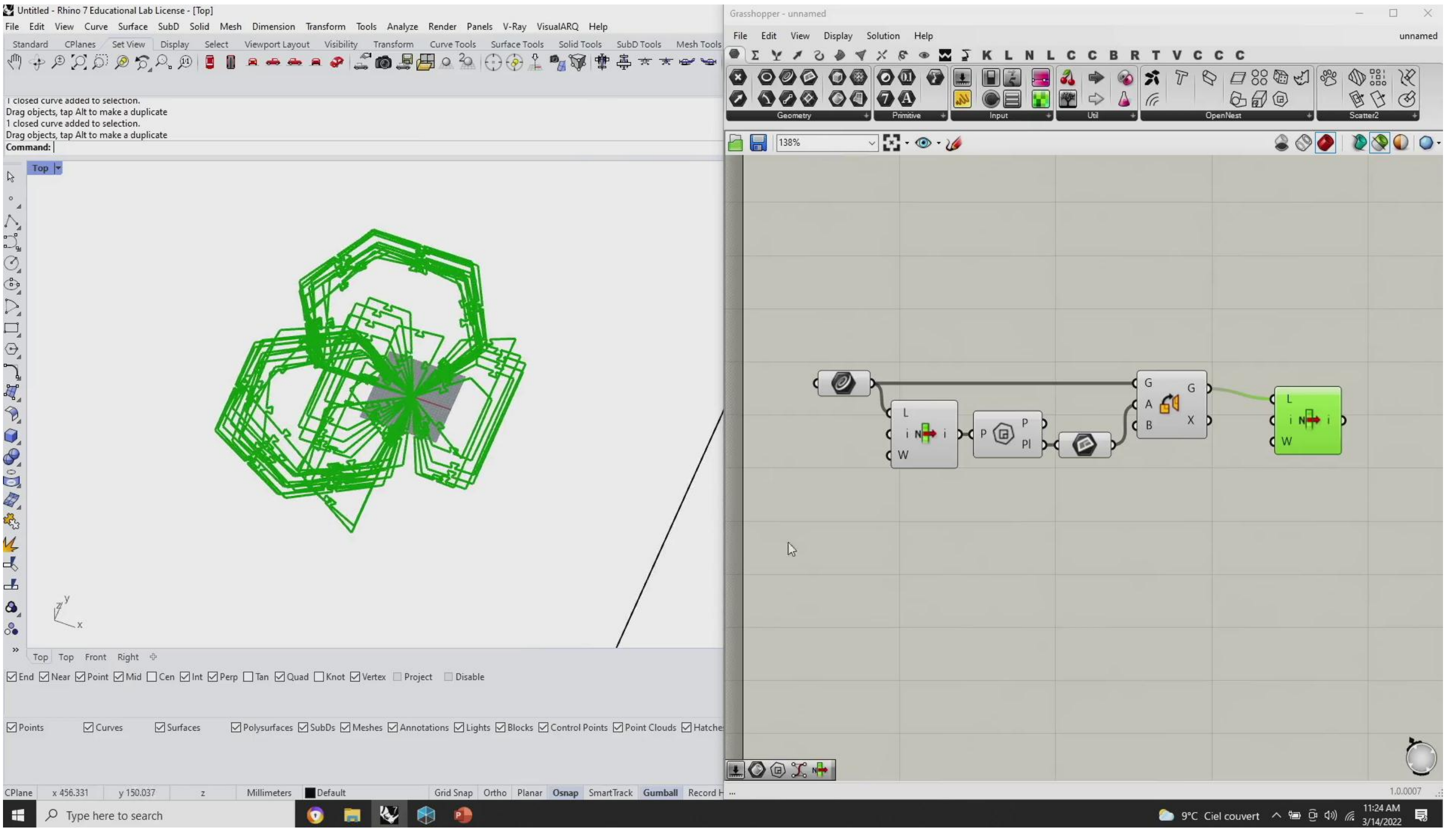



For instance, if we have the rotation which is one, it will keep objects initial rotation. If we have number two, that represents an element that can be rotated in space 180 degree. For instance, we have this element that you see here. Also next to it, it is the same element but rotated to 180 degrees. You can see that next to it there is an element that's rotated 180 degrees. Then there are a number of Iterations that might give better result or might be the same depending, if there's a better solution or not. Then there is a Seed value in case you want to explore different possibilities. There are a series of Boolean toggles to enable or disable the solver. The output of the separation can be seen in geometry output which is represented by these purple polygons. Then there is a second output that is a Transformation output that gives essentially the transformation between original shape and target shape by a series of transformation matrices. In order to explore the transform output I'm giving you the second example so I'll disable this geometry. For instance if you have an example from previous exercise of compass wood.

Notes

Summary

4m 10s

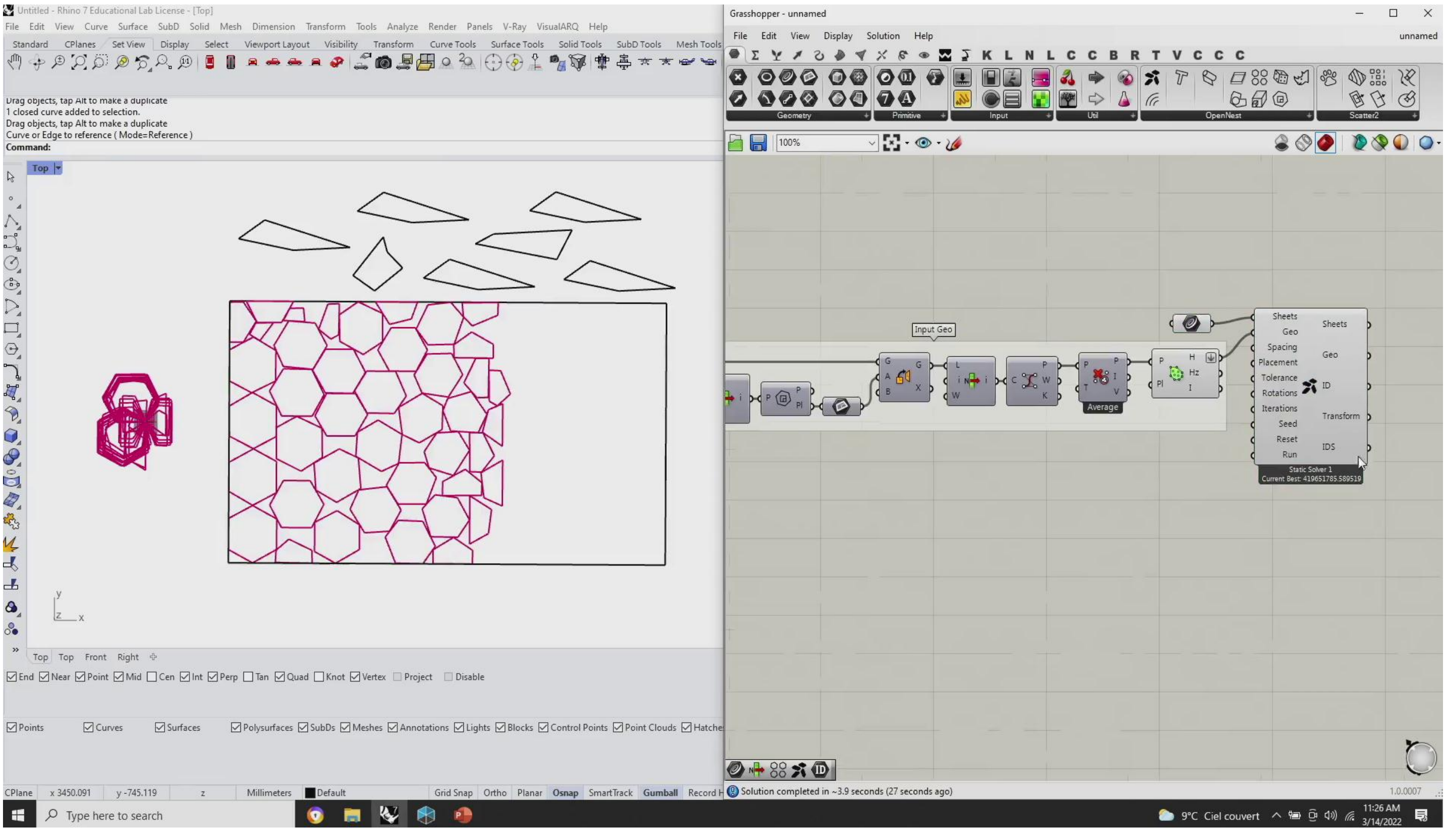



This has series of internalised polylines that have butterfly keys for connecting those plates we would like for instance, to nest those elements. The problem that you can see right now is that the geometry of OpenNest only works for the 2D polylines, how we can nest those pair of polylines with additional geometry? The first point that we need to follow, we need to orient this geometry from 3D space to 2D space. How we can do it? We can select the first item which represents the first polyline and we oriented to the X and Y plane. We can use a component Orient. Let's orient this geometry to XY plane but first we need to find somehow the plane position. For this we can use a component Project. It projects a polyline to a plane. If you would see a series of planes, it tries to project the geometry to a list of planes. We can use this input as an A parameter, we would like to orient from. Those proofs of polylines we oriented from 3D space to XY plane. There's orientation from 3D to 2D space happening here. Now the second part is we need to take again, the first polyline. That for instance can be like this. If you will try to run this OpenNest method using a series of polylines like that, the operation can be quite slow because each polyline contains joints.

Notes

Summary

5m 51s

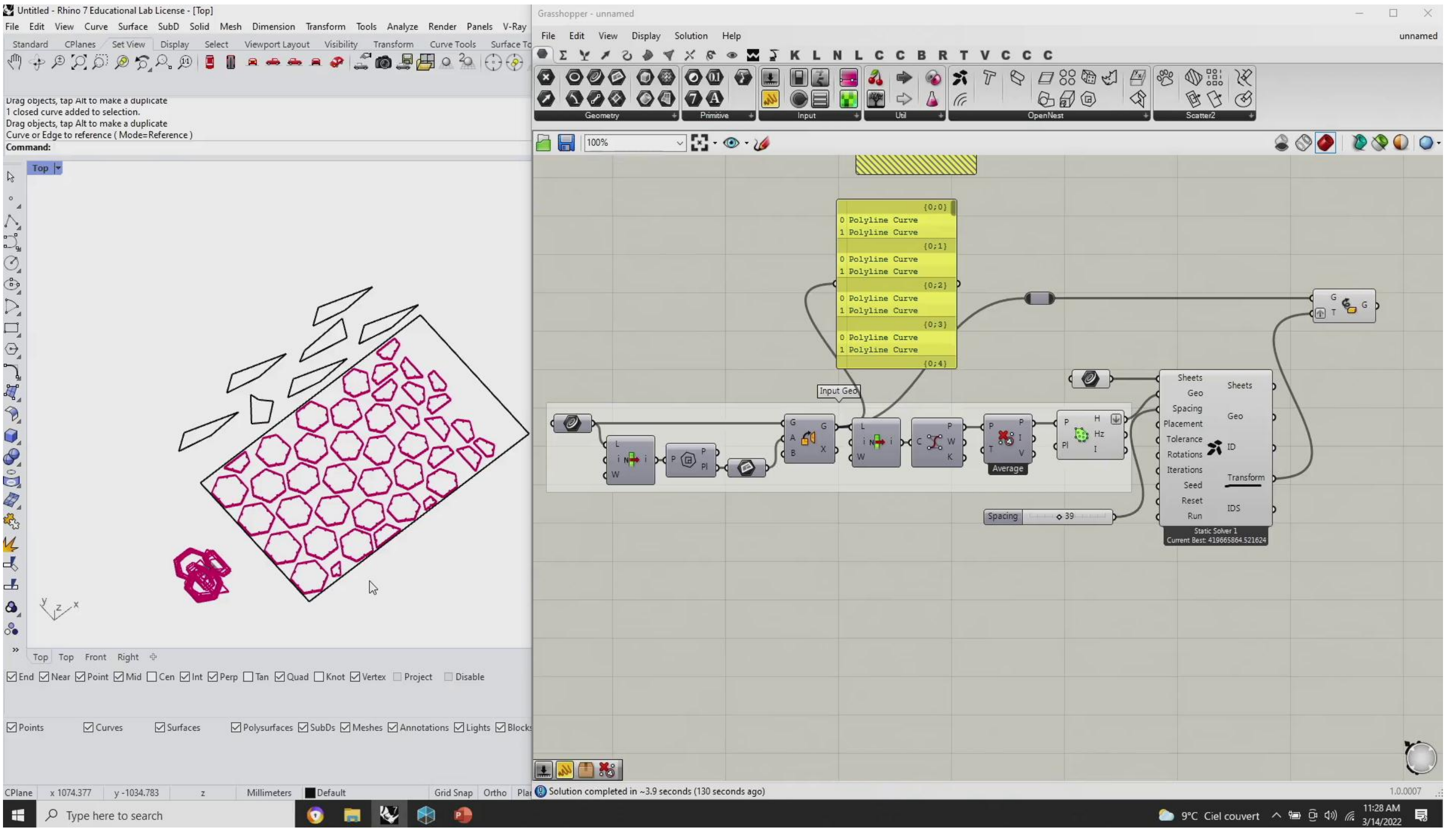



In consequence we have a lot of geometries with a lot of points that will run quite slowly. Therefore we need to simplify this geometry. I will hide the previous geometry so that it's more visible, better. Let's take a series of control points from this geometry using Control Points component. Then let's try to remove duplicates. We are performing the simplification of the number of points. Then we can compute the Convex Hull. What you can see right now if I hide this geometry, the initial geometry is much more simplified. All this part will be used for the input of the OpenNest. It will be Input Geo, which is input geometry. We use the same sheets to nest. Again, we can select this rectangle. We can use OpenNest for nesting, select sheets and select geometry. Now it is important that we need to graft the input, elsewise nesting operation runs multiple times for each individual sheet. Simply I will click Flatten. Now we can see multiple geometries nested. But if you understood the difference, we actually don't have the actual geometry here because we simplified those timber joints. We need to somehow use this Transformation output in order to orient the joint geometry to the nested geometry.

Notes

Summary

7m 48s




In this output we had series of polylines that already represent this. We can use the Transform component from Grasshopper in order to perform transformation on the original geometry. Let's connect G to G which is geometry and geometry and let's take Transformation but before that, we need to graft the input. Now we can see that series of geometries have been oriented. I will just clean up the example. The same way you can perform the orientation of NDC additional geometry by simply using this Transformation output. That is it for this example. In the following video, we are going to show how to use the CNC fabrication. For instance, in this geometry we can simply increase the Spacing. For instance, In the next video we're going to show how actually to cut these geometries using IBOIS CNC plugin.

Notes

Summary

9m 48s