HOW VIRTUAL BECOMES REAL

Parametrise a grid of timber laths for post-forming (G_02)

Creative team
Alberto Pugnale and Stanislav Roudavski

Content developed by
Alberto Pugnale, Alessandro Liuti and Louis Gadd

Supported by the faculty
"Strategic Initiative Funding"

THE UNIVERSITY OF MELBOURNE

msd
Melbourne School of Design

FACULTY OF ARCHITECTURE, BUILDING AND PLANNING
www.msd.unimelb.edu.au

0:00 SLIDE - Cover

- In this tutorial, you are going to parametrise a grid of orthogonal polylines, which are meant to simulate the laths of a post-formed timber gridshell, prior to erection. At the end of this tutorial, follow the instructions of tutorial G02.1 to perform the actual post-forming

0:16 Slide - Intro

- Just a couple of notes before starting. Timber gridshells are generally constructed using 4 layers of orthogonal laths, 2 per each direction. To simulate post-forming in Grasshopper, however, 1 layer per direction is generally enough. The same strategy can also be applied for the realisation of simple physical models
- Now, let's start

0:37 GRASSHOPPER

- First, let's open the Rhino **Options** panel and make sure the **Units** are set to **Metres**
- Now, you are going to draw a grid of laths which is oriented according to the XY plane
- Get a **Series** component to define the number of laths you will have in the X direction. For instance, set the number *Count* to 20, to generate a list of numbers ranging from 0 to 19
- Get another **Series** component to define the number of laths in the Y direction. You are going to define a square grid, so keep the *Count* value to 20
- You now want to generate 20 polylines in the X direction and 20 polylines in the Y direction. To do so:
    - Get a **ConstructPoint** and feed the first list of numbers into the X input. Then, feed the second list of numbers into the Y input. At this stage, you have only generated 20 points, as the two sequences of numbers are univocally combined as X and Y coordinates. You can check the data structure using a **PointList** and a **ParamViewer**. You need to *Graft* the X input to define a new branch for each item in the list. This will associate each X coordinate with the 20 Y coordinate values, therefore generating 20 columns of points
    - Get a **Polyline** component and feed the **ConstructPoint** output to generate the first 20 laths along the Y direction
    - Repeat the same procedure to generate the laths along the X direction, but in this case remember to *Graft* the items of the Y input

2:24 Slides – Grid trimming

- In most of the cases, you don't want to use a square grid of laths, but a different cut
- Furthermore, if the gridshell has to be supported at the 4 corners, you are going to face a structural and technological issue. In this case, you will need to trim small portions of the grid to generate 4 edge beams, which can better transfer the forces to the ground

2:44 GRASSHOPPER

- To preserve the lath data structures, and make sure the cut doesn't trim between polyline vertices:
    - Draw in **Rhino** a closed curve, polyline or region. You can define a simple symmetrical cut for the purpose of this exercise
    - Now get a **Curve** component in Grasshopper and reference your closed curve
    - Now use **PointInCurve** to test which lath vertices are included in the boundary you have just defined
    - Use **Dispatch** to extract these points from the **ConstructPoint** lists
    - Finally, replace the **Polyline** component input with this new set of points
    - You need to repeat the same operation for the laths in the orthogonal direction, just copy and paste **PointInCurve** and **Dispatch**

4:02 Slides – Final code

- This is the end of the tutorial. In the support master file, you will find the refined Grasshopper definition, which is linked to the code of tutorial G02.1:
    - The two Sliders highlighted in pink allow you to define the number of laths along the X and Y directions
    - The Curve component highlighted in green can be used to define a trimming boundary, which you can import directly from the Rhino interface
    - The groups of components on the right generate the actual laths in the X and Y direction, providing, as outputs, two lists of segments and respective vertices

4:39 END OF TUTORIAL G02