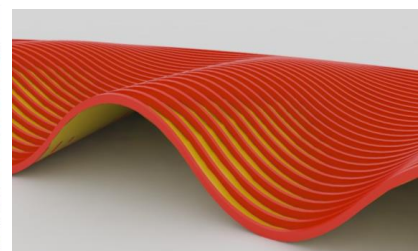


ZENTRUM PAUL KLEE



What you will learn

1. Organize and handle data through tree data structure
2. Use and handle a large set of NURBS curve through their control points

List of relevant components used

- Point Decompose
- Flip Matrix
- Evaluate Curve
- Range
- Graft
- Graph Mapper

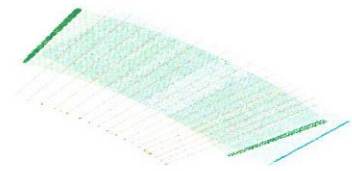
Introduction

A series of diminishing “Waves” on a concentric grid forms the roof structure of the “Zentrum Paul Klee ” in Berne/Switzerland, designed by Renzo Piano (2004). Because of their changing curvature and size, the final geometry is the result of quite complex interdependencies.

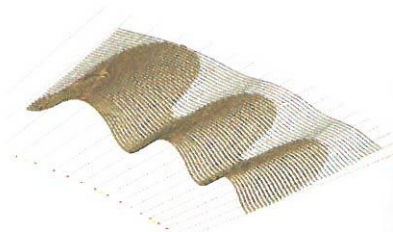
To accompany and support the design process, a parametric model was developed. It enabled the architects to test countless alternative solutions until the final design was determined. And in a final step, it was used to generate the construction documents for the steel contractors.

The development of the whole and its parts was achieved by an integrated process in three steps:

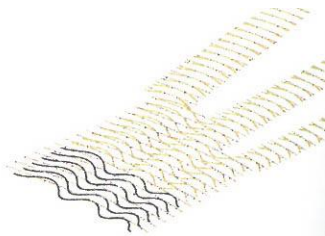
1. **2D parametric grid.** The design concept is translated into a two dimensional parametric grid, which determines the positions of the ridges and valleys of the roof



2. **3D parametric waves:** the three-dimensional geometries of the waves are built on the grid. From the result, any kind of section and view can be derived to check the geometry during the design phase

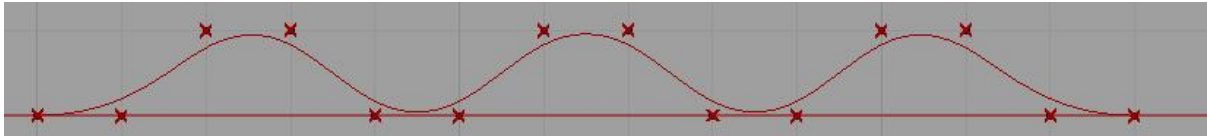


3. **Deduction of steel geometries.** The geometry of the curved I-beams is mapped to two-dimensional plans for the contractor, including webs and flanges. Also the geometry for the sheet metal cladding of the I-beams and the roof is unfolded



We will implement step 1, 2d parametric grid, and step 2, 3d parametric waves

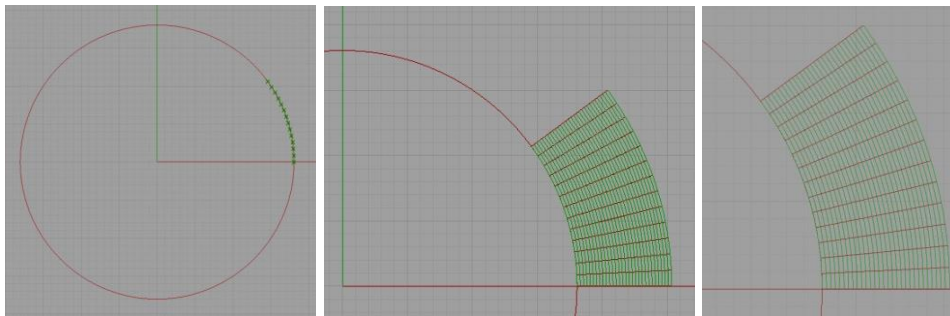
Implementation of 2d parametric grid



The roof geometry is defined by 30 undulated beams, each one describing in plan an arc of circumference. The beam geometry is defined with the use of NURBS curves each with 14 control points. Therefore the base grid will be composed by $14 \cdot 30 = 420$ points.

- 1- Draw a circumference with an arbitrary radius (use a slider).
- 2- On this circumference draw 14 points enclosed in an arch of $1/10$ of the circumference.

HINT: use the **Evaluate Curve** component to generate the points on the circumference. Which numbers you should use as t parameter values in order to define 14 points enclosed in an arch that measure one tenth of the circumference? It is convenient to use the component **Range** to generate the numbers.



- 3- Starting from each of the 14 points enclosed in the $1/10$ of the circumference, draw a segment of fixed length perpendicular to the circle and radiating outwards the circle. The length of the segments define the depth of the building.

HINT: use the components "**vect2Pt**" and "**line SDL**" to draw these segments.

- 4- Define 30 points in each of the radiating segments; the number of points you define will generate a correspondent number of undulated beams. It is convenient to create the points using **Evaluate Curve** and **Range** component.

HINT: Be sure to **Graft** the segments before you input them in the **Evaluate Curve** component in order to generate the 30 points in each segment. Points will be organized in 14 branches, one for each segment, each one containing 30 points (see below, Figure 1). With the component **Flip Matrix** you can transform the tree data structure to be made of 30 branches, each one containing 14 points (see below, Figure 2). By doing so, each branch now contains 14 control points of one of the 30 undulated beam.

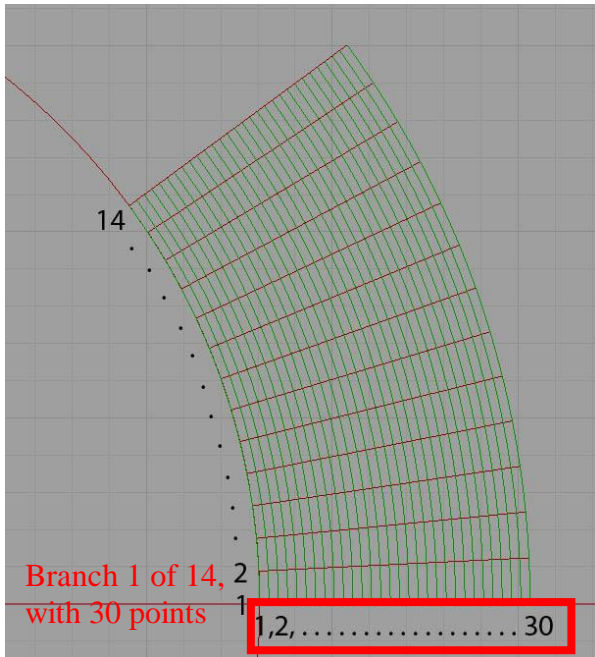
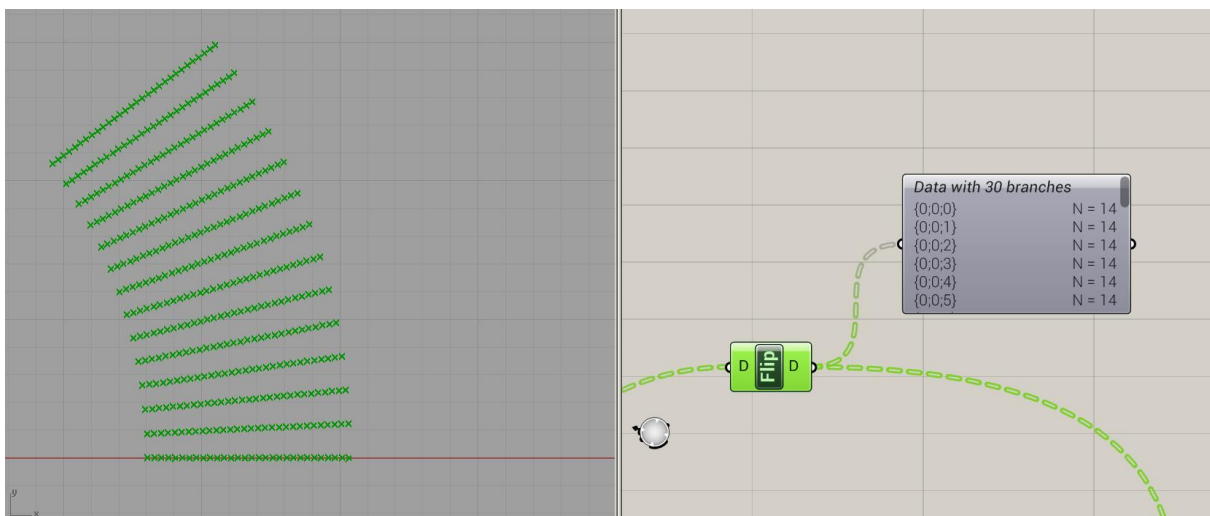


Figure 1. 14 branches with 30 points



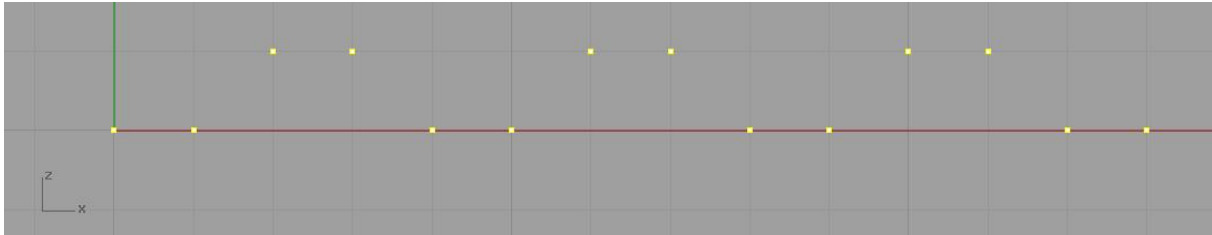
Figure 2. 30 branches with 14 points

Review – This is how your drawing should look like at this point



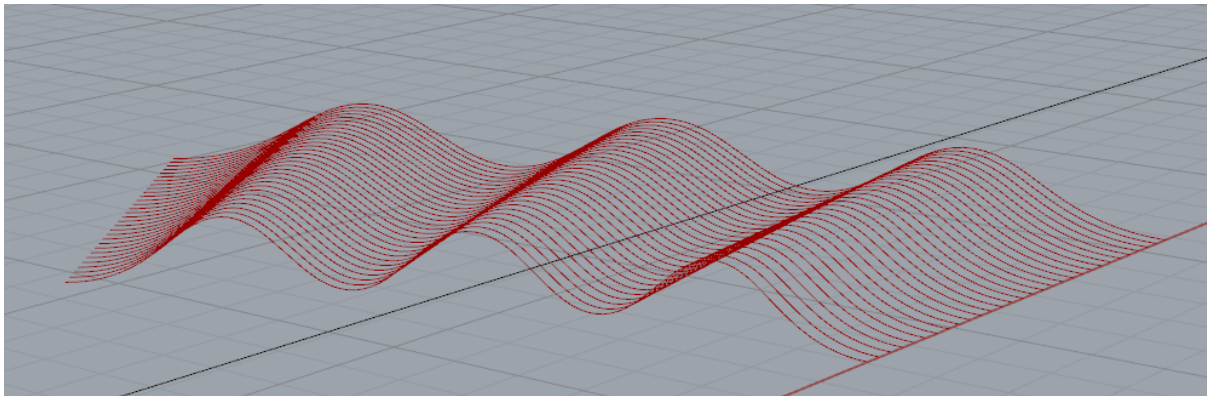
Implementation of 3D parametric waves

- 1- We first execute few operations directly in the Rhino canvas (not Grasshopper). Select the Front viewport in Rhino. In this viewport draw with Rhino single points command 14 points, equally spaced, two by two at two different heights following as a reference the figure below.



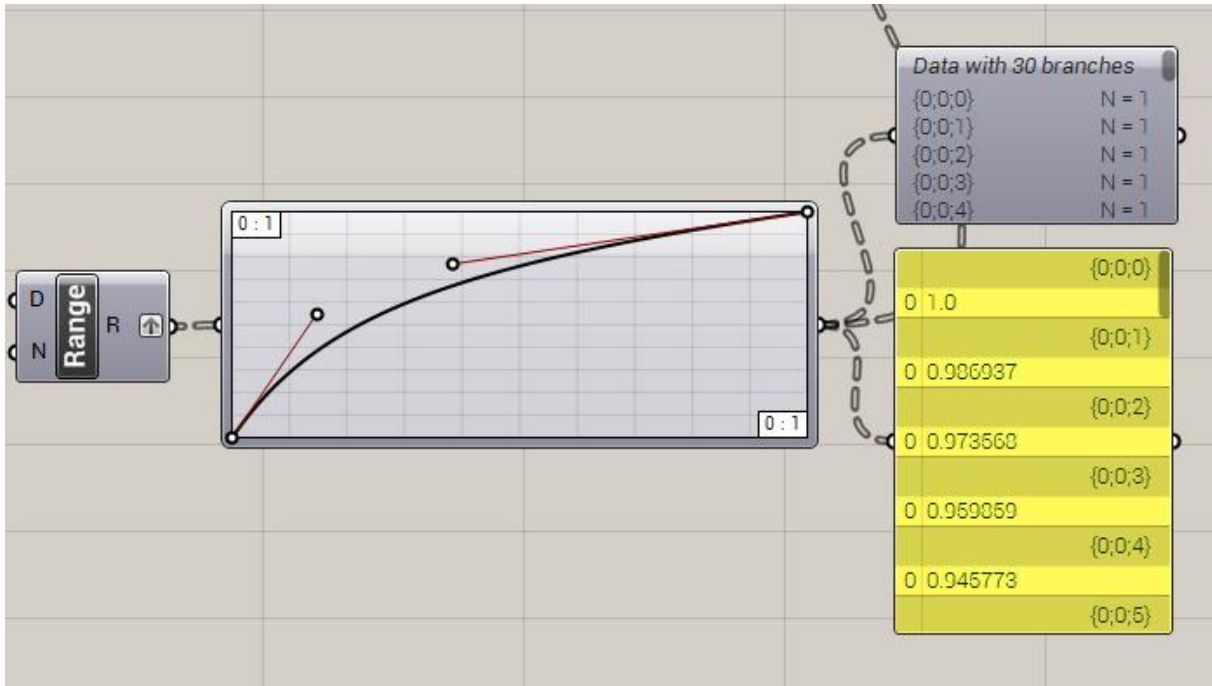
- 1- Now select them and reference them into a grasshopper **Point** component (Params>Geometry). Now you can use the z coordinate of these points as the z coordinate of the points of each of the 30 undulated beams. With the component "Curve" (Curve>Spline) generate the geometry of the arches (the component uses a NURBS curve).

HINT: Use **Point decompose** "component and then **Point xyz** component.



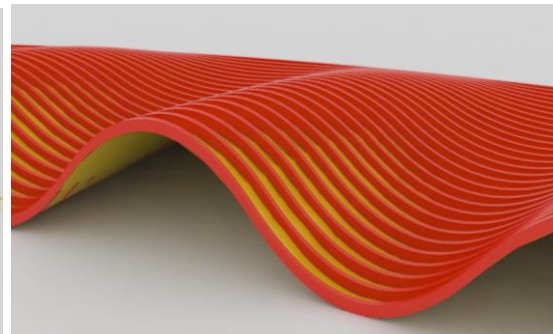
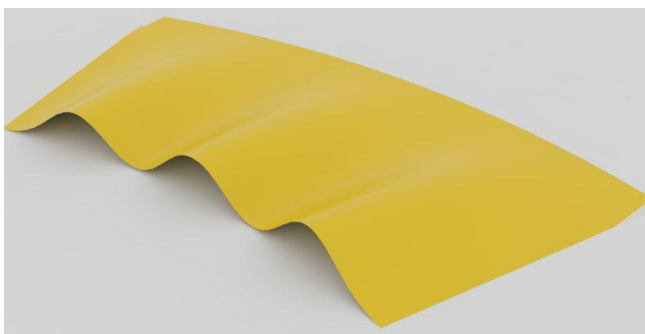
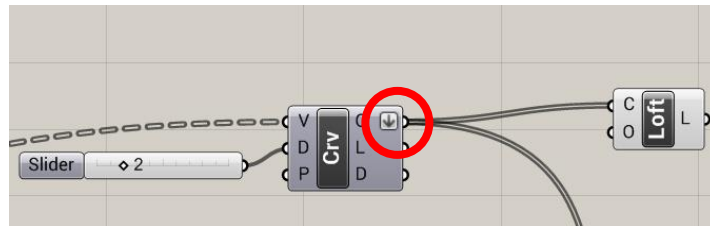
- 2- The idea is however to have the possibility to modify the z coordinate of the 14 points drawn in Rhino before using it as the z coordinate for the 14 control points of the undulated beams. Use a "**multiplication**" component to increase or decrease their values with a slider acting as the factor of the multiplication.
- 3- In this project the height of the undulated beams is also decreasing gradually. We control the diminishing height with a reference "**Bezier**" curve. Use the component **Graph mapper** (Params>Input) then right click> graph type> Bezier. The Bezier curve is not generating a real geometry in this case, but it is used only for generating coefficients that are used to set the diminishing height of the undulated beams.

HINT: use the output of **Graph Mapper**" as multiplier of the z coordinates of the 14 points before using them as the z coordinates of the undulated beams. You will need to Graft in order to provide each arch (i.e. each branch) with different coefficients, decreasing gradually.



OPTIONAL

1- Use **Loft** component to generate the surface. The curves should be "flattened" before.



2- Use **Extrude** to turn the curves of the beams into solid geometry.

You will need to use the component two times consecutively, one to define the beams width, the other to define the beams height.