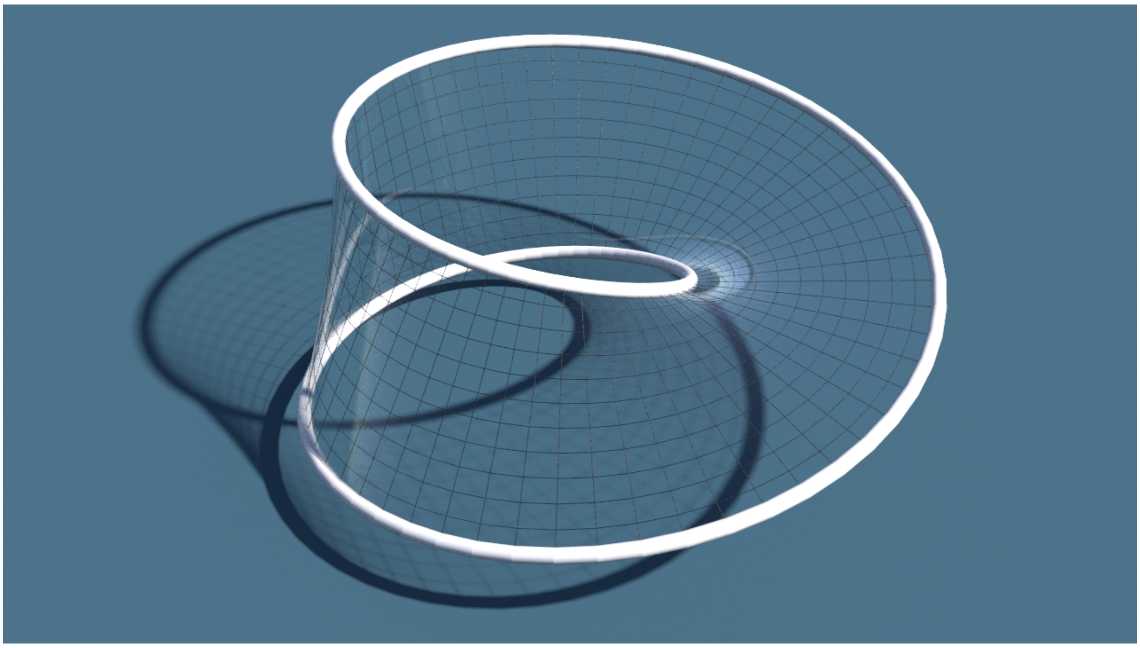
Technische Universität Berlin
Institut für Mathematik
Sullivan / Knöppel
http://www3.math.tu-berlin.de/geometrie/Lehre/SS17/MathVis SS 17

Mathematical Visualization

Assignment 1 - Rendering a Möbius strip[1]

As a warm up have a look at *Visualizing Discrete Geometry I* and *II*. There some general things are explained as how to draw tubes around edges, spheres around points, how to color them and how to combine these geometries again. To create a ambient scene with light see *Simple Ambient Scenes*. Also have a look at *Scenes with White Background* to get an impression how to use materials. These posts already contain enough information to create pictures like the one below.



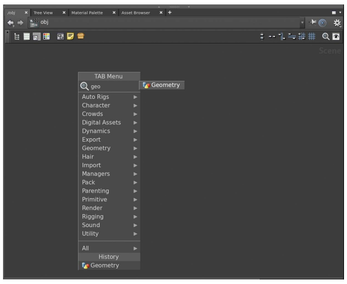
Here we basically used the parametrization $f\colon [0, 2\pi] \times [-1, 1] \to \mathbb{R}^3$ given by

$$f(\varphi, t) = \begin{pmatrix} (1 + \frac{t}{2}\cos\frac{\varphi}{2})\cos\varphi \\ (1 + \frac{t}{2}\cos\frac{\varphi}{2})\sin\varphi \\ \frac{t}{2}\sin\frac{\varphi}{2} \end{pmatrix}.$$

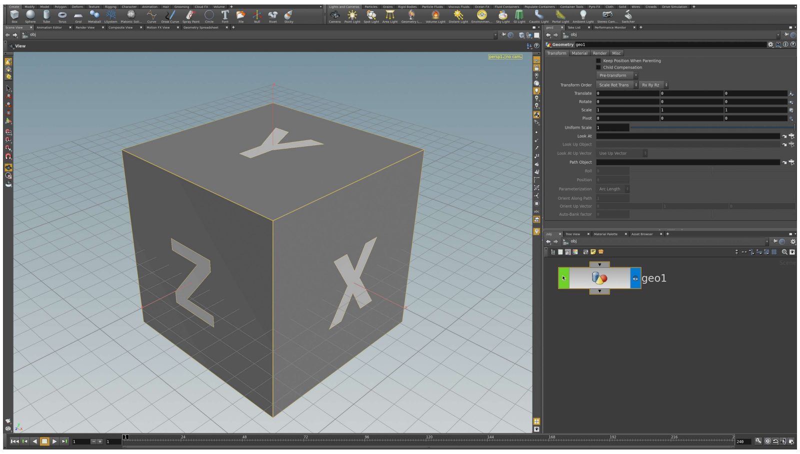
Once one got used to Houdini the implementation is easy. For now we proceed in small steps:

1) Create a geometry: Therefore move the mouse over the *Network view* and hit *tab*. This opens a panel which offers you a variety of nodes. We want to create a geometry. Typing *geo* offers you single choice - a *geometry object node*.

[1] due 7.5.2017

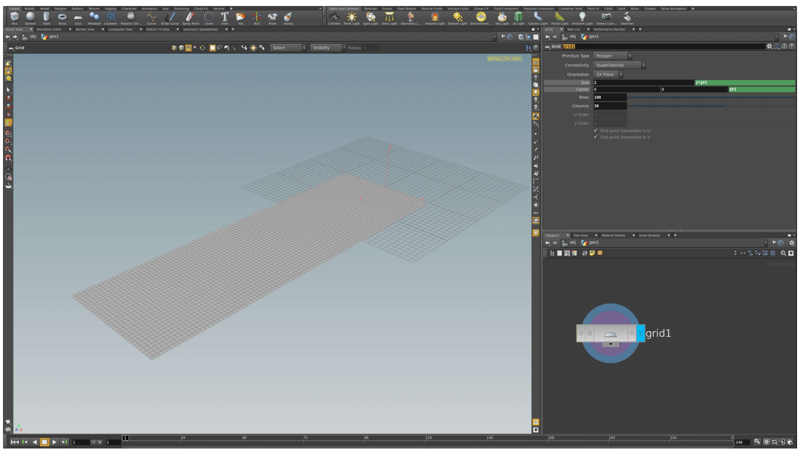


If you hit enter the panel disappears and you can place the node where you want. Move the mouse to the position you want the node to be placed and right-click or hit enter again. A node with the name `geo1` appears which contains a default geometry. A textured cube is shown in the *Scene View*.



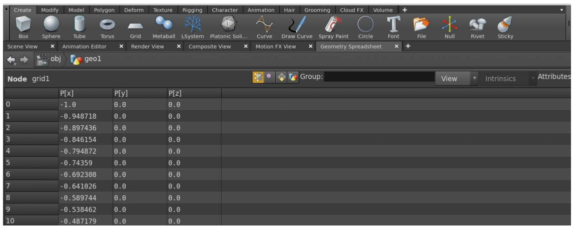
Double click on `geo1` to get into the node. There you find another node called `file1`. Click on the node to select it and hit delete. Now the default geometry has disappeared and we are ready to place our own geometry.

2) Create a parameter domain: A planar geometry can be created using a *Grid node* - move the mouse to the place where the node shall appear in network panel hit *tab*, type `grid` and hit *enter* twice. A node called `grid1` is created and a rectangular quad mesh appears in the *Scene View*. Click on the node `grid1`. If the node is selected its parameters are shown in the parameter view:

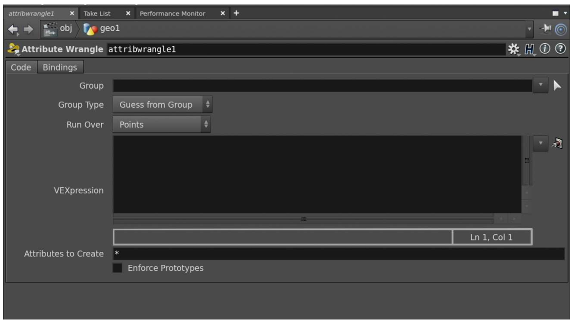


Choose the dimensions such that the grid fits with the domain of f (in the parameter panel π can be accessed via HScript by `$PI`) and choose the resolution you want.

3) Modify its point positions: A *geometry object* consists of points, vertices and primitives on each of which so called *attributes* can be stored - as well as on the object itself (*detail attributes*). Attributes can be created by the user to store the data needed in the particular application at hand. But there are also default attributes which are needed anyway, e.g. simply to display the geometry. So the points come with a position attribute `P` as you can also see in the *Geometry Spread Sheet*.



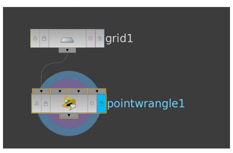
This is what we want to modify, which can be done by an Attribute Wrangle node. In the *Parameter View* you find a text field called *VEXpression* where you can write code in *VEX* (a list of all VEX functions is found *here*).



The code block is is executed in a loop. A drop down menu in the parameter panel offers you 4 options what to loop over - points, vertices, primitives or detail (in the latter case the code block is executed only once).

Note: There are also nodes called Point Wrangle and Primitive Wrangle - they are somewhat redundant as they are just an Attribute Wrangle with a particular preset what to loop over. Though after a while one might find this quite convenient.

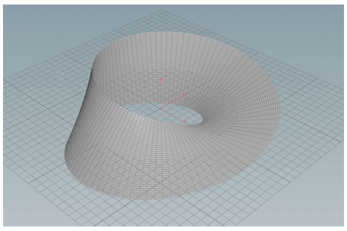
As we want to modify the point positions we want to wire a *Point Wrangle node* to `grid1`. Therefore click on the rectangle at the bottom of the *Grid node* and drag the mouse. This pulls out a wire. Hit *tab*, type `pointwrangle` and hit *enter* twice. It appears a node with the name `pointwrangle1`. (You can also create the node first and wire afterwards.) Select the button at its right end (*display*) so that the *Scene View* displays the output geometry of `pointwrangle1`.



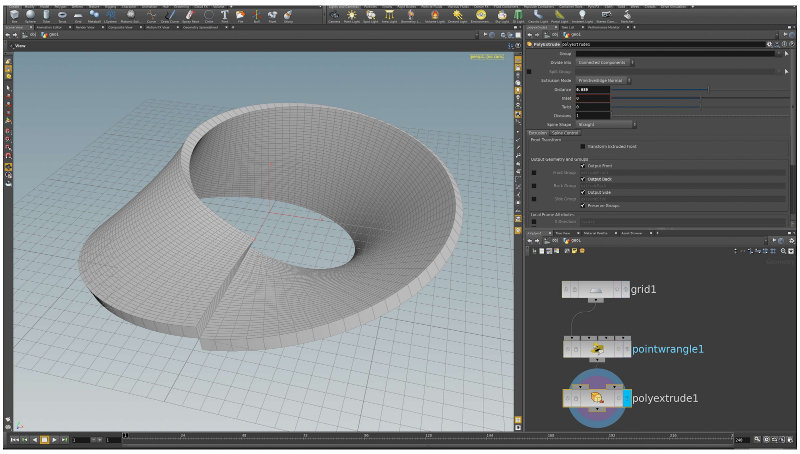
Now we can finally implement the parametrization f: In a Point Wrangle node we can access the point attribute `P` by writing `@P`. So the VEX code implementing f could look as follows:

```
// method that implements f
vector f(float phi; float t){
    // init a 3-vector
        vector p;
        // note that the y-axis points upwards
        p.z = (1+ t*cos(phi/2.)/2.)*cos(phi);
        p.x = (1+ t*cos(phi/2.)/2.)*sin(phi);
        p.y =  t*sin(phi/2.)/2.;
        return p;
}

// coordinates in the zx-plane
float phi = @P.z;
float t = @P.x;

// modify point positions
@P = f(phi,t);
```

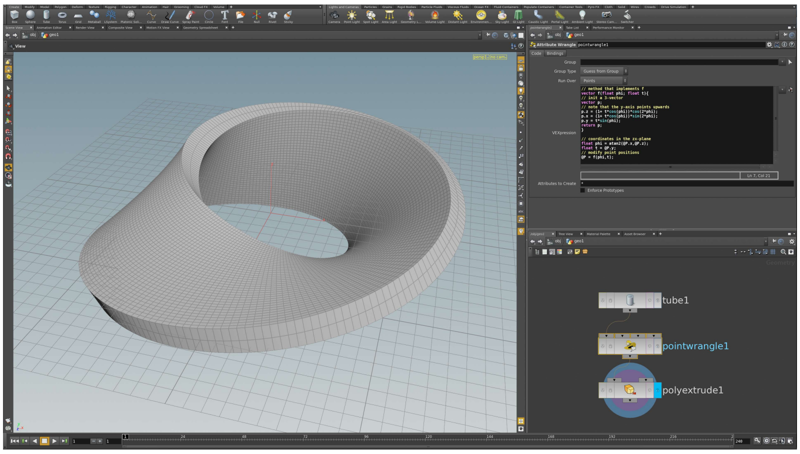
The result looks as follows:



Though if we e.g. want to thicken the surface with a *Poly Extrude node* we observe the jump of the normals. Actually this is also a problem for extracting the boundary curve or for applying certain materials.

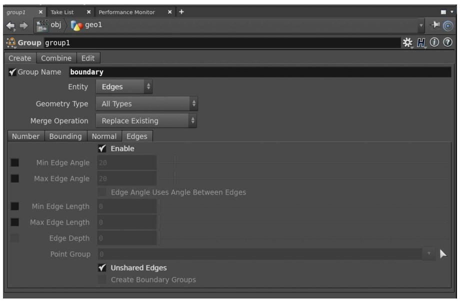


4) A possible solution: Instead of using a square we could use a cylider which just covers the Möbius strip twice. Though we then have a doubly-covered strip, we have fixed the discontinuity in the normals. So let us wire up a different network as shown in the picture below.

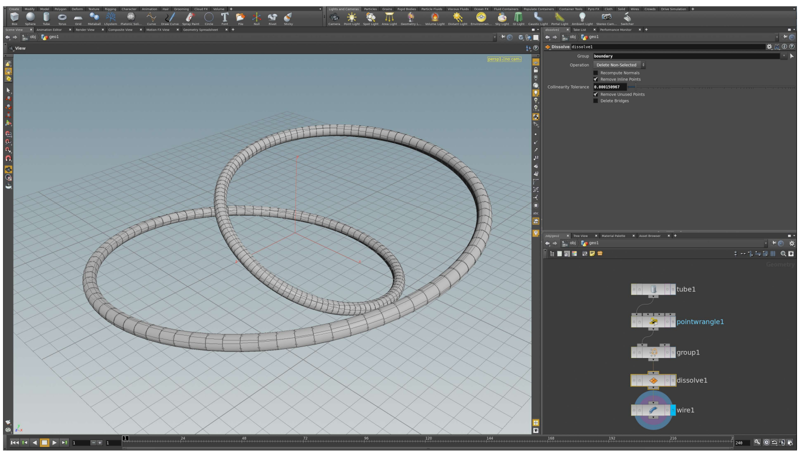


The names of the nodes in the picture are self-explanatory. The resolution should be good enough to see everything when you zoom in. Just make sure that in the parameter panel of the *tube node* the primitive type is set to *Polygon*.

5) Extracting the boundary curve: Let us finish with a way to extract the boundary curve. This can be done by a *Dissolve node*. First wire a *group node*. This node offers the possibility to group all edges that are not glued to another face, i.e. which are boundary edges. Therefore select *Edges* in the *Entity* drop down menu uncheck the checkbox in the *Number pane* pane and check instead the Enable-checkbox in the *Edges pane*. There you also check *Unshared Edges*. Specify a group name, e.g. *boundary*.



Write the group name into the *Group*-textfield of the *dissolve node* and select *Delete Non-Selected* in the *Operation* drop down menu. This probably will need a relatively small *Colinearity Tolerance* (and unfortunately sometimes even then fails for too high resolutions). Below a picture of the curve we get.



Homework 1: Wire up the network, assign materials, add lights and render a nice ray-traced Möbius strip. Be creative!