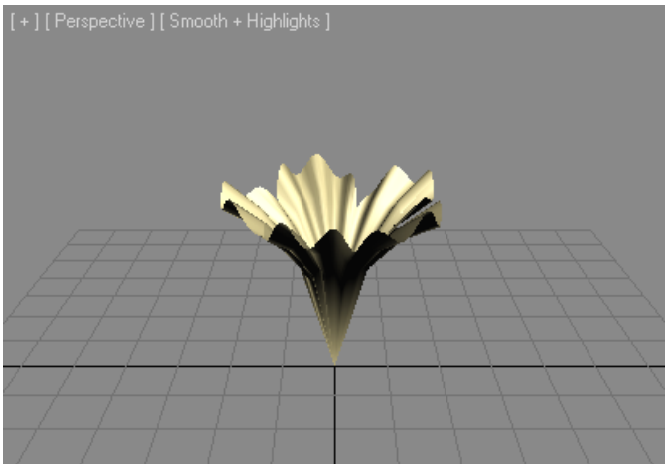
MAXSCRIPT

FLOWER CREATOR

The **Flower Creator** maxscript (whose complete name is **Flower Creator Geo**[metric] **Object**) is used, obviously, for creating flowers. Also, all flower's parameters are animatable.

The installation of **Flower Creator** is very simple: just copy the maxscript in the **Scripts\Startup** folder of the Max's root directory. Copying **Flower Creator** in the **Startup** sub-folder guarantees the automatic loading of the maxscript –like a plugin– every time you start 3ds Max. Once Max is started, you'll find **Flower Creator** in the **Create** panel > **Geometry** > **havardsc** category.

For creating a flower, click on the **Flower Creator** button, and click and drag in any viewport. The initial click will position the flower, while the dragging determines its overall dimensions. The default flower is shown in this picture:



In order to see both sides of the flower, it's recommended to set the **Backface Cull** option. This option maybe globally set for all the scene's objects through:

- **Customize** menu **> Preferences > Viewport** tab **> Backface Cull on Object Creation**

Or this option may be individually set for each flower by any of these procedures:

- **Display** panel **> Display Properties** rollout **> Backface Cull**
- **Tools** menu **> Display Floater >Object Level** tab **> Backface Cull**

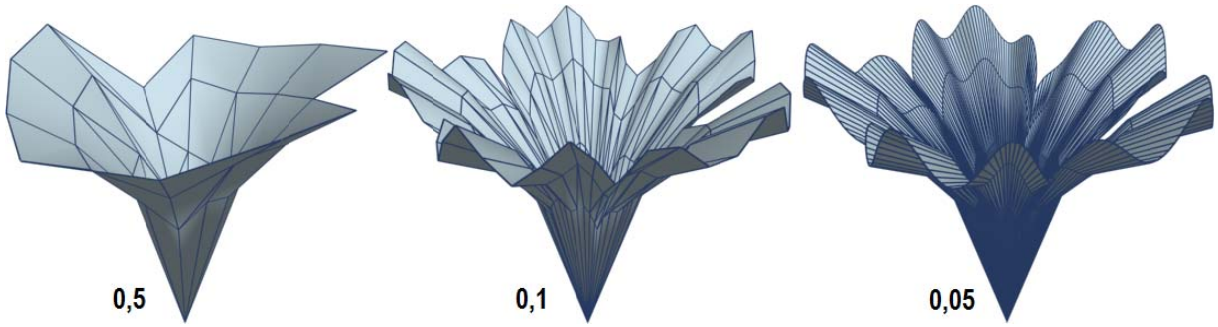
Once the flower is created, you'll find the following parameters in the **Modify** panel:

Mesh Options group

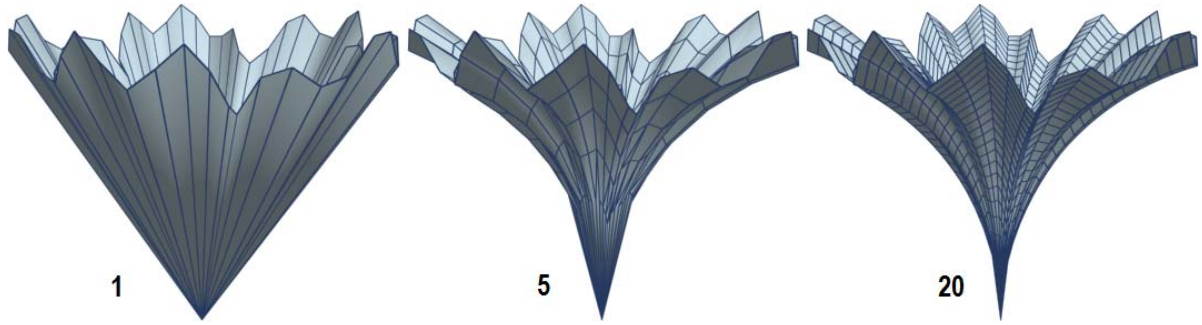
In this group you set the global dimensions of the flower.

The **Size** spinner controls the size of the flower in current Max's units.

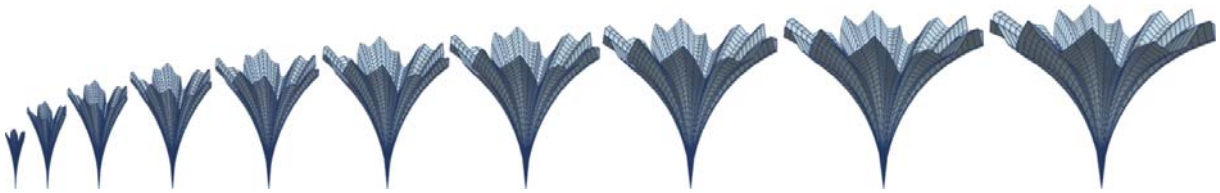
The **Details** spinner controls level of details of the mesh by establishing the distance between generatrixes or "meridians". Its values go from **0.01** to **0.5**. The lower the value, the higher the number of meridians:



The **Cap Segments** spinner controls the level of details of the mesh by establishing the number of "parallels" (similar to the **Cap Segment** parameter in a cylinder). This is a very sensitive parameter, so you should change it discreetly in order to avoid a too-much dense mesh and the possible crash. A reasonable value softens the curve that defines the flower's lateral profile:



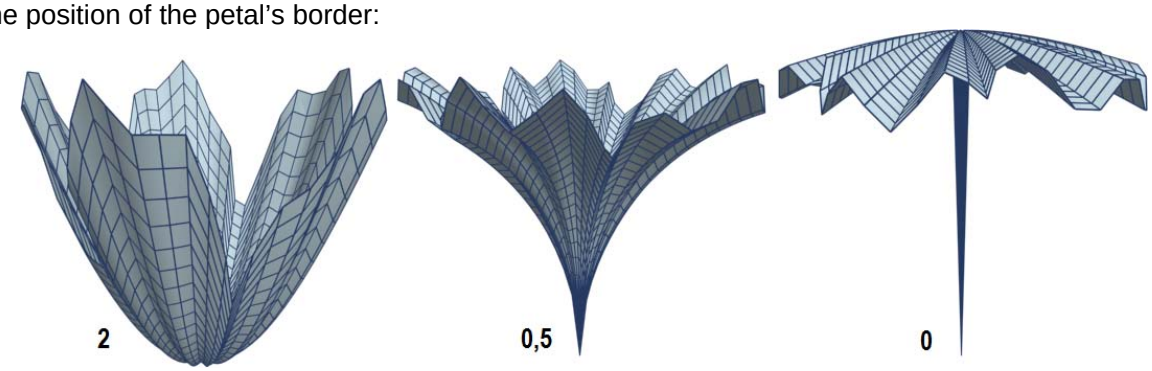
The **Growth** spinner controls the flower's growing from zero to the value set in the **Size** parameter. This growing is exponential, and follows the lateral profile of the flower. This imitates the opening of the real flowers. The values of the **Growth** spinner go from **0** to **1**:



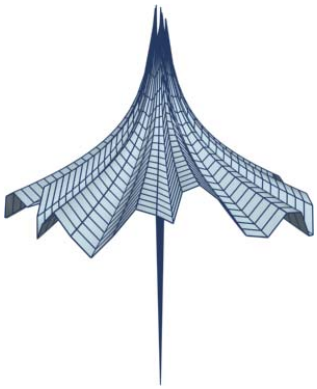
Vertical Shape group

This group controls the lateral profile of the flower. Besides, the script author gives us an equation that links a set of parameters. These parameters have no specific names for identifying their functions:

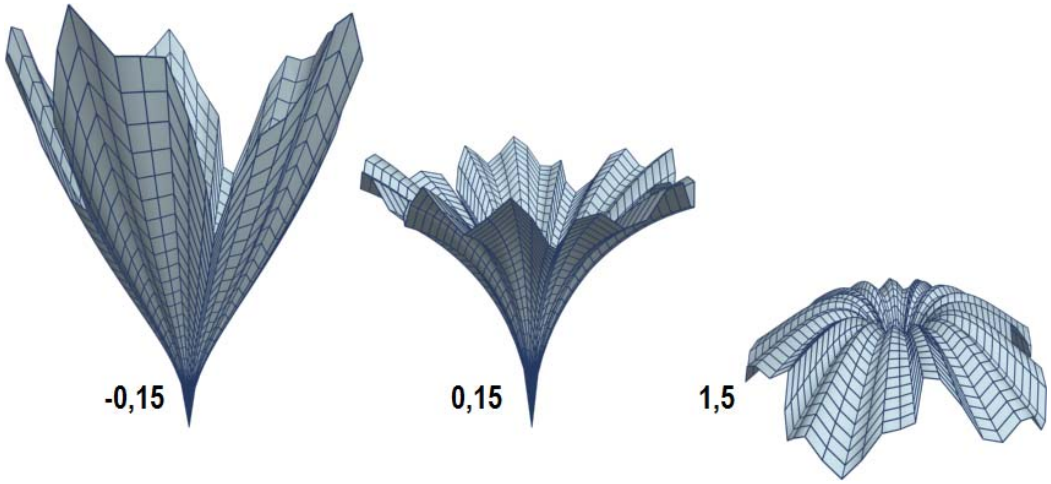
a parameter (default value: **0.5**) pushes the central zone of the flower up or down while retaining the position of the petal's border:



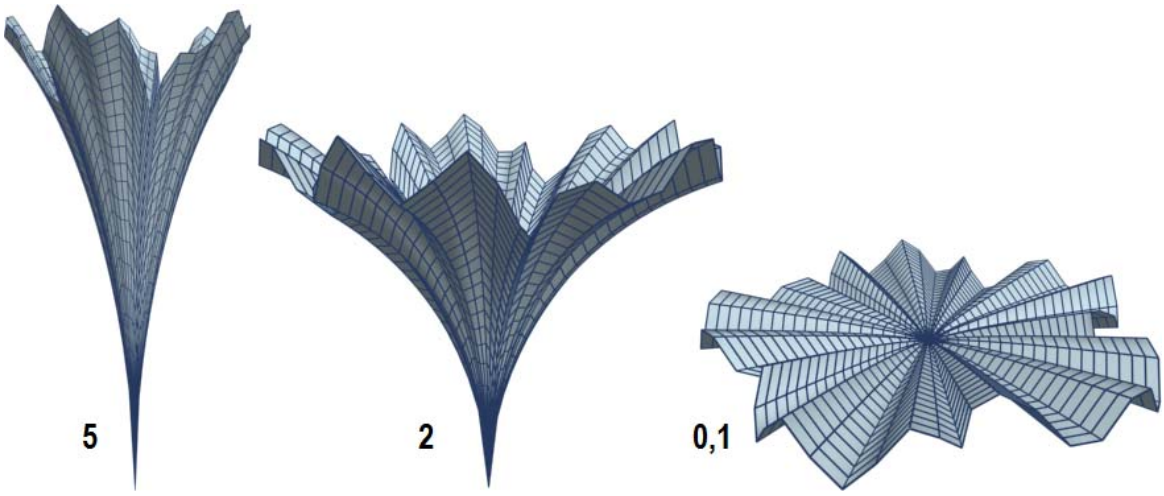
Negative values in this parameter turn over the petals downwards:



b parameter (default value: **0.15**) causes the opposite effect: it raises or descents the petal's borders while keeping the center intact:



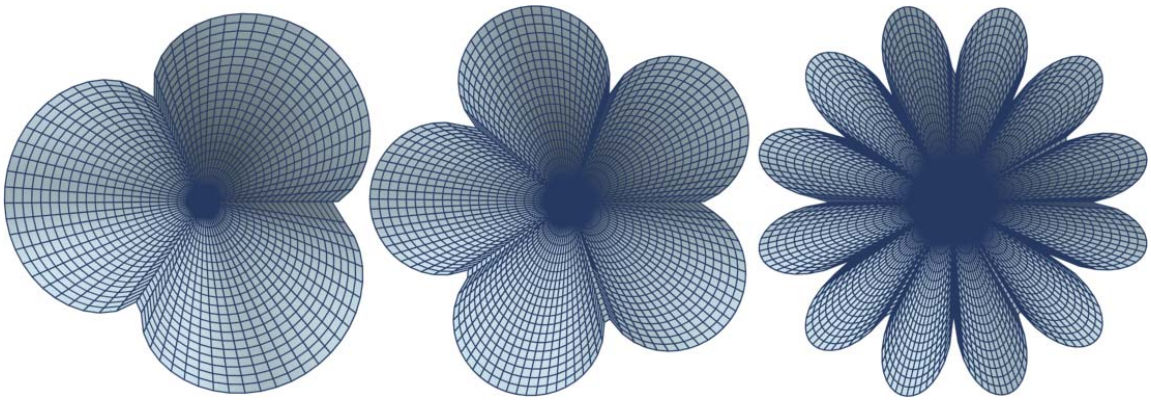
Aa parameter (default value: **2**) establishes the slope of the curve that defines the lateral profile of the flower. Its values go from **0.1** to **5**:



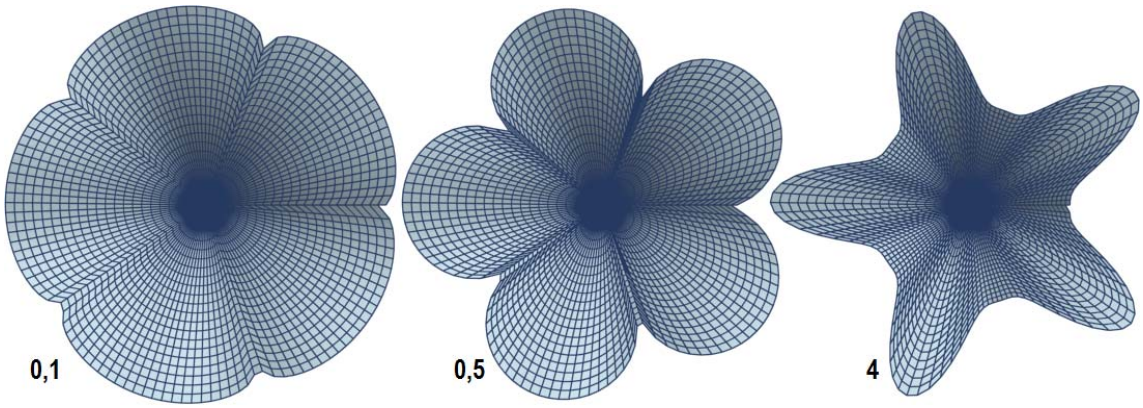
- **Horizontal Shape** group

This group is responsible for the general shape of the corolla —in other words, the shape of the flower when it's seen from the top. Here we have another equation that links a set of parameters. Those parameters say nothing…until you experiment with them.

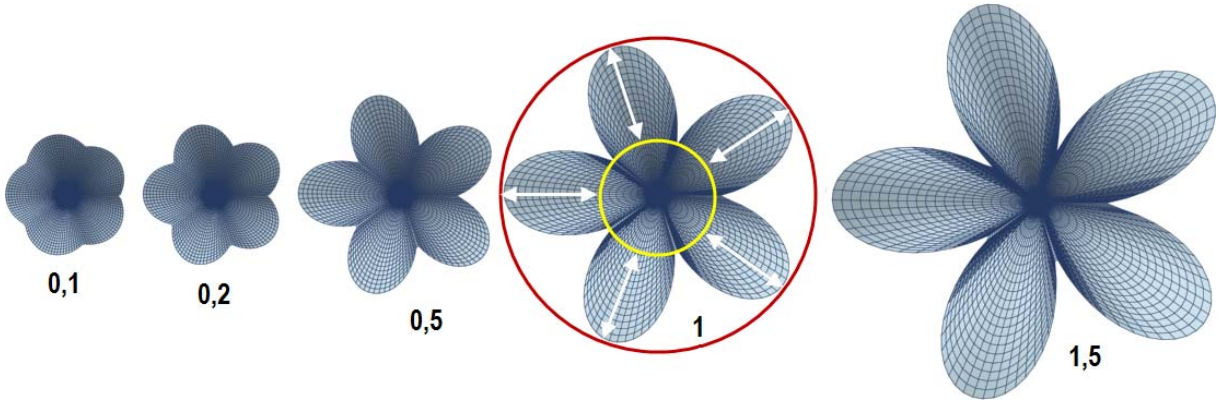
c parameter sets the number of petals. Its values go from **1** to **12**. If you don't want "fractional" petals, you should use integers. Of course, if you want well-rounded petals, you should lower the value of **Detail** while you increase **c**.



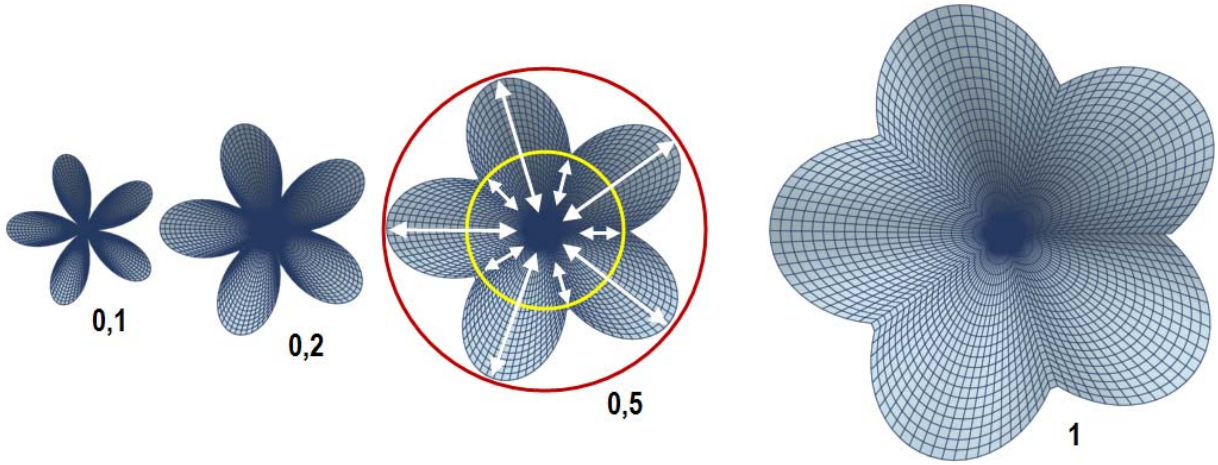
d parameter (default value: **0.5**) sets the width of the petals' tips. Its maximum value is **4**. For **d** = 0 the petals tend to fuse, so they shape a circular corolla:



R1 parameter (default value: **0.5**) sets the distance between the petals' tips and the petals' points of contact. With a value of 0 you'll get an almost circular corolla. Its maximum value is **2**. Of course, this value also affects the overall dimensions of the corolla:



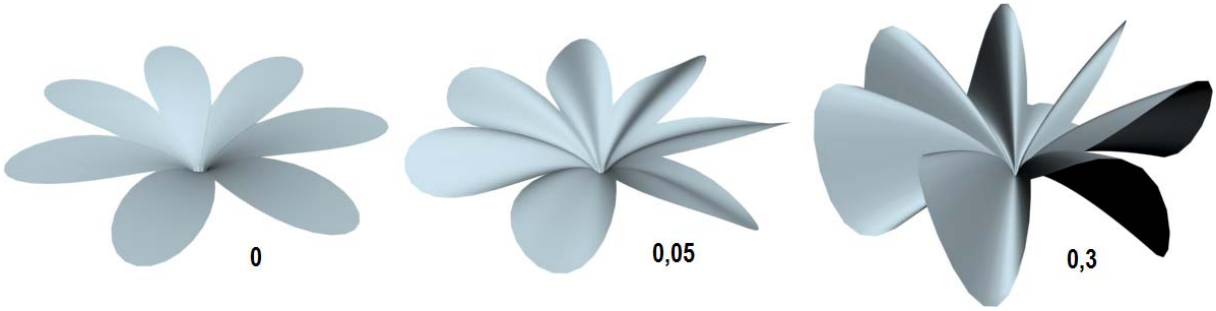
R0 parameter (default value: **0.5**) sets the distance between the petals' tips *and* the petals' points of contact respect to the center of the flower. This is useful for enhancing the "notches" between petals. Its maximum value is **1**, and it also affects the overall dimensions of the corolla:



- **Vertical Perturbation** group

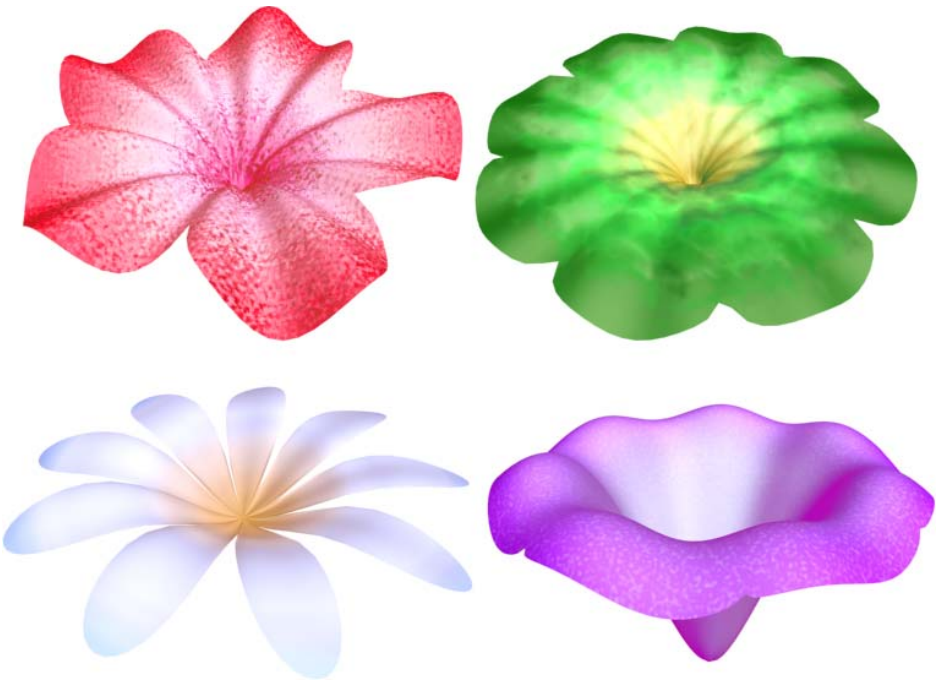
This group determines the petals' bending.

A1 parameter (default value: **0.05**) sets the amplitude of this bending. For a value of 0 the petals become flat. Values near to the maximum (**0.5**) generate exaggerated curvatures:



Lastly, the **p** parameter sets the undulation frequency. In order to get the same curvature for all the petals, if you don't want to collide with math, you might tentatively set **p** equal to **c** (i.e. **p** equal to the number of petals). Otherwise, you'll get a quasi-randomly undulation distributed among the petals. Negative values reverse the curvature.

All that remains is: applying an **UVW Map** modifier with the **Planar** option, giving the flower some texture and… expanding your imagination:



If you want to animate the flowering, the modifier should be applied when the flower is completely developed: go to the frame in which the flower reaches that state, and then click on the **Fit** button of the **UVW Map** modifier. You may also animate the scale of the modifier's gizmo with a **Slow**-type curve.

Ariel Núñez-Morera