Topology Optimization Modeling

Ameba

User Manual

for Version 1.0.0

April 3, 2019

谢亿民设计

XIE Archi-Structure Design

目　　录

About Ameba

Ameba is a type of single-celled organism which is capable of changing its shape in different environments.

Based on the Bi-directional Evolutionary Structural Optimization (BESO) technique originally proposed by Professor Yi-Min (Mike) Xie, his team at XIE Technologies has developed a topology optimization software tool called Ameba. According to their design requirements,the users can apply different loading and boundary conditions to the initial design domain. During the computational process by this software, the design domain can evolve, similar to an ameba, into various shapes, and eventually reach an organic form which is also structurally efficient

The team at XIE Technologies is committed to providing an advanced, then easy-to-use topology optimization tool, than can enhance your creative designs and accelerate your product development.

User agreement

User agreement

Welcome to Ameba

1. Thank you for using the "Ameba" topology optimization plug-in and online computing services.
2. Please read the following terms carefully. If you disagree with any of the terms of the agreement, you may choose not to enter "Ameba". When you register successfully, whether you enter the "Ameba" website or use the "Ameba" computing service, or publish any content on the "Ameba" website, it means that you (ie "users") completely accept all terms under the agreement.

Privacy and Intellectual Property

1. The "Ameba" topology optimization allows the user to submit a model file for optimization calculations. The calculation results are transmitted over the Internet to the user's home location and the service itself is not responsible for storing user-related models and calculation results.
2. The user’s design and calculation of the intellectual property of the optimization result are owned by the user. We will protect the user’s personal information and design data as much as possible to prevent the user data from being leaked or infringed.

About Using the Product

1. After the success of user's registration, the "Ameba" account number and corresponding user name and password and other account information will be generated. After the user completed the application registration process, the user will be entitled to use the "Ameba" account. This right belongs only to the original applicant, and is prohibited from giving, borrowing, renting, transferring or selling. The user should save and use its user name and password carefully and reasonably. Improper user care may result in hacking or password theft. The responsibility lies with the user.
2. In the process of using the "Ameba" service, user may not upload, copy, publish, transmit, or reproduce the following:
   a) Oppose the basic principles set by the Constitution;
   b) Endangering national security, leaking state secrets, subverting state power, and undermining national unity;
   c) Damage to national honors and interests;
   d) Inciting ethnic hatred, ethnic discrimination, and undermining national unity;

e) Undermining the country’ s religious policy and promoting cults and feudal superstitions;

f) spread rumors, disrupt social order, and undermine social stability;

g) Dissemination of obscenity, pornography, gambling, violence, homicide, terror or abetment of crimes;

h) insulting or slandering others and infringing the legal rights and interests of others;

i) Information containing other contents prohibited by laws and administrative regulations.

3. When using "Ameba" to optimize computing services, the user needs to generate a temporary authorization code on the site. The use right of the authorization code's belongs only to the initial registrant and is prohibited from granting, borrowing, renting, transferring or selling.

Disclaimer

1. "Ameba" will try its best to ensure the reliability of calculation results and optimization results through various means. At the same time, please be noted that there are some unavoidable position defects in this software. When the user uses the calculation results for engineering practice, please judge the rationality of the structure and conduct the necessary design review.
2. "Ameba" assumes no responsibility for service interruptions or other defects caused by force majeure or "Ameba" failure to control, but will endeavor to reduce the resulting loss and impact to the user.

About the Agreement

In accordance with the development of the Internet and changes in relevant laws, regulations and regulatory documents, or due to business development needs, "Ameba" will amend or change the terms of service of this agreement when necessary. Users can log in to the “Ameba” website to check the latest version of the relevant agreement terms. The revised service terms will effectively replace the original service terms once they are published. If the user continues to use the services provided by "Ameba", it means that the user has accepted the modified service terms; if the user does not accept the modified service terms, it shall stop using the services provided by "Ameba".

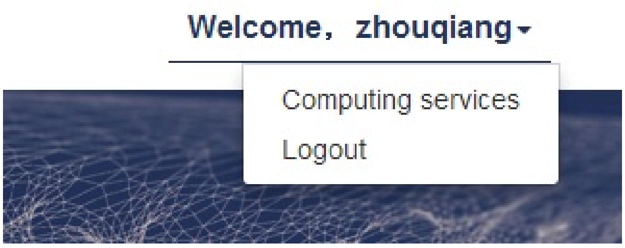
Register

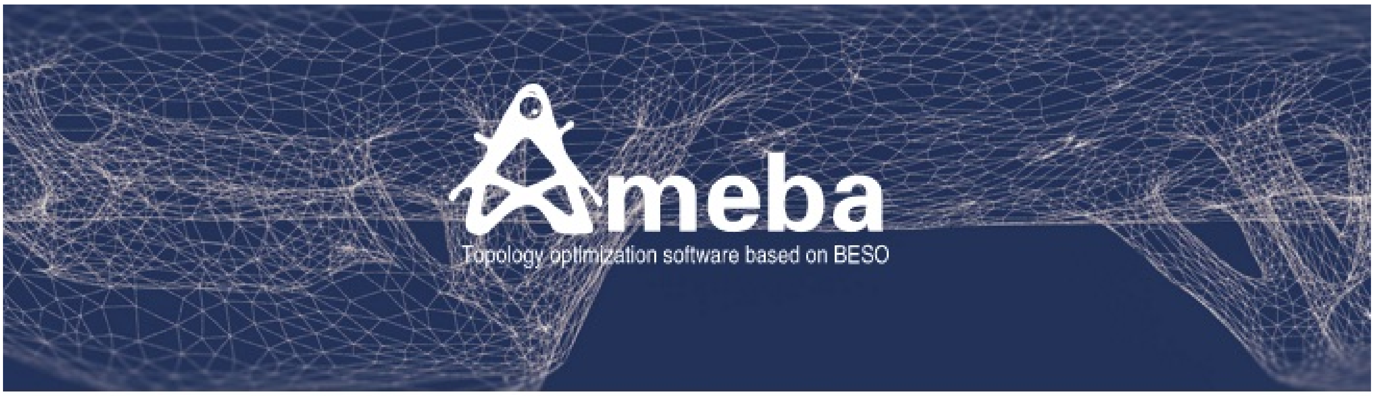
Register and Login

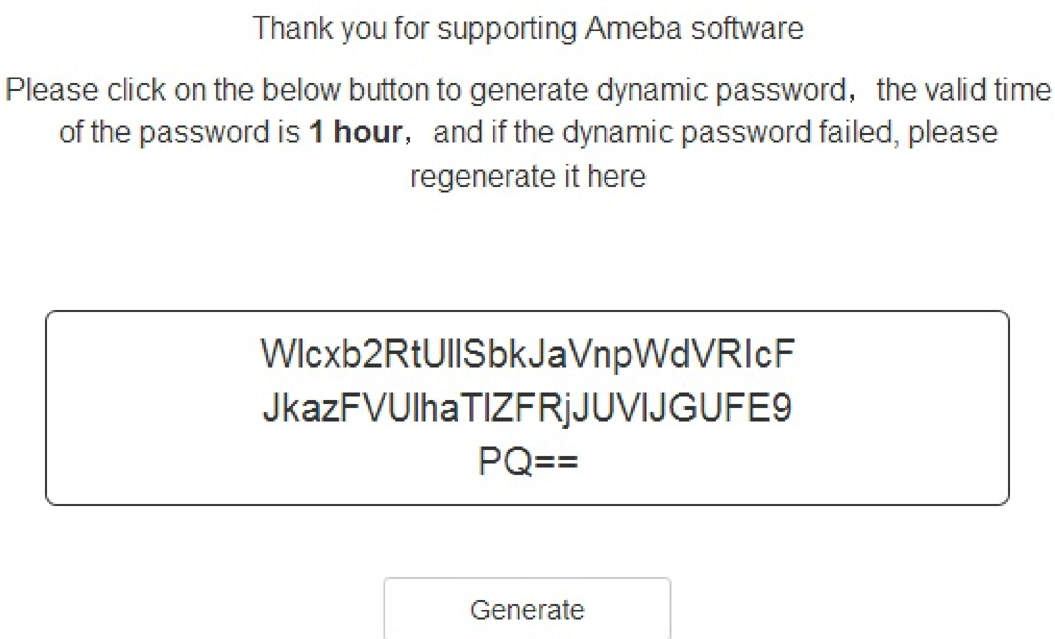
User login http://ameba.xieym.com

Click the top right corner Login | Register , click the top right corner panel after login ,

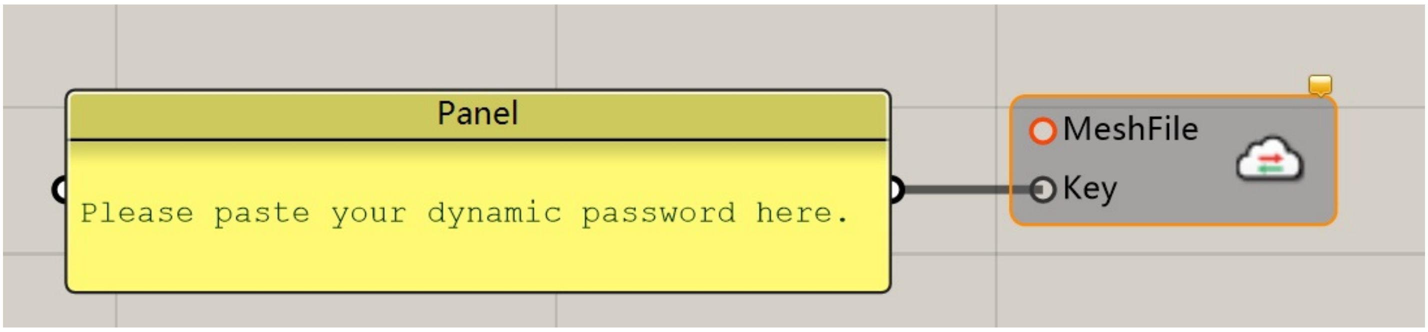
click【computing services】







Switch to dynamic password web page , click Generate button , a dynamic password will be generated in the upper box, which is used to login cloud server 。



Logout

Click the top right panel, logout。

Start a new project

Start a new project

Preparation

Download the Ameba.exe file and run the installation.

The unit system of Ameba is: N, mm, Mpa. Therefore, it is best to keep mm as the unit of Rhino's model.

Requirements of plug-in environment :

.NET Framework 4.5.2

Rhinoceros 6 SR7 (Windows) or later

Install plug-in

Ameba plug-in Download

Software operation interface

Start a 2D optimization project

Start a 2D optimization project (Rhino)

Calculation instructions :

2D calculation refers to the calculation of a plane object to approximate the simulated 3D environment. The advantage is that the calculation speed is faster.

Attention :

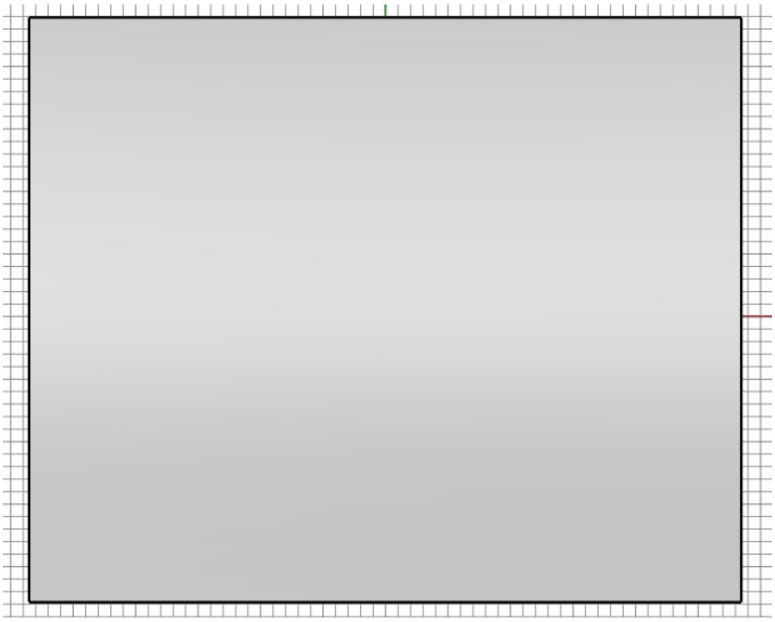
All 2D objects must be closed 2D objects (a flat surface or a closed curve)

Example :

2D calculations can only be performed in Rhino's Top view.

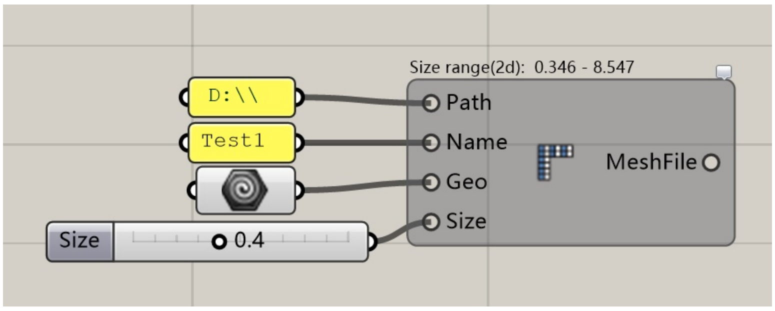
1. Define a design domain

A plane represents a closed design domain.

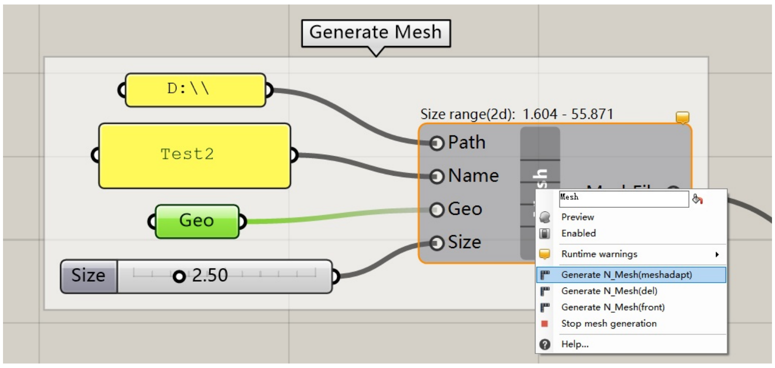


2. Generate mesh

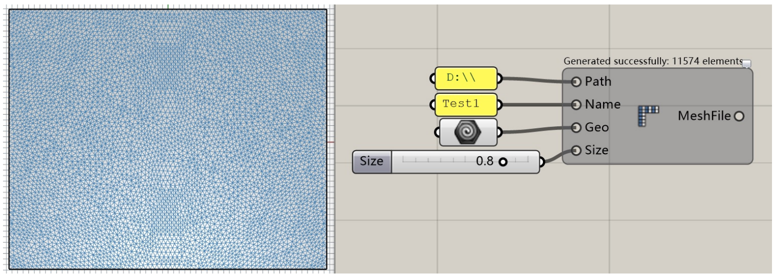
Use the AmebaMesh component to pick up the model, then right click on the component button Generate N_Mesh.



Wait for a while, if the component is still yellow or red, you can click the small tab in the upper right corner of the component to view the error information. Usually the mesh becomes white when the step completes (Attention: Size must be set within the Size range). It is noticed that there are three alternative method to generate meshes, and users don't need to distinguish the features of them (try the other two if one doesn't work).



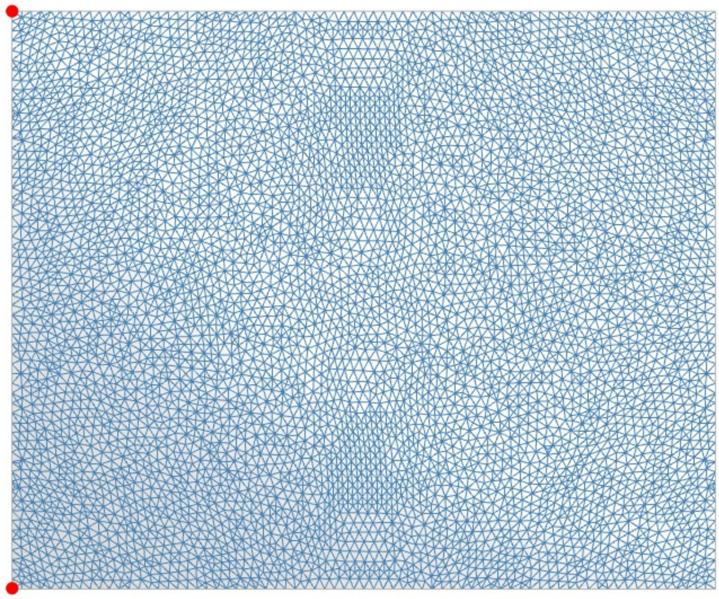
After the success of dividing the mesh, we will see that the model is covered with blue mesh edges, and the top of the component shows how many mesh cells have been generated.



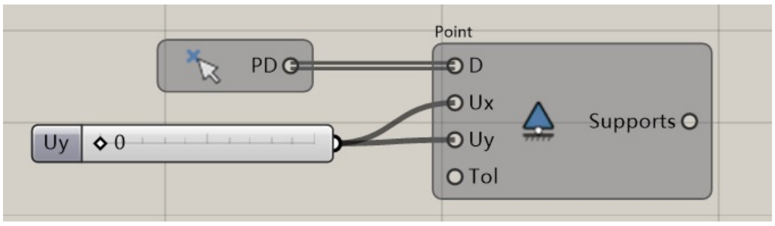
If you encounter problems in this step, please click Error Report and Solution

3. Define the support

The support defines the red point below.

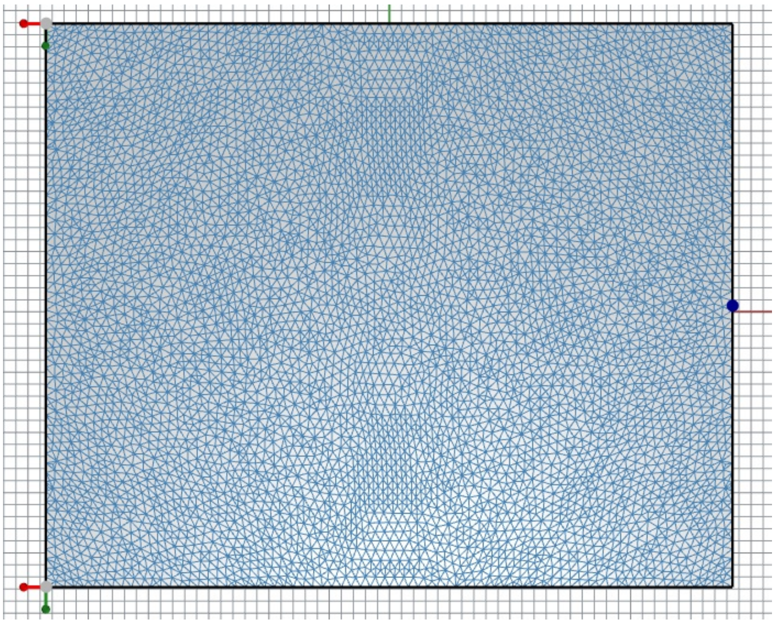


The following figure defines the fixed support in the XY direction.

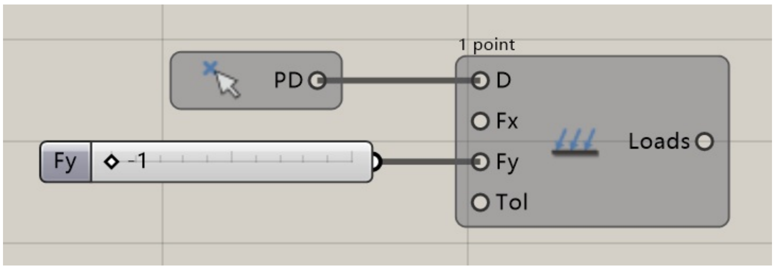


4. Define the load

Define the load at the blue point below.

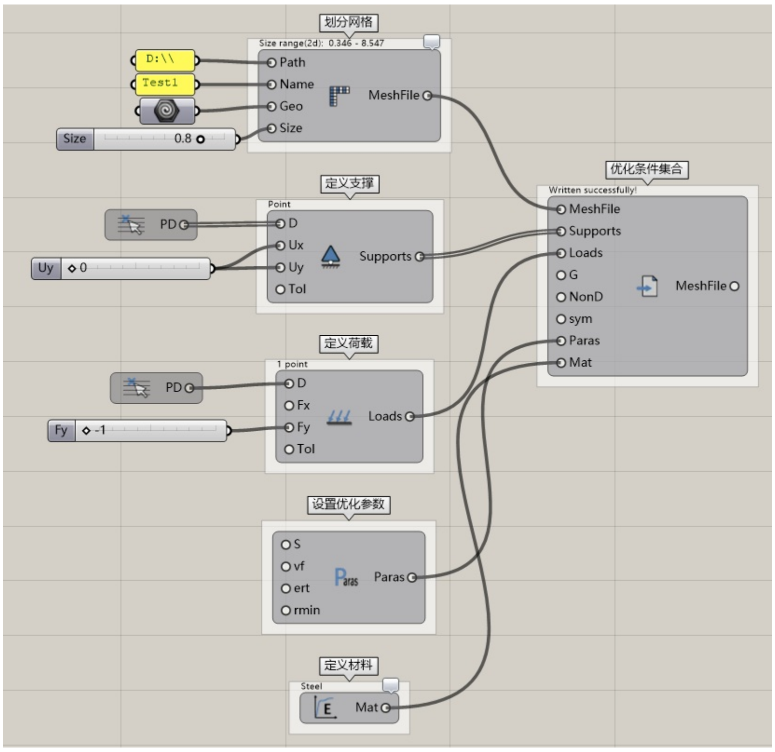


Define the point load in the Y-axis direction



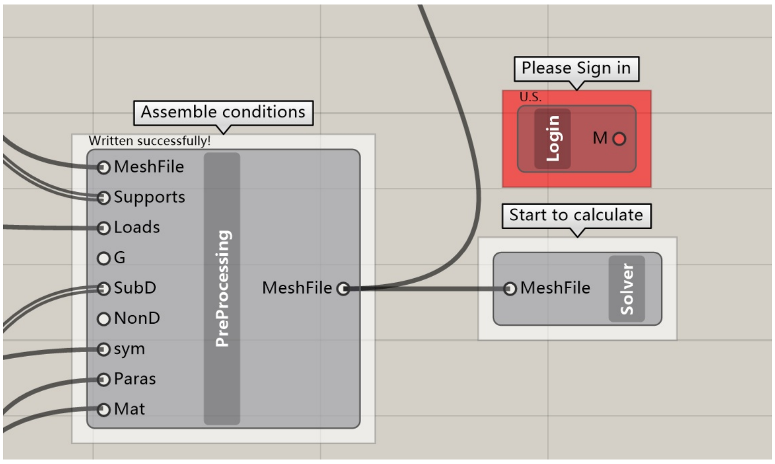
5. Define other parameters

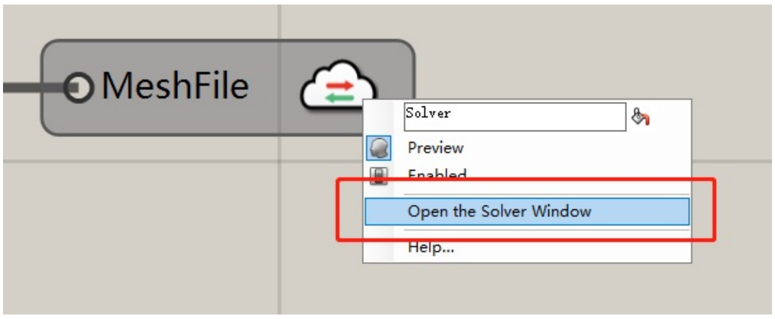
This example all stays default.



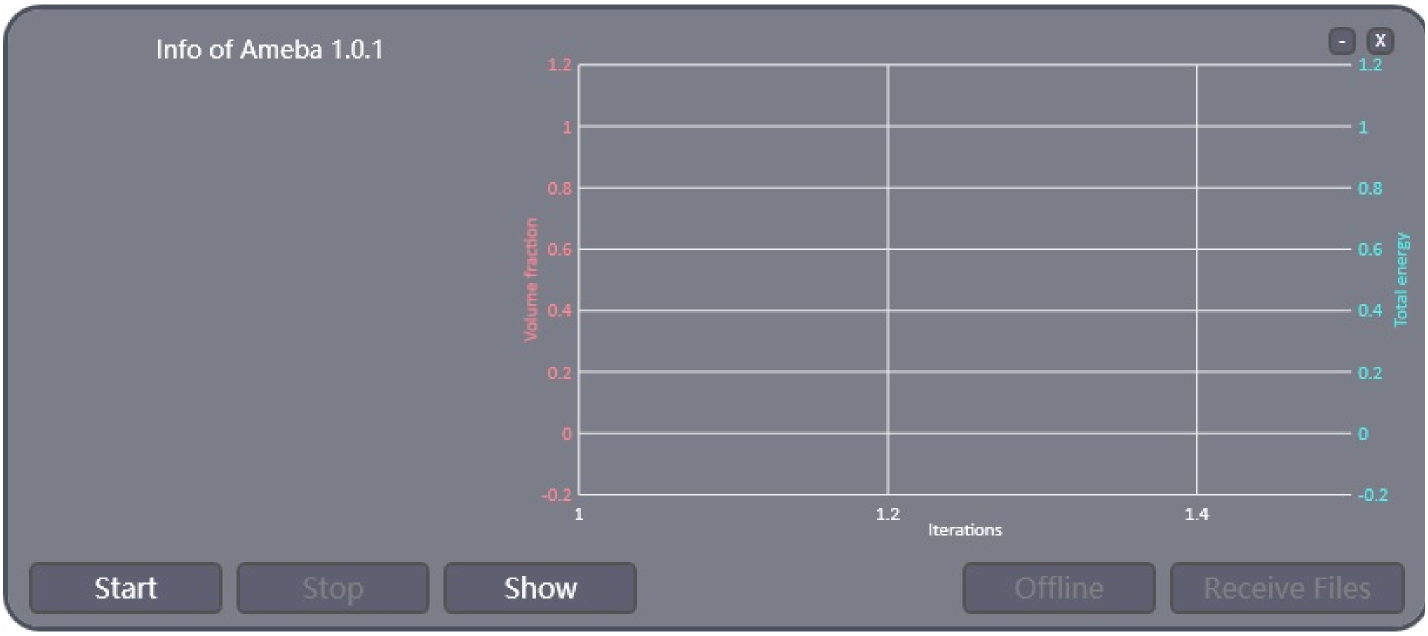
6. Computing services

First of all, you are supposed to insure PreProcessing component has revealed "Written successfully!" . Then users need to verify the license status by using Login component.





Right-click Solve component and select "Open the Solve Window" to open the Solver Window (or double-click this component).



Notes: The key has just an hour of validity. But it can't be shut downwhen your project is under calculation. You need to regenerate it from Ameba's website.

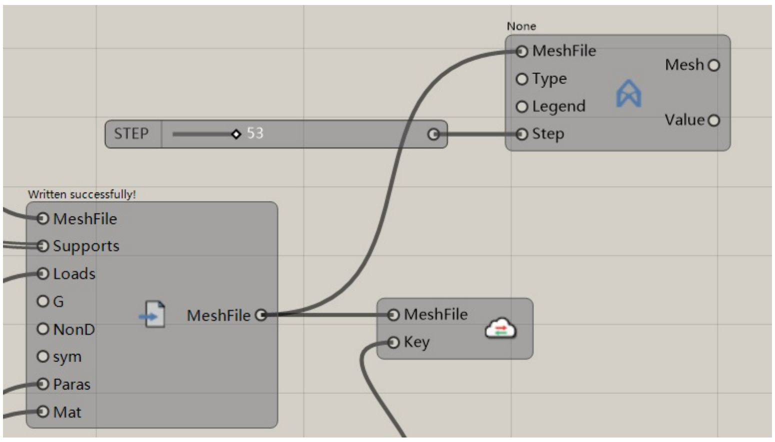
Click Start button and click Show button if you wanna review the results in real-time (you must connect Display component before computing). And then, please wait a moment until the Solver Information Panel displays "The calculation has already started. Please wait patiently." Next, you can have a cup of coffee and wait about ten minutes(it depends on your mesh model, load case and internet speed). After completion, you can receive the result model using Display component.

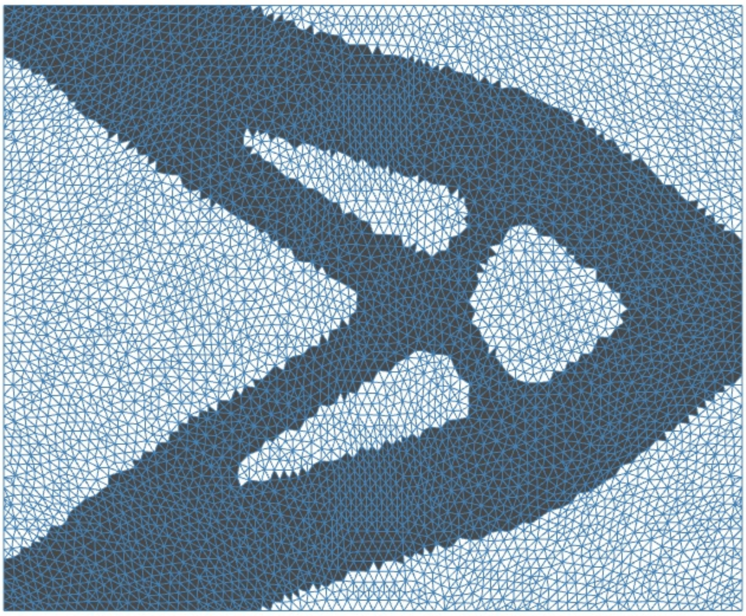
You could click the Solver Window and get more details.

If you encounter problems in this step, please click Error Report and Solution

7. Visualize calculation results

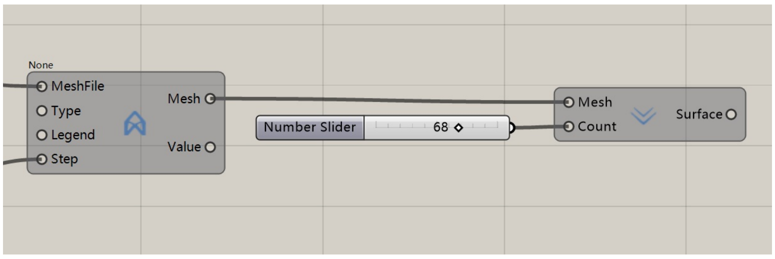
Access to the Display and Step component. In the above, the Last Result Button of the calculation window can directly adjust the Step to the number of steps in the latest calculation. After the calculation is completed, you can click the Last Result Button to view the final calculation result.

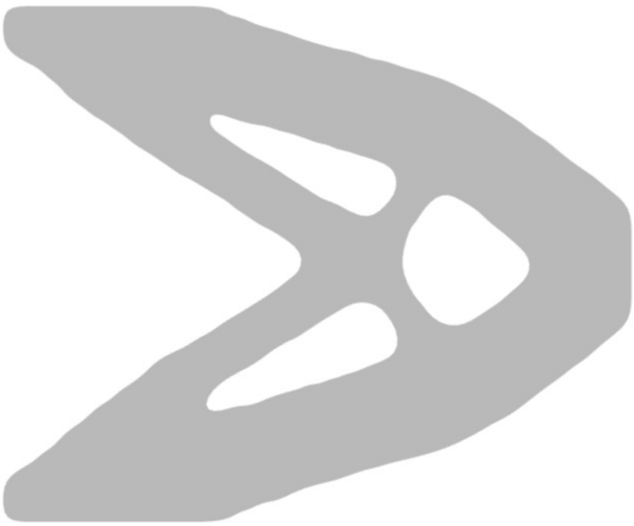




8. Post-processing of two-dimensional topology optimization

Post-process the resulting mesh model with the Rebuilding2D component to achieve higher model quality.





Start a 3D optimization project

Start a 3D optimization project

Calculation instructions :

Three-dimensional calculation refers to carry out optimization calculation of a closed three-dimensional object, can be any closed three-dimensional entity.

Matters need attention :

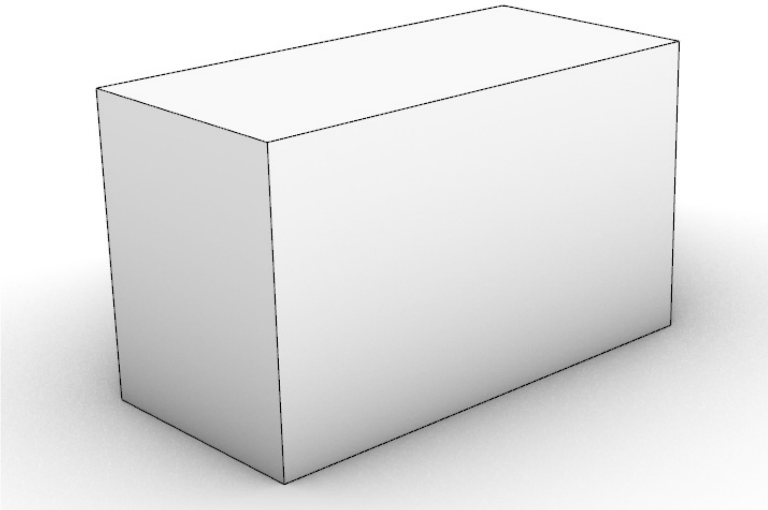
All 3D objects must be closed 3D geometry.

Example :

All 3D calculations are done in the perspective view of Rhino。

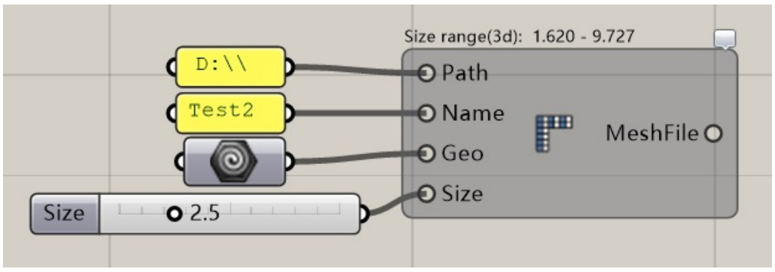
1. Define a design domain

A box represents a closed design domain.

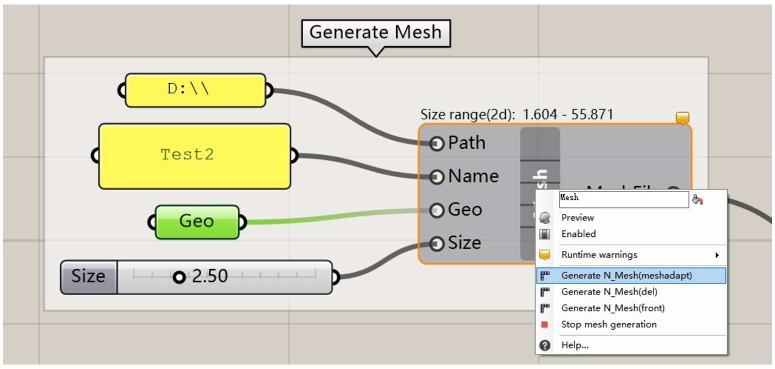


2. Generate mesh

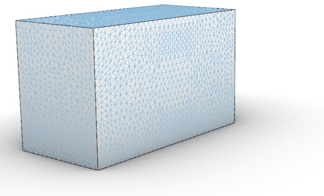
Use the AmebaMesh component to pick up the model, then right click on the component button Generate N_Mesh.

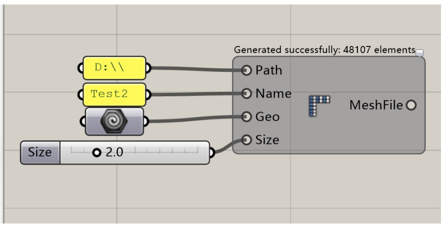


Wait for a while, if the component is still yellow or red, you can click the small tab in the upper right corner of the component to view the error information. Usually the mesh becomes white when the step completes (Attention: Size must be set within the Size range).It is noticed that there are three alternative method to generate meshes, and users don't need to distinguish the features of them (try the other two if one doesn't work).



After the success of dividing the mesh, we will see that the model is covered with blue mesh edges, and the top of the component shows how many mesh cells have been generated.



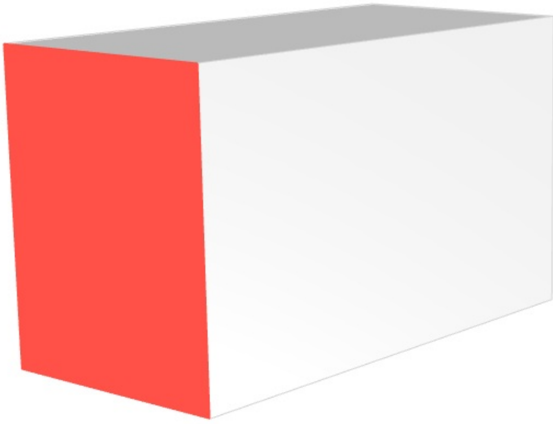


If you encounter problems in this step, please click Error Report and Solution

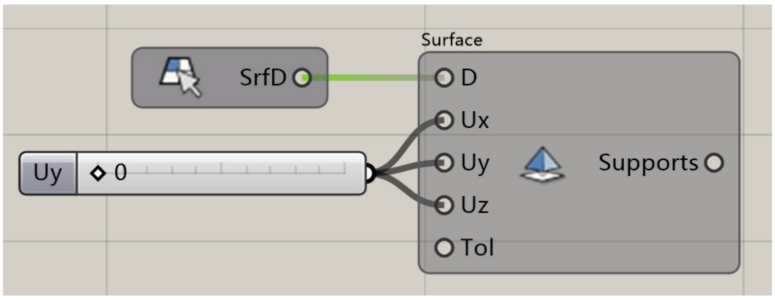
---

3. Define the support

The supports are defined at the red surface of the figure below.



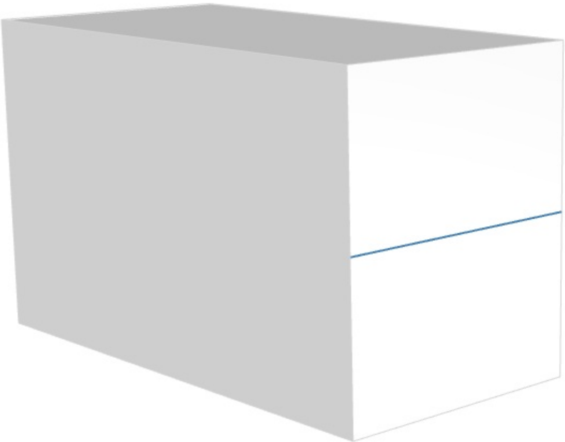
The following figure defines the fixed support in the XYZ direction.



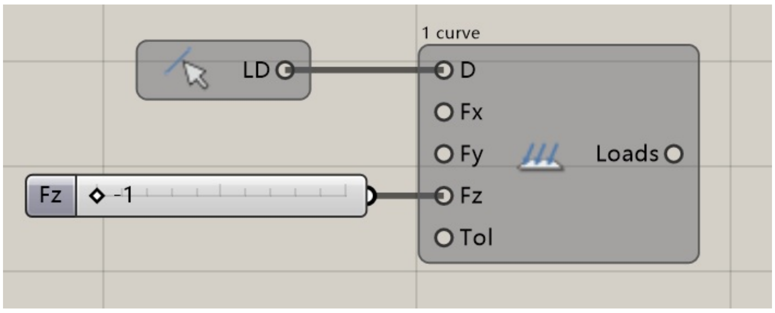
---

4. Define the load

Define the load at the blue point below.

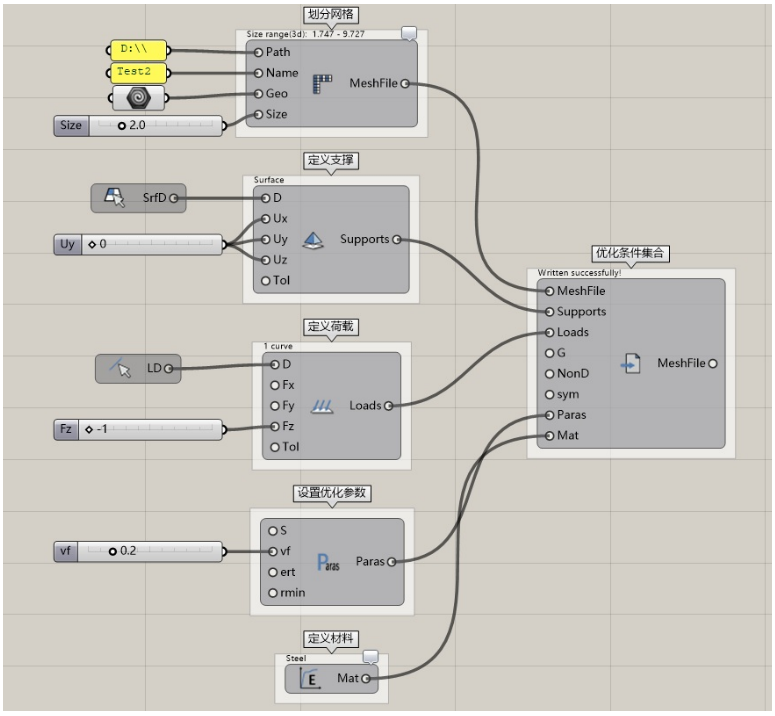


Define the point load in the Z-axis direction



5. Define other parameters

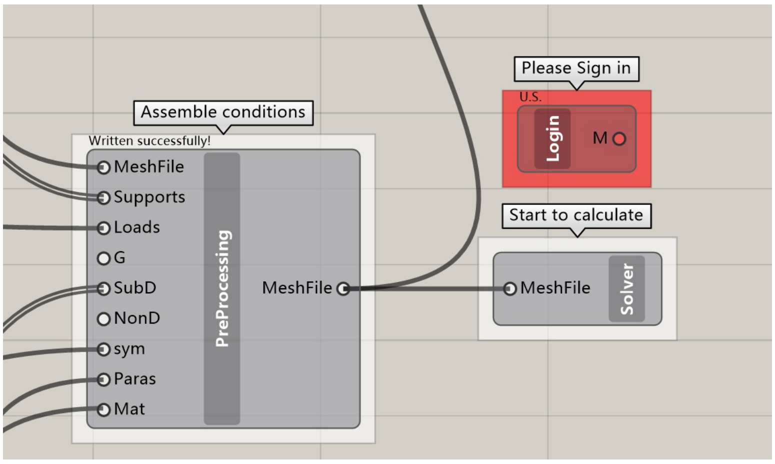
The volume fraction of this example is 20%, and all other parameters remain the default.

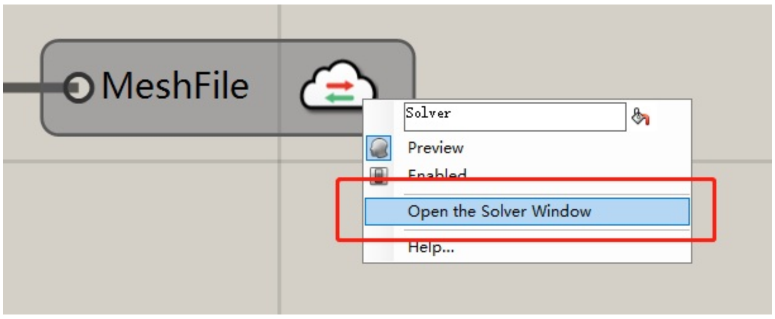


6. Computing services

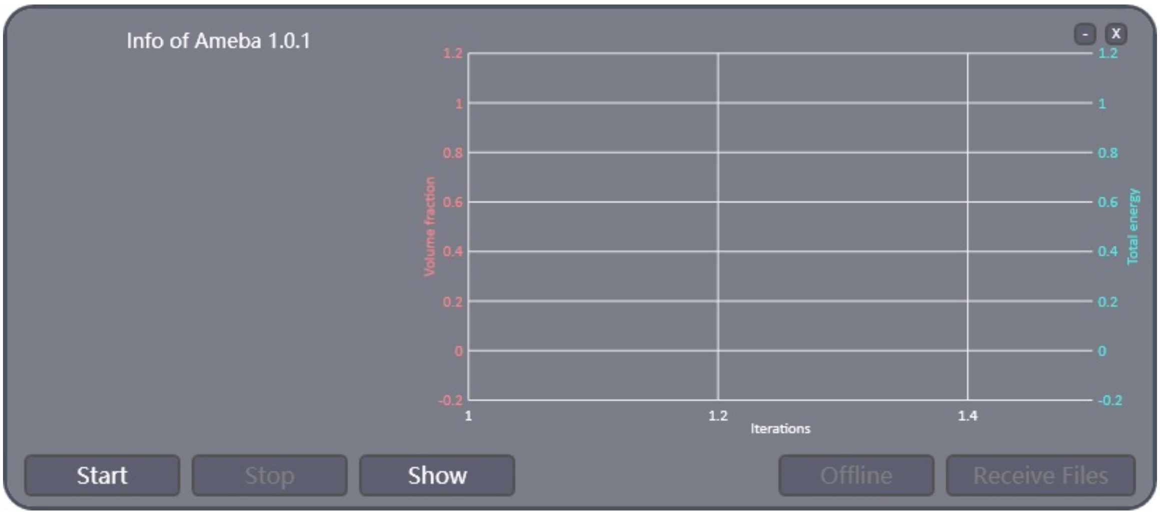
First of all, you are supposed to insure PreProcessing component has revealed "Written successfully!" .

Then users need to verify the license status by using Login component.





Right-click Solve component and select "Open the Solve Window" to open the Solver Window (or double-click this component).



Notes: The key has just an hour of validity. But it can't be shut downwhen your project is under calculation. You need to regenerate it from Ameba's website.

Click Start button and click Show button if you wanna review the results in real-time (you must connect Display component before computing). And then, please wait a moment until the Solver Information Panel displays "The calculation has already started. Please wait patiently." Next, you can have a cup of coffee and wait about ten minutes(it depends on your mesh model, load case and internet speed). After completion, you can receive the result model using Display component.

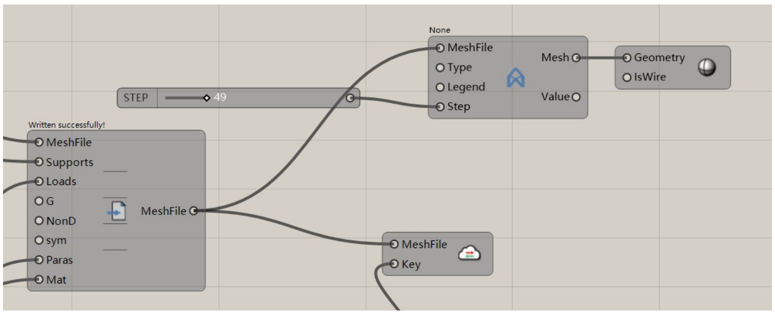
You could click the Solver Window and get more details.

If you encounter problems in this step, please click Error Report and Solution

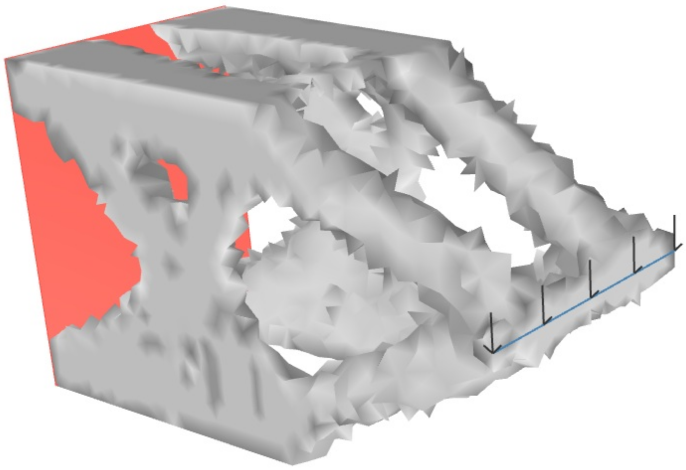
---

7. Visualize calculation results

Access to the Display and Step component. In the above, the Last Result Button of the calculation window can directly adjust the Step to the number of steps in the latest calculation. After the calculation is completed, you can click the Last Result Button to view the final calculation result.



As shown in the figure below, the result model of 3D topology optimization is obtained.

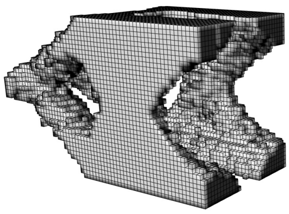


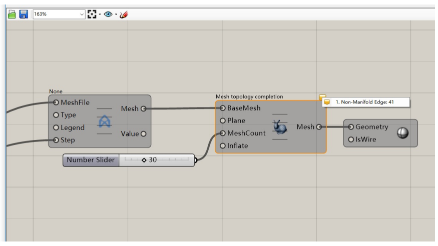
8. Post-processing of three-dimensional topology optimization

Since the calculated mesh model is rough, it is difficult for designers to continue editing. We can use Ameba's MeshTools to post-process the model.

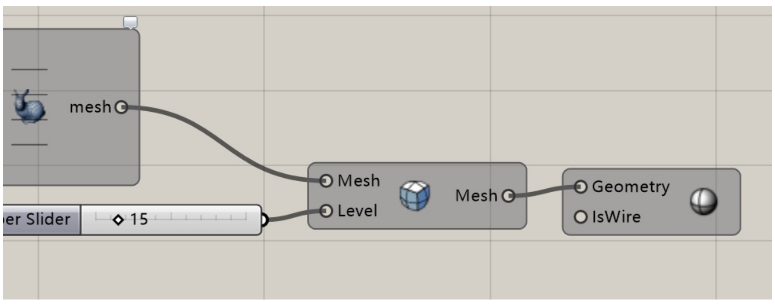
Use Remesh and QuadMesh or TriMesh to perform mesh reconstruction on a three-dimensional topology-optimized model.

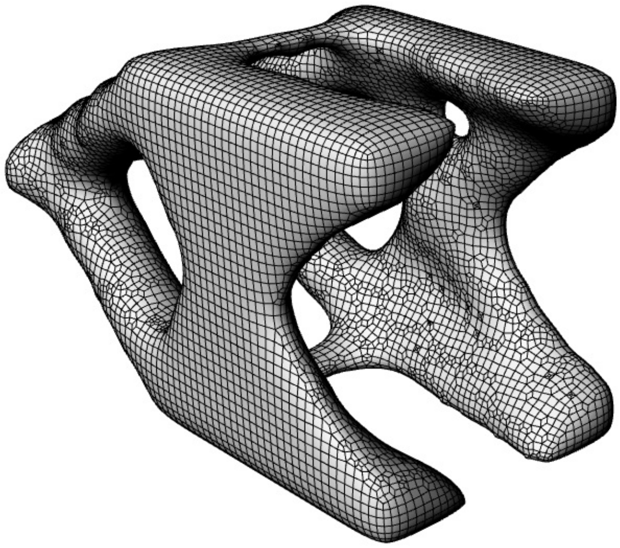
After the calculation is completed, a structured mesh model stacked by Mesh Box will be obtained. Due to the limitation of the Free version, this model may generate unsmooth edges, and the MeshCount parameter can only be input within 30.





Finally, we access the AmebaSmooth component to smooth the model to get a smooth and re-editable mesh model.





Function Introduction

Function Introduction

Domain

A group of components for getting Rhinoceros objects.

Ameba2D

A group of components used to set two-dimensional topology optimization conditions. Users can use these components to set loads, non-design areas, and supports.

Ameba3D

A group of components used to set three-dimensional topology optimization conditions. Users can use these components to set loads, non-design areas, and supports.

Mesh

Convert the user's model into a mesh model that can be used for calculations.

Preprocess

For the component group of calculating the result pre-processing, the user can use the module under the group to set the symmetry constraint, material, optimization parameters, sensitivity options, multiple working conditions, and also collect the pre-processing parameters to connect the cloud server.

Postprocess

For the component group used for post-processing of calculation results, the user can use the module under the group to query and display the calculation result, and also reconstruct the mesh model.

Cloud

A component group used to interact with the cloud server. Users can use the components under the group to perform cloud computing or log in to the server.

MeshTools

Optimized mesh model.

Cloud

Cloud

A component group used to interact with the cloud server. Users can use the components under the group to perform cloud computing or log in to the server.

Solve

Deliver cloud computing and monitor cloud computing results.

Login

Log in to the cloud servers.

Solver

Solver

Description：

Start cloud computing.

Usage：

The default server is "Shanghai", if it is an overseas user, you can switch to "Virginia".

Connect all the inputs, then right click on the component, select Display calculation window, open the server calculation window, press the window button Start to start the calculation, press Stop to stop the operation.

Hide：

Hide the window.

Last Result：

Automatically jump to the AmebaStep component to the latest number of files returned.

Attention：

The new version is that the calculation feedback is calculated and returned in real time, so when the calculation converges to its own needs, you can press Stop to end it, instead of waiting for all calculations to complete like the old version.

Input：

MeshFile: Transfer the data from the Mesh component to this component .

Key: dynamic password.

Login

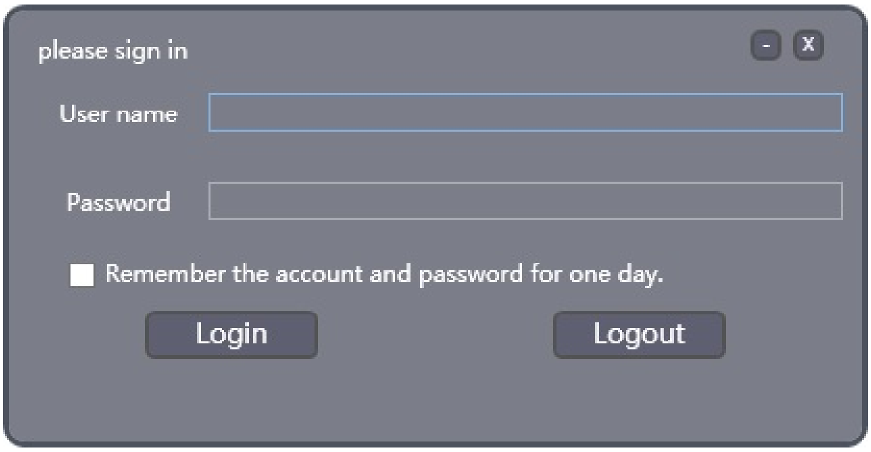
Login

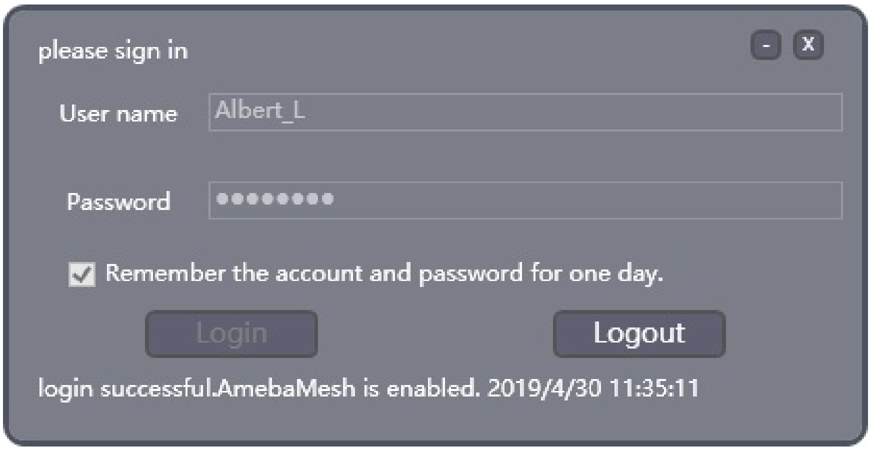
Description：

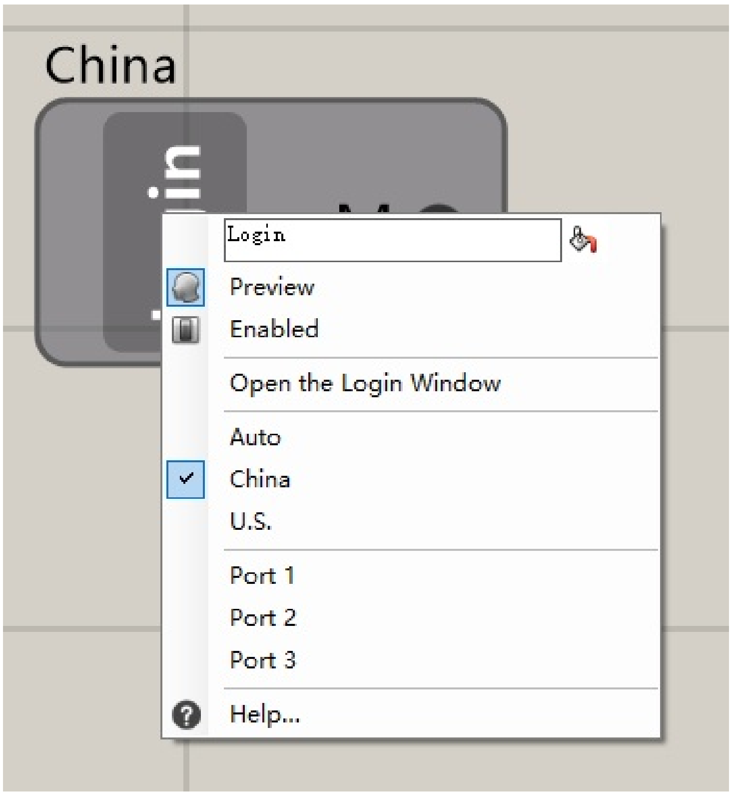
Log in to the server.

Usage：

Double-click the component to enter the Ameba login screen, and then register to log in (This is what needs to be done before cloud computing).







Output：

M: Some login information about ameba account.

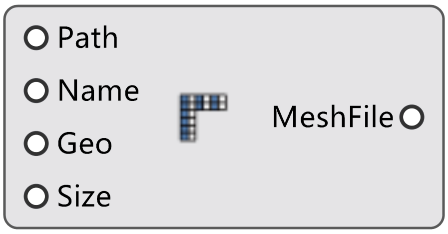
Note:

If your rhino get a crash after click Login button, you can try to change the port. Just right-click the component and select other ports. If you still have any problems, please let us know through leave message on https://ameba.xieym.com/Manage/LeaveMessage

Mesh

Mesh

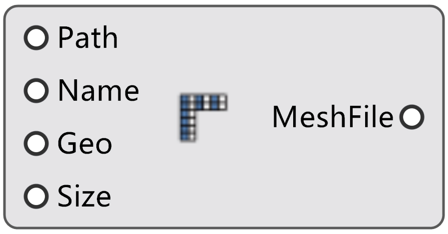
Convert the user's model into a mesh model that can be used for calculations.



Mesh

The objects to be analyzed are meshed for cloud operations.

AmebaMesh



AmebaMesh

Description：

(If it is a two-dimensional calculation, the plane must be XY plane, the Z value of the coordinates of all the vertices of the plane must be 0)

Usage：

Set all the inputs, then right click on the battery and select Generate N_Mesh to mesh.There are three alternative method to generate meshes, and users don't need to distinguish the features of them (try the other two if one doesn't work).

Input：

Path: Define a directory to store project files.

Name: Define a project name.

Geo: Define an analysis area.

Size: Divide the size of the grid.

Attention：

After settingPath, Name, Geo, Size range will appear, and Size can only be set to within the Size range below the component.

Output：

MeshFile: Connect Display, PreProcessing and Solver for data transmission.

Error Report and Solution：

click me and jump to Error Report and Solution

Mesh2D

Mesh2D

Description：

The plane must be XY plane, the Z value of the coordinates of all the vertices of the plane must be 0.

Usage：

The default value of the path is "My Documents", and the name and surface are set at the same time. The Size range will appear in the lower part of the component. This range is recommended (not required to be within this range), because the Size may be too small. Rhino.exe crashes.

Input：

Path: Define a directory to store project files.

Name: Define a project name.

Surface: Define an analysis area.

Size: Divide the size of the grid.

Attention：

After settingPath, Name, Surface, Size range will appear, and Size can only be set to within the Size range below the component.

Output：

MeshFile: Connect Display, PreProcessing and Solver for data transmission.

Error Report and Solution：

click me and jump to Error Report and Solution

Mesh3D

Mesh3D

Description：

Tetrahedral meshing of closed geometry.

Usage：

The default value of the path is "My Documents", and the name and surface are set at the same time.

The Size range will appear in the lower part of the component. This range is recommended (not required to be within this range), because the Size may be too small. Rhino.exe crashes.

Input：

Path: Define a directory to store project files.

Name: Define a project name.

Brep: Define an analysis area.

Size: Divide the size of the grid.

Attention：

After settingPath, Name, Brep, Size range will appear, and Size can only be set to within the Size range below the component.

Output：

MeshFile: Connect Display, PreProcessing and Solver for data transmission.

Error Report and Solution：

click me and jump to Error Report and Solution

ReadMesh

CustomTriangleMesh

CustomQuadMesh

Mesh3DFromMesh

Mesh3DFromMesh

Description：

Tetrahedral meshing of closed mesh.

Usage：

The default value of the path is "My Documents", and the name and surface are set at the same time.

The Size range will appear in the lower part of the component. This range is recommended (not required to be within this range), because the Size may be too small. Rhino.exe crashes.

Input：

Path: Define a directory to store project files.

Name: Define a project name.

Brep: Define an analysis area.

Angle:

Size: Divide the size of the grid.

Attention：

After settingPath, Name, Mesh, Size range will appear, and Size can only be set to within the Size range below the component.

Output：

MeshFile: Connect Display, PreProcessing and Solver for data transmission.

Error Report and Solution：

click me and jump to Error Report and Solution

Domain

GeometryDomain

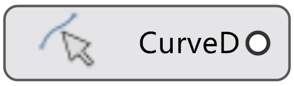
A group of components for getting Rhinoceros objects.

PointsDomain

Get the selected point object.

LinesDomain

Get the selected line.



CurvesDomain

Get the selected curve.

SurfacesDomain

Get the selected surface.

PointsDomain

PointsDomain

Description:

Get the selected point object.

Usage:

Right-click on the battery, then select 【Set points】, and draw points directly in Rhino.

Output:

PD: point objects.

LinesDomain

LinesDomain

Description：

Get the selected line.

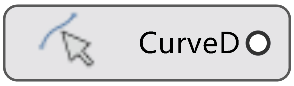
Usage：

Right-click the battery, then select 【Set lines】, and draw lines directly in Rhino.

Output:

LD:lines domain.

CurvesDomain



CurvesDomain

Description:

Get the selected curve.

Usage:

Right-button battery, then select 【Select lines】, and pick up straight objects directly in Rhino.

Output:

CurveD: curves domain.

SurfacesDomain

SurfacesDomain

Description:

Get the selected surface.

Usage:

Right-click the battery, then select 【Select surface】, and pick up the plane surface directly in Rhino.

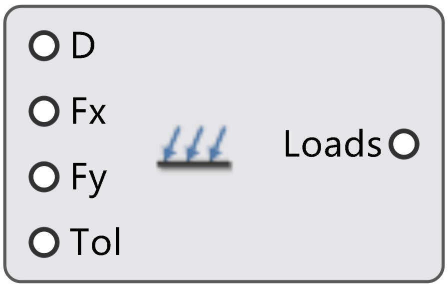
Output:

SrfD: surfaces domain.

Ameba2D

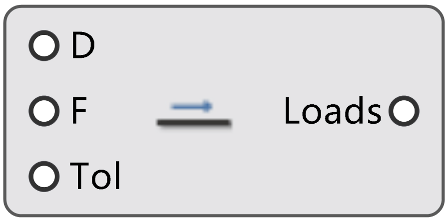
Ameba2D

A group of components used to set two-dimensional topology optimization conditions. Users can use these components to set loads, non-design areas, and supports.



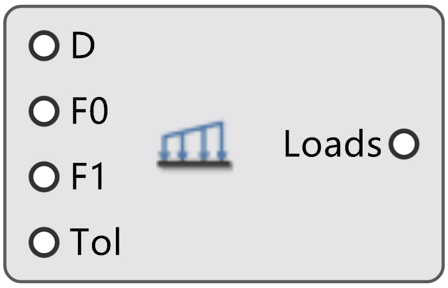
Load2D

Apply a two-dimensional load along a certain direction.



TangentLoad2D

Define the two-dimensional shear load on the surface of the object.

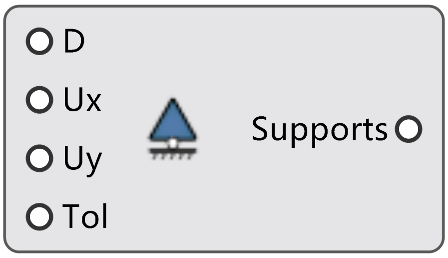


[NormalLoad2D =230x]](NormalLoad2D.md)(NormalLoad2D.md)

Apply a two-dimensional load along the normal to the surface of the object.

NonDesignDomain2D

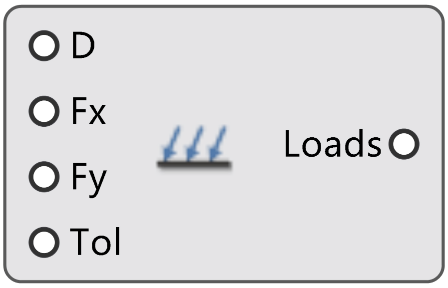
Define a two-dimensional non-design area.



Support2D

Define two-dimensional support.

Load2D



Load2D

Description：

In the two-dimensional analysis, a general load is defined and the load can only be applied to the surface of the object.

(This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display the cell to which the load is applied and the cell center point to represent the cell.

Input：

D: Enter the area of a load. You can input points, lines or surfaces.

Fx: Enter the load in the X-axis direction.

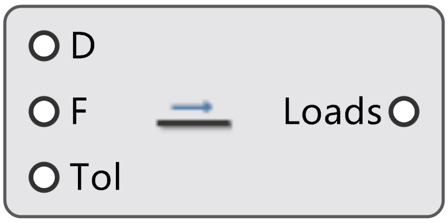
Fy: Enter the load in the Y-axis direction.

Tol: The tolerance for area determination. This parameter will be invalid when you input surfaces.

Output：

Loads: Output a load.

TangentLoad2D



TangentLoad2D

Description：

In the two-dimensional analysis, a shear load is defined and the load can only be applied to the surface of the object.

(This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display the cell to which the load is applied and the cell center point to represent the cell.

Input：

D: Enter the area of a load. You can input circles or line.

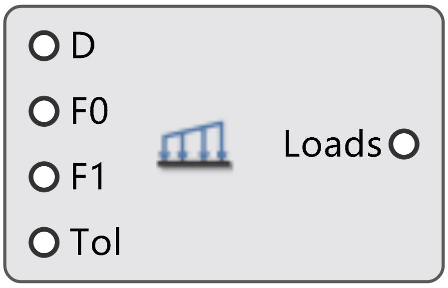
F: The size of the line load.

Tol: The tolerance for area determination. This parameter will be invalid when you input surfaces.

Output：

Loads: Output a load.

NormalLoad2D



NormalLoad2D

Description：

Define a normal uniform load along a curve in 2D problem.

(This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display the cell to which the load is applied and the cell center point to represent the cell.

Input：

D: Enter the area of a load. You can input circles or lines.

F0: Load size of left end.

F1: Load size of right end.

Tol: The tolerance for area determination. This parameter will be invalid when you input surfaces.

Output：

Loads: Output a load.

NondesignDomain2D

NondesignDomain2D

Description：

Design a non-desgin domain in 2D problem.

(This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display cells that are not designed, and use cell center points to represent cells.

Input：

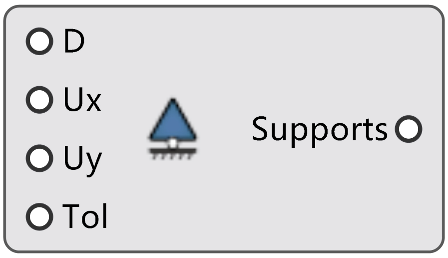
D: Reference line of non-desgin domain. You can input lines or surfaces.

Tol: The tolerance for area determination. This parameter will be invalid when you input surfaces.

Output：

NonD: Non-desgin domain.

Support2D



Support2D

Description：

Define a support in 2D problem.

(This component must be used after generating the mesh.)

Input：

D: Enter a support domain. You can input points, lines or surfaces.

Ux: The displacement in the X-axis direction. If you don't input any vaule, there will be no constrains in this direction. If you input 0, there will be constrained in this direction. If you input a non-zero value, this value will mean displacement in this direction.

Uy: The displacement in the Y-axis direction. If you don't input any vaule, there will be no constrains in this direction. If you input 0, there will be constrained in this direction. If you input a non-zero value, this value will mean displacement in this direction.

Tol: The tolerance for area determination. This parameter will be invalid when you input surfaces.

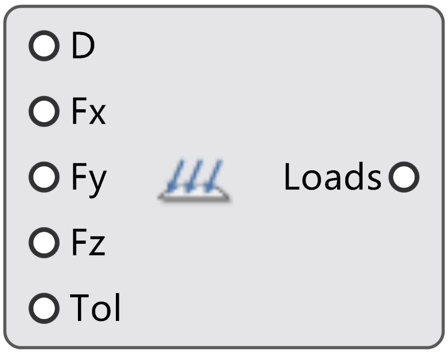
Output：

Supports: Output a support.

Ameba3D

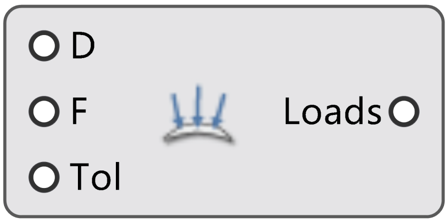
Ameba3D

A group of components used to set three-dimensional topology optimization conditions. Users can use these components to set loads, non-design areas, and supports.



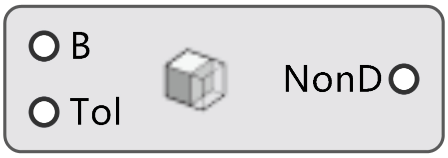
Load3D

Apply a three-dimensional load along a certain direction.



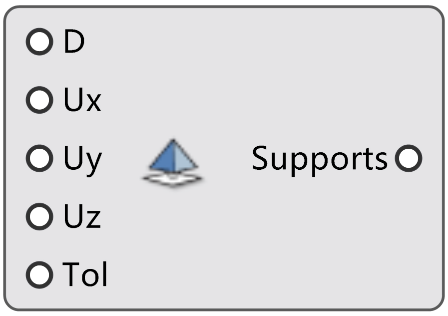
NormalLoad3D

Apply a three-dimensional load along the normal to the surface of the object.



NonDesignDomain3D

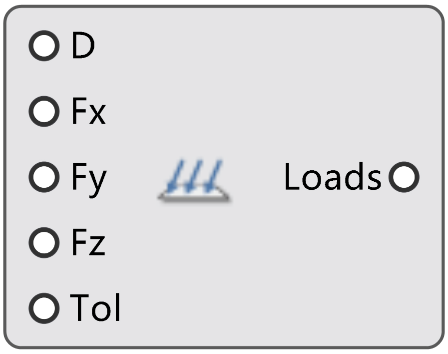
Define three-dimensional non-design areas.



Support3D

Define three-dimensional supports.

Load3D



Load3D

Description：

Define a genaral load in 3D problem.

(This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display the cell to which the load is applied and the cell center point to represent the cell.

Input：

D: Enter the area of a load. You can input points, lines, surfaces or closed soilds.

Fx: Enter the load in the X-axis direction.

Fy: Enter the load in the Y-axis direction.

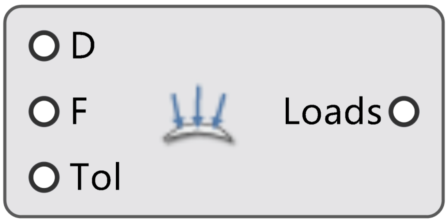
Fz: Enter the load in the Z-axis direction.

Tol: The tolerance for area determination. This parameter will be invalid when you input closed soilds.

Output：

Loads: Output a load.

NormalLoad3D



NormalLoad3D

Description：

In the three-dimensional analysis, define a load that is perpendicular to the surface and the load can only be applied to the surface of the object.

(This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display the cell to which the load is applied and the cell center point to represent the cell.

Input：

D: Enter the area of a load. You have to input surfaces.

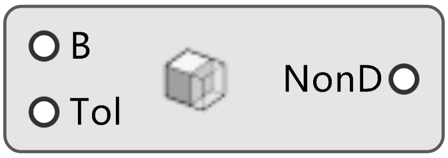
F: Load size.

Tol: The tolerance for area determination. This parameter will be invalid when you input closed soilds.

Output：

Loads: Output a load.

NondesignDomain3D



NonDesignDomain3D

Description：

In the two-dimensional analysis, define a non-designed area (only valid in two-dimensional analysis). (This component must be used after generating the mesh.)

Usage：

Right-click on ShowSelElement to display cells that are not designed, and use cell center points to represent cells.

Input：

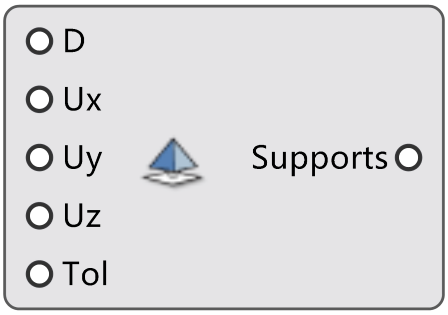
D: Enter the area of a load. You can input lines, surfaces or closed soilds.

Tol: The tolerance for area determination. This parameter will be invalid when you input closed soilds.

Output：

NonD: Output non-design area.

Support3D



Support3D

Description：

In the three-dimensional analysis, define a support (only valid in three-dimensional analysis).

(This component must be used after generating the mesh.)

Input：

D: Enter the area of a load. You can input points, lines, surfaces or closed soilds.

Ux: The displacement in the X-axis direction. If you don't input any vaule, there will be no constrains in this direction. If you input 0, there will be constrained in this direction. If you input a non-zero value, this value will mean displacement in this direction.

Uy: The displacement in the Y-axis direction. If you don't input any vaule, there will be no constrains in this direction. If you input 0, there will be constrained in this direction. If you input a non-zero value, this value will mean displacement in this direction.

Uz: The displacement in the Z-axis direction. If you don't input any vaule, there will be no constrains in this direction. If you input 0, there will be constrained in this direction. If you input a non-zero value, this value will mean displacement in this direction.

Tol: The tolerance for area determination. This parameter will be invalid when you input closed soilds.

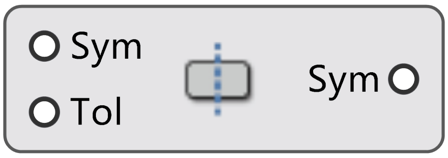
Output：

Supports: Output a support.

PreProcess

Preprocess

For the component group of calculating the result pre-processing, the user can use the module under the group to set the symmetry constraint, material, optimization parameters, sensitivity options, multiple working conditions, and also collect the pre-processing parameters to connect the cloud server.

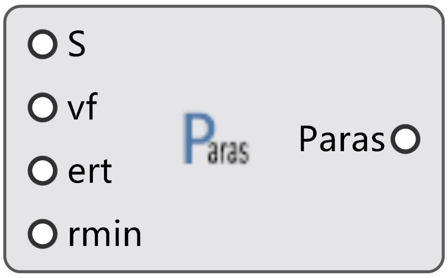


Symmetry

Symmetric constraints.

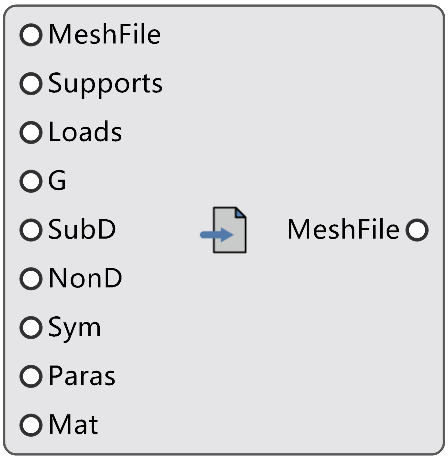
Material

Material settings.



OptParameters

Optimize parameter settings.

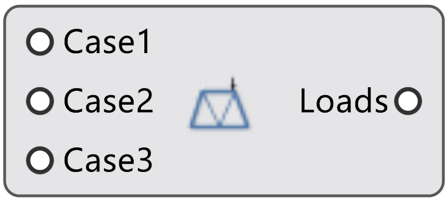


PreProcessing

Pre-processing parameter set.

Sensitivity

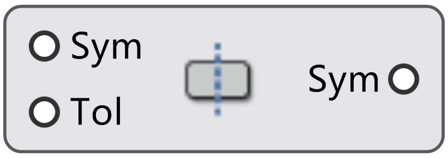
Sensitivity option.



MultipleLoadCase

Multiple settings.

Symmetry



Symmetry

Description：

If a symmetric constraint is imposed, define a line (two-dimensional) or a plane (three-dimensional) as the axis of symmetry or face.

(This component must be used after generating the mesh.)

Input：

Sym: Enter the axis of symmetry (two-dimensional) or the plane of symmetry (three-dimensional).

Tol: Tolerance value.

Output：

Syms: Symmetric constraint.

Material

Material

Description：

Define the material of the object.

Attention：

E and u inputs are valid only when set to User.

Usage：

Right-click on the battery and select some of the preset materials.

Input：

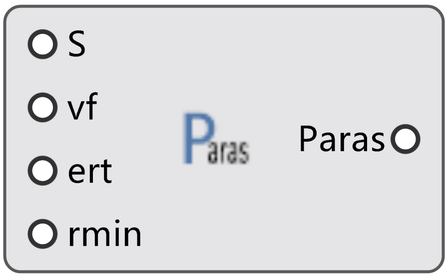
E: modulus of elasticity.

u: Poisson's ratio.

Output：

Mat: Material parameters.

OptParameters



OptParameters

Description :

Define some parameters of the BESO algorithm.

Input :

S: Sensitivity. Presets can be selected from PreProcessing.

```
In short, sensitivity is the basis for the increase and decrease of the unit.
At present, the strain energy density and the Mises stress can be selected.
When the unit volumes are all the same, the calculation results of the strain
energy density or the Mises stress are the same, but the unit sizes are
generally different. In order to consider the influence of the unit volume,
the strain energy density is selected.
```

vf: Constrained volume fraction.

```
The constrained volume fraction is the percentage of the volume of
the material area to be retained in the original design area (0-1).
This value cannot be too small. Otherwise, the material is not enough
to establish a structure. The calculated structure is not established,
and the calculation will not be convergence.
```

ert: Evolution rate.

```
Evolution rate is the volume percentage of material cut at each step.
(recommended 0.01-0.05)
```

rmin:过滤半径。（默认为3倍的网格）

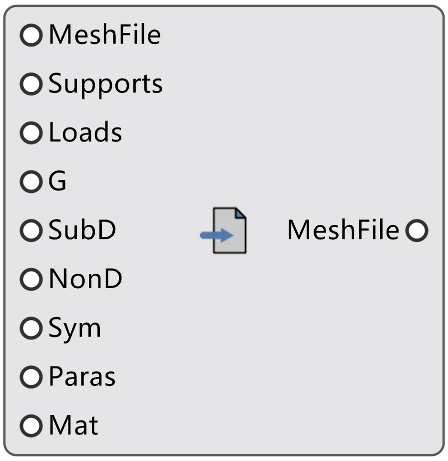
```
It can be understood that the center of the i-th unit is the center,
the radius is Rmin (the three-dimensional is the ball), and the
sensitivity of the i-th unit is determined according to the distance
```

```
between the inner unit and the i-th unit.
```

Output :

Paras: Parameter set.

PreProcessing



PreProcessing

Description：

Write all the definitions of the analysis to a file and read the file as a cloud.

输入端：

MeshFile: Transfer the data from the Mesh battery to this battery.

Supports: Connect all the supports as a List.

Loads: All loads are accessed as a List.

G: Gravity (defined by gravity, you can customize the direction and size of gravity).

NonD: Non-designed area. (If not, keep it as default).

Sym: Symmetric constraint. (If not, keep it as default).

Paras: Optimization parameter set.

Mat: Material.

Output：

MeshFile:Mesh File.

Sensitivity

Sensitivity

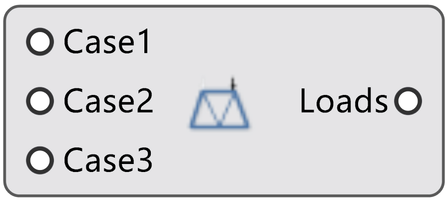
Description：

Some presets for the sensitivity of the BESO algorithm are given.

Output：

S: Sensitivity。

MultipleLoadCase



MultipleLoadCase

Description：

The input end of the component is a load case, and one load case can be composed of one or more loads;

Attention：

The input end of the component is a load case, and one load case can be composed of one or more loads;

输入端：

Case1: The first condition.

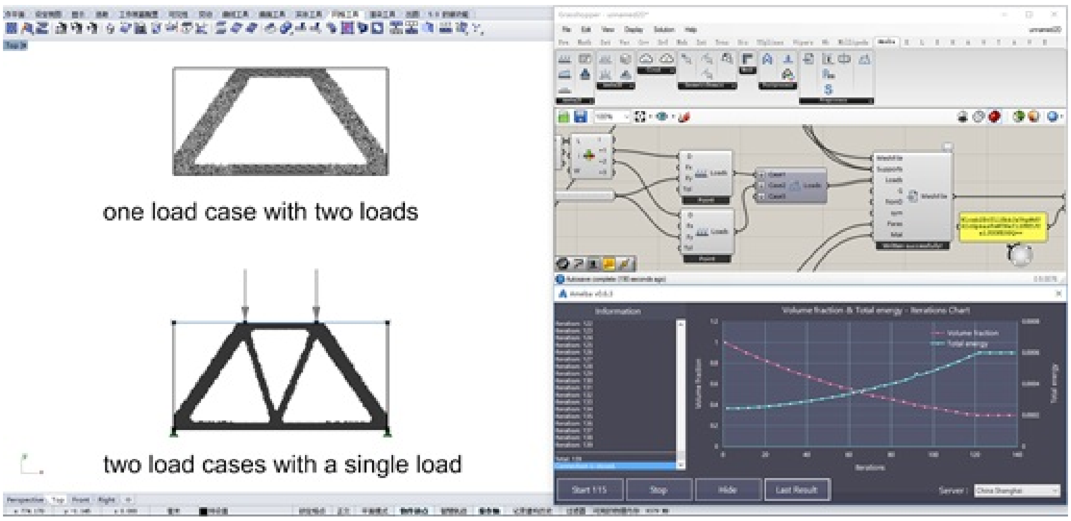
Case2: The second condition.

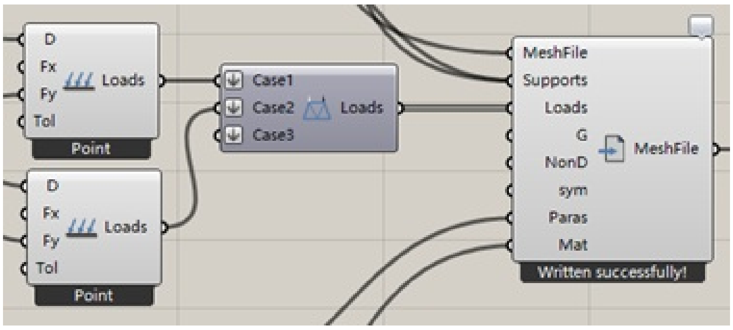
...

输出端：

Loads: Multiple working conditions collection.

示例：

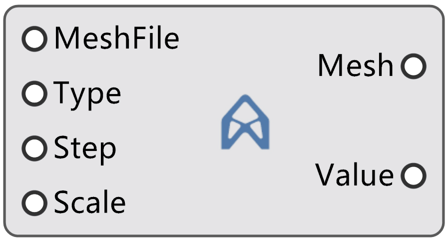



D
Fx
Fy
Tol
Loads
Point
D
Fx
Fy
Tol
Loads
Point
Case1
Case2
Case3
Loads
MeshFile
Supports
Loads
G
NonD
sym
Paras
Mat
MeshFile
Written successfully!

PostProcess

Postprocess

For the component group used for post-processing of calculation results, the user can use the module under the group to query and display the calculation result, and also reconstruct the mesh model.

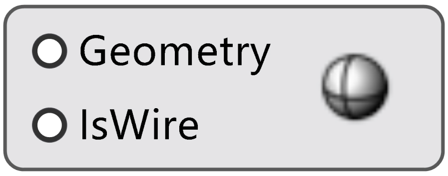


Display

Display the result.

Step

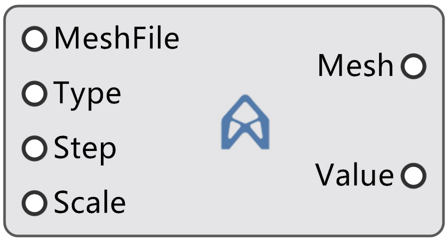
The number of iteration steps.



RenderDisplay

Display the model.

Display



Display

Description：

Display the calculation result.

After the server completes the calculation, you will receive many files (topology optimizated result) from server. You need to connect Step component with Display component. Then, you can pull Step component's button for looking through these files. If you want to have a real-time view, you just left-click "Show" button in the calculation window.

Input：

MeshFile: Transfer the data from the Mesh component to this componenet.

Type: Display type ("None" means no cloud image, "Mises" means Mises stress, "Principal" means main stress)

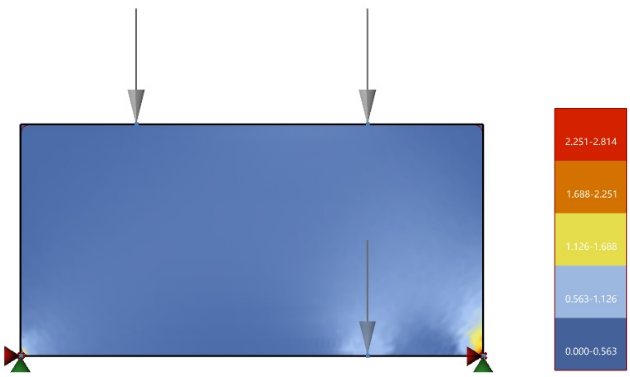
Step: Each step of the calculation (the total number of steps is different depending on the input mesh)

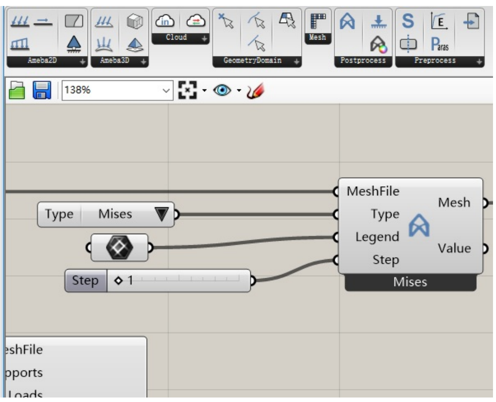
Output：

Mesh: Corresponding to the mesh of Step.

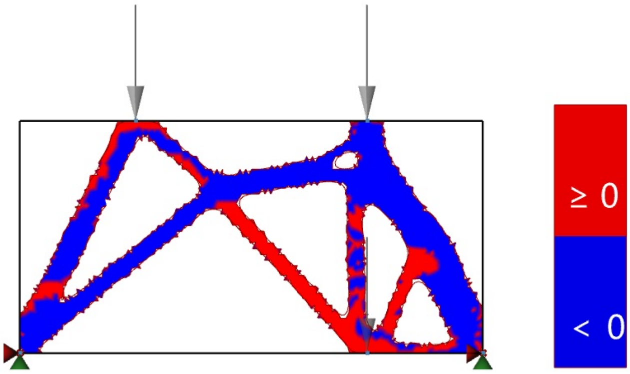
Value: The result value of the display type corresponding to the mesh vertex.

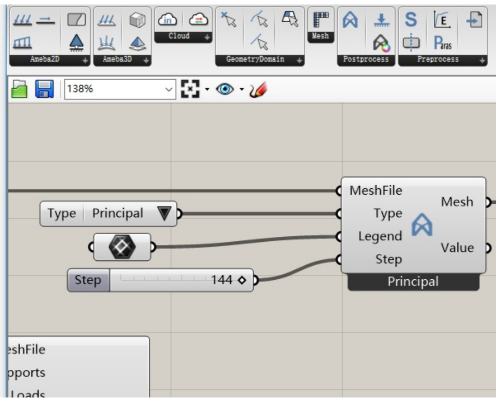
Example：





Display


≥ 0
< 0


138%
Type
Principal
Step
144
MeshFile
Type
Legend
Step
Mesh
Value
Principal

Step

Step

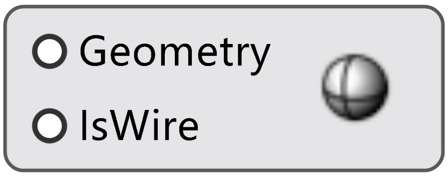
Description：

After the server completes the calculation, you will receive many files (topology optimizated result) from server. You need to connect Step component with Display component. Then, you can pull Step component's button for looking through these files. If you want to have a real-time view, you just left-click "Show" button in the calculation window after connecting Step component with Display component.

Output：

Step: The number of steps calculated.

RenderDisplay



RenderDisplay

Description：

Display the model as a white model with black edges.

Input：

Geometry: The model to be displayed.

IsWire: Whether to display mesh edges or surface boundaries.

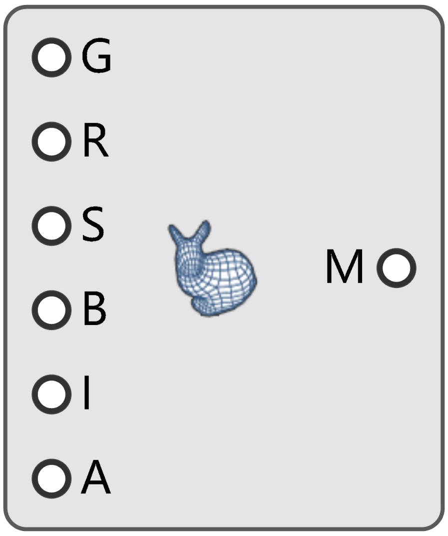
MeshOpt

MeshOpt

Tools for Mesh Optimization.

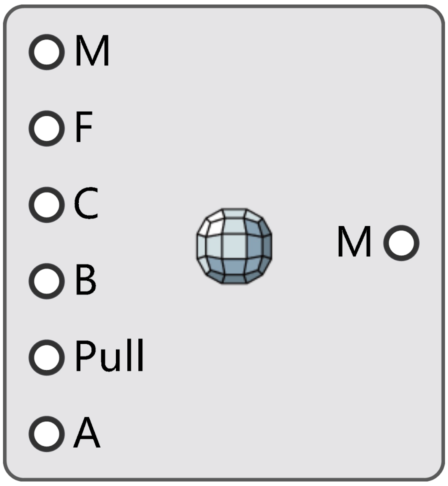
Rebuild2D

This component has the ability to reconstructe the mesh calculated by Ameba2D into a Trimmed surface. In most cases, it is used for repairing 2d topology optimization results.



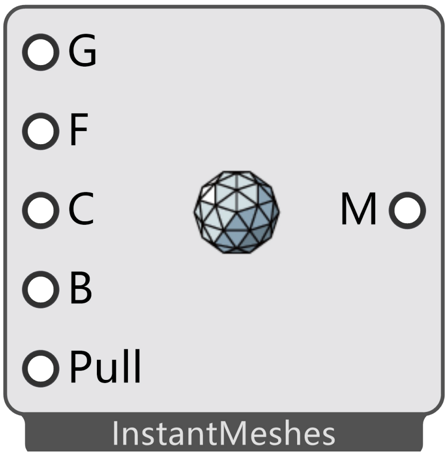
Remesh

Reconstruct the mesh from arbitrary geometry using OpenVDB.It is able to be used for repairing non-manifold mesh since it works according to voxels. But sometimes it also generates some non-manifold edges. You should adjust "Size" and "Iso" to optimize mesh in this case.



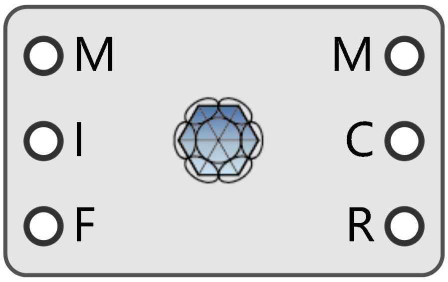
QuadMesh

This component has the ability to create pure quad meshes from closed geometry using InstantMeshes and Quadriflow. To get started, please right-click the component and click "Solve". If you wanna stop it, you can select "Stop" to kill the process.



TriMesh

This component has the ability to create pure triangle meshes from arbitrary meshes using InstantMeshes and pmp-library. It offers three algorithms which are used for opened meshes and closed meshes respectively.



CPMesh

CP mesh (Circle Packing Mesh) developed to rationalize freeform surfaces in architecture and design originally, a CP mesh is a triangle mesh whose incircles form a packing.

Rebuild2D

Rebuild2D

Description：

This component has the ability to reconstructe the mesh calculated by Ameba2D into a Trimmed surface. In most cases, it is used for repairing 2d topology optimization results.

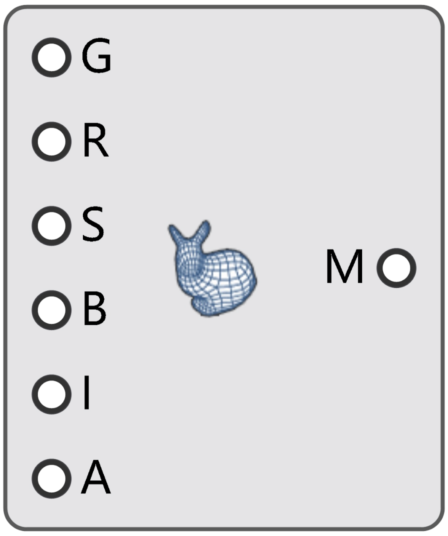
Input：

- Mesh: The mesh model to be optimized.
- Count: The number of subdivision points. The larger the value, the closer it is to the original mesh.

Output：

- Surface: Output a Trimmed Surface.

Remesh



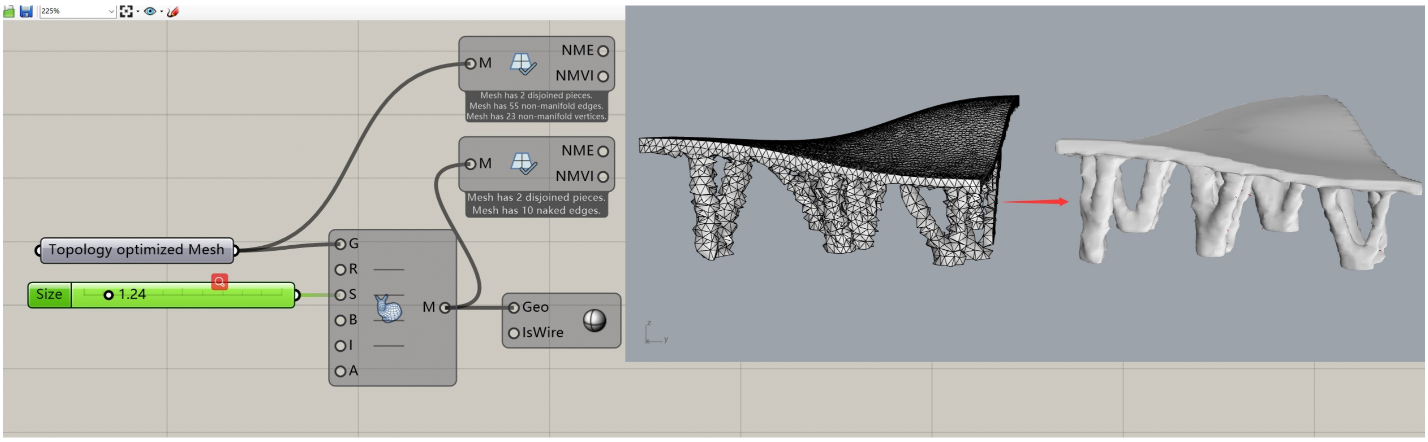
Remesh

Description :

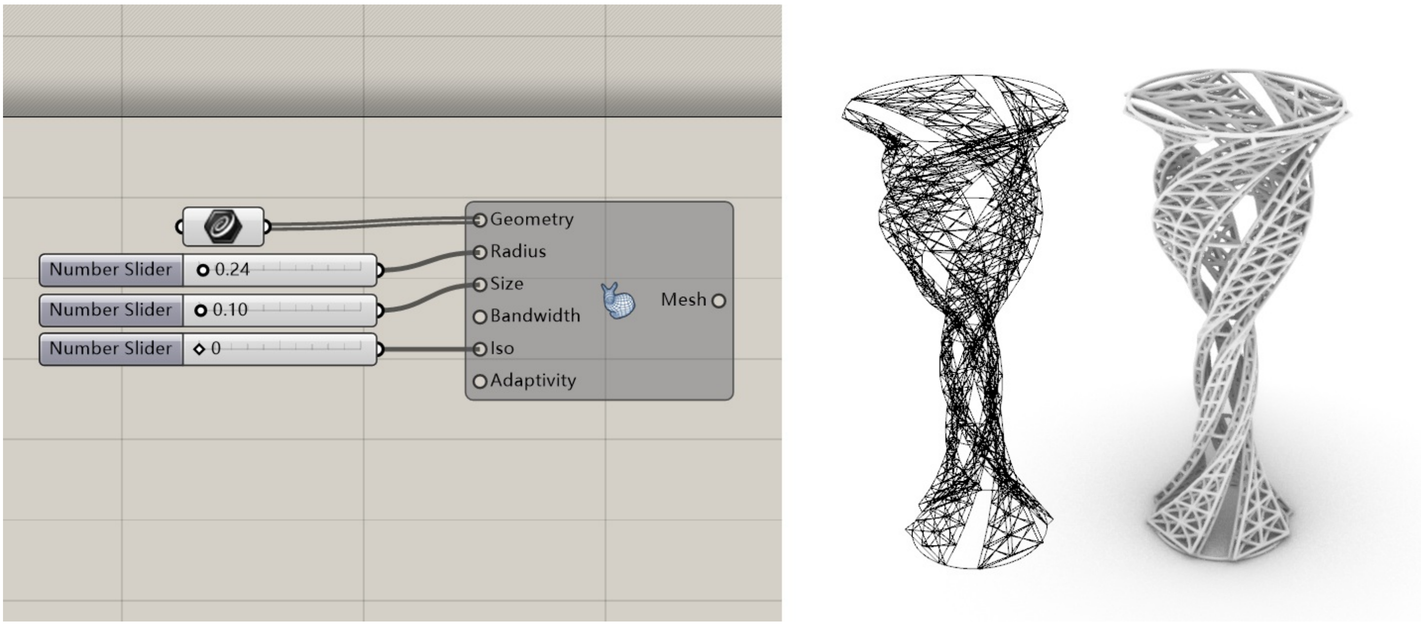
Reconstruct the mesh from arbitrary geometry using OpenVDB.It is able to be used for repairing non-manifold mesh since it works according to voxels. But sometimes it also generates some non-manifold edges. You should adjust "Size" and "Iso" to optimize mesh in this case.

BTW: You can input arbitrary geometry in this component such as points, curves, surfaces, meshes or breps. There are some instantce as shown below.

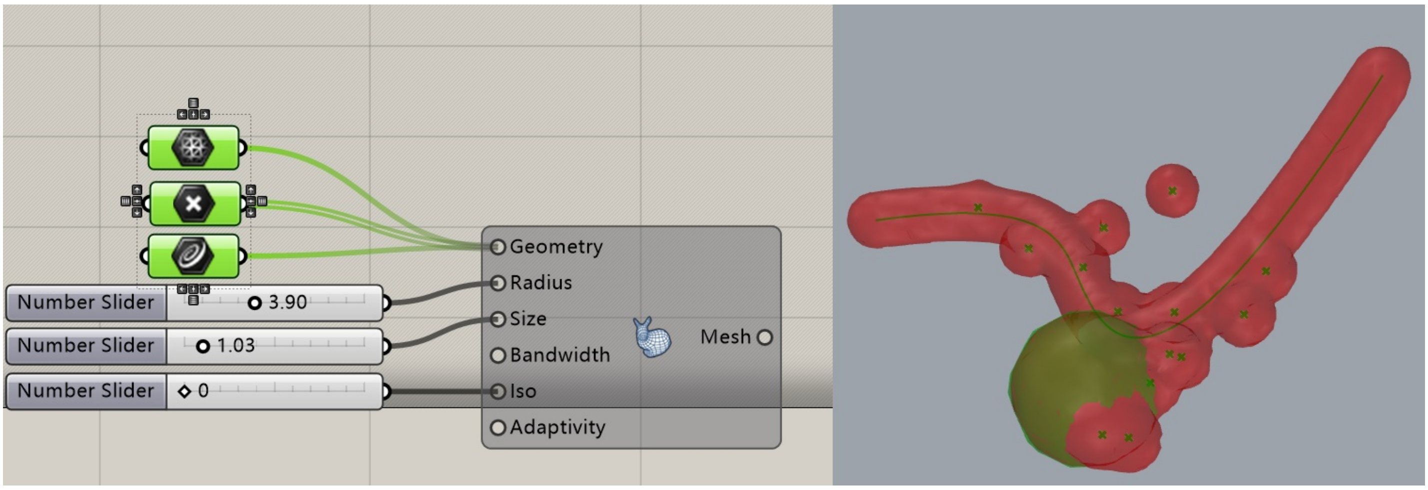
(1) Repair topology optimized results.



(2) Create pipe mesh from curves.



(3) Remesh component combines all kinds of geometry into one mesh.



Input：

- Geometry: Input a geometry.
- Radius: Supply one value or a list of values equal to the number of points or curves supplied. If you only input a mesh in 'Geometry', this parameter will be unvalid.
- Size: Voxel size is the x, y, z dimensions of the individual voxels filling the volume. Think of this as the resolution of the volume.
- Bandwidth: Bandwidth extends the available voxel field around your volume. Voxels within this band are set active, everything else is inactive.
- Iso: Isovalue is the accuracy of the resulting mesh to the original value. it can be abstractly thought of as a positive or negative offset.
- Adaptivity: Adaptivity sets the uniformity of mesh faces. Values can range from 0-1, with a value 0 being more equalized and dense.

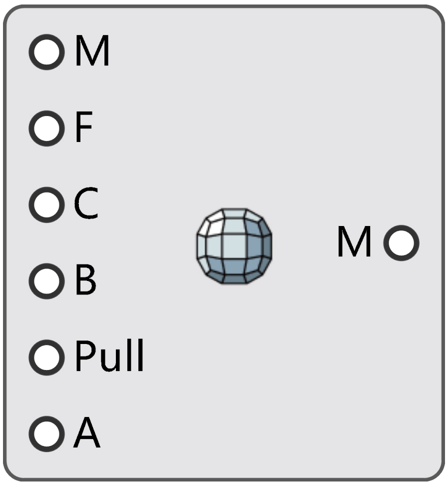
Output：

- Mesh: Return remeshing mesh.

Thanks：

- Remesh is based developed on top of the OpenVDB library which is developed and maintained by DreamWorks
  Animation for use in special effects applications for feature film productions. For more information please visit:
  www.openvdb.org

QuadMesh

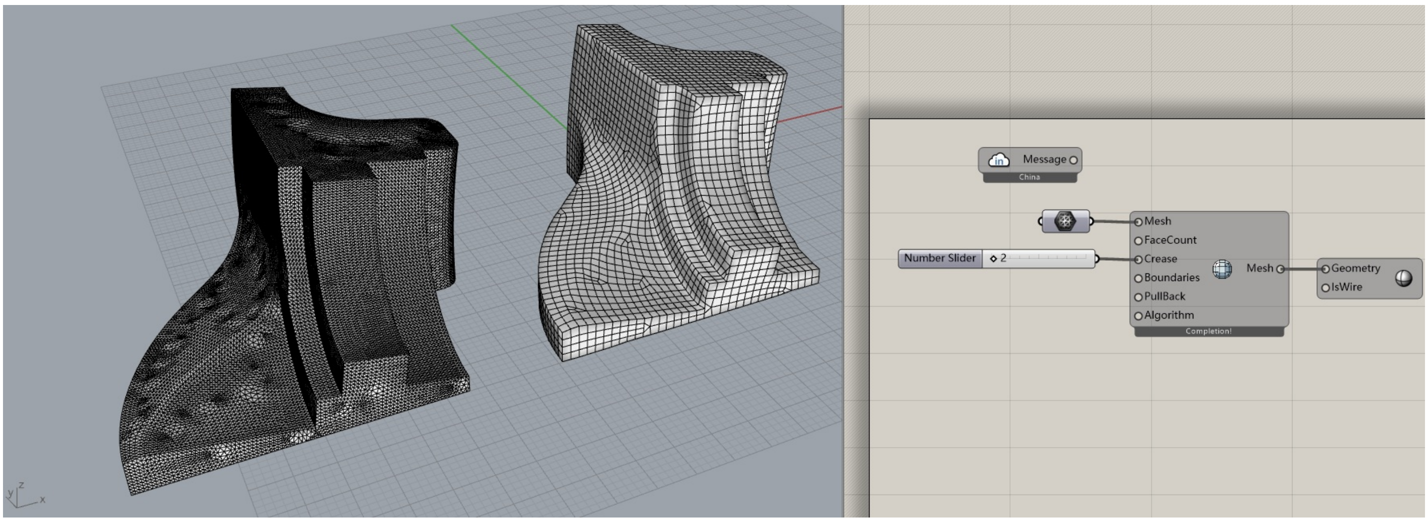


QuadMesh

Description:

This component has the ability to create pure quad meshes from closed geometry using InstantMeshes and Quadriflow. To get started, please right-click the component and click "Solve". If you wanna stop it, you can select "Stop" to kill the process.

BTW: This component is a function in Ameba Pro. Please login first before use. It is notices that this function has not effective in the processing of opened meshes.



Input:

- Mesh: A mesh model that needs to be remeshed.
- FaceCount: The number of faces.
- Crease: The dihedral angle of the crease.
- Boundaries: Align boundaries (used only when the grid is not closed).
- PullBack: Pulls the generated mesh vertices back to the original model.

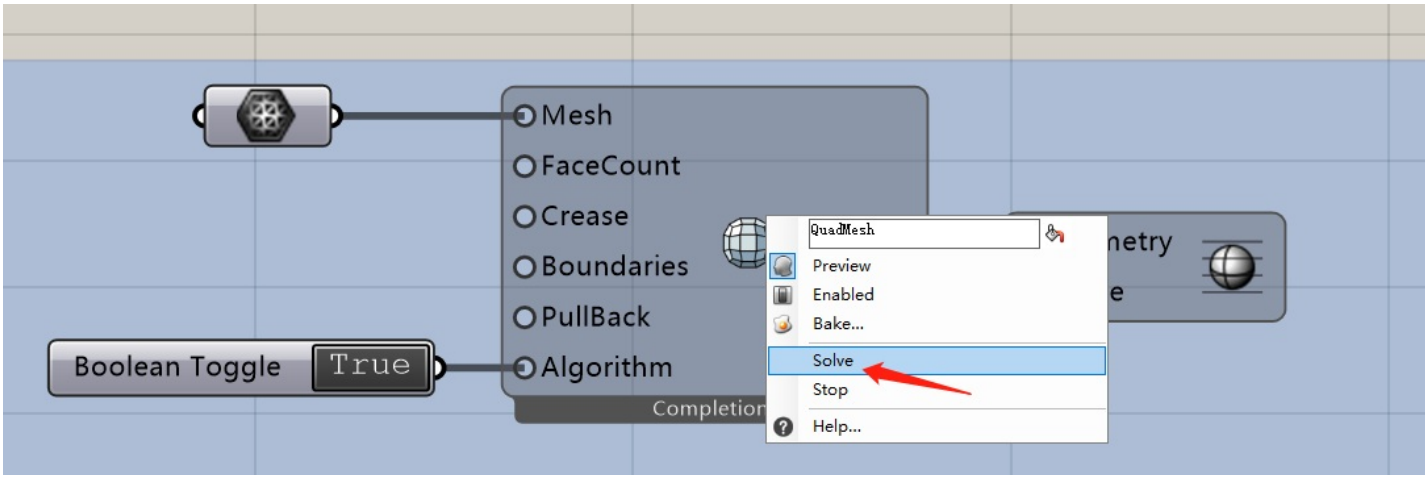
- Algorithm: Algorithm switching, using the InstantMesh algorithm for False and Quadriflow algorithm for True (slow, good quality, but may fail to calculate).

Output:

- Mesh: Outputs the reconstructed quad mesh.

Right click menu:

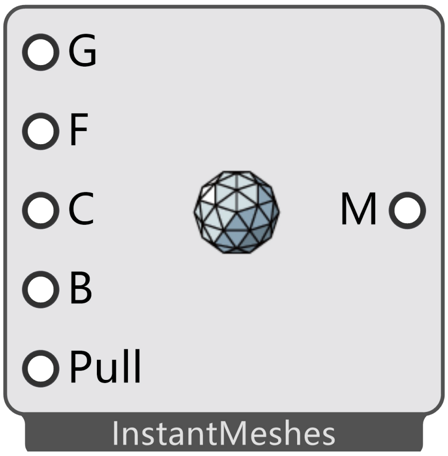
- Solve: Start the calculation.
- Stop: Terminate the calculation.



Thanks：

- QuadMesh's Instant Meshes algorithm is based on the publication:
  [Instant Field-Aligned Meshes]
  Wenzel Jakob, Marco Tarini, Daniele Panozzo, Olga Sorkine-Hornung
  In ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2015)
- QuadMesh's Quadriflow algorithm is based on the publication:
  [QuadriFlow: A Scalable and Robust Method for Quadrangulation]

TriMesh



TriMesh

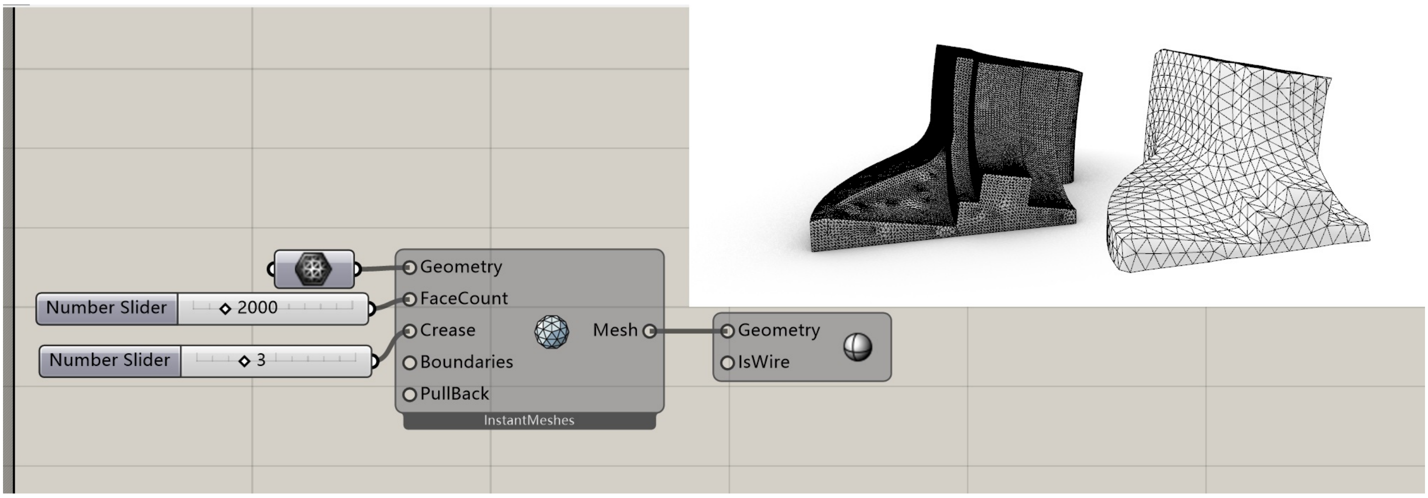
Description:

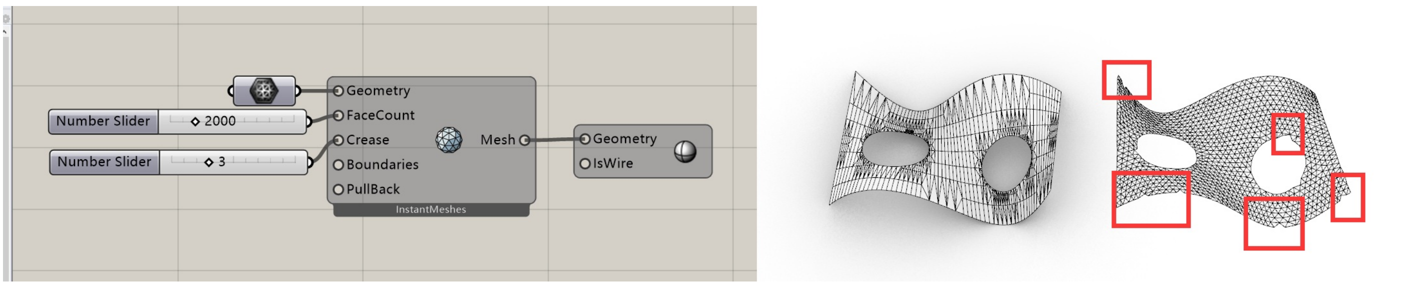
This component has the ability to create pure triangle meshes from arbitrary meshes using InstantMeshes and pmp-library. It offers three algorithms which are used for opened meshes and closed meshes respectively.

InstantMeshes

InstantMeshes algorithm is good at create pure triangle meshes from closed meshes. It is able to produce mesh faces with uniform edge lenths. But it is notices that this function has not effective in the processing of opened meshes.

:-:





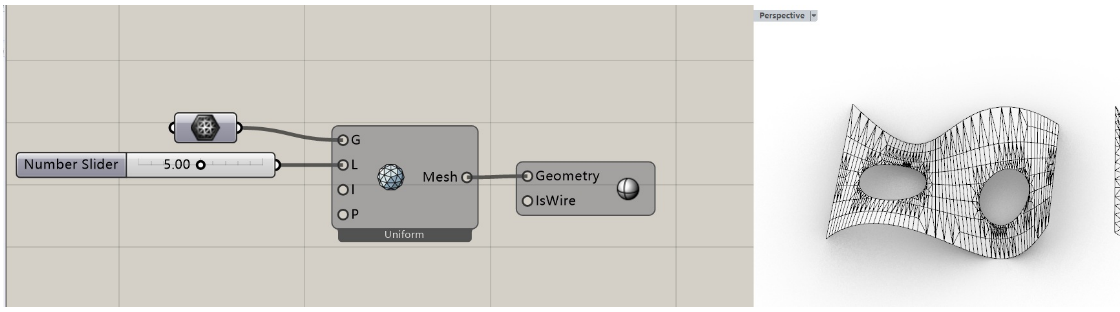
Input:

- Geometry: A geometry model that needs to be remeshed.
- FaceCount: The number of faces.
- Crease: The dihedral angle of the crease.
- Boundaries: Align boundaries (used only when the grid is not closed).
- PullBack: Pulls the generated mesh vertices back to the original model.

Uniform

Uniform mode can solve opened meshes very well. But the speed of this mode is slower than the speed of InstantMeshes.

:-:

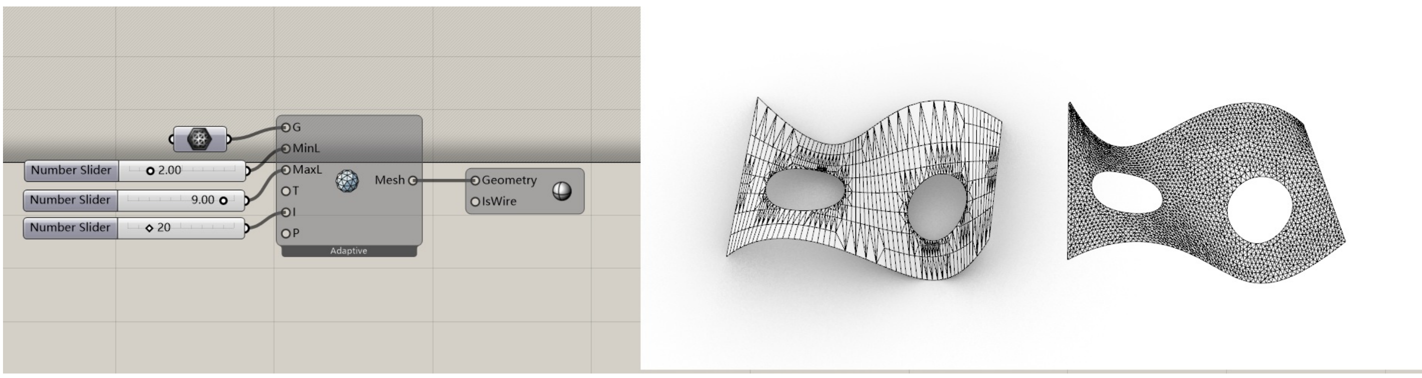


Input:

- Geometry: A geometry model that needs to be remeshed.
- Length: Target Edge Length.
- Iteration: A maximum number of remesh iterations.
- Projection: Project vertices to the original model.

Adaptive

Adaptive mode can also solve opened meshes very well. But the speed of this mode is slower than the speed of InstantMeshes and Uniform Mode.



Input:

- Geometry: A geometry model that needs to be remeshed.
- MinimumLength: The minimum length of mesh edge.

- MaximumLength: The maximum length of mesh edge.
- Tolerance: The tolerance of algorithm.
- Iteration: A maximum number of remesh iterations.
- Projection: Project vertices to the original model.

Output:

- Mesh: Outputs the reconstructed mesh of the triangle.

Thanks：

TriMesh's Instant Meshes algorithm is based on the publication:

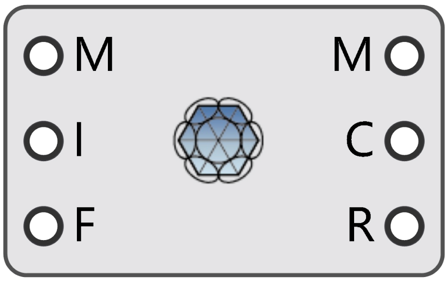
- TriMesh's Instant Meshes algorithm is based on the publication:

  Instant Field-Aligned Meshes

  Wenzel Jakob, Marco Tarini, Daniele Panozzo, Olga Sorkine-Hornung

  In ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2015)

CPMesh



CPMesh (Circle Packing Mesh)

Description：
CP mesh (Circle Packing Mesh) developed to rationalize freeform surfaces in architecture and design originally, a CP mesh is a triangle mesh whose incircles form a packing. A CP mesh is a triangle mesh whose incircles (orange) form a packing. Then spheres (blue), which are centered at mesh vertices and are orthogonal to the incircles of neighboring triangles form a packing, too. Centers and axes of incircles defifine a hexagonal support structure with Torsion-Free Nodes (red, see Figure 1).

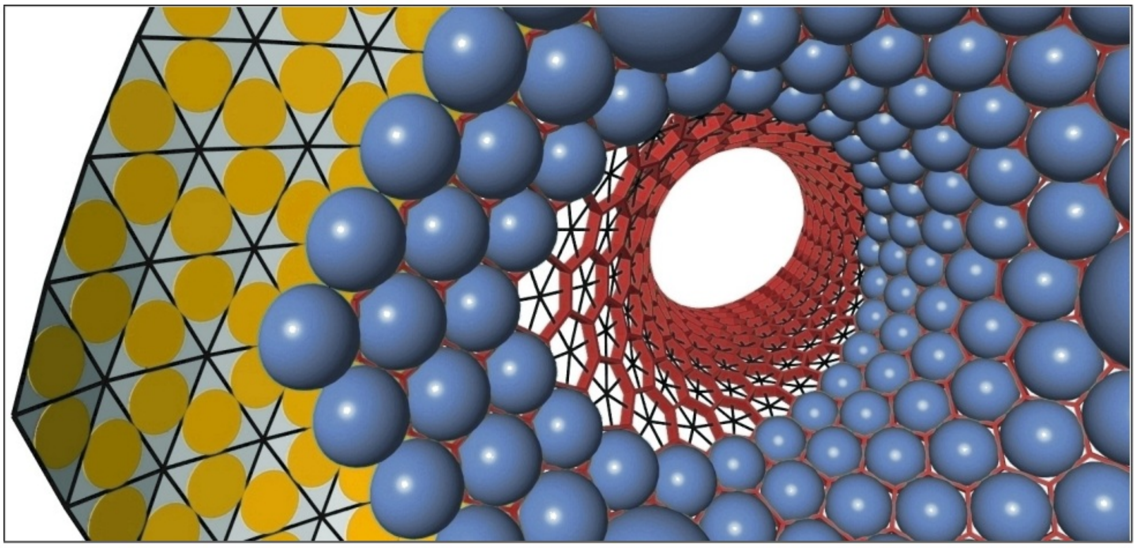


Figure 1. Graph of CP Mesh. Adapted from 'Packing circles and spheres on surfaces'.

"CP meshes have some useful properties which are quite relevant to various algorithms and applications. One of these properties is that they form an ideal primal/dual structure which guarantees orthogonality, non-intersection and convexity. This is important in case we are using discrete differential geometry operators: Orthogonality ensures good numerical qualities of the Laplacian and furthermore it ensures that we can safely compute primal values based on the dual values and dual values based on primal values respectively. Another important property of a CP mesh is that if we extrude the edges of the dual polygon we get a Torsion Free Structure. In other words, its physical realization could be build out of beams which intersect in a common axis passing through the incircles center. And last but not least, CP meshes can be used to approximate circle packings on a surface. This could be done by either fitting a circle into each vertex gap between the incircle packing or by using the contact points of the sphere packing. Both methods generally produce a fairly good packing. A real circle packing however, is not possible on an arbitrary surface."

——Houdini Gubbins

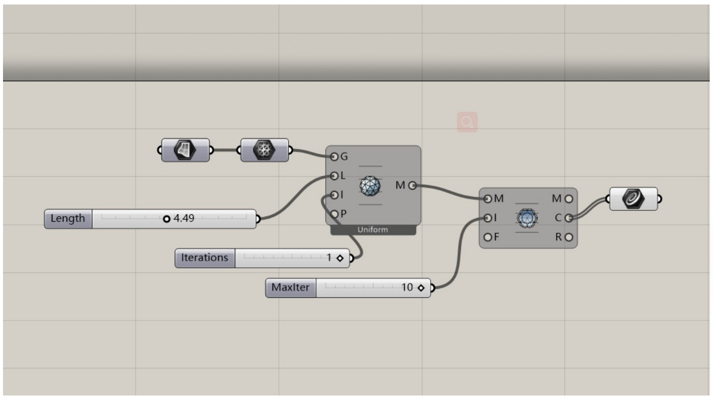
If you wanna learn more about CP mesh, please read references as follows or visit Houdini Gubbins

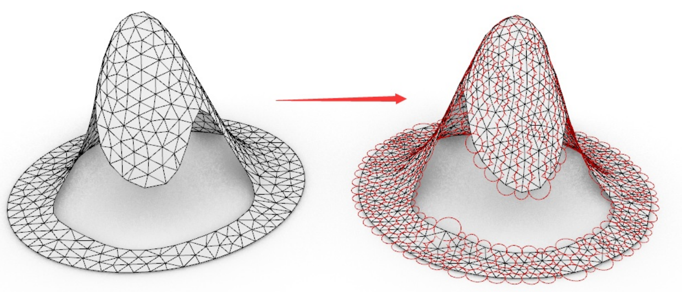
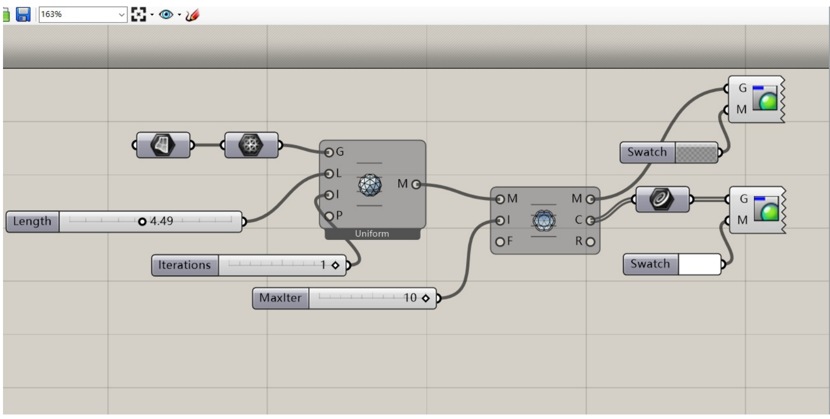
Input：

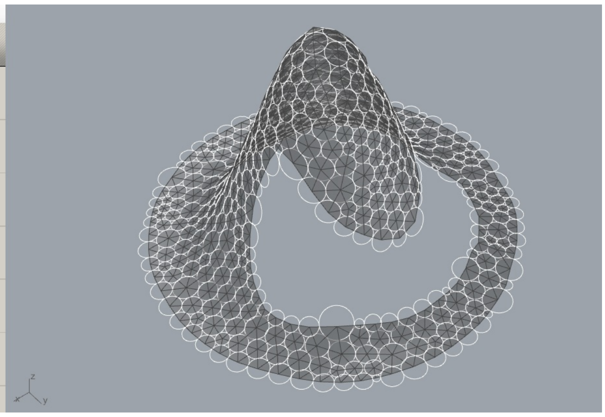
- Mesh: The mesh model to be optimized.
- Iteration: A maximum number of remesh iterations.
- Fixed: If ture, boundaries will be fixed in the process of optimization. Or fix the original mesh.

Output：

- Mesh: Return the optimized CP Mesh.
- Circles: Return a Circle Packing from Ball Packing.
- Radii: Return all radii of the Circle Packing.





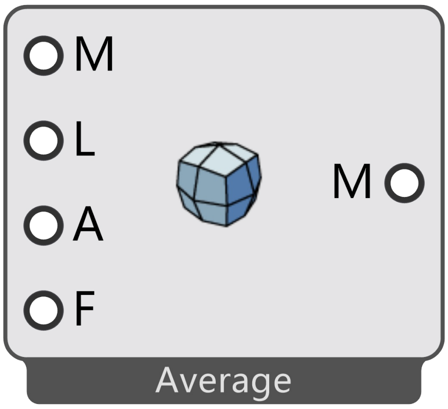
References

[1] Pottmann H, Eigensatz M, Vaxman A, Wallner J. Architectural geometry. Computers & graphics. 2015 Apr 1;47:145-64.

MeshTools

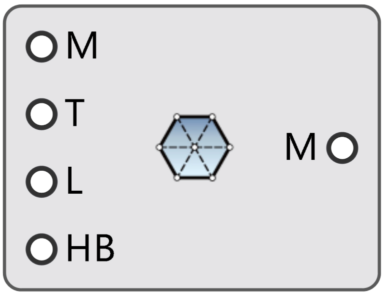
MeshTools

Optimized mesh model.



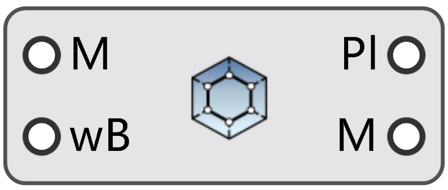
Smooth

Smooth a mesh. We can set those points which are set as supports or loads as anchors in smooth processing to keep sharp features of the model.



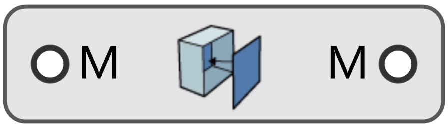
Subdivision

Subdivision are simple yet powerful ways to generate smooth surfaces from arbitrary polygonal meshes. A subdivision method recursively refines a coarse mesh and generates an ever closer approximation to a smooth surface. The coarse mesh can have arbitrary shape, but it has to be a 2-manifold. The coarse mesh is repeatedly refined by a quadrisection pattern, and new points are generated to approximate a smooth surface. This component contains two popular subdivision methods, Catmull-Clark and Loop.Variations of these methods can be easily extended by substituting the geometry computation of the refinement host.



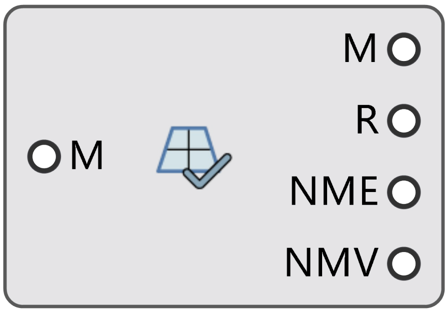
MeshDual

This component computes the dual of the mesh. Triangle meshes' dual graph are hexagons, quadrangle meshes' dual graph are still quadrangles (expressed by polylines).



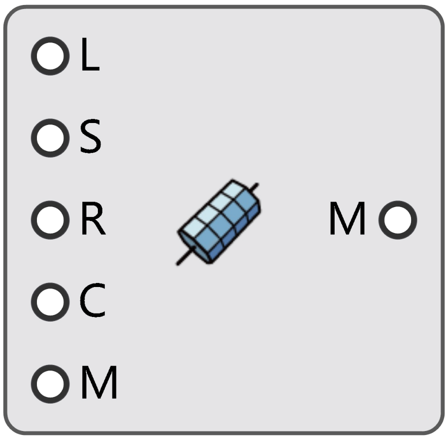
FillHoles

Create simply some mesh faces to fill the holes.



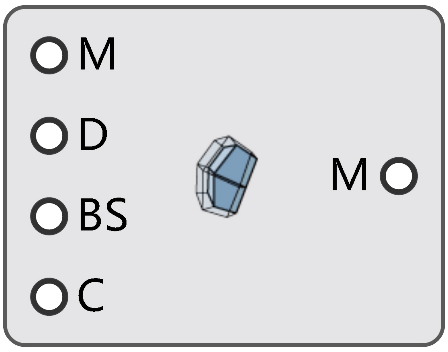
MeshChecker

Check and report informations that whether inputting mesh has non-manifold edges(or non-manifold vertices) or not.



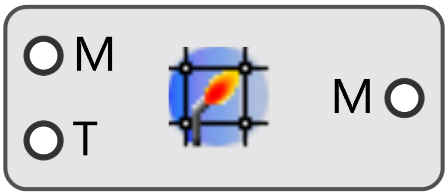
MeshPipe

Create pipe meshes according lines.



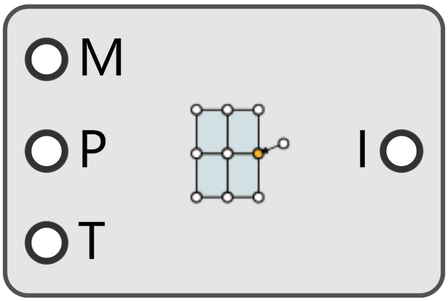
OffsetMesh

Offset mesh.



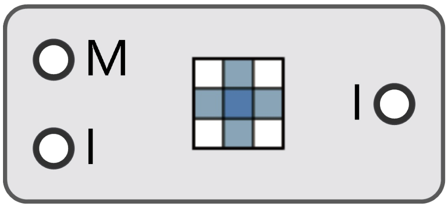
MeshWeld

This component has the ability to weld mesh and clean unused vertices.



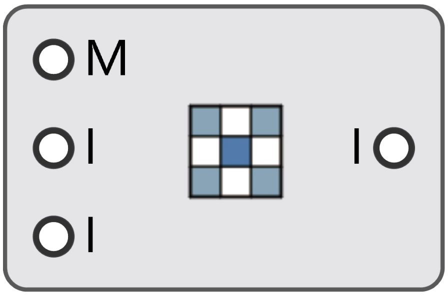
CloestVertex

Get the index of the mesh vertices corresponding to each point in the point cloud.



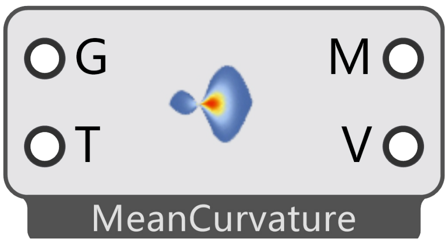
AdjancentFaces

Find adjancent faces' indices according to a face's index.



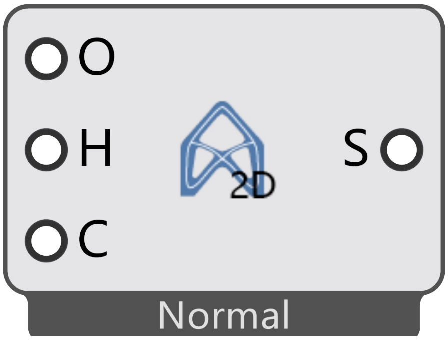
CornerFaces

Find corner faces' indices according to a face's index.



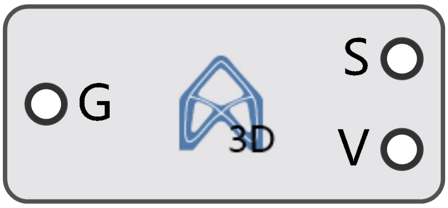
CurvatureAnalysis

Compute the curvature value of per-vertex (Mean,Gaussian,MaxAbsCurvature).



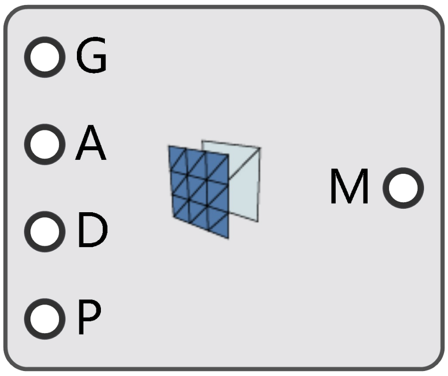
Skeleton2d

Extract the skeleton from a 2D closed curves(or planar mesh, planar surface in "Robust" mode). Skeletons are effective shape abstractions used in segmentation, shape matching, reconstruction, virtual navigation, etc. As the name implies, a curve skeleton is a graph of curvilinear structures (1D). This component provides two algorithms to achieve this function.



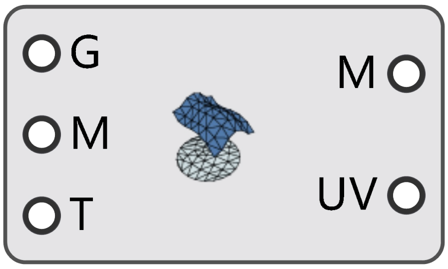
Skeleton3d

Extract the skeleton from a 3D mesh using CGAL. Skeletons are effective shape abstractions used in segmentation, shape matching, reconstruction, virtual navigation, etc. As the name implies, a curve skeleton is a graph of curvilinear structures (1D). It is not a medial axis that for a 3D geometry is composed of surfaces (2D).



Simplification

Surface mesh simplification is the process of reducing the number of faces used in a surface mesh while keeping the overall shape, volume and boundaries preserved as much as possible. It is the opposite of subdivision.

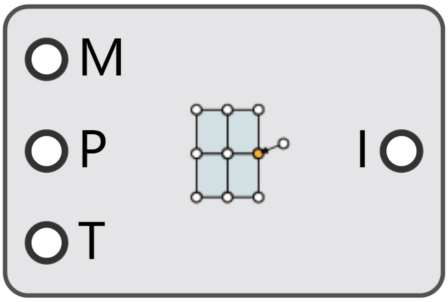


Parameterization

Parameterizing a surface amounts to finding a one-to-one mapping from a suitable domain to the surface. A good mapping is the one which minimizes either angle distortions (conformal parameterization) or area distortions (equiareal parameterization) in some sense. In this component, we focus on parameterizing triangulated surfaces which are homeomorphic to a disk or a sphere, and on piecewise linear mappings onto a planar domain.

Although the main motivation behind the first parameterization methods was the application to texture mapping, it is now frequently used for mapping more sophisticated modulation signals (such as normal, transparency, reflection or light modulation maps), fitting scattered data, re-parameterizing spline surfaces, repairing CAD models, approximating surfaces and remeshing.

ClosetVertex



CloestVertex

Description:

Get the index of the mesh vertices corresponding to each point in the point cloud.

Input:

Mesh: Input a mesh.

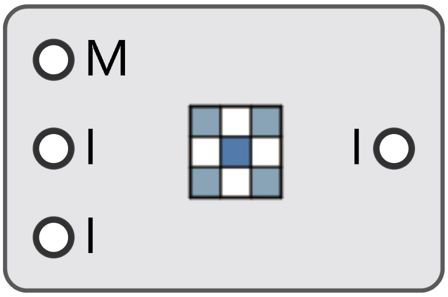
Points: Input a point list.

Threshold: Distance threshold.

Output:

Indices: Output the point indices.

CornerFaces



CornerFaces

Description:

A simple tool for finding indices of corner faces according to an index of a face.

Input:

Mesh: Input a mesh.

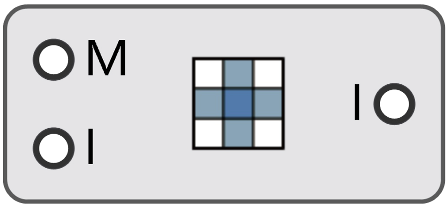
Index: Input the index of a face.

IncludeSelf: If true, it will return the input index.

Output:

Indices: Output indices of corner faces.

AdjancentFaces



AdjancentFaces

Description:

A simple tool for finding indices of adjancent faces according to an index of a face.

Input:

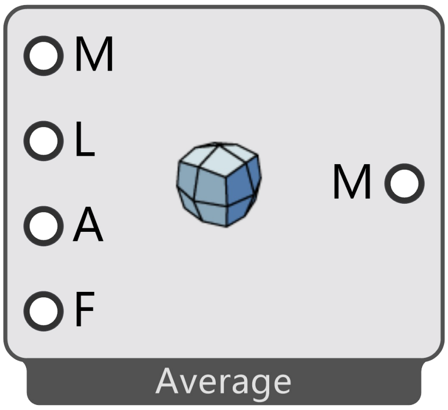
Mesh: Input a mesh.

Index: Input the index of a face.

Output:

Indices: Output indices of adjancent faces.

Smooth



Smooth

Description：

Smooth a mesh. We can set those points which are set as supports or loads as anchors in smooth processing to keep sharp features of the model.

Input：

Mesh: Input a welded mesh.

Level: The number of smoothing iterations for mesh.

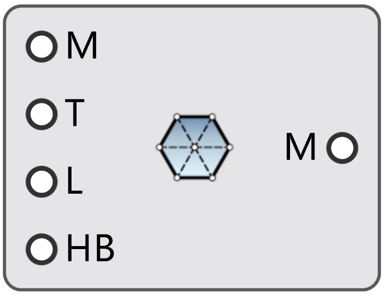
Anchors: Users could set some anchors which are not soomthed though inputing indices of mesh vertices.

Fixed: If this parameter is true, the boundaries of opened meshes will be set as anchors automatically.

Output：

Mesh: Output the mesh.

Subdivision



Subdivision

Description：

Subdivision are simple yet powerful ways to generate smooth surfaces from arbitrary polygonal meshes. A subdivision method recursively refines a coarse mesh and generates an ever closer approximation to a smooth surface. The coarse mesh can have arbitrary shape, but it has to be a 2-manifold. The coarse mesh is repeatedly refined by a quadrisection pattern, and new points are generated to approximate a smooth surface. This component contains two popular subdivision methods, Catmull-Clark and Loop.Variations of these methods can be easily extended by substituting the geometry computation of the refinement host.

Input：

Mesh: Enter a mesh.

Types: Defines how to subdivide the mesh:

- 0: Catmull-Clark Subdivision. This component will subdivide the mesh using Catmull-Clark algorithm.
- 1: Loop Subdivision. This component will subdivide the mesh using Loop algorithm.

Level: The number of subdiving iterations for each mesh faces.

Output：

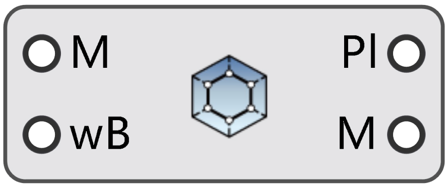
Mesh: Output the mesh.

Algorithm Details：

Catmull-Clark Subdivision Algorithm Details：https://en.wikipedia.org/wiki/Catmull–Clark_subdivision_surface

Loop Subdivision Algorithm Details：https://en.wikipedia.org/wiki/Loop_subdivision_surface

MeshDual



MeshDual

Description :

This component computes the dual graph of a mesh.

Input :

Mesh: Input a mesh.

withBoundary: Whether you want to create a dual graph with boundary or not.

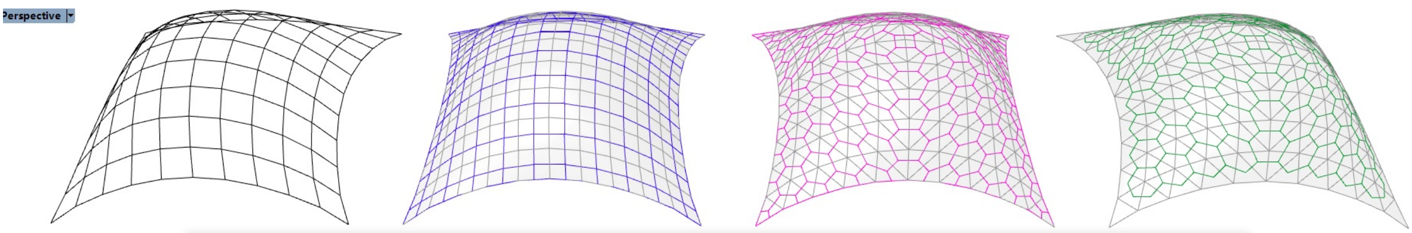
Output :

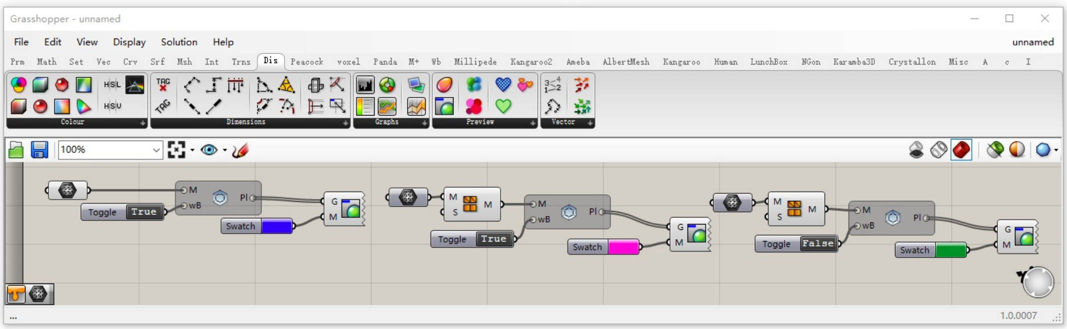
Polyline: Return a polyline list as a dual graph.

Ngon: Return a ngon list as a dual graph.

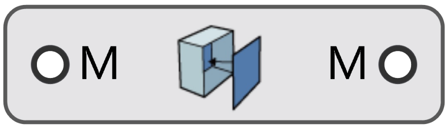
Algorithm Details :

Sometimes, we use it for designing building's curtain walls. The edge count of a polygon depends on its center point, which is a vertex from your original mesh. For example, if that vertex's valence is 6 (means this vertex is shared by 6 triangles) , you will get a hexagon though MeshDual. So, if you have a uniform triangular mesh (Every vertex valence is close to 6), you will get many hexagon polylines via this algorithm. However, if you input a quadrilateral mesh, you will get a new set of quadrilateral polylines, which are generated by interconnecting the center points of original quadrilateral mesh faces.





FillHoles



FillHoles

Description：

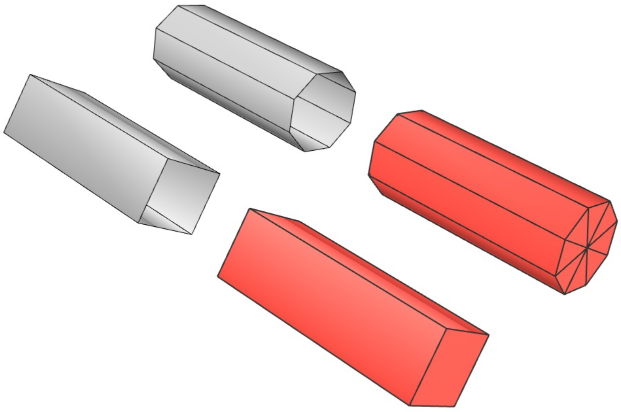
Create simply some mesh faces to fill the holes.

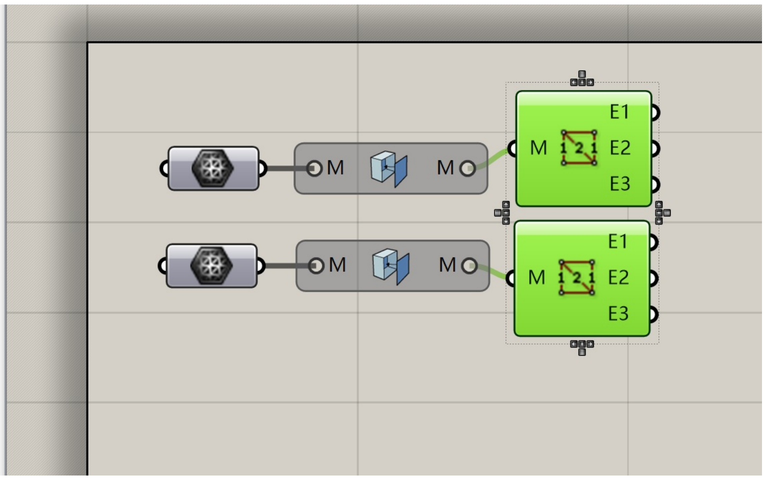
Input：

Mesh: Input a mesh.

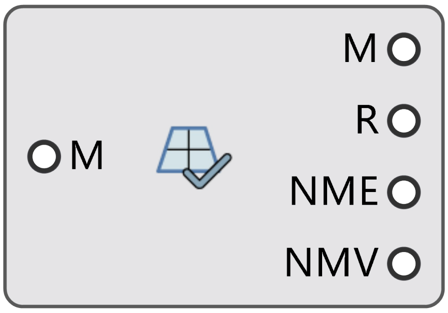
Output：

Mesh: Fill holes of the mesh.





MeshChecker



MeshChecker

Description：

Check and report informations that whether inputting mesh has non-manifold edges(or non-manifold vertices) or not.

Input：

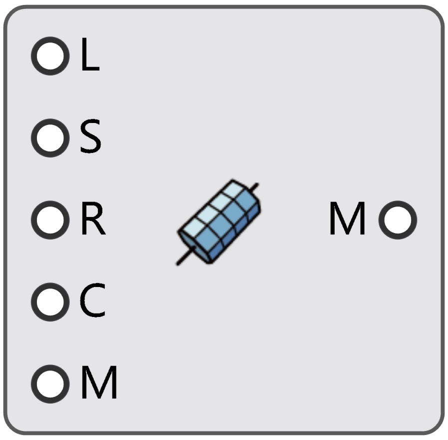
Mesh: Input a mesh.

Output：

Non-Manifold Edges: If your mesh has some non-manifold edges, it will output them.

Non-Manifold Vertices Indices: If your mesh has some non-manifold vertices, it will output them.

MeshPipe



MeshPipe

Description：

Create mesh pipes from lines.

Input：

Line: Input a set of lines.

Sides: The side count of pipe's section.

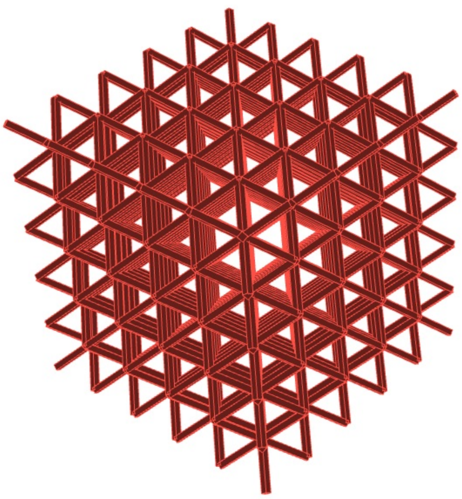
Radius: The radius of pipe's section.

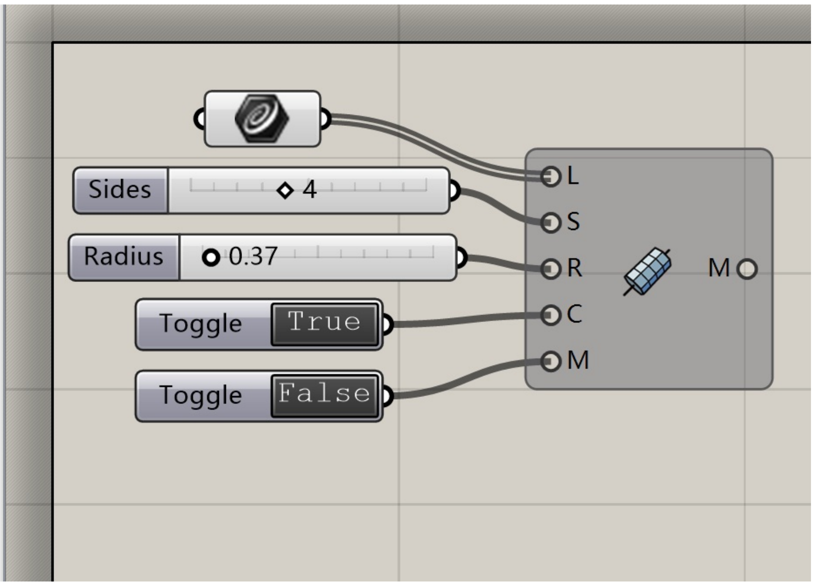
Cap: If true, closed pipes can be generated.

Merge: If true, these pipes can be merged.

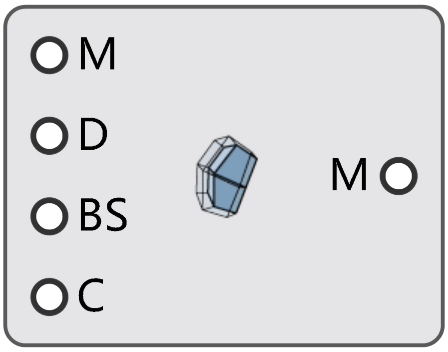
Output：

Mesh: Output mesh pipes.





OffsetMesh



OffsetMesh

Description：

Offset mesh.

Input：

Mesh: Input a mesh.

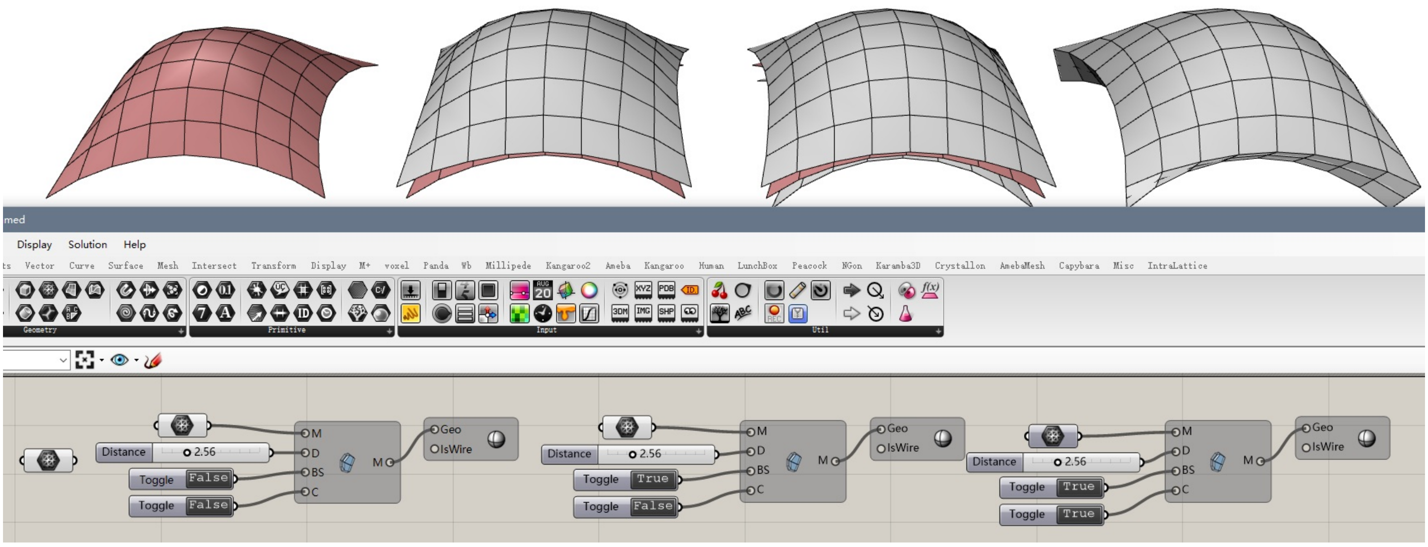
Distance: Input a number as offset distance.

IsBothSides: If true, the mesh will offset on both sides.

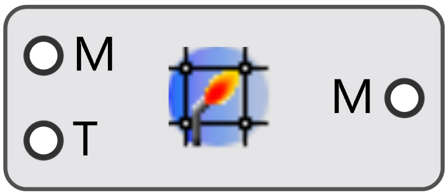
IsClosed: If true, you can get a soild mesh.

Output：

Mesh: Output a mesh.



MeshWeld



MeshWeld

Description:

This component has the ability to weld mesh and clean unused vertices.

Input:

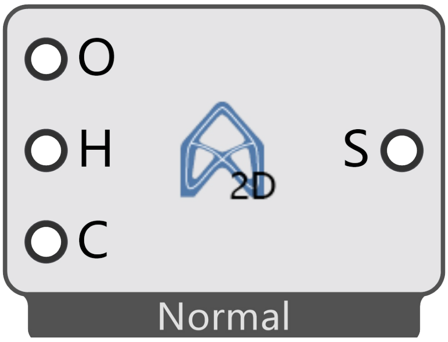
Mesh: Input a mesh.

Threshold: The threshold of the weld mesh (usually kept the default).

Output:

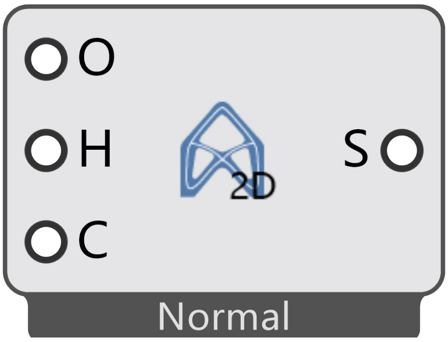
Mesh: Outputs the welded mesh.

Skeleton2D

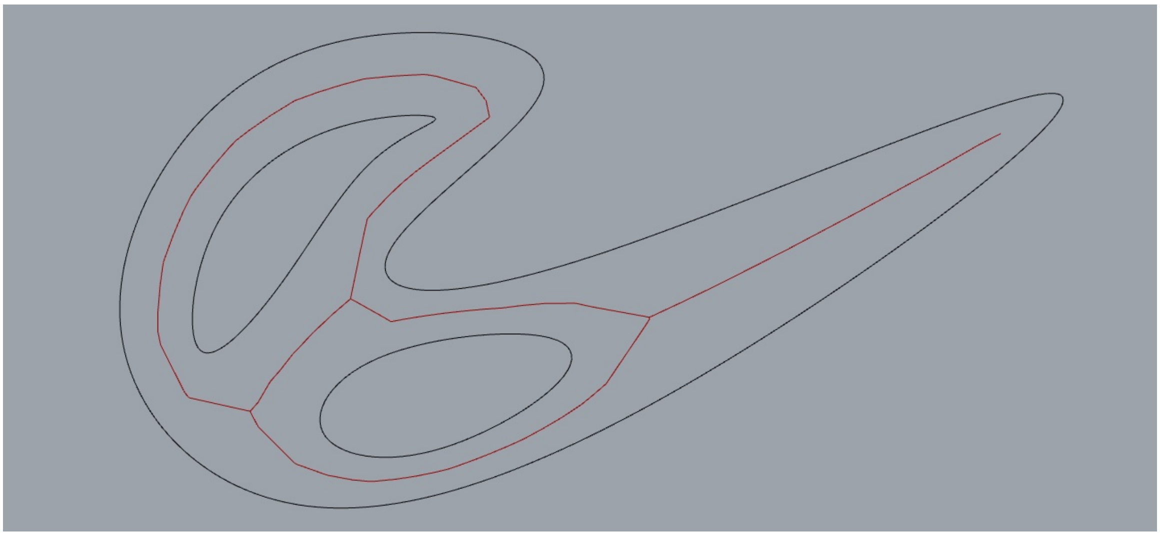


Skeleton2d

Description :
Extract the skeleton from a 2D closed curves(or planar mesh, planar surface in "Robust" mode). Skeletons are effective shape abstractions used in segmentation, shape matching, reconstruction, virtual navigation, etc. As the name implies, a curve skeleton is a graph of curvilinear structures (1D). This component provides two algorithms to achieve this function.



Normal
Normal represents an algorithm that generates skeletons of as few lines as possible. But it's not robust enough.

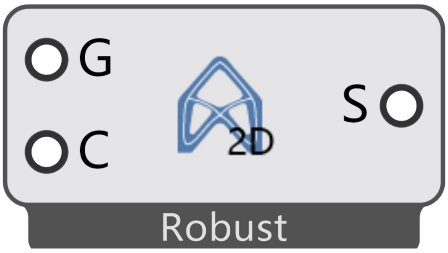


Input：

Outer: Input a outer closed polyline.

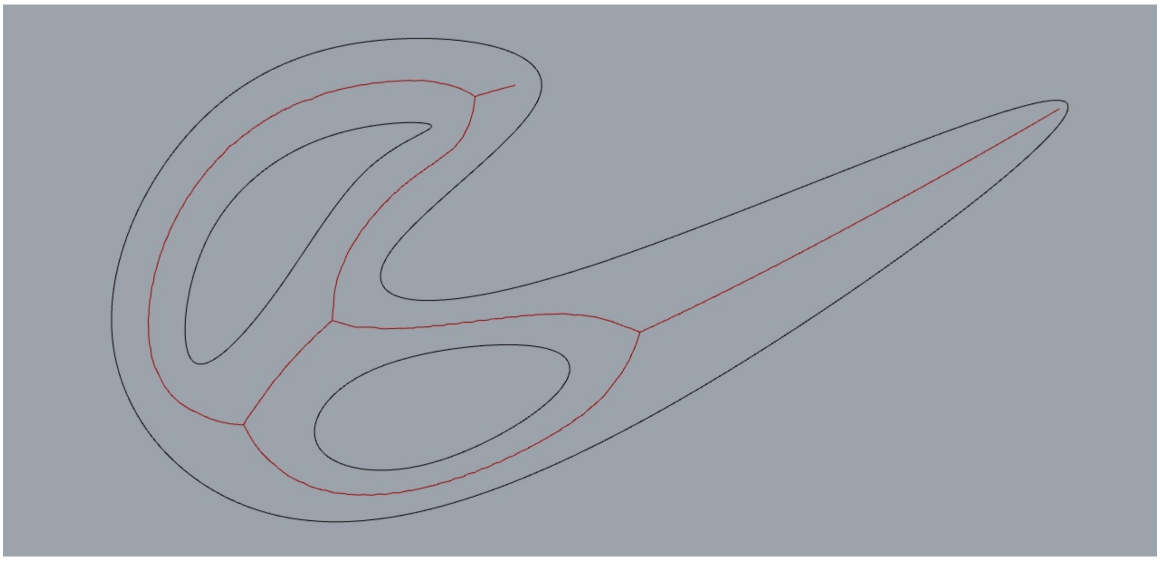
Holes: Input closed holes.

Count: Number of sampling points. This parameter is closely related to the accuracy of skeleton extraction.



Robust

Robust represents an algorithm that generates skeletons of as robustly as possible.



Input：

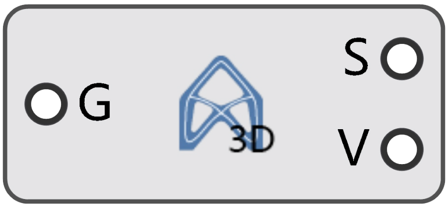
Mesh: Input a 2D mesh or curves for skeleton extraction.

Count: Number of sampling points. This parameter is closely related to the accuracy of skeleton extraction.

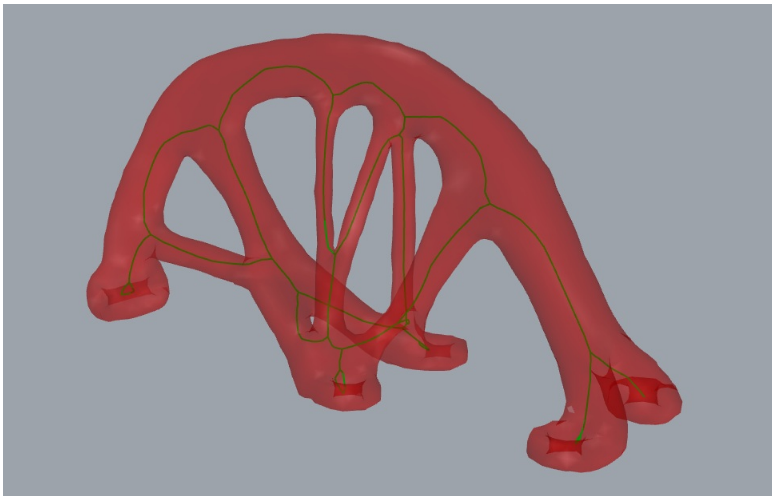
Output：

Skeleton: Output all the edges of the skeleton.

Skeleton3D



Skeleton3d



Description：

Extract the skeleton from a 3D mesh using CGAL. Skeletons are effective shape abstractions used in segmentation, shape matching, reconstruction, virtual navigation, etc. As the name implies, a curve skeleton is a graph of curvilinear structures (1D). It is not a medial axis that for a 3D geometry is composed of surfaces (2D). As illustrated in example, the curve skeleton of a shape captures its essential topology.

Input：

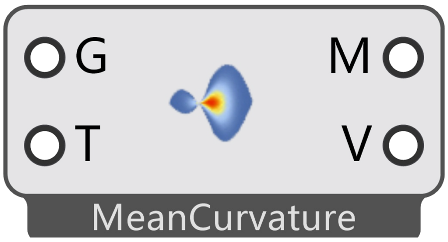
Geometry: Input a Geometry.

Output：

Skeleton: Output all the edges of the skeleton.

Verterbrae: Output skeleton points and the corresponding surface points.

CurvatureAnalysis



CurvatureAnalysis

Description :

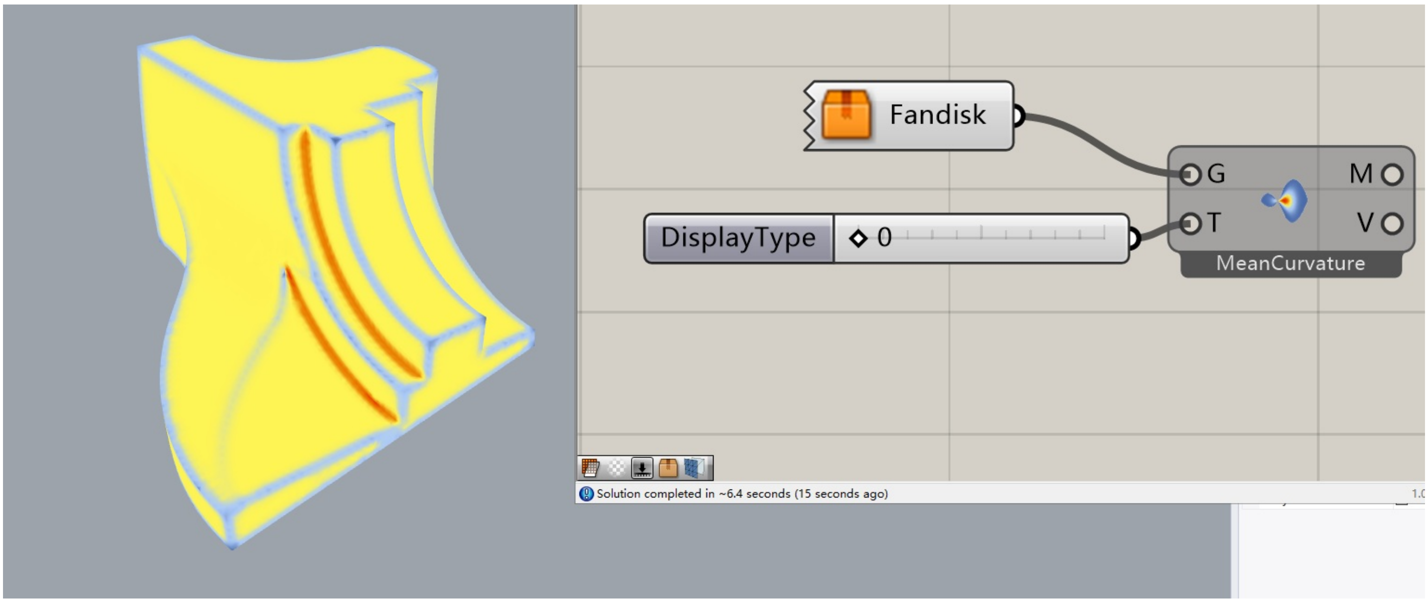
Compute the curvature value of per-vertex (Mean,Gaussian,MaxAbsCurvature).

Input :

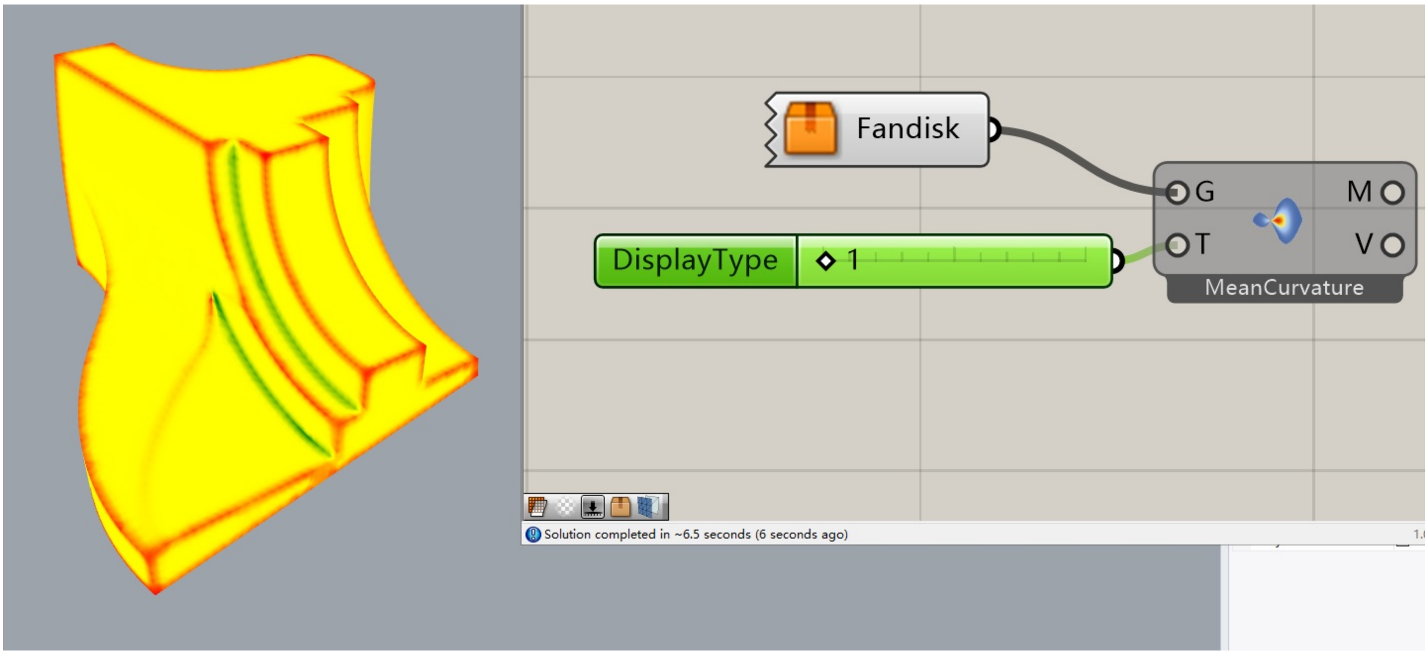
Geometry: Input a Geometry.

DisplayType: Display Type. 0 -> HeatMapper, 1 -> TrafficMapper, 2 -> GreyMapper.

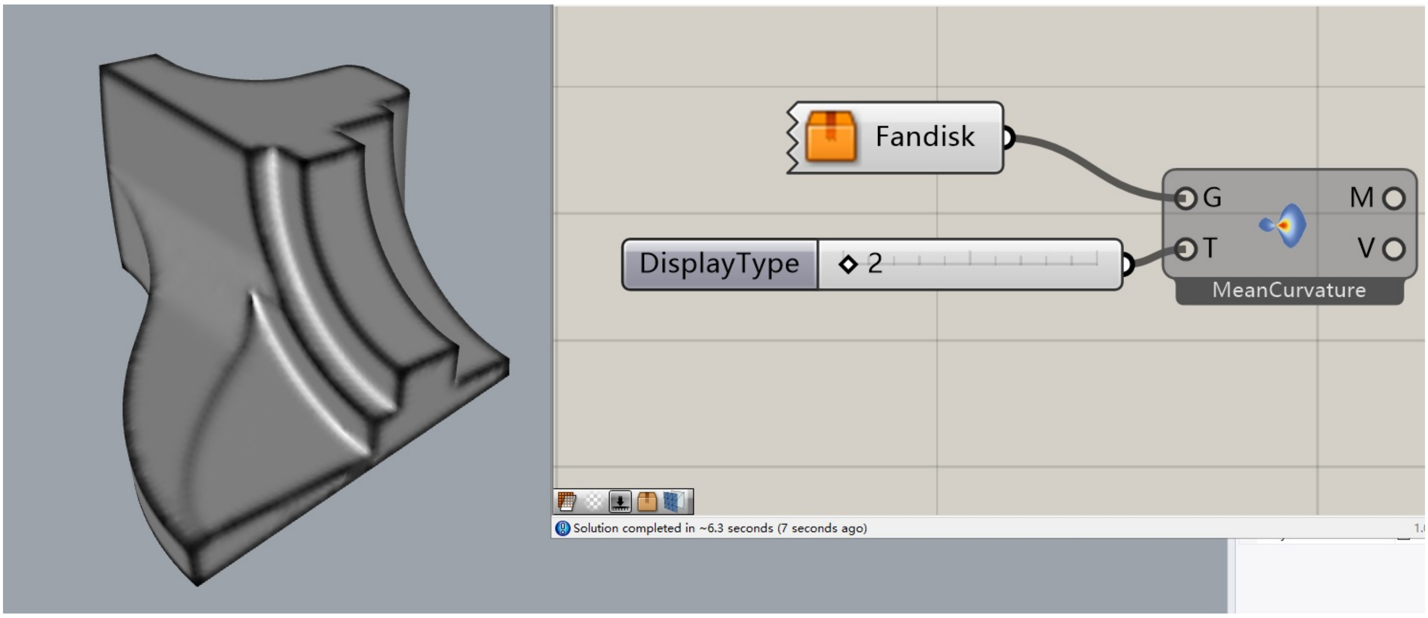
HeatMapper-Type = 0



TrafficMapper-Type = 1



GreyMapper-Type = 2



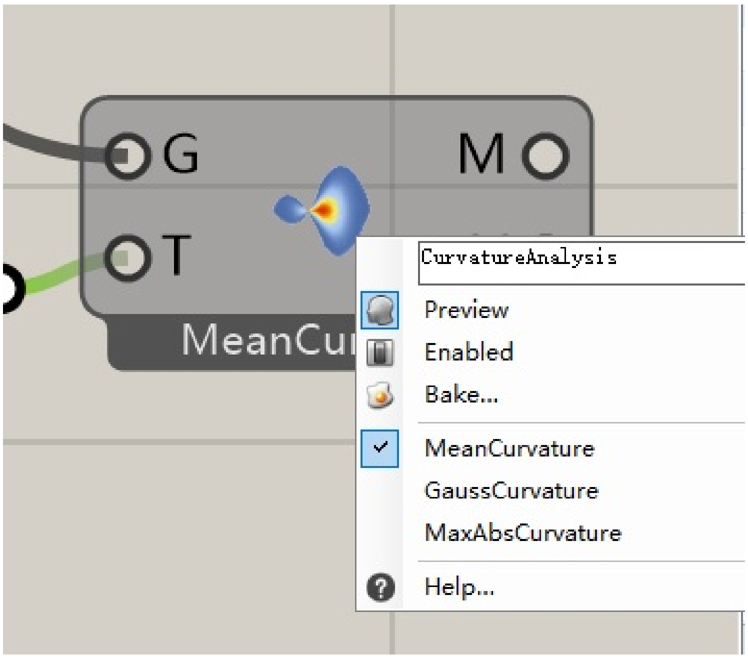
Output：

Mesh: Output the mesh.

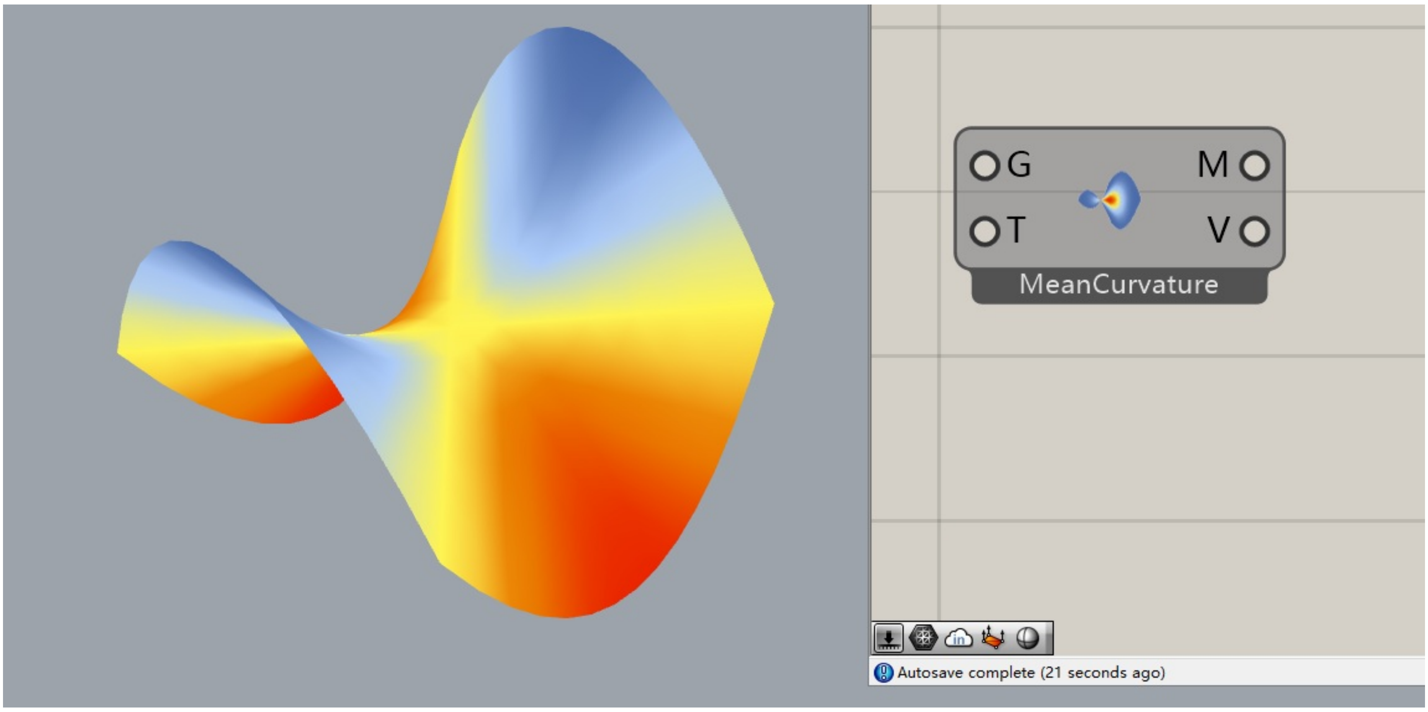
Value: Output the curvature value of pre-vertex.

Options:

CurvatureAnalysis

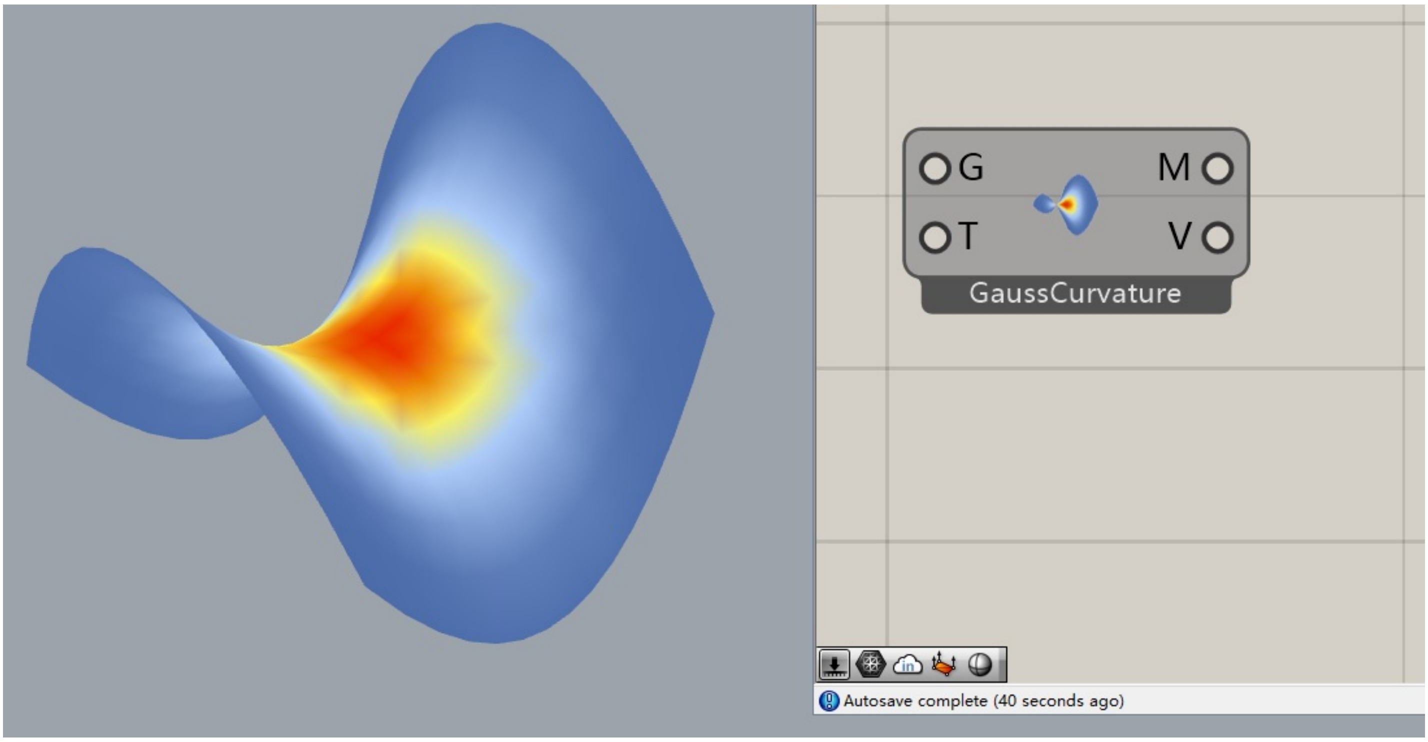


MeanCurvature

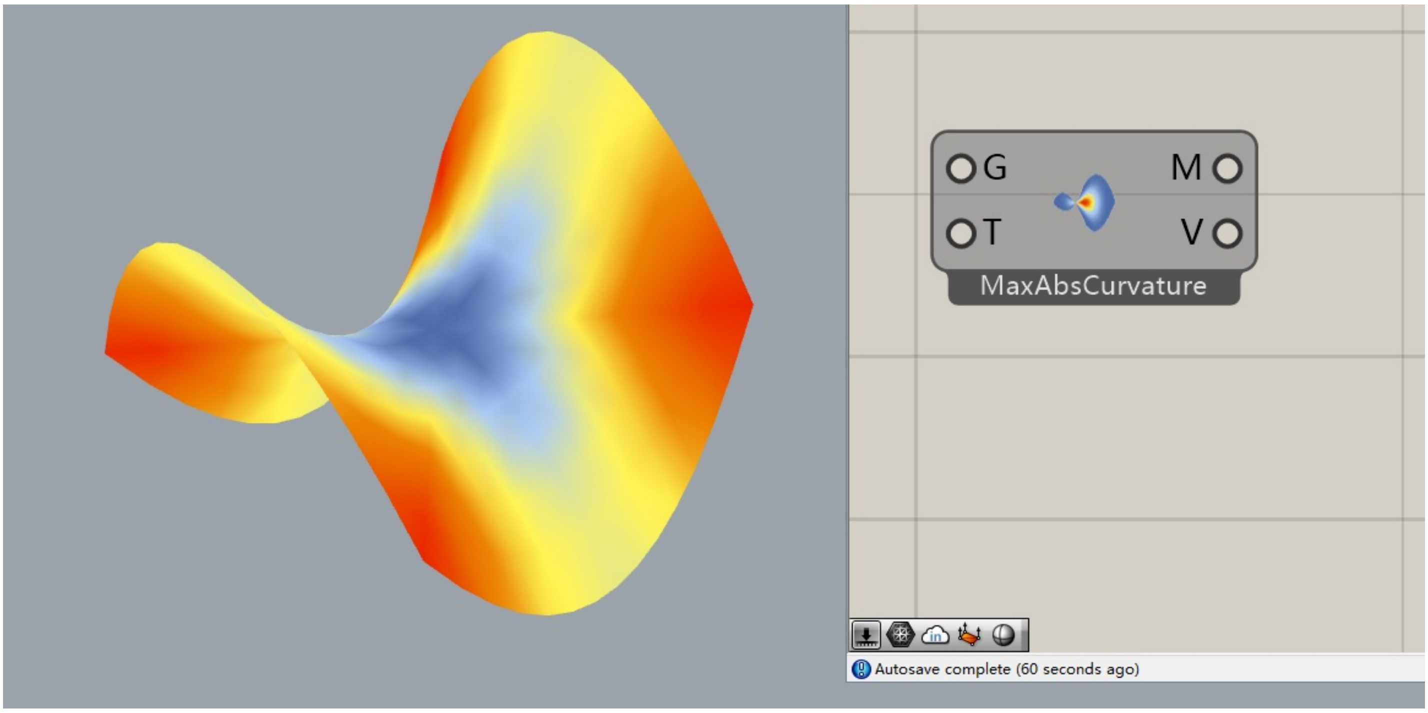


GaussCurvature

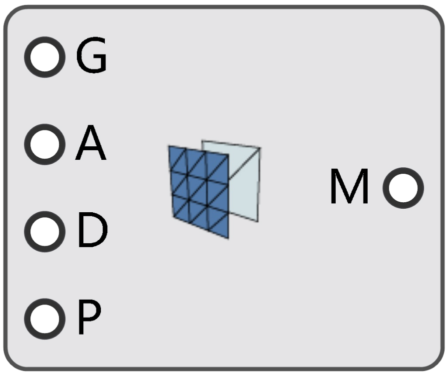
CurvatureAnalysis



MaxAbsCurvature



Simplification



Simplification

Description：

Surface mesh simplification is the process of reducing the number of faces used in a surface mesh while keeping the overall shape, volume and boundaries preserved as much as possible. It is the opposite of subdivision.

Input：

Geometry: Input a Geometry.

Aspect: Aspect Ratio.

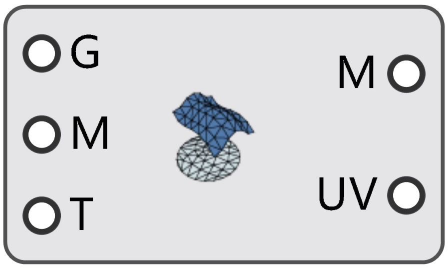
NormalDeviation: Normal Deviation.

TargetPercentage: Target Percentage (0 - 1).

Output：

Mesh: Output the mesh.

Parameterization



Parameterization

Description：
Parameterizing a surface amounts to finding a one-to-one mapping from a suitable domain to the surface. A good mapping is the one which minimizes either angle distortions (conformal parameterization) or area distortions (equiareal parameterization) in some sense. In this component, we focus on parameterizing triangulated surfaces which are homeomorphic to a disk or a sphere, and on piecewise linear mappings onto a planar domain.
Although the main motivation behind the first parameterization methods was the application to texture mapping, it is now frequently used for mapping more sophisticated modulation signals (such as normal, transparency, reflection or light modulation maps), fitting scattered data, re-parameterizing spline surfaces, repairing CAD models, approximating surfaces and remeshing.
Input：

- Geometry: Input a Geometry.
- Method: Parameterization methods.
  '0' = Tutte Barycentric Mapping;
  '1' = Discrete Authalic Parameterization;
  '2' = Discrete Conformal Map;
  '3' = Floater Mean Value Coordinates;
  '4' = Least Squares Conformal Maps;
  '5' = As Rigid As Possible Parameterization.
  If you wanna learn more about these algorithms please click CGAL_Parameterization
- BorderType：Border Types.
  '0' = Circular_border_arc_length: It is able to parameterize the border of a 3D surface onto a circle, with an arc-length parameterization: the (u,v) values are proportional to the length of border edges;
  '1' = Circular_border_uniform: It is able to parameterize the border of a 3D surface onto a circle in a uniform manner: points are equally spaced;

'2' = Square_border_arc_length: It is able to parameterize the border of a 3D surface onto a square, with an arc-length parameterization: (u,v) values are proportional to the length of border edges;
'3' = Square_border_uniform: It is able to parameterize the border of a 3D surface onto a square in a uniform manner: points are equally spaced."

Output：

Mesh: Output the mesh.

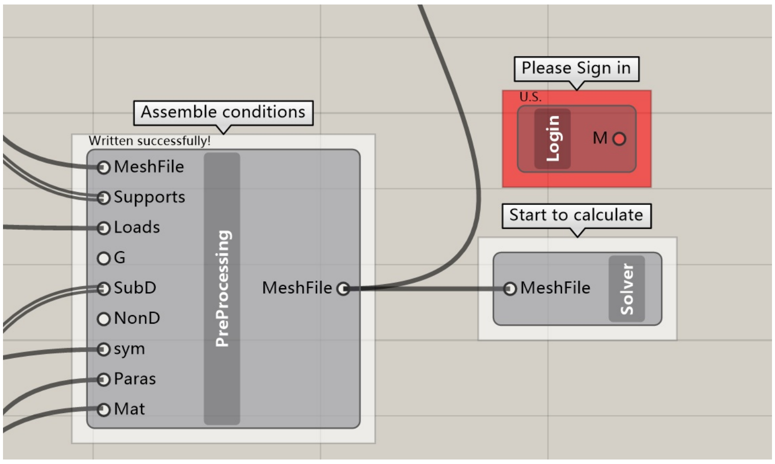
UVPosition: Output UV Positions.

Solver Window

Preparatory Work

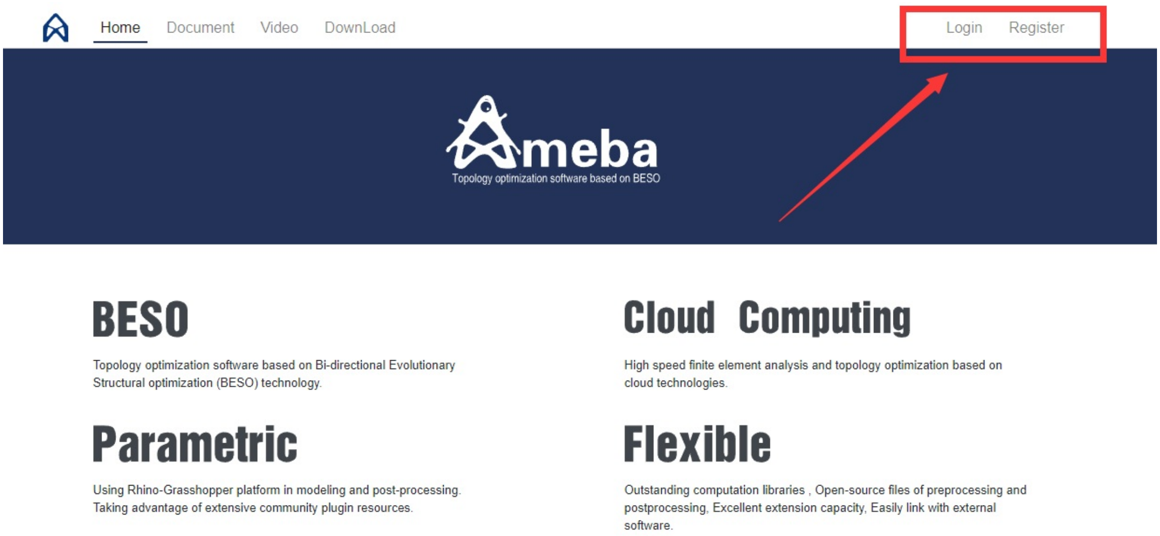
First of all, you are supposed to insure PreProcessing component has revealed "Written successfully!" .

Then users need to verify the license status by using Login component.



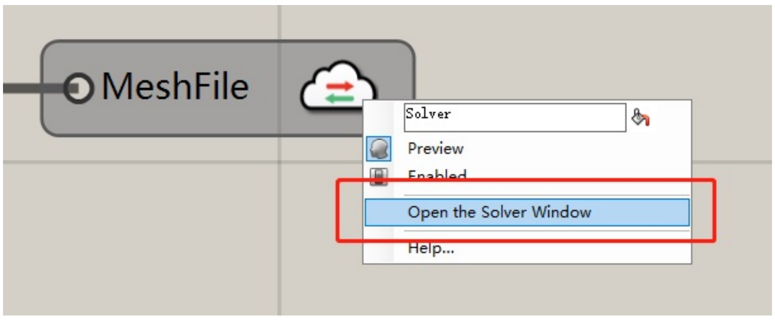
Notes:

- Users can register our account on ameba.xieym.com and learn about our license information.

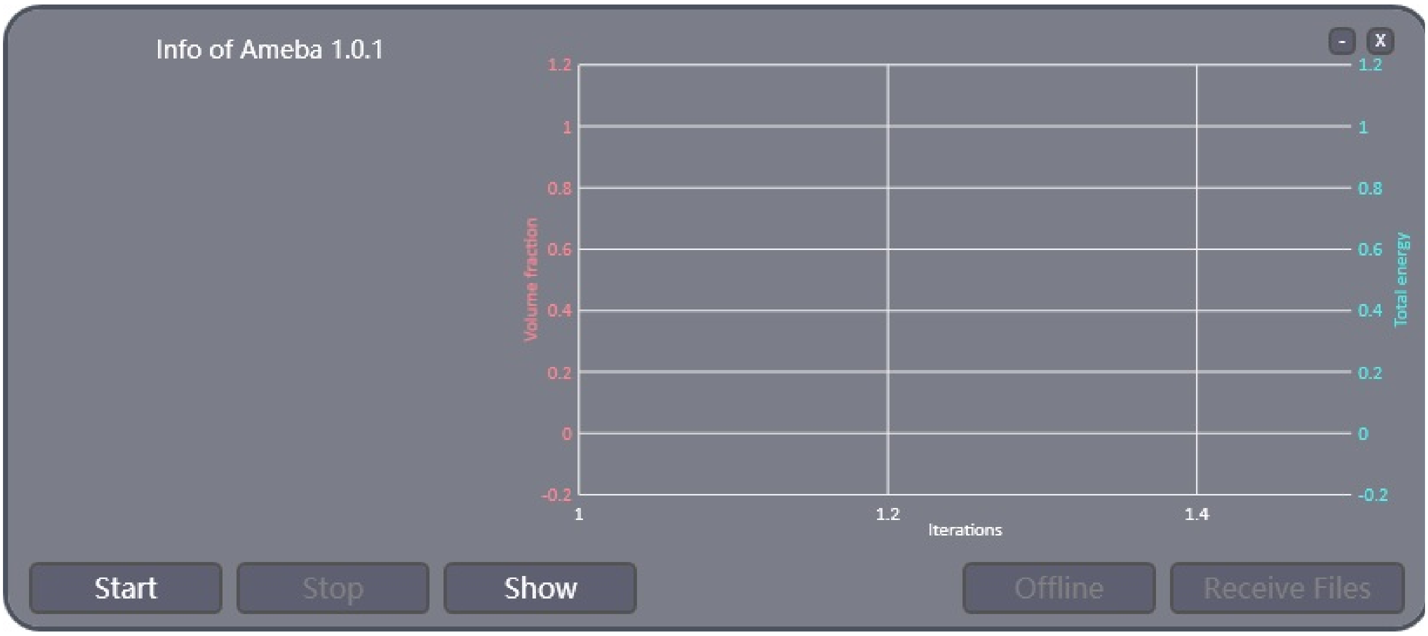


- The login information will be automatically reserved for 24h. You will need to log in again every time if you restart Rhinoceros (but you do not need to enter an account or password).
- Unless you manually click Stop to stop the calculation, the calculation will keep run whether the login expiration period expires or Rhinoceros is closed.

How to start it



The one way is right-click Solve component and select "Open the Solve Window". You can also double-click this component. After operations, Solver Window will be opened.

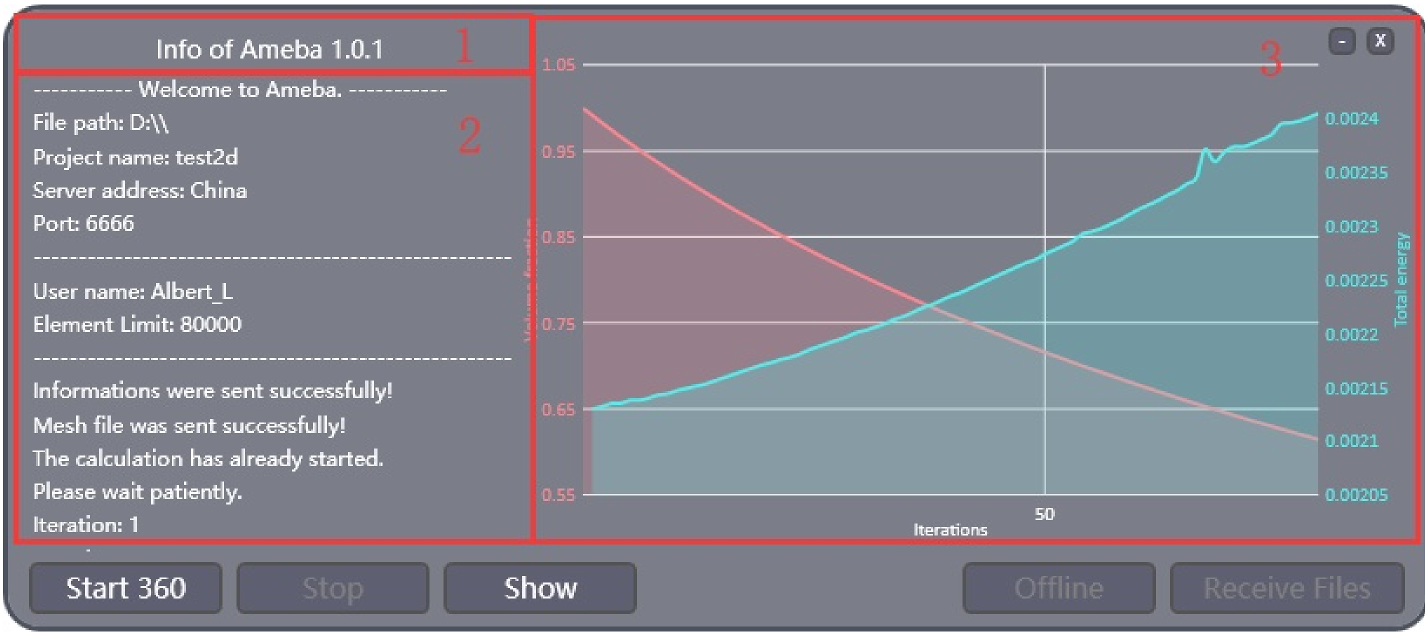


Then, you are supposed to select a server, China Shanghai or US.Virginia, and click Start button. Please wait a moment until the Solver Information Panel displays "The calculation has started. Please wait patiently." Next, you can have a cup of coffee and wait about ten minutes(depends on your mesh model, load case and internet speed). After completion, you can review the result model using Display component.

Window Introduction

The Solve Window

1. Title: The title of the information panel. You can get the version number from it.
2. Solver Information Panel: It will display some informations for account, the file transfer status, calculation process and error reporting.
3. Broken Line Graph: A broken line graph which is about the Volume fraction,Total energy and Iterations. The Volume fraction will be reduced to your appointed value according to OptParameters component. The Total energy means the gross strain energy. The more stable this parameter is, the easier calculation result converges.

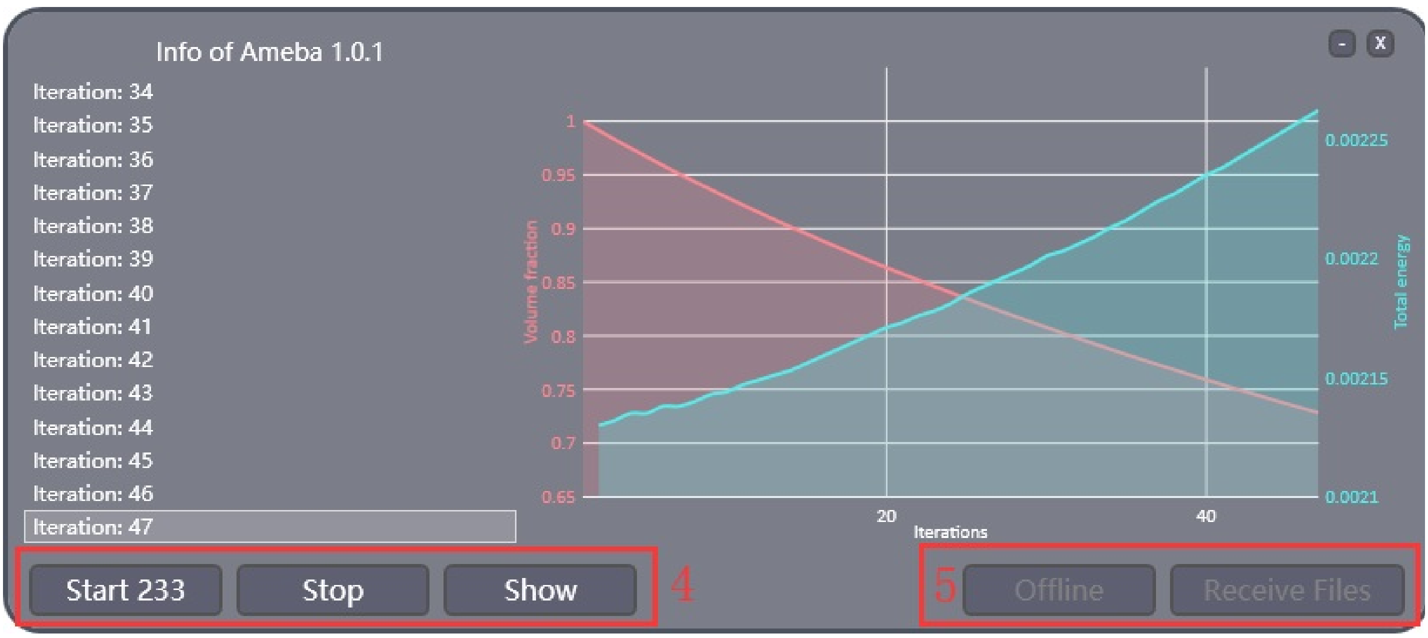


4. Buttons:

- Start: Start to solving. Once the calculation is started, a counter will appear to monitor if the server connection is normal. If the counter stoped for a long time, there is a problem with the server connection.
- Stop: Shut down the solving process.
- Show: If you click this button, it will turn into "Auto" and show solving process model in real-time.

5. Offline Mode (Under Development):

- Offline: Start to calculate offline. Even if your computer has shut down, solving process is still continue. It is being developed at the moment.
- Receive Files: This function is used to receive the returned files in offline mode. It is being developed at the moment.



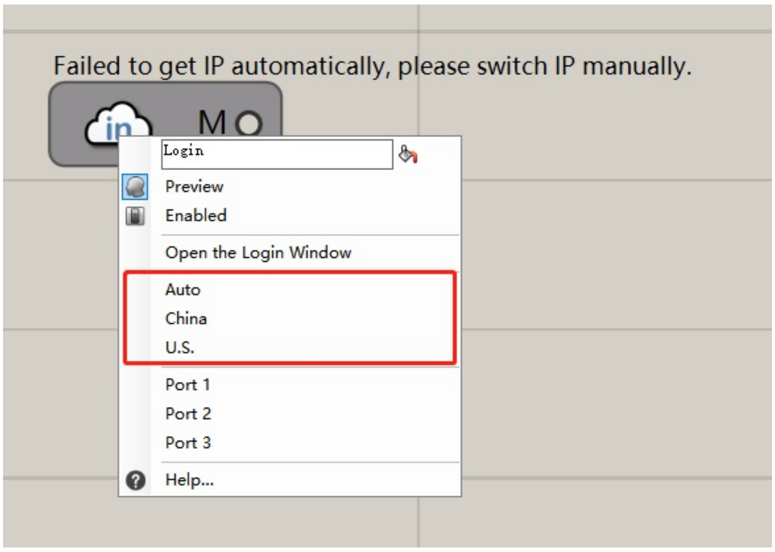
Error Report and Solution

Error Report and Solution

Login

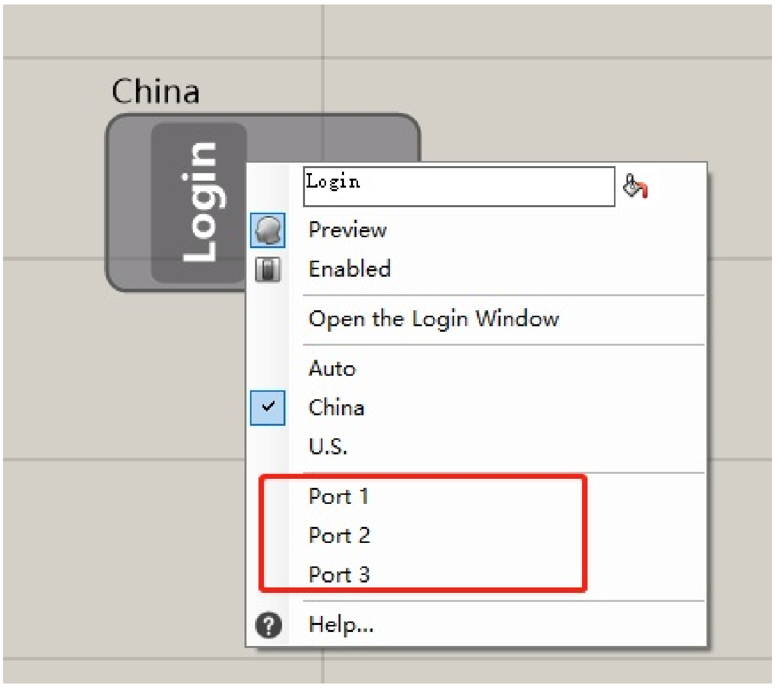
- Ameba cloud computing needs to be hosted on cloud servers, so users must log in before computing. When communicating with the cloud server, the user's IP address must be obtained and assigned to the corresponding server according to the user's local area.

Failed to get IP automatically, please swith IP manually.



- In general, getting the user's IP address is automatic, but something might prevent it. Users need to right-click the Login component and manually select a server (China or U.S.).

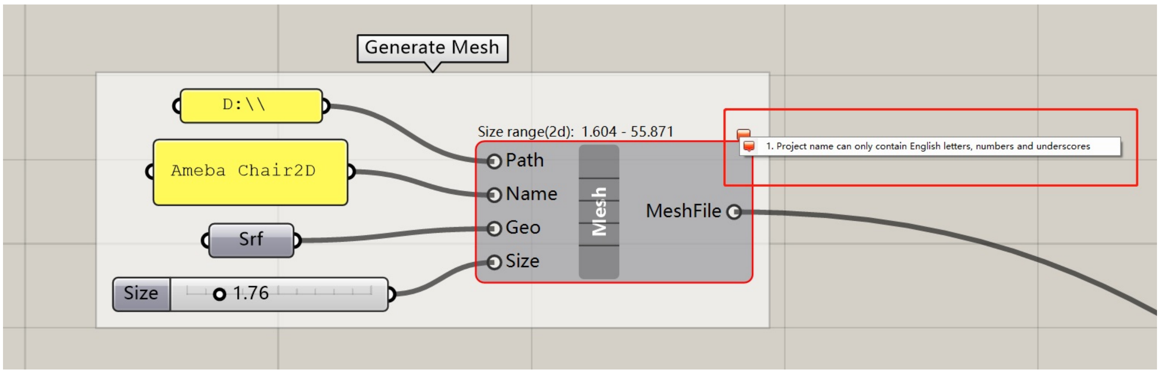
About Network connection problems (Could cause Rhinoceros to crash)



- Since some agencies or organizations may block Internet by blocking ports. In this case, users have to switch ports manually. If you still cannot address this problem, please let us know (Leave message or send an email to ameba@xieym.com). When your Rhinoceros crashes and you find that a "RhinoDotNetCrash.txt" file has been generated on your desktop, please first try to update .Net framework to version 4.5.2 or later and then try login again. If it still doesn't work, please send the "RhinoDotNetCrash.txt" file to us via email.

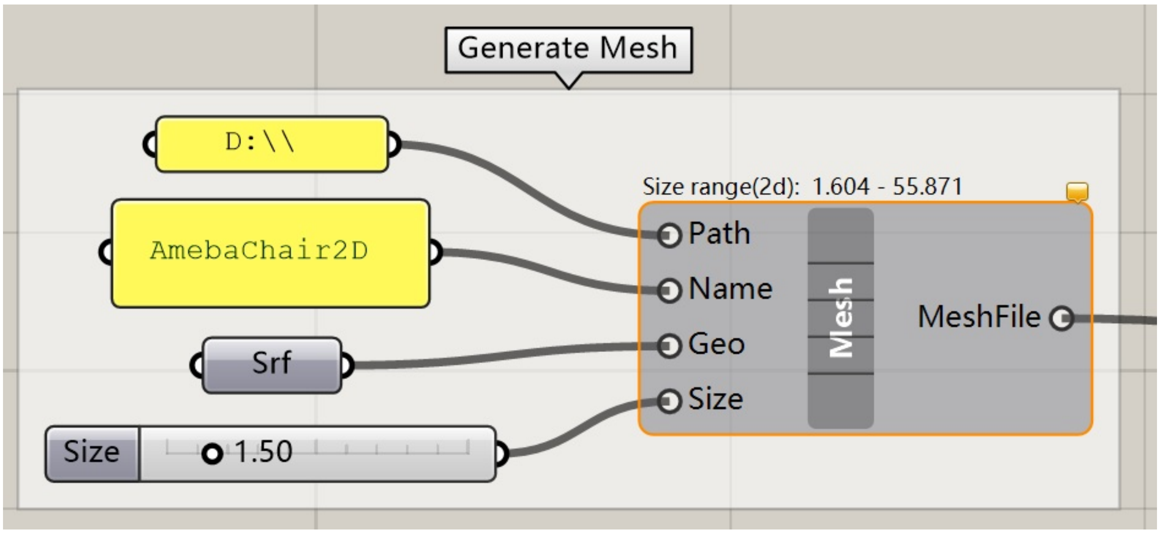
AmebaMesh

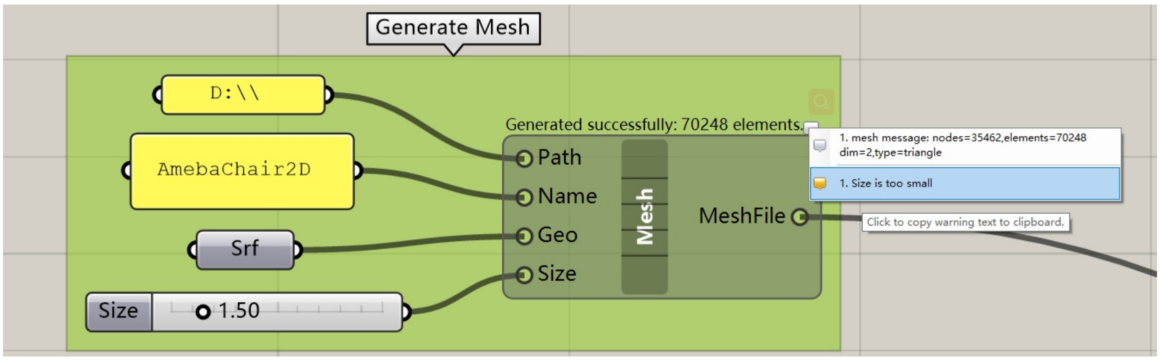
"Project name can only contain English letters, numbers and underscores"



- It means only English letters, numbers and underscores can be allowed as a project's name. Therefore, please check your panel component carefully.

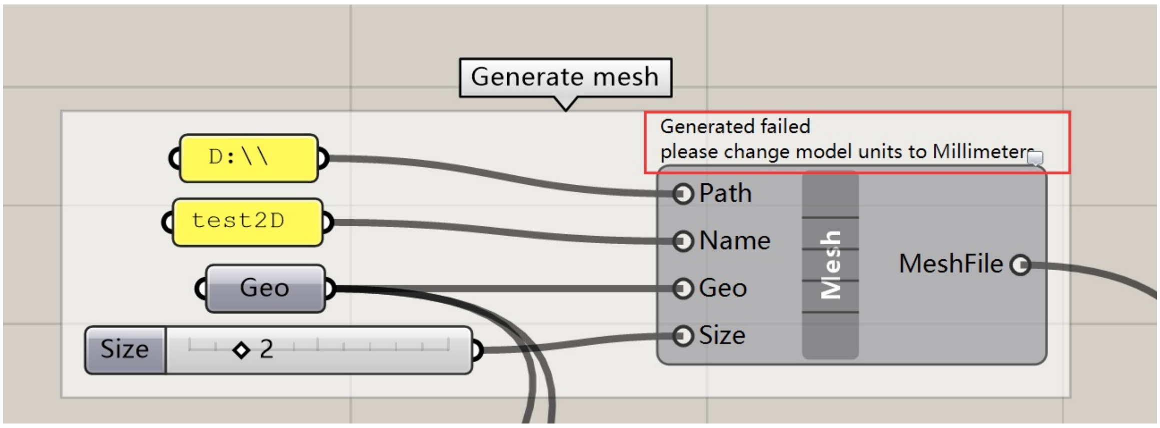
"Size is too small"





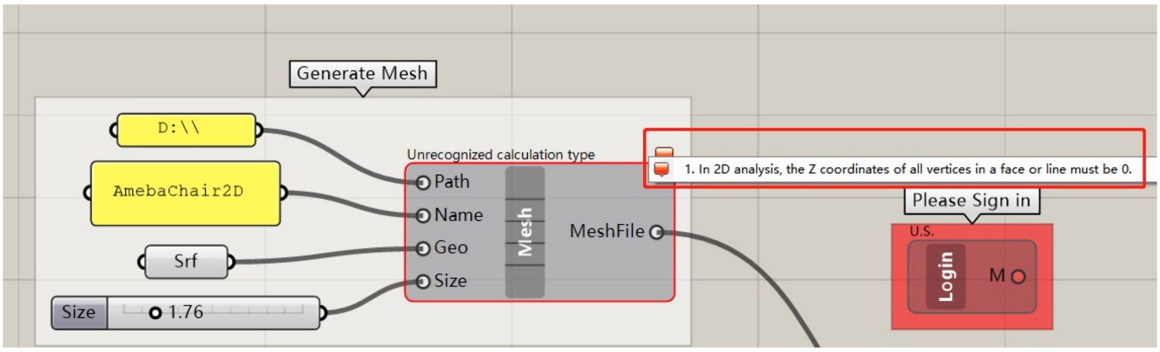
- Before genertating meshes, a recommanded range of mesh size is displayed above the component. If you input a value that is less than the recommended size range, this component will issue a warning and it is possible that the size of the meshes will cause your project to enter a large scale computing cloud server that is only available to users who have pro licenses.

"Generated failed, please change model units to Millimeters"

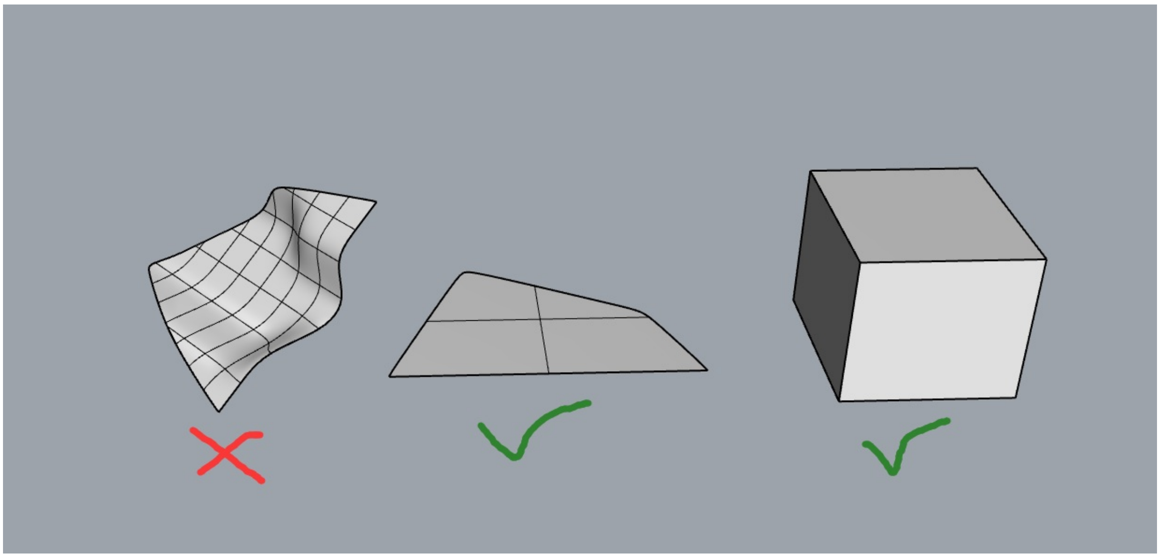


- Before topology optimization, mesh subdivision of the model is required. Gmsh is used as the subdivision program. It have to read user's model units. Sometimes Gmsh can cause mesh failure if the model size is too small or the model units are too large. Therefore, the solution is to enlarge the model or change the model units (Millimeters is recommended).

"Unrecognized calculation type"

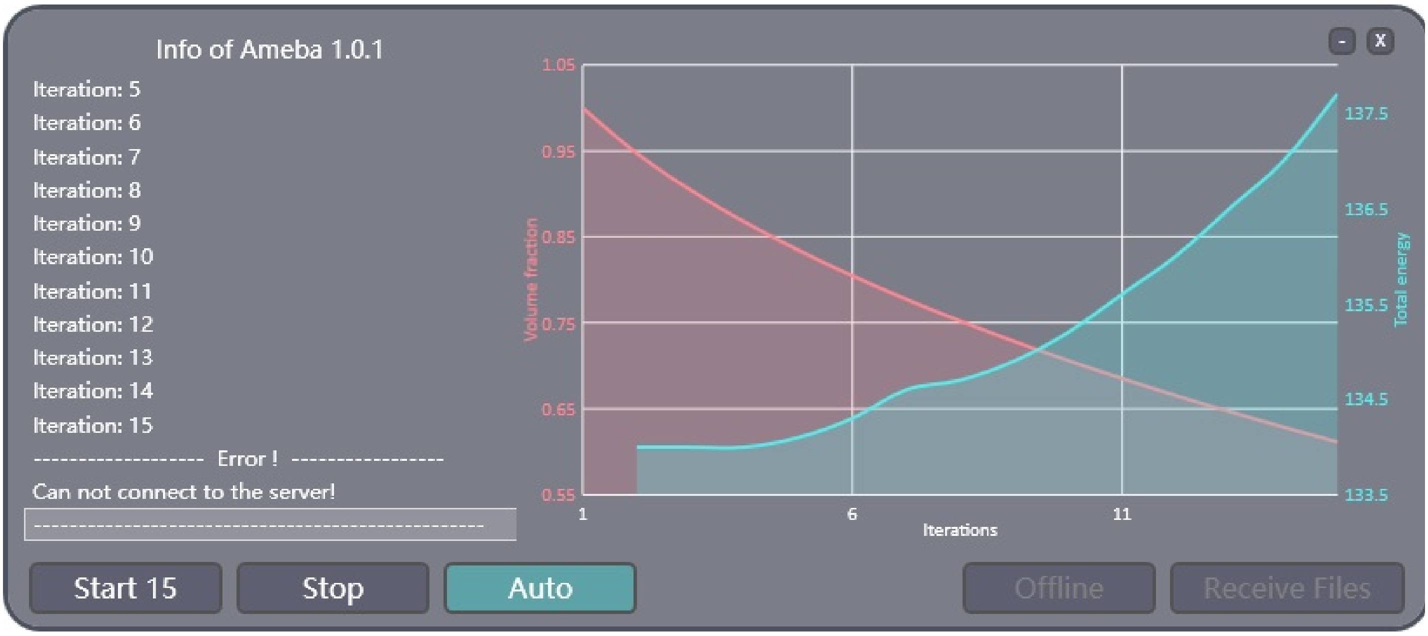


- At present, Only 3d elements (closed brep) and 2d elements (planar surface on the xy plane) can be supported. The calculation of arbitrary surfaces (Shell elements) is being developing. Since the 3d meshes are finite elements (tetrahedral meshes), users are not allowed to conduct mesh division by themseleves.

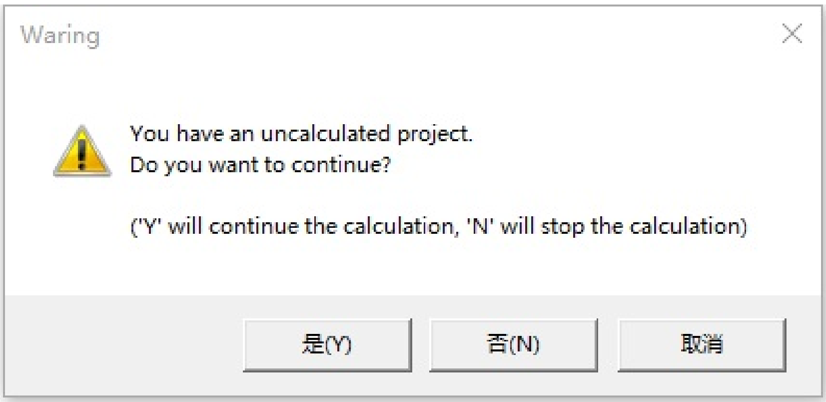


Solver Window

Can not connect to the server! ("Impossible to read over the end of the flow")



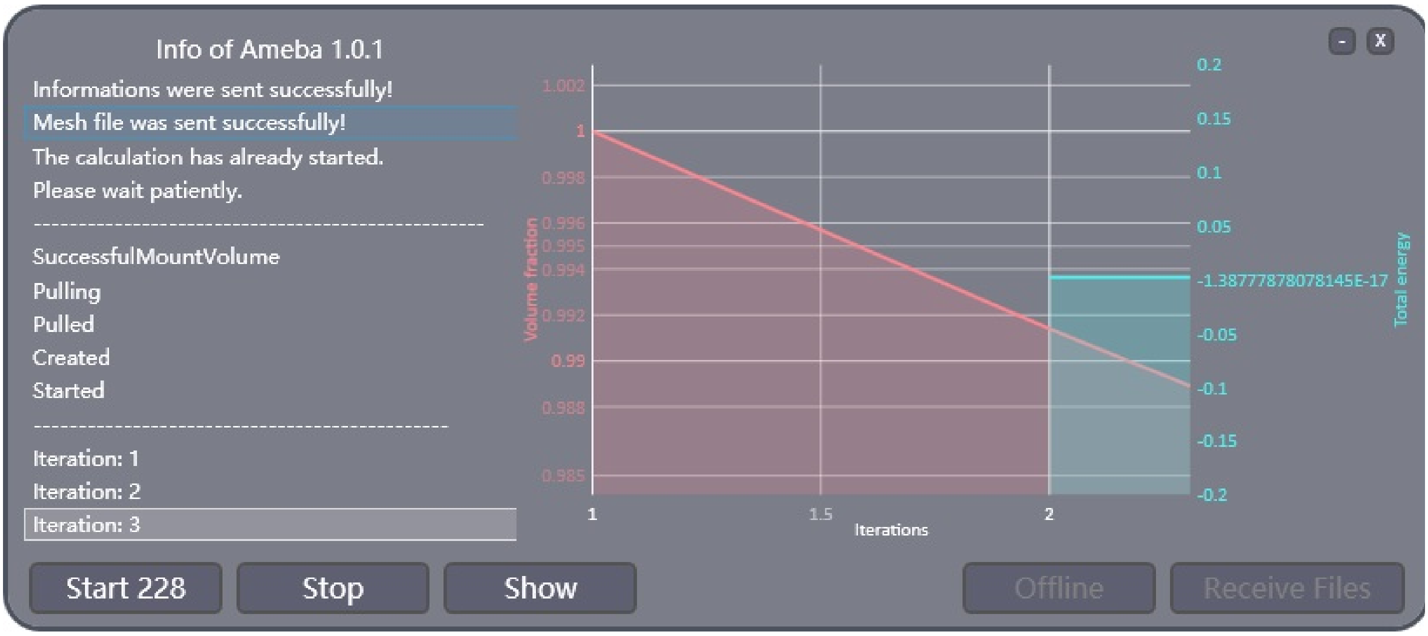
- If your network is disconnected suddenly, the Solver Window will show this report. Don't worry! Just click Start again, you can continue the disconnected calculation.



About Normal Computing

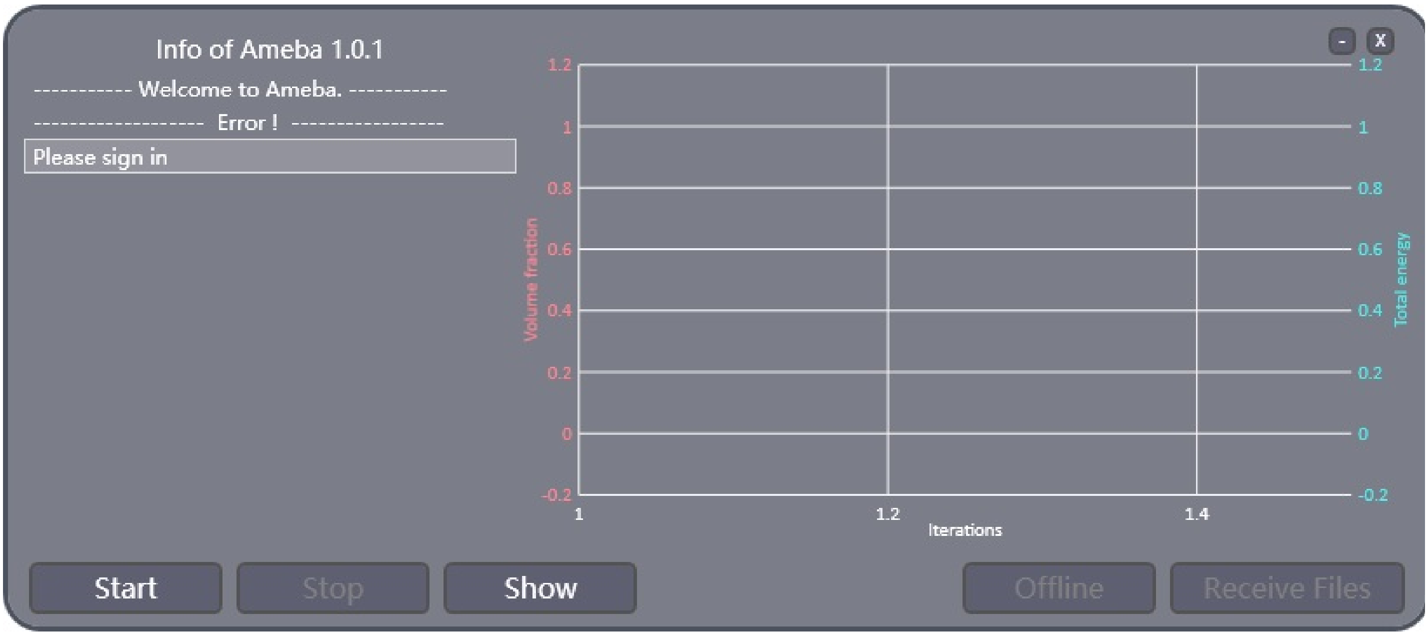
- The computation time of topology optimization depends on the computation amount. The computation amount is determined by the type of element (2D or 3D), the number of elements, and the complexity of the input conditions (load, support, etc.). So it could take a minute or it could take many hours.
- Generally speaking,** the precision of topology optimization is affected by the number of elements**. The denser the elements, the richer the details of the model.

About Large-scale Computing



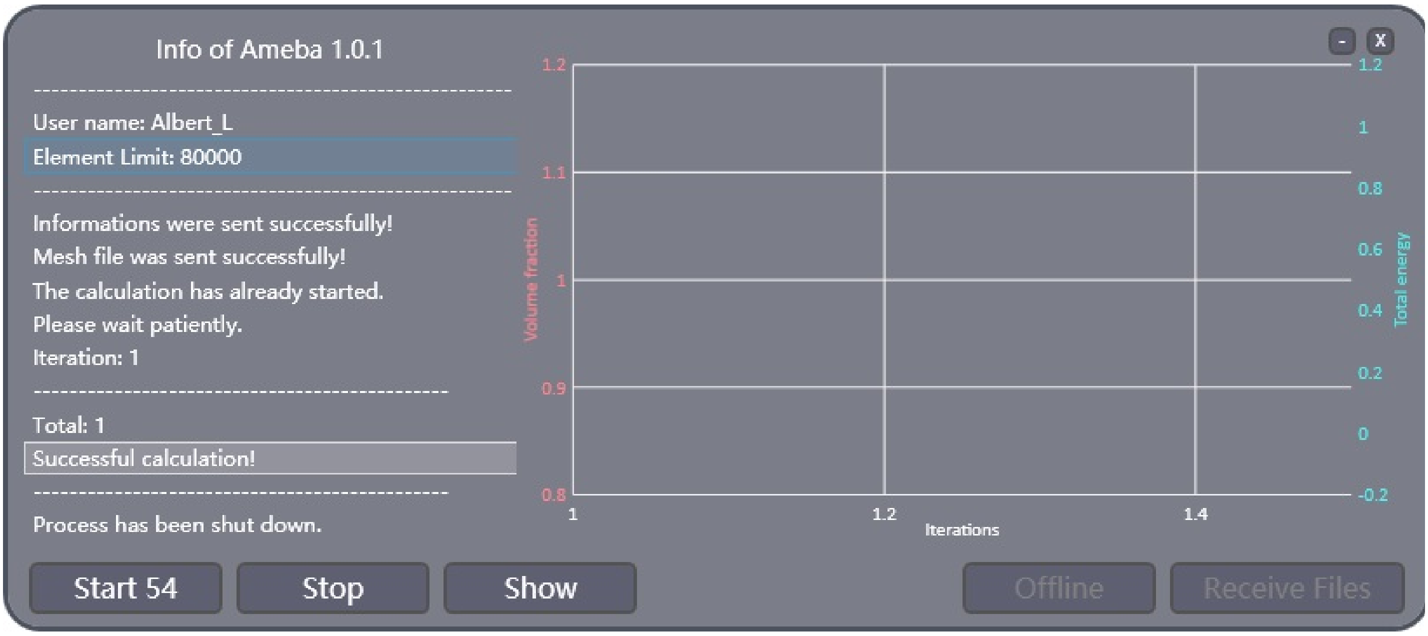
- When you have too many elements, the server automatically assigns your peoject to a cloud server that supports large-scale computing. This process will take some time, so please be patient.

"Please sign in"



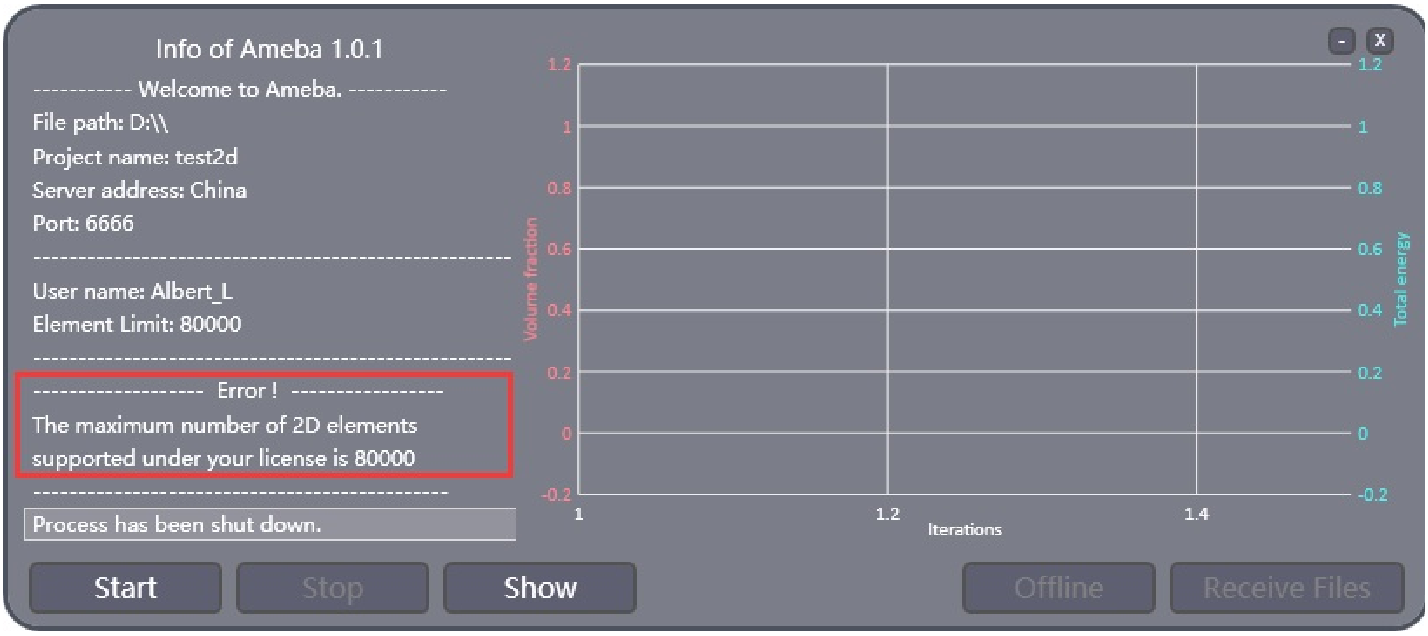
- Users have to sign in to verify the license status by using Login component. before start to calculate.

"Computing condition error"



- If the result is "Computing condition error", a error report file, called "Ameba.log", will be generated. Please send this file to our email (ameba@xieym.com) or copy the content of report file and leave message on here

"The maximum number of 2D (or 3D) elements supported under your license is"



- The count of topology optimization elements is limited according to license status. If you want to calculate more elements, please purchase Ameba license via our website.

"Out of memory."

- It means your model and conditions are too complex. You need to decrease your mesh count or simplify your case of loads or supports. If you still cannot address this problem, please let us know

(Leave message or send an email to ameba@xieym.com).

Logs

2018.03.31 Ameba 0.6.0 released.

2018.03.31 Ameba 0.6.0 user manual(Chinese version) released

2018.04.16 Ameba 0.6.0 user manual(English version) released

2018.04.23 Ameba 0.6.1 released:

1. Solved recieving interruptions;
2. optimized the speed of recieving;
3. optimized the speed of 【Display】component.

2018.06.22 Ameba 0.6.2 released:

1. Mises stress and principal stress cloud image display function;
2. Add a US server;
3. Add surface Load and Support;
4. The calculation of the component is no longer affect main interface.

2018.07.27 Ameba 0.6.3 released:

1. Add multi-case optimization function;
2. Enable visualize and return the calculated information and results in real time;
3. Optimized the calculation speed.

2018.08.30 Ameba 0.6.4 released:

1. New component display style;
2. Add RenderDisplay, AmebaSmooth,Rebuilding2D and Remeshing3D components;
3. Calculation can support arbitrary load area;
4. Calculation can support any non-designed area;
5. Adopt more resource-saving support and load display methods.