

Liquid Grid

This page provides information on the Grid rollout for a LiquidSim object.

Overview

The Liquid Grid rollout lets you set the boundaries, size and resolution quality of the sim. Regarding size, the Phoenix Simulator works best when the scale of the container matches the real-world size of the simulated effect. For example, if you are simulating water pouring into a glass cup, the cup should be in a typical size range, such as 9-12cm tall and 8-9cm wide, while your container grid might be a bit larger.

Note that it does not matter if the units are displayed as two meters or two thousand millimeters - the way you view the units is irrelevant. Phoenix always converts the units to a common world-size length, so the only important thing is the size of the container. If you are simulating a large waterfall, for example, the container might be up to several hundred meters tall, or several hundred thousand millimeters.

If your scene is structured in a way that makes it hard for you to scale the objects to their real-world size, you can use the **Scene Scale** parameter to tell Phoenix to treat the container as larger / smaller than it actually is when measured in Scene Units. This will influence the dynamics of the simulation, allowing to you achieve the correct behavior for your simulation without the need to tweak the size of the objects in your scene.

Using the parameters on this rollout you can:

- Specify the **Size** and **Resolution** of the Simulator.
- Enable / Disable **Adaptive Grid** which is a performance optimization allowing you to keep the size of the Simulator as small as possible thus reducing RAM usage.
- Specify which **Walls** of the Simulator will be considered **Open** (infinite) or **Jammed** (i.e. as solid obstacles, closed).
- Link multiple Simulators in a **Cascade** setup - for more information, please check the following article: [Transferring fluid between Simulators using a Cascade Connection](#).
- Specify a **Confine Geometry** to limit the fluid calculations only to the volume of the specified object.

UI Path: ||Select Liquid Simulator object|| > **Modify panel > Grid rollout**

Parameters

General

Expand – Opens a floating dialog that contains the selected rollout and automatically folds the command panel rollout.

Re-Center – Resets the position of the floating rollout.

? – Opens up the help documents for the Liquid Grid.

Scene Scale | *scenescale* – Specifies a multiplier for the original units of the scene. Phoenix works best when the container size is close to the real-world size of the desired effect. You can use this parameter to make the fluid solver see the container as bigger or smaller than it actually is in the scene, in case you cannot change the general scene units. Check the sizes shown in the Total Cells field - this is the size the fluid solver will use. These sizes will change as you change the **Scene Scale** and should be close to the real world size of the effect you are simulating. It does not matter how you view the units - in meters, centimeters, inches, etc. For example, a candle simulation can be 20 cm tall, or 0.2 m tall - it's the same.

Larger scale would make the fluid move more slowly because it needs to travel a greater distance, while smaller scale makes the fluid move faster and more chaotic. Compared to the effect of the **Time Scale** option in the [Dynamics rollout](#), **Time Scale** makes everything slower and things continue to work in the same way (except that more simulation steps will inevitably make the fluid dissipate some velocity and detail). **Scene Scale** will affect some simulation parameters like gravity, cooling, burning, surface tension, the rate of birth of splashes and mist, and make them behave like this is a much larger or smaller effect. *For more information on how changing the Scene Scale affects the simulation, see the Scene Scale example below.*

Voxel Size | *cellsz* – The size of a single voxel, in scene units. *For more information, see the [Grid Resolution](#) example below.*

X, Y, Z | *xc, yc, zc* – The grid size in cells. The dimensions shown next to **X, Y, Z** are the grid sizes in the scene, multiplied by the **Scene Scale** parameter - these sizes show how the solver will see the grid box and you can use the **Scene Scale** to cheat the solver into simulating as if the grid box was larger or smaller. In case you want to see how big the container for the loaded cache is in the scene without accounting for the **Scene Scale**, see the [Container Dimensions](#) in the Simulation rollout.

Increase/Decrease resolution – Changes the resolution of the grid while maintaining its size. *For more information, see the [Grid Resolution](#) example below.*

Expand

Re-Center

?

Scene Scale	0.113	↕
Voxel Size	1.897	↕
X 18.797	88	↕
Y 18.797	88	↕
Z 51.692	241	↕
Total Voxels	1 874 048	

Increase Resolution

Decrease Resolution

Example: Scene Scale

The following video provides examples to show the differences of **Scene Scale** with values of *0.1*, *5.0* and *15.0*.

Software used: Phoenix FD 4.30.01 Nightly (02 Oct 2020)

[Download Example File](#)

Example: Grid Resolution

The following video provides examples to show the differences when the Total cells from the **Grid's Resolution** is at *570,000*, *4,000,000* and *16,000,000*.

Software used: Phoenix FD 4.30.01 Nightly (02 Oct 2020)

[Download Example File](#)

Container Walls

X, Y, Z | x_bound, y_bound, z_bound – Select between different container wall conditions for the simulation grid.

Open – The fluid is allowed to leave the bounding box of the Simulator through this wall. If **Fill Up for Ocean** is enabled, the Wall is treated as if there is infinite liquid below the **Initial Fill Up** level.

Jammed(-) – The simulation behaves as if there is a solid boundary in the negative direction. When **Adaptive** is enabled, the grid will not expand in this direction.

Jammed(+) – The simulation behaves as if there is a solid boundary in the positive direction. When **Adaptive** is enabled, the grid will not expand in this direction.

Jammed Both – The simulation behaves as if there is a solid boundary in both directions. When **Adaptive** is enabled, the grid will not expand in this direction.

Wrap – The left and right boundaries are connected (toroidal topology). E.g. Fluid leaving the Simulator from the +X wall will enter it again from the -X wall.

Confine Geom | *usegridgizmo, gridgizmo* – You can specify a closed geometry object with normals pointing outwards, and the simulation will run only inside this object. The rest of the cells will be frozen as if a solid body was covering them. This way you can fill irregular shapes with liquid, or generally speed up your simulation by chopping off empty cells when you have an irregular fluid shape, e.g. a rocket launch.

While using a **Confine Geometry** can speed up a simulation, it will not reduce RAM usage.

Cascade Simulator | *usecascade, cascade* – Specifies the source [LiquidSim](#) to connect this Simulator to, forming a cascading simulation. This allows you to join several Simulators into a structure with a complex shape. This can help you reduce memory usage by using many smaller Simulators in place of a single large Simulator. For more information, see the [Connecting Two Simulators in a Cascade Setup](#) section on the [Tips and Tricks](#) page.

- In order to use a **Cascade Setup** you must have a fluid which flows from one Simulator into another. For more information, see the Transferring fluid between Simulators using a Cascade Connection section on the Interactions Between Simulators page.
- The Simulators must be run sequentially and each one should be started only after the previous one has finished simulating. The **Cascade Simulator** parameter points to the previous Simulator in the sequence.
- For the simulation to function correctly, you need to have the **Velocity Grid Channel** and all **Particle Groups** that are simulated in the **Source Simulator** exported to its cache files - otherwise the connection will not work properly.
- If you intend to use any additional channels such as **RGB**, particle IDs or **Age**s, etc., **they also need to be exported** from the Source Simulator's Output rollout before running the current Simulator.

Container Walls

X Open ▼

Y Open ▼

Z Open ▼

☐ Confine Geom None

☐ Cascade Simulator None

Adaptive Grid

Either keep **Adaptive Grid** disabled or set the Container Walls: Z to **Jammed Both** when simulating Oceans. The **Ocean Level** parameter in the Rendering rollout depends on the vertical size of your Simulator.

Adaptive Grid | *adaptive* – The grid will resize automatically during the simulation in order to prevent the liquid from leaving the bounds of the Simulator box. Note that only the **Open Container Walls** will expand and contract when Adaptive Grid is enabled.

Adapt By | *adaptby* – Specifies the particle system that will be used for the grid adaptation. More than one particle system can be used by using a comma to separate the different systems. For example **Liquid,Foam** will resize the grid based on the movement of both the liquid and the foam particle systems. Available systems are *Liquid, Foam, Splash, Mist, Wetmap*. Make sure to enable Particle Velocity from the [Output rollout](#) for all particle systems you want to use for the adaptive grid.

Extra Margin | *adapt_margin* – Specifies the number of cells between the end of the grid and the active zone. You can use this to give the fluid a bit more room if the adaptive grid can't keep up with the simulation.

No Smaller Than Initial Grid | *nbigrid* – When enabled, the **Adaptive Grid** can't contract to a smaller size than what is given as the initial X,Y,Z size for the Simulator. Note that this way the initial grid box is always included, even if the fluid has moved farther from it. If this option is disabled, the grid will always encompass only the active fluid and will move together with it if needed.

Expand and Don't Shrink | *onlyexpand* – When enabled the **Adaptive Grid** will expand without shrinking.

Maximum expansion | *maxexp, expx/y/z/neg/pos* – Specifies maximum growth sizes for each side of the grid, in cells. Using this, you can stop the expansion in certain directions.

Shrink to View | *usegridfitcamera, grid_fit_camera* – Species a camera whose frustum will be used to determine the maximum expansion. The Adaptive Grid will not resize beyond the frustum.

When a **Shrink to View Camera** is provided, the **Adaptive Grid** will expand no further than the already specified **Maximum Expansion Limits**.

The image shows a screenshot of the 'Adaptive Grid' settings panel in a software interface. The panel has a dark gray background with white text. At the top, there is a checkbox labeled 'Adaptive Grid' which is checked. Below it, the 'Adapt By' dropdown menu is set to 'Liquid'. The 'Extra Margin' is a numeric input field set to '0'. There are three checkboxes below: 'No Smaller Than Initial Grid' (checked), 'Expand and Don't Shrink' (unchecked), and 'Maximum Expansion' (unchecked). Under 'Maximum Expansion', there are three rows of controls for X, Y, and Z axes. Each row has a '-' sign, a numeric input field set to '100', a '+' sign, and another numeric input field set to '100'. At the bottom, there is a 'Max Cells' section with a 'Shrink to View' checkbox (unchecked) and a dropdown menu set to 'None'.