5.8.6.2 Modeling - Modifiers - Simulate - Ocean Simulation
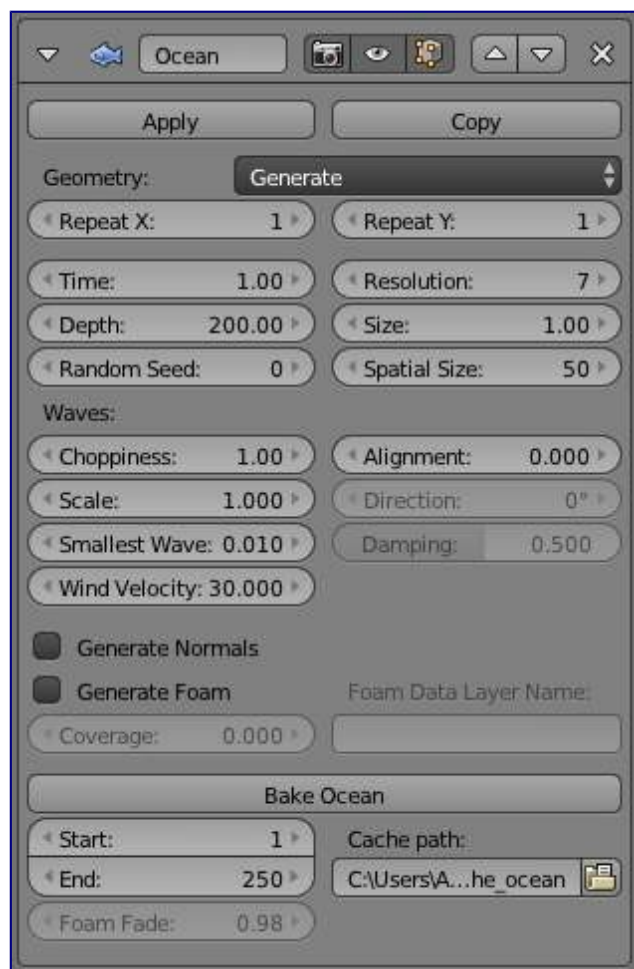
Ocean Simulation

Blender's ocean simulation tools take the form of a modifier, to simulate and generate a deforming ocean surface, and associated texture, used to render the simulation data.

Ported from the open source Houdini Ocean Toolkit, it is intended to simulate deep ocean waves and foam.

Options

Ocean Modifier Panel

Geometry Options

Geometry
 Generate

  Creates a tiled mesh grid that exactly corresponds with the resolution of the simulation data

  When generating a mesh surface, the existing mesh object is completely overridden with the ocean grid. A UV channel is also added, mapping the `0.0 - 1.0` UV space to the simulation grid.

 Displace
  Uses the existing geometry rather than replacing it. Vertices are displaced along the local Z-axis.

Repeat X, Repeat Y
 When generating a mesh surface, controls the number of times the grid is tiled in X and Y directions. UVs for these tiled mesh areas continue outside of the `0.0 - 1.0` UV space.

Simulator Options

Time
 The time at which the ocean surface is being evaluated. To make an animated ocean, you will need to insert keyframes (RMB) and animate this time value - the speed that the time value is changing will determine the speed of the wave animation

Depth

    The constant depth of the ocean floor under the simulated area. Lower values simulate shallower waters by producing higher frequency details and smaller waves.

Random Seed

    A different seed will produce a different simulation result.

Resolution

    The main control of quality vs speed in the simulation engine. This determines the resolution of the internal 2D grids generated by the simulation.

    The internal grids are powers of two of the resolution value, so a resolution value of 16 will create simulation data of size 256 x 256. The higher the resolution, the more detail will be produced, but the slower it will be to calculate.

> **Note**
>
> When using the 'Generate' modifier geometry option, this resolution value also determines the resolution of the generated mesh surface, equal to the resolution of the internal simulation data.

Size

    A simple scaling factor that does not affect the height of the waves or behavior of the simulation.

Spatial Size

    The width of the ocean surface area being simulated, in meters. This also determines the size of the generated mesh, or the displaced area, in Blender units. Of course you can scale the object with ocean modifier in object mode to tweak the apparent size in your scene.

Wave Options

Choppiness

    The choppiness of the wave peaks. With a choppiness of 0, the ocean surface is only displaced up and down in the Z direction, but with higher choppiness, the waves are also displaced laterally in X and Y, to create sharper wave peaks.

Scale

    An overall scale control for the amplitude of the waves. It approximates the height or depth of the waves above or below zero. Rather than just scaling the ocean object in Z, it scales all aspects of the simulation, displacement in X and Y, and corresponding foam and normals too.

Alignment

    The directionality of the wave shapes due to wind. At a value of 0, the wind and waves are randomly, uniformly oriented. With higher Alignment values, the wind is blowing in a more constant direction, making the waves appear more compressed and aligned to a single direction.

Direction

    When using Alignment, the direction in degrees that the waves are aligned to.

Damping

    When using Alignment, amount that inter-reflected waves are damped out. This has the effect of making the wave motion more directional (not just the wave shape). With damping of 0.0, waves are reflected off each other every direction, with damping of 1.0, these inter-reflected waves are damped out, leaving only waves traveling in the direction of the wind.
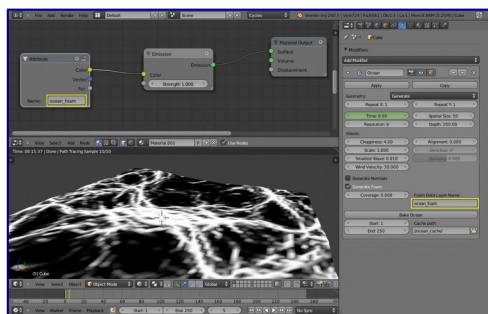
Smallest Wave

    A minimum limit for the size of generated waves. Acts similarly to a low-pass filter, removing higher frequency wave detail.

Wind Velocity

Wind speed in meters/second. With a low velocity, waves are restricted to smaller surface waves.

Simulation Data Generation Options

Using foam vertex colors with a named data layer

By default, the simulator only generates displacement data, since it takes the least amount of work and gives the fastest feedback. Additional simulation data can be generated for rendering as well.

Generate Normals
> Simulates additional normal map data. This can be used by the Ocean texture, when mapped to Normals, as a bump map, and enables generating normal map image sequences when baking.

Generate Foam
> Simulates additional foam data. This can be retrieved by the Ocean texture for use in texturing (perhaps as a mask), and enables generating foam map image sequences when baking.

Coverage
> Tweaks the amount of foam covering the waves, negative values will reduce the amount of foam (leaving only the topmost peaks), positive values will add it. Typically ranges from `-1.0` to `1.0`

Foam Data Layer Name
> Optional name for the vertex data layer, used by the Ocean modifier to store foam maps as vertex colors. This is required for accessing the foam data in the renderer.

Baking

Rather than simulating the ocean data live, the ocean data can be baked to disk. When a simulation is baked, the simulator engine is completely bypassed, and the modifier/texture retrieves all information from the baked files.

Baking can be advantageous for a few reasons:

- It's faster to use the stored data rather than re-calculating it
- Allows rendering ocean data in external renderers
- Enables more advanced foam maps

Data Files

Simulation data is stored in disk as sequences of OpenEXR image maps, one for each of displacement, normal and foam (if enabled to be generated). Upon loading the data from these baked files, when a frame of the bake sequence is read from disk, it is cached in memory. This means that accessing loaded frames subsequent times is fast, not incurring the overhead of disk access.

Since these baked files are plain OpenEXRs, they can also be opened and rendered in any other application or renderer that supports them.

Baking Foam

Baking also provides improved foam capabilities. When simulating live, the ocean simulator retrieves data for that current frame only. In the case of the foam map, this represents the tips of wave crests for that given frame. In reality, after foam is created by wave interactions, it remains sitting on the top of the wave surface for a while, as it dissipates. With baking, it's possible to approximate that behaviour, by accumulating foam from previous frames, leaving it remaining on the surface.

https://vimeo.com/17517981

Baking Options

Start, End
    Frames of the simulation to bake (inclusive). The start and end frames of the bake are repeated when accessing frames outside the baked range.
Cache Path
    Folder to store the baked EXR files in. The sequences will be in the form `disp_####.exr`, `normal_####.exr`, and `foam_####.exr` where #### is the four digit frame number. If the cache path folder does not exist, it will be created.

Simulation Internals

The simulator itself uses FFT methods to generate 2D grids of simulation information internally, very similar to 2D texture maps. The simulator can generate three types of data - displacement, normals, and extra data that is used to calculate wave crest intersections (i.e. foam). After simulation, these maps are used to displace the ocean surface geometry in 3D, and also can be used for shading via the Ocean texture. The internal simulation engine is multi threaded with OpenMP to take advantage of multiple cores.

Examples

https://vimeo.com/18911131

Simulated and baked to image maps in Blender, rendered in 3Delight.