ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2020
J. Bender and T. Popa
(Guest Editors)

Volume 39 (2020), Number 8

Supplemental Material for ”Making Procedural Water Waves Boundary-aware”

S. Jeschke[1] C. Hafner[2] N. Chentanez[1] M. Macklin[1] M. Müller-Fischer[1] C. Wojtan[2]

[1]NVIDIA [2]IST Austria

1. Particle System

Our animation framework provides a tuneable particle system that improves visual fidelity independently of the wave-cage method. Inputs to the system are the terrain and the animated water surface, as well as an optional user-supplied “details” heightfield $b(x,y)$ that may contain geometry such as sharp rocks, which are too small to affect the water waves. This document describes the two types of particles that are used, when they are emitted, and how they are advected.

The particle system has a global budget of how many particles may exist in total (100k in our implementation). Active particles are advected in every frame until they reach the end of their lifetime and are marked inactive. Inactive particles randomly pick a point in screen space and check if the corresponding point on the water surface is a suitable location for being re-emitted. If this check fails, the particle stays inactive and tries again in the next frame. Thus, particles are seeded uniformly in screen space to maximize the visual impact given the current camera angle.

There are two types of particles: spray particles, which emulate the behavior of wave crests folding in on themselves, and foam particles. Spray particles are emitted from the lip of a sharp wave crest under the condition that it will not be immediately reabsorbed by the water surface in the next frame. This is formalized as

$$\left(\frac{\partial^2 \boldsymbol{\sigma}}{\partial t^2} - \boldsymbol{g}\right) \cdot \boldsymbol{n} < 0,$$

where $\boldsymbol{n}$ is the normal of $\boldsymbol{\sigma}$, and $\boldsymbol{g}$ is acceleration due to gravity. This ensures that the water surface decelerates relative to the particle. Thus, the particle can escape the wave with its initial velocity. An alternative condition is to detect wave lips from the Jacobian of $\boldsymbol{\sigma}$ as described by Tessendorf [Tes01].

A spray particle is initialized with the velocity of $\boldsymbol{\sigma}$ at the seed point and follows a ballistic trajectory until it collides with the water surface or a different obstacle. At this point it is converted to a foam particle.

Foam particles are created by conversion of spray particles, and direct seeding at points where the water surface intersects the user-supplied heightfield $b(x,y)$. The condition for direct seeding is that the water surface is sufficiently close to the heightfield along the z-coordinate, and that the wave moves towards it, formalized as

$$|\boldsymbol{\sigma} \cdot \boldsymbol{e}_3 - b| < \varepsilon, \quad \text{and} \quad \frac{\partial \boldsymbol{\sigma}}{\partial t} \cdot \boldsymbol{n}_b > 0,$$

where $\boldsymbol{n}_b$ is the normal vector of the heightfield. During advection, foam particles behave like small rigid bodies sliding on the water surface with high friction. Wave crests traveling towards the seashore will thus push them to the beach, where they stick and eventually despawn.

Foam particles are additionally used to drive a wetness map on the terrain and a “foaminess” map on the water surface, as shown in the supplemental video and the teaser image of the main document. The wetness map is updated by splatting foam particles onto a texture which is then used to darken the terrain during rendering. The map “dries” a little in every frame to emulate the brightening of the terrain after a wave draws back.

The foaminess map is also updated by splatting foam particles onto it. This map drives the opacity of a foam texture that is projected onto the water surface at rest and then transformed together with the surface itself. The opacity of this texture is further multiplied with the inverse Jacobian of $\boldsymbol{\sigma}$. This reduces foaminess in regions where the surface is stretched and increases it where the surface is compressed.

Fig. 1 shows a breakdown of the water surface, particle types, and final result.

References

[Tes01] TESSENDORF J.: Simulating ocean water. *SIGGRAPH’99 Course Note* (01 2001). 1

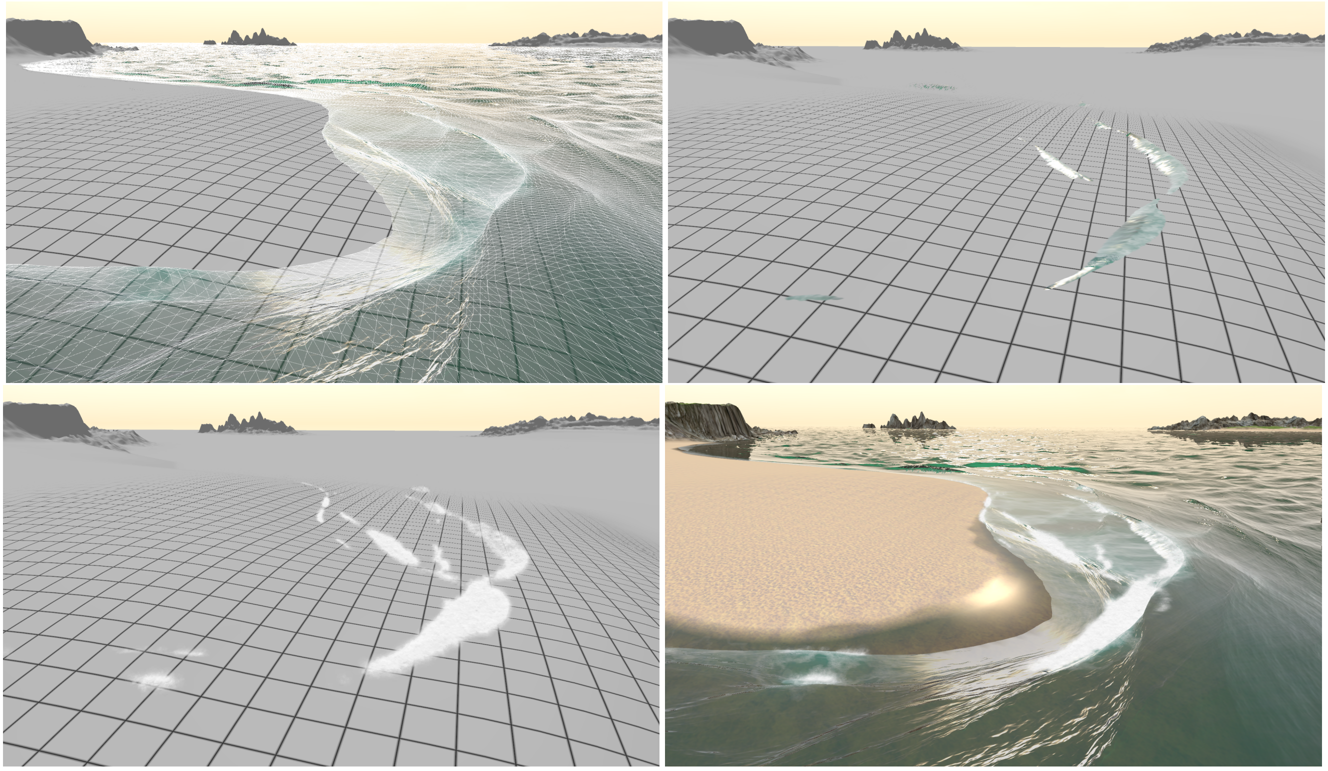

Figure 1: *Breakdown of the rendering system. (a) water surface mesh, (b) spray particles, (c) foam particles, (d) final result.*