On the Simulation of Ocean Waves in Real-Time Using the GPU

Alex Salgado[1], Aura Conci[2]

Universidade Federal Fluminense

[1]*asalgado@ic.uff.br*, [2]*aconci@ic.uff.br*

Abstract

Nowadays, due to GPU processing power, it is possible to use advanced rendering techniques with great realism in real-time applications. This work simulates the ocean wave behavior processing all geometric computation and rendering in GPU. The shape is defined by the Gerstner´s equation considering the movement of each particles of water. The deep sea topology is considered. It is possible to tackles the representation in deep water as well as in shallow water. The ocean floor profoundness and inclination influence on the shape and form of the particle displacement are considered. The model represents also the wave refraction and the wind or gale effect on the wave. This novel approach simulates the breaking waves near the shore. The real-time rendering technique implemented in this work uses combinations of advanced tangent-space reflective bump mapping and environment mapping plus Fresnel reflection and HDR.

1. Introduction

Representation of natural phenomena is one of the most challenging tasks in computer graphics. The ocean wave simulation is included in these [1,2]. Most of previous work considers the waves geometric modeling in empirical forms [3,4]. In this work the classic model of Gerstner waves is used [5]. The objective is to create realistic ocean visualization in real time representing complex effect as the sea floor and wind influence (from breeze to gale) and also the effect of breaking waves. We develop a model of geometry representation and a wave visualization system using several techniques of rendering. Animation and rendering use a shading model programmed in the NVIDIA Cg language, applying techniques of tangent-space reflective bump mapping, environment mapping to Fresnel reflection [6] and HDR (High Dynamic Range).

2. Stages of simulation system

The proposed system to simulate and animate ocean waves is composed of 4 stages: wave generation, surface modeling, optic computation and water rendering. The Figure 1 shows these stages and theirs connections.

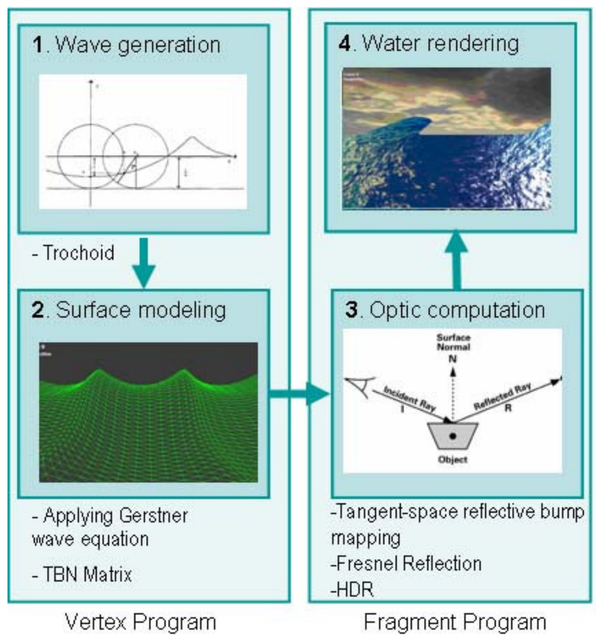

Figure 1 – Stages of the system

The first two stages (figure1) are performed in vertex programming. The geometry wave generation is defined using the Gerstner equation [7] with circular orbit to simulate the deep water waves and elliptical orbit, with major axis parallel to the sea floor slope to simulate shallow water waves as shows in equation (1) and equation (2).

$$\begin{aligned} x &= x_0 + r\sin(kx_0 - \omega t) \\ y &= y_0 - r\cos(kx_0 - \omega t) \end{aligned} \qquad (1)$$

where **r=H/2** is the height of the wave, **k=2π/L** is the wave number, $\mathbf{L=2\pi/k=gT^2/2\pi}$ is the wavelength, **ω=2π/T** is the frequency and **ω/k=L/T=c** is the wave celerity. A trochoid surface equation (stage 1 in figure1) is used for definition of the particle motion on the ocean surface [3]:

$$x = x_0 + r\cos\alpha.S_x.\sin\Phi + \sin\alpha.S_z.\cos\Phi$$
$$y = y_0 - r\cos\alpha.S_z.\cos\Phi + \sin\alpha.S_x.\sin\Phi \quad (2)$$

where **Φ** is the phase angle; $\mathbf{\sin\alpha = \sin\gamma e^{-k_0 h}}$, **γ** is the bottom slope in the direction of the wave travel; $\mathbf{S_x = 1/(1 - e^{-k_x h})}$ and $\mathbf{S_y = S_x(1 - e^{-k_y h})}$ are the major axis increment and the minor axis decrement respectively; $\mathbf{k_0}$ determines the depth influence on the ellipse inclination; $\mathbf{k_x}$ is a enlargement factor of the ellipse major axis; $\mathbf{k_y}$ is the minor axis reduction factor (figure 2). It is important to note that $S_x \rightarrow \infty$ when the depth h $\rightarrow 0$.

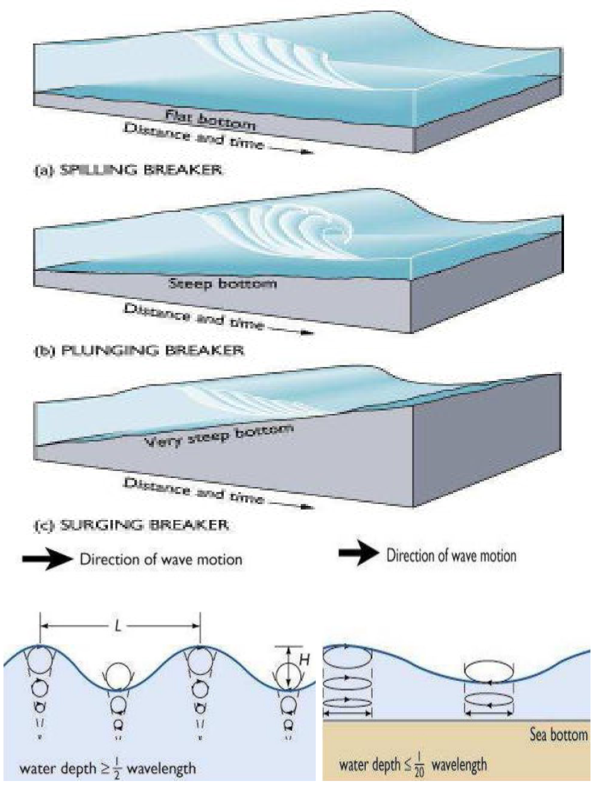

Figure 2 – Influence of the ocean floor

In this work, the surface is first modeled and then the shader is applied. The surface is considered endless on the XY plane that represents the animated ocean (stage 2 in figure 1). Bump mapping is used to provide the effect of the capillary waves (stage 3 in figure 1). Stages 3 and 4 were developed using a fragment program. The optic (stage 3) calculates illumination, Fresnel reflexion and HDR. In the last stage, the ocean is rendered by joining the previous three stages and the water color contribution.

3. Results

The developed system creates a realistic simulation of the ocean waves (figure 3). It can be an applicable tool in realistic animations. Since this was made as an effect (fx file), it is simple to be used for programmers or 3D artists. The FX Compose 1.8 from NVIDIA® was the shader IDE used to program in Cg (shader 2.0). Also was tested as an effect file in the 3D MAX® from Discreet®. Figure 2 images are examples of generated scenes running in real-time.

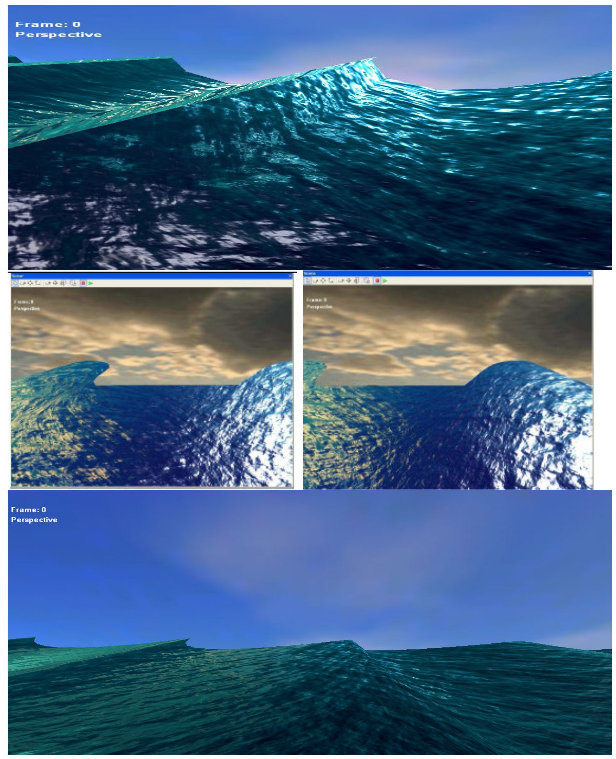

Figure 3 – Real-time results

References

[1] R .Fedkiw, J. Stam., H. W. Jensen. *Visual Simulation of Smoke*, Proceedings of SIGGRAPH, pp. 15-22, 2001.

[2] N. Foster, R. Fedkiw, *Practical Animation of Liquids*, Proceedings of SIGGRAPH, pp. 23-30, 2001.

[3] A. Fournier, W. T. Reeves, *A simple model of ocean waves*, SIGGRAPH, volume 20, pp. 75-84, 1986.

[4] D. R. Peachey, *Modeling waves and surf*, SIGGRAPH, volume 20, pp. 65-70, 1986.

[5] B. Kinsman, *Wind Waves: Their Generation and Propagation on the Ocean Surface*, Prentice Hall, 1965.

[6] M. Wloka, Technical report: Fresnel Reflexion, NVIDIA Corporation, (Jun 2002).

[7] J. Tessendorf, *Simulating ocean water*, SIGGRAPH'99 Course Notes, 1999.