Real-time Ocean Wave in Multi-channel Marine Simulator

Xie Cui* Jin Yi-cheng† Liu Xiu-wen‡

Nautical Science and Technology Institute, Dalian Maritime University, Dalian, China

Abstract

We present an adaptive scheme for real-time simulating irregular long crest waves in multi-channel marine simulator. The method restricts computations to the visible part of the ocean surface in each view channel, adapts the geometric resolution to the viewing distance and only considers the visible waves wavelengths. The method allows the user to move over an unbounded ocean following the own-ship, and immersed in a real-time updated photo-realistic ocean environment.

Keywords: multi-channel marine simulator, adaptive scheme, ocean waves spectrum

1 Introduction

Developing real-time ocean wave models for marine simulator is difficult: First the ocean wave is both very wide and very detailed. Second, ocean wave and stream conditions are an important guideline for safety navigation, which requires to reproduce ocean waves as accurately as possible. Third, the marine simulator is a complex multi-channel visual system, so we must consider how to seamlessly stitch up each channel's image. Finally, because marine virtual ocean environment is composed of many objects (such as ships, islands, docks, buildings and etc.), we must make sure to enable real-time rendering at a high frame speed, and create an immersing ocean environment.

----

*e-mail: xiecuidlmu@sohu.com

† e-mail: jycdmu@dlmu.edu.cn

‡ e-mail: liuxw@dlmu.edu.cn

Our main contribution is generating a predetermined surface mesh with adaptive geometric resolution related to the current eye-point, and its location in world space will change dynamically together with the own-ship's movement. Combining the different settings of different channels in marine simulator, we seamlessly stitch up each channel's surface mesh and generate a consecutive dynamic mesh for simulating a virtual unbounded ocean.

The remainder of this paper develops as follows: Section 2 reviews the existing models for real-time simulating ocean waves. Then we present our wave model in Section 3. The adaptive scheme is described in Section 4. We show and discuss the ocean wave rendering results in Section 5, and conclude in Section 6.

2 Previous Work

Up to now, several real-time ocean wave models have been developed in CG, which can be separated in two families. The first is based on geometric models. The second is based on statistical models, and the third is the mixed models.

The start works of Fournier and Reeves[Fournier et al. 1986] as well as Peachey [Peachey et al.1986] can be seen as forerunners of the geometric-based category. Generally speaking, their methods are simple and intuitionistic, the calculation can be conducted in near real-time. But the generated scenes are less realistic, especially when only one or two wave trains are simulated.

The second family corresponds to spectral approaches, first introduced in CG by Mastin et al. [Mastin et al. 1987]. The basic idea is to produce a height field having the same spectrum as the ocean surface. This can be done by filtering a white noise with Pierson-Moskowitz's or Hasselmann's filter, and then calculating its Fast Fourier Transform (FFT). Tessendorf [Tessendorf 2001] explained simulating ocean water based on FFT in detail. The main benefits of this approach are that many different waves are simultaneously simulated, with visually pleasing results. However, real-time animating the resulting ocean surface remains challenging when eye-point continuously moves over an unbounded ocean.

The third family is a hybrid approach: Premože [Premože et al. 2000] used wave spectrums and the linear combination of sine function to simulate natural water. Changbo Wang et al. [Changbo Wang et al.2003] proposed a real-time ocean wave model based on cellular automata. Their model's frequencies are also come from the wave spectrum and satisfy the requirements of real-time simulation of ocean waves. Yin Yong et al. [YinYong et al. 2003] used a dynamic rectangular mesh model and adaptive wave spectrum sample scheme to simulate real-time irregular long crest waves with different wave scale.

But for all above families of approaches, the ocean surface mesh is regular without continuous LOD, so the quality isn't pleasing when eye-point is moving on the surface. Moreover the visible surface area is large and a high resolution is needed in regions close to the viewer, this requires very large meshes, yielding costly simulations. These methods can only be adapted to real-time for very crude resolutions.

In the same spirit of [Hinsinger et al. 02], We also adapt the wave model in spectral space by locally filtering the waves according to the local geometric sampling rate. In the same time, we improve the wave spectrum sample scheme to create an adaptive set of independent unit waves. This method not only saves computational time but also reduces geometric aliasing. Another improvement is an eye-related adaptive scheme applied to an independent ocean surface mesh, which adapt the mesh sampling to the projected size of each surface grid unit, thus generate a predetermined region of the ocean surface with continuous LOD. In simulation, we transform the location of surface mesh in world space according to the current eye-point position and the current navigation direction of each channel, Combining the height evaluation of all grid nodes by irregular long crest wave model, we finally obtain a seamlessly tiled consistent ocean scene with very high frame rate.

3 Irregular long crest wave model

Since ocean waves created based on FFT usually restrict computations to a rectangular area, which can be used as a tile for simulating a larger surface. The computation cost only allows crude resolution when real-time rates are required. Moreover, enabling the trainee or pilot to move over the ocean on a ship is impossible, since avoiding obvious repetitively would require the use of an extremely large simulated region.

This paper focuses on the ocean waves created by the wind in deep water. They are also called the irregular long crest wave. In general instances, long crest wave can be composed by multiple regular long-crested waves (unit waves) with different amplitudes and different wavelengths. The following formula [YinYong et al. 2003] can be adopted to calculate the instantaneous wave height:

$$y = y_0 + \sum_{i=1}^{n} \zeta_{ai} \cos[k_i(x\sin\theta + z\cos\theta) + \omega_i t + \varphi_i] \qquad (1)$$

here：

y_0：Tide height

k_i: $k_i = \frac{\omega_i^2}{g}$ ， the wave number of i-th unit wave

g： $g=9.81\text{m/s}^2$，acceleration of gravity

θ：propagation direction of unit regular wave， $\theta \in [0,2\pi]$

t：time

Φ_i：phase angle of i-th unit wave (Radian)

4 Adaptive Scheme

Since the visible ocean surface area is large, a high mesh resolution is needed in regions close to the eye-point, and a crude mesh resolution is adequate for the distant regions. So we create a predetermined mesh with continuous LOD and we can dynamically change its location in world space according to the different channels. In simulation, we update the mesh's position according to both the channel's setting and the current own-ship's position. This yields real-time ocean surface mesh in each channel of marine simulator and create a fluid simulations.

4.1 An adaptive surface mesh model

The basic idea [Hinsinger et al. 02] is to generate the mesh representing the ocean surface such that every grid covers, at rest, the same area on the screen. This is done by subdividing the part of the screen into a grid of quads, which are back-projected on the plane modeling the ocean surface at rest. The resulting mesh points are adequate for representing the ocean surface.

In our multi-channel marine simulator, the eye-point of visual system is located just above steering wheel and moves with the own-ship. Viewing direction is along the course heading and the relative position between own-ship and eye-point does not change. Generally the vertical distances between the eye-point and ocean surface is predetermined upon the different types of the ship's height. Depending on the condition of installing place, we can select suitable distance from eye-point to screen, width of screen and height of screen to get correct geometry relation of the scene. Thus a suitable angle of field of view (VOF) and the correct setting parameters of viewing frustum of each channel can be obtained.

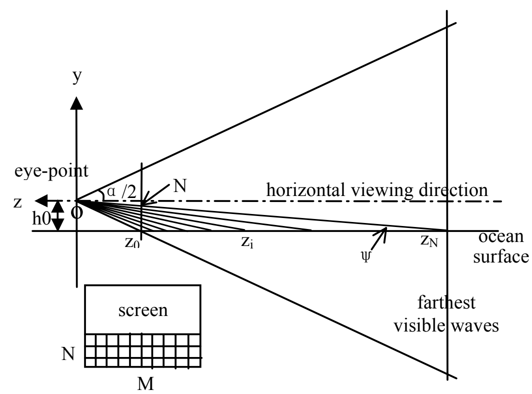

Figure 1. Generate Adaptive Surface Mesh

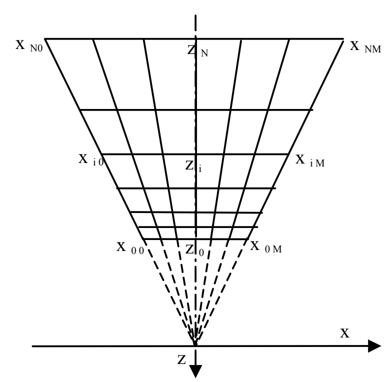

Figure 2. Adaptive Ocean Surface

Through simple trigonometric function computation, all M×N nodes coordinates (x_{ij}, z_i) will be obtained in object space (see Figure 1 and 2). u is the number of marine simulator's channels, φ is the horizontal angle of VOF, and α is the vertical angle of VOF. h_0 is the vertical distances between the eye-point and ocean surface .

Our improvement is that a surface mesh with continuous LOD can be predetermined and precomputed in initialization phase without reducing the rendering speed. You may note the advantage of the method: the generated mesh density transition is continuous and an adequate resolution being maintained everywhere in the computed ocean image.

In simulation, we take the resulting surface mesh as an object participating in all the geometry transformation together with other objects .In section 4.2 we will show you how the mesh object being transformed in world space with different channels and how a dynamic continuous multi-channel ocean surface mesh being constructed.

4.2 Consecutive seamless multi-channel ocean wave mesh

In marine simulator, three different kinds of screens are widely used, which are flat screen, polygonal screen and cylindrical screen etc. The choice is determined by specific application. Some key technologies of multi-channel visual system in marine simulator such as system configuration, synchronization and dynamically setting of viewing direction and the position of eye-point for each channel was introduced in detail in [Jin Yi-cheng et al. 2003]. An illustration is figure 3.

Figure 3. Multi-channel Own-ship Bridge

Here take three-channel visual system as an example. Considering the horizontal plane y=0, Supposing at time t the current eye-point point is at (S_x,S_z), the course heading angle is β， then run the following steps to get the instantaneous position of the surface mesh of all channels in marine simulator. First, we rotate the resulting mesh coordinates with an angle (β+k•φ/u) in its' object space, where k denotes the different channel of marine simulator. k ＝0 denotes the center channel，k=1 denotes the first right channel and k= -1 denotes the first left channel. Second we shift the mesh's origin to the current eye-point position in world space, and compute the new position of each mesh nodes (see figure 4).

$$x_{wij} = x_{ij} \cdot \cos(\beta + k \cdot \frac{\varphi}{u}) + z_i \cdot \sin(\beta + k \cdot \frac{\varphi}{u}) + S_x \qquad (2)$$

$$z_{wi} = -x_{ij} \cdot \sin(\beta + k \cdot \frac{\varphi}{u}) + z_i \cdot \cos(\beta + k \cdot \frac{\varphi}{u}) + S_z \qquad (3)$$

Because the joint mesh nodes between the center channel and side channels is the same coordinates in world space, so their heights are the same. Now we get a consecutive seamless multi-channel ocean wave mesh.

4.3 Adaptive set of simulated unit waves

Since the grid defines the highest available resolution, we do not display waves smaller than the grid quads. In addition to reducing aliasing, this saves much computational time, since we don't need to simulate the associated unit waves for this frame and location, especially for the wide surface area near the horizon since the resolution there is coarse. Most of the unit waves are thus pruned .

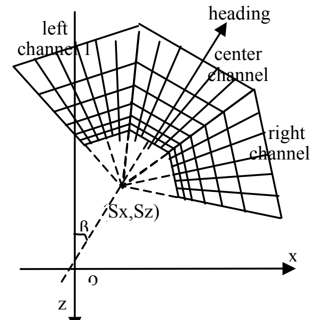

Figure 4. Seamless Meshes

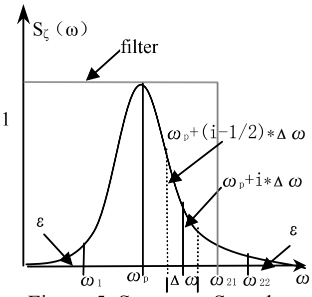

Figure 5. Spectrum Sampling

In addition to implementation the filtering of waves with distance, we adaptively sample the wave spectrum to different unit frequency and use as the unit waves for wave simulation. Thus our algorithm provides a flexible control of the quality/cost ratio by the number of simulated unit waves. The following is our implementation (see figure 5).

1) We first determine a finite spectrum range for our simulation by a desired precision parameter ε to select the primary spectrum range[ω_1, ω_{22}] reserving the mostly energy of spectrum.
2) According to the visible wavelength size of all meshes computed by the spaces between adjacent mesh nodes, we can decide the maximal visible frequency ω_{21} for all grid nodes.Therefore waves appearing smaller than the grid quads do not display and the associated unit waves whose wavelength is smaller than the grid quads don't need to be simulated.
3)Compare ω_{21} with ω_{22}, and select the smaller one as a conclusive frequency $\omega_2=\min(\omega_{21}, \omega_{22})$.
4) Range [ω_1, ω_2] is adaptively sampled according to the number n of simulated unit waves.

4.4 Calculation of wave height in multi-channel marine simulator

After calculating the amplitude ζ_{ai} ,phase frequency ω_i and wave propagation direction θ in above section, we can evaluate the instantaneous wave height of each channel's grid nodes by combining the equation 1 、2 and 3. Finally we obtain a dynamic continuous ocean surface in multi-channel marine simulator.

But now we have to note two things for generating a seamless multi-channel image. One is the edge blending between adjoining channels and the other is the synchronization between different channels. To solve the first problem, we extend the angle of horizontal VOF one to three degrees between adjoining channels edge. As for the second problem, a synchronous time is sent to all side channels by the center channel at the beginning of simulation. As a result, the ocean waves are consistency and fast updated when continuously moving the eye-point with a ship.

5 Results and Discussion

A detailed discussion of rendering issues is out of the scope of this paper. Here we emphasize the fast rendering with many complex objects to create an immersing virtual ocean environment. In addition to using an optimized mesh with continuous LOD, we apply two high-resolution textures to the ocean surface to improve the ocean fidelity. One is used for adding surface details, and the other is used to add reflections of the sky. Using texture-mapping method saves much time on rendering and display with nVidia graphic chip set. We can still maintain a better image quality as shown in figure 6-7 even with an extremely coarse grid-size.

We rendered several kinds of the ocean waves with other simple objects (including ocean waves, two islands and own-ship) on a PC of Pentium Ⅳ 1.9GHz GeForce2 graphics board , with the horizon near the middle of the Image. Rendering is implemented in OpenGVS 4.4. 16 unit waves were computed. As illustrated in Figure 6: more than 160 fps are obtained with a 40×30 screen-mesh resolution. Good quality images can still be obtained more than 45 fps (Figure 6,right) with 80×60. Two beautiful scenes are presented on Figures 7.

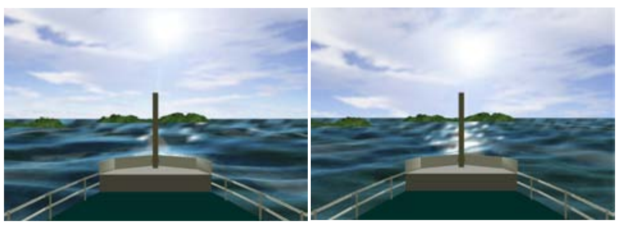

Figure 6. Mesh Resolution (left) 40×30 (right) 80×60

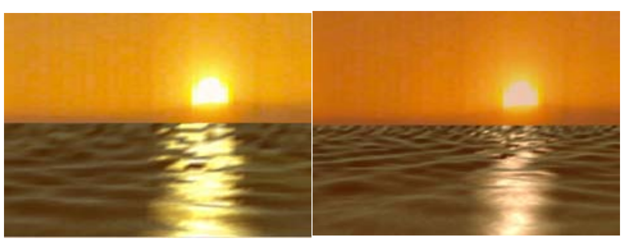

Figure 7. Mesh Resolution (left) 40×30 (right) 80×60

Compromises have to be chosen between speed and quality: using smaller quads allows for figuring thinner details with a higher cost. If the quads are large, the ocean surface might look too smooth. We can decide the aptitude resolution of surface mesh to a specific application according to the hardware performance of image generating system.

In addition to increasing efficiency, our method is more flexible than the spectral approach: the mesh density on the screen can be tuned to increase resolution for better image quality. Adjusting the number n of unit waves provides a flexible control of the quality/cost ratio. Moreover, our method is more efficient than [YinYong et al. 2003]: the number of computed mesh points is about 1/4 of his method for each channel mesh. We also improved the method of [Hinsinger et al. 02] in that we precreate an independent ocean surface mesh and can used for any multi-channel marine simulator. Lastly, our method works at no extra cost whatever the eye-point motion with the own-ship, which is a foundation for simulator training and scientific research etc.

6 Conclusion and Future Work

We have proposed an adaptive scheme for real-time simulating irregular long crest wave in multi-channel marine simulator. Relying on an ocean wave spectrum sampling and a dynamic adaptive surface mesh, all computations are exclusively concentrated onto the visible part of the ocean surface and the visible wavelengths in each view channel. Thus allows the user to move over an unbounded ocean following the own-ship, and immersed in a real-time updated photo-realistic ocean environment.

Concerning future work, we plan to implement the high-speed ship waves simulation and to study how to introduce it to our simulator, since we have achieved the meddle-speed and the low-speed ship wave effect [YinYong et al. 2001]. Moreover, our method eases the addition of ship waves.

References

Fournier, A. and Reeves, W. 1986.A Simple Model of Ocean Waves. *Computer Graphics and ACM Proceedings of SIGGRAPH* , vol. 20, no. 4, 75-84.

Gary A. Mastin, Peter A. Watterger, and John F. Mareda. 1987. Fourier Synthesis of Ocean Scenes.IEEE CG&A, March, 16-23.

Hinsinger, D., Neyret, F., Cani, M.-P. 2002. Interactive Animation of Ocean Waves. Proceedings of *ACM SIGGRAPH Symposium on Computer Animation*, 161–166

Jin Yi-cheng, Yin Yong, Li Zhi-hua.2003 Key Technologies of Multi-channel visual system used in marine simulation. *Proceedings of MARSIM'03,* vol. 1, RA-27-1–RA-27-8

Peachey, D. R. 1986. Modeling Waves and Surf. *Computer Graphics. ACM Proceedings of SIGGRAPH*, vol. 20, no. 4, 65-74.

Premože, S. and Ashikhmin, M. 2000. Rendering Natural Waters. Proceedings of Pacific Graphics, *Eighth Pacific Conference on Computer Graphics and Applications*, Oct.2000

Tessendorf, J. Simulating Ocean waves.1999.*SIGGRAPH'99 Course Notes&SIGGRAPH'2000*: Course Notes 25:Simulating Nature: From Theory to Practice,.3.1-3.18

Wang Changbo, Wang Zhangye, Peng Qunsheng. 2003.Real-time Simulation of Ocean Wave Based on Cellular Automata, *Proc. Of CAD/Graphics'2003,* Macao, China.

Yin Yong, Jin Yi-cheng, Ren Hong-xiang. 2003. Wave Simulation of Visual System in Marine Simulator Based on Wave Spectrums. *Proceedings of MARSIM'03,* vol. 1, RA-28-1–RA-28-8

Yin Yong, Ren Hong-xiang, Zhang Xiu-feng, Jin Yi-cheng,. 2002. Wave Simulation Technology Applied to virtual Environment of Navigation . Journal of system simulation,(in Chinese) 2002 , v 14, n 3, March 2002, 313-315