DYNAMIC FLUIDS APPLIED TO A SURF ZONE ENVIRONMENT

Guru Prasad and Glenn Martin

Institute for Simulation and Training, University of Central Florida

Orlando, Florida

ABSTRACT

To date, simulators including fluids in their environment model have restricted themselves to the representation of sea states alone. The behavior of fluids in the littoral area or surf zone is more dynamic and includes the interaction of terrain and fluid surfaces. The surf zone refers to the 10 meter curve at the low tide to the high water mark at the highest high tide. The representation of dynamic waves, wakes and eddies have been mimicked by uniform elevation change of the 2D fluid surface along with texture animations.

This paper discusses the work of real-time dynamic fluid flow, applied to a networked dynamic terrain environment, and then shows how it all applies to a surf zone environment. Surf zone issues include incorporating and integrating dynamic terrain and dynamic fluids, dynamics of amphibious vehicles, and the problem of underwater terrain. This will support the modeling of improved vehicle dynamics models (to support buoyant and ground driven behaviors), dynamic wakes, design and mobility modeling, and in training.

ABOUT THE AUTHORS

Guru Prasad is a Research Scientist at the Visual Systems Laboratory at the Institute for Simulation and Training, University of Central Florida. He has worked on the Dynamic Terrain project developing mobility models and physically-based models. He received his Master's in mechanical engineering from the University of Central Florida (U.C.F.) and is currently pursuing his Ph.D. in simulation under industrial engineering at U.C.F.

Glenn Martin is a Visual Systems Scientist in the Visual Systems Laboratory at the Institute for Simulation and Training, University of Central Florida. He worked on the Dynamic Terrain project where he implemented a second revision of a software architecture for distributed, dynamic environments. He holds the BS and MS degrees in computer science from the University of Central Florida and is pursuing his Ph.D. in computer science at U.C.F.

DYNAMIC FLUIDS APPLIED TO A SURF ZONE ENVIRONMENT

Guru Prasad and Glenn Martin

Institute for Simulation and Training, University of Central Florida

Orlando, Florida

INTRODUCTION

The surf zone area consists of the littoral zone and involves complex interaction between the land, sea and atmosphere. The representation of the surf zone environment in distributed simulation is important for mission planning, mission rehearsal, training, amphibious vehicle testing, education and engineering analysis. The necessity of a surf zone environment for various programs in DIS and the factors affecting them (Biano, 1996) indicate the necessity of a dynamic fluid surface and representation of various attributes to support the same. Some of these effects are waves, wakes, acoustic propagation and visibility. In addition, there is also a need for bathymetry, capacity to represent mines, and a desire for terrain topography. The work done on dynamic terrain and fluids at the Institute for Simulation and Training (IST) provided a motivation and served as a testbed to expand on some ideas of a shared environment to provide a proving ground for implementing surf zone models.

OBJECTIVES AND ISSUES

The objective of this effort is to conduct initial implementation of some of the physics-based models and study their interactions and implications in a distributed simulation. The results of the above should lead to:

- systematic model development to incorporate vehicle dynamics, hydrodynamics, ocean models, and mobility
- a testbed for transforming existing ocean models to the application required by the end user
- development of relationships between model fidelity, database resolutions and real-time requirements
- a basis for terrain/fluid interaction models
- improved architectural support in the shared environment to facilitate the requirements of the surf zone
- API for simulations running reasoning models

Modeling

The models serving the surf zone have to cater to all three zones (land, sea and atmosphere). Most of the models catering to ocean effects are time varying functions. Some of the dynamic models catering to the surf zone are thermal, circulation, wave, surf and tidal models (Haeger, 1988). Certain terrain attributes which were more or less constant are also time varying in the surf zone and affect the mobility of amphibious vehicles. Dynamic models, such as a wake model, are a function of time, number of wake sources in the region, the vehicle parameters, water depth, and wave models.

The modeling of the above effects in a distributed environment calls for representation of additional data which are temporal and spatial in nature and are distributed along different scales. Since the incorporation of surf zone into military simulations is in its inceptual stages, provisions have to be made to incorporate new data in a manner compatible to the scientific and simulation community. Most of the ocean models are prohibitive for current real-time simulation applications in terms of data and computational requirements.

The objective of this effort is to ease some of these requirements by providing structure to abstract the data from user applications so that some of the models developed currently need not change due to a changes in network, addition of new data, change of database resolution, or the use of faster computing resources. This also provides a basis to add new models to the testbed before their aggregation/disaggregation to suit the end user needs. This testbed could also provide basis to develop new dead reckoning schemes and protocols for a surf zone simulation. Many of the fluid dynamics calculations use finite difference schemes for their simulation and necessitate the representation of gridded data for their use. This paper discusses the effects of extending some of the previous work to support amphibious vehicle operation in a shared surf zone environment.

Visualization

Visualizing the surf zone area provides unique problems. Unlike dynamic terrain, the entire surf zone is constantly changing as waves move towards the shore. A first attempt at visualizing the surf zone might consider only waves which will not necessarily require an extreme number of polygons to represent. However, if breaking waves and other effects such as wakes are considered, then certainly visualization will be a concern.

Networking

Networking is one of the most difficult aspects of the surf zone. The motion of amphibious vehicles is affected by wakes (AAAV, 1995) which makes the simulation of wakes critical to a training scenario. This deters the use of a simple wave model with a few parameters.

Ordinarily, wakes would cause the use of a small resolution (or a large number of polygons) which would force many updates to be sent on the network for something as dynamic as the ocean. Limiting the scope to the surf zone will help, but the networking cost will still be high. Instead, as will be shown, another approach (using spline surfaces) can be used.

PREVIOUS WORK

Shallow Water Models

Most of the surface wave representations in simulation follow Airy's theory (Kinsman, 1965). Graphic researchers (Peachey 1986; Fournier 1986) were interested in rendering animation frames with refraction and breaking of waves. Efforts were made to represent these waves in a hovercraft trainee environment (Donner, 1991). The sea states were animated by recycling the waveforms and the surf zone was animated using curl and spill polygons. The elevation profiles used in the above were generated by off-line programs. Computational bounds and rendering algorithms were primarily responsible in keeping the hydrodynamics away from the surf zone simulation.

Particle System Models

Heuristics were employed in representing particle motions to simulate ship wakes (Goss, 1990). The work done at IST looked into incorporating 2D Navier-Stokes equation to provide a physical basis for fluid flow using a synthetic surface in a real-time environment. Fluid flow around boundaries, turbulent behavior of fluids and fluid blending could be simulated.

Ocean Models

Ocean models typically fall into one of five categories:

- Thermal models
- Circulation models
- Wave models
- Tide models
- Surf models

Some are probabilistic and others are based on static "animated" databases of parameters including wave height, depth, distance from shore line, etc. They are not well suited for real-time, interactive simulation. However, a model could be built that runs in interactive-time that uses these parameters.

OUR OCEAN MODEL

Description

The ocean surface is currently represented by a 2-1/2D mathematical surface. This is a logical step in extending existing terrain databases and mobility models used in distributed simulation before implementing 3D databases. This provides a basis to implement wave theories, ocean models and computational fluid dynamics models. The wave profiles used here are generated by Fourier sum of sinusoidal functions and hold good for deep, intermediate and portions of shallow water. The (x,y,z) points form the defining points for a spline surface. The spline surface is essential to support continuity and refraction properties of wave fronts (Tso, 1987). Incidentally, the dynamic terrain (Altman, 1994) surface developed at IST was represented by a mathematical surface supporting three different curve types. The curve types used in the present implementation do not support wave breaking phenomenon.

Bathymetry data is essential for gravity wave theories and various ocean models. One method is to extend the current terrain surface under the ocean surface and the other is to provide a separate surface and store bathymetry data. The former surface was easier to extend since extreme care has to be taken to represent the transition region to provide continuity. The latter is already implemented at IST (AAAV, 1995) to represent ocean surfaces supporting amphibious vehicles in ModSAF.

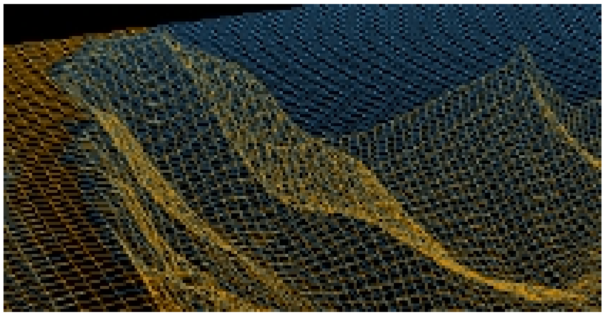

Figure 1. Ambient Ocean and Terrain skin showing bathymetry data.

This paper discusses the implementation of extending the terrain surface under water where the dynamic ocean surface is represented by adding an attribute layer. This is illustrated in Figure 1.

Effects

Waves, wakes, buoyancy are some of the important dynamic effects studied in our work.

Waves. Most of the surface waves discussed in this paper refer to gravity waves. The definition of the surf zone region has been fairly loose. For the application of wave theories, the ocean surface has to fall under shallow water, intermediate water and deep water (Kinsman 1965) where,

$d/\lambda_d \leq 1/200$ in shallow water,

$1/200 < d/\lambda_d < 1/2$ for intermediate water

and $d/\lambda_d \geq 1/2$ for deep water,

where $\lambda_d = gT^2/2\pi$ = wavelength in deep water,

d = depth of water.

Linear wave theory developed by Airy's and Stokes' 5th theorem are still followed by many modelers for deep, intermediate and some regions of shallow water (Patel, 1989). Linear wave theory is used here to model small amplitude waves for small to moderate sea states. Complex waves could still be created using superposition. The choice of wave theory in simulation is a compromise between accuracy and computational speed.

The architecture chosen in this paper is not affected significantly due to either one of the above changes and lets one implement any model. This is illustrated by the use of two different models: one for waves and the other for wakes created on the same surface. The elevation of the ocean surface is defined by $\xi(x,y,t)$ where,

$$\xi(x,y,t) = \sum_{i=1}^{N} a_i \sin\{k_i \cos(\theta_i)x + k_i \sin(\theta_i)y - \omega_i t\}$$

a_i = amplitude,

k_i = wave number,

ω_i = frequency,

θ_i = initial angle of wave front with x axis,

t = time, and

N= number of waves in the wave front.

A combination of two to three waves differing in phase and amplitude are used to meet the spectral energy requirements of the wave front. The amplitude, velocity and phase are functions of water depth (Kinsman, 1965) and have to be recalculated to obtain the correct wave height. The refraction of the waves are also implemented using Snell's law so that the waves align with the beach irrespective of the initial direction of the wavefront. The wave model is implemented by a client application logging on to an Ocean Service which maintains the wave, wake and other ocean effect updates which could be running on the same or different machines. Detailed description of the Ocean Service is given in the architecture section.

The wave client sends updates of the new elevation profile $\xi(x,y)$ at a specified rate to the Ocean Service which maintains those updates in the database. The patch size conforming to a specific wavefront could vary dynamically due to wave refraction, number of waves in a wavetrain, wavelength, or wave celerity.

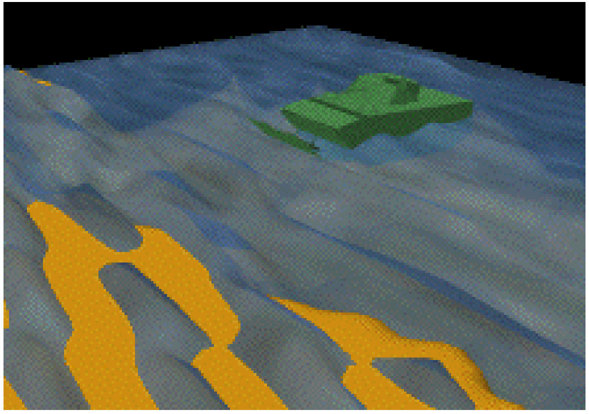

Figure 2. Dynamic ocean surface along a beach.

In the current implementation the patch size is retained constant since the surf zone is a small area and the coastal profile did not have significant undulations. Figure 2 illustrates the waves due to one wave client handling a wavetrain and refraction of waves. The architecture does not inhibit the number of wave clients logged on to the service. Tests on scalability conforming to the delay in updates vs. patch size will be similar to that of terrain updates (refer to section on Terrain Service).

Wakes. Bow wakes and stern wakes have been animated before (Goss, 1990; Donner, 1991) to provide visual cues in a trainee environment. The stern wakes generated by fast amphibious vehicles are enormous and dictate their distance of separation, number of sorties, and their formation for a small landing or surf zone. The methods described in the above papers can be easily implemented using the wave client with the orientation of elevation patch attached to a sea vessel. But this architecture provides a mechanism to use hydrodynamic formulations using continuity and momentum relations like Navier-Stokes and Boussinesq type models to represent wakes and other turbulence in a distributed environment. The work done at IST using Navier-Stokes formulations (Chen, 1995) to represent wakes (see Figure 3) are currently being implemented in a shared environment.

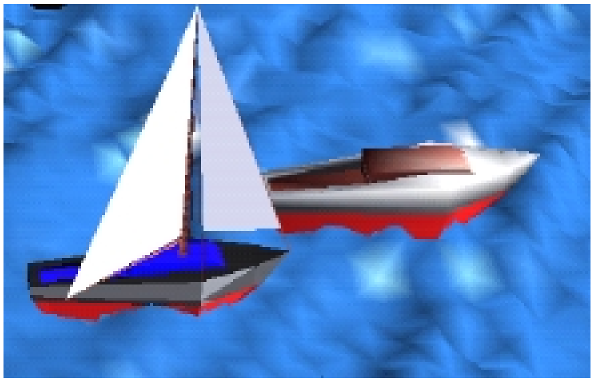

Figure 3. Effects of stern wakes from a speed boat.

It is true that stability and accuracy of finite difference schemes depends on the boundary conditions, grid size and the difference scheme used. However, the shared architecture does provide means to test these important physical phenomenon and help in establishing relationships between database resolutions, delay in updating and other reasons useful in validation and development of dead reckoning schemes. This provides the basis to test new vehicles or other sources of wakes in a shared environment.

Buoyancy. Hydrodynamic stability is important for the motion analysis of the vessels. The equations of motion of the sea vessel including buoyancy and wave forces are coupled with the wave equations. In a distributed simulation, explicit time integration schemes are essential to calculate the state of the sea vessel. In the current implementation, a simplified motion analysis is performed by decoupling the vehicle equations of motion from the waves by querying the Ocean Service for information on the sea state information at the location of interest. The forces of the sea state model, wakes created by other vehicles, drag and wind are some of the external forces necessary to derive accurate heave, pitch and roll motion of vehicle. The vehicle uses depth information by querying the services and makes a smooth transition to land.

Terrain

To represent the underwater terrain, the terrain skin is extended to represent the surface under water. Contour maps or other forms of bathymetry data could be used to draw the elevations. The datum line for sea elevation is chosen as the low tide level. The terrain surface (whether it is above or below water level) is still dynamic and can represent underwater explosions and erosion. The role of the Terrain Service in a shared environment and how it was extended into the Ocean Service is discussed below.

THE SHARED ENVIRONMENT

In order to network a dynamic environment, a set of "services" jointly called the "Shared Environment" has been built. The Shared Environment is covered in detail in Altman (1994) but it simply provides the functionality of communicating changes on the network and the protocols used. For example, Figure 4 shows a typical use of the Shared Environment. On each machine, the Services provide the state information that each client needs within a client/server relationship.

The Entity Service provides the client applications with state regarding DIS entities. Each client is provided with its own private channel to the service which allows the decoupling of applications from each other. In addition, the Entity Service is responsible for dead reckoning entities, performing coordinate conversion, and providing the "heartbeat" for any entities that a client might create. The Entity Service is meant to encapsulate all the details of the DIS standard away from the various client applications. Therefore, duplicate functionality is kept to a minimum and only one application (the Entity Service) is impacted as the standard changes.

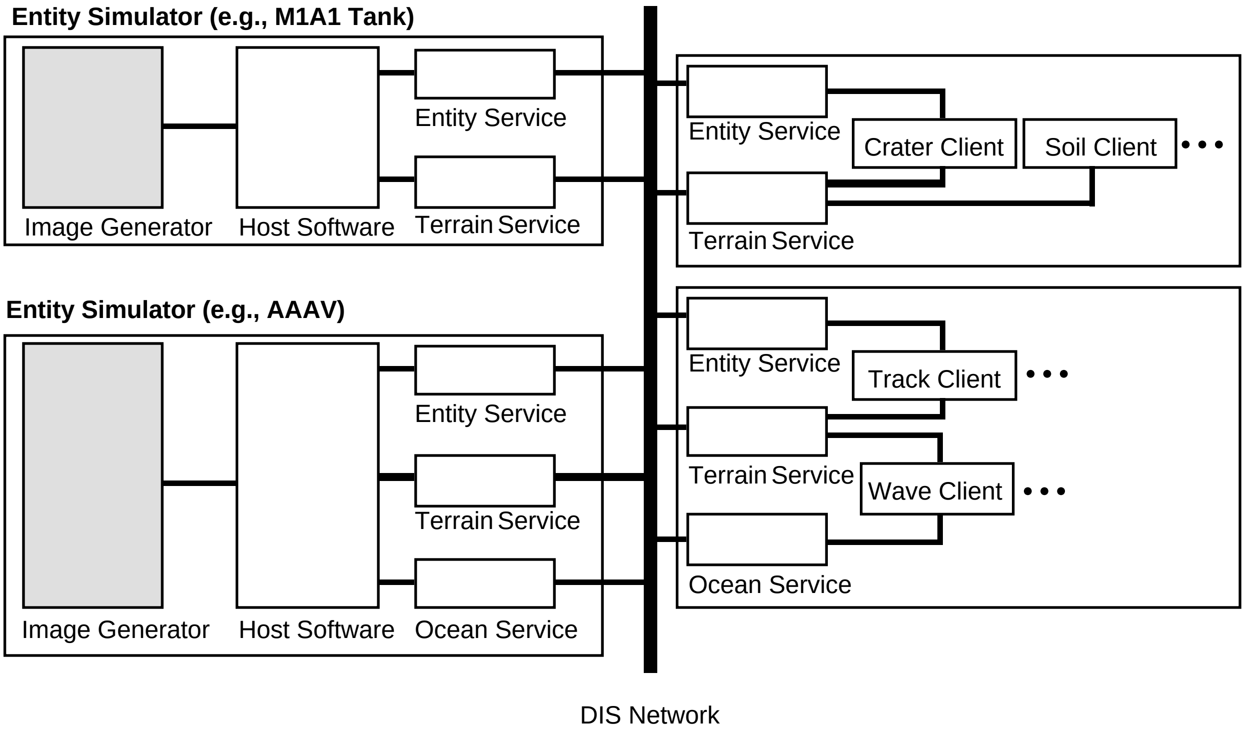

Figure 4. The Shared Environment (with example clients).

The Terrain Service was originally built as part of a dynamic terrain project. Like the Entity Service, it encapsulated networked terrain updates away from the clients. The Terrain Service is responsible for sending terrain changes on the network and for notifying clients when an update has been received (the client can then decide whether or not to requery that piece of terrain depending on its own requirements). To overcome the limitations of polygons and to provide multiple terrain attributes, a database abstraction called the Dynamic Terrain Database (DTDB) was built using higher-order mathematical surfaces.

The Dynamic Terrain Database

The DTDB is a software database abstraction that was constructed to support an arbitrary number of terrain attributes (soil strength, friction angle, etc.) for a particular area in a terrain database. All attributes are layers in the conceptual model of the DTDB (see Figure 5). Therefore, the DTDB answers the question "What is the value of attribute *n* at location *x,y*?" for all queries since mathematical surfaces can be sampled anywhere, not just where there are control points. This mechanism intentionally decouples the form of the query from the underlying representation. Clients of the DTDB do not have, nor should they have, any preconceived notions about how the data are stored.

Currently, the DTDB is developed around higher-order (spline-based) mathematical surfaces. This avoids harsh discontinuities (that can exist in polygonal databases) which must be avoided in complex dynamic systems such as vehicles. Surfaces implemented include bitmaps, standard triangle meshes, uniform non-

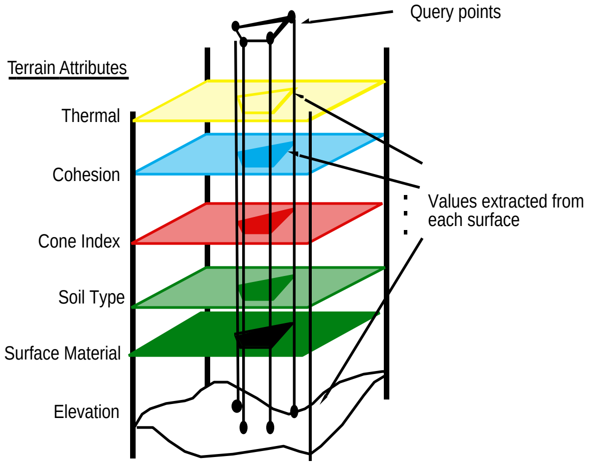

Figure 5. The Conceptual Model of the DTDB.

rational bilinear B-splines, and uniform non-rational bicubic B-splines (Bartels, 1986). However, the DTDB has been designed to allow the addition of other models of terrain. For example, models that use wavelet-based multiresolution surfaces (Martin, 1995) or spatiotemporal data (Worboys, 1994) could be added.

As part of its original development, the DTDB and the Terrain Service were studied to determine their scalability (Altman, 1994). A series of experiments were constructed to test the latency of the query and update processes. The results showed that a linear relationship existed between the number of points queried and/or updated and the latency caused by the process. Altman (1994) concluded that the Terrain Service and DTDB are indeed scalable due to the linear relationship involved (instead of a cubic or quadratic relationship).

The Ocean Service

As a simple experiment, the Terrain Service was modified to create a new Ocean Service that used the DTDB to represent the surface of the water. Using this easy extension, we can begin to study some of the problems in simulating the surf zone environment without some of the costly expense of building an entirely new system.

As an example, the DTDB allows client applications to abstractly represent queries and updates. In an application such as a wave model, the application can represent each wave in an update with only six points (a 2x3 update within the bounding quadrilateral) by using a large resolution value for the update. Although the client will not know it, the DTDB in the Ocean Service will use these six points to generate a temporary surface that will be merged with the actual database surface. This is another example of the advantages of decoupling the query/update mechanism from the underlying representation.

Note that the DTDB currently only supports 2-1/2D surfaces. With some mathematics work, it could be updated to handle other surface types to represent breaking waves. Ultimately, the DTDB could be altered to support 3D volume databases to truly represent the ocean (or the atmosphere) although the memory storage and CPU processing needed would be a concern. However, new techniques such as using wavelets to represent terrain (Martin, 1995) could be used to help minimize the data needed to represent the actual ocean.

The Surf Zone Database

In order to generate a simple demonstration, a sample surf zone database was created. The beach is stored within the DTDB of the Terrain Service, and the ocean surface is stored within the DTDB of the Ocean Service. Using a common technique in computer graphics, the two surfaces intersect each other so that the beach and ocean are properly visible where they should be. This creates some unnecessary polygons for rendering but it allows some early proof of concept work. For example, the fact that the ocean is considered separately from the terrain (rather than part of it as is common in today's databases), the underwater terrain can actually be represented and used by simulations of amphibious vehicles. Figure 1 contains a small portion of the database used and shows the combination of the underwater terrain and ocean surface.

SUMMARY

The above implementation is a proof of concept of how a complex and dynamic surf zone environment can be brought into an interactive arena. The Shared Environment has attempted to provide key structures for distributed simulation and look into some of the future challenges of representing physics-based models, data abstraction, and networking. The nature of the DTDB demonstrates many advantages over previous methods for terrain representation (such as decoupling the data from the clients), supports additional terrain representations (such as methods using wavelets), and provides insight into how a 3D database might be constructed. Furthermore, the testbed provides means for systematic model development, transformation of different ocean models, and representation of floating or partially submerged objects in an interactive and dynamic field required by various water crafts.

REFERENCES

(1995). AAAV: Demonstrating the feasibility of using virtual simulation for test and evaluation. Final report IST-CR-95-32, Institute for Simulation and Training, University of Central Florida, Orlando.

Altman, M., Kilby, M., & Lisle, C. (1994 Sep.). On a shared environment concept for distributed simulation. Proceedings of the 11th Workshop on Standards for the Interoperability of Distributed Simulations, Orlando, Florida, 535 - 543.

Bartels, R., Beatty, J., & Barsky, B. (1986). An Introduction to Splines for use in Computer Graphics and Geometric Modeling. M. Kaufmann Publishers, Los Altos, California.

Biancardi C.G. (1988 August). A simplified mathematical model for an onboard maneuvering simulator. Simulation 51 (2) .

Biano R., Croft R., Leete R., & Littlejohn W. (1996 March). Factors affecting modeling and simulating the surf zone region. 14th workshop on standards for the interoperability of distributed simulations I. Paper 96-14-050. Orlando.

Chen J.X. (1995 Spring). Physically-based modeling and real-time simulation of fluids. Ph.D. dissertation, University of Central Florida, Orlando.

Fournier A., and Reeves W.T. (1986 August). A simple model of ocean waves. Computer Graphics (Proc. SIGGRAPH). 20. (4).

Goss M.E. (1990 May). A real time particle system for display of ship wakes. IEEE Computer Graphics & Applications.

Haeger S.D. (1994). Modeling the littoral ocean for military applications. Proceedings of the 16th I/ITSEC conference. Paper 4-12.

Donner M.E. (1991). The challenges of simulating a Hovercraft ocean environment. Proceedings of the 13th I/ITSC conference.

Kilby, M., Lisle, C., Altman, M., & Sartor, M. (1993). Dynamic environment simulation with DIS technology. Proceedings of the 15th Interservice/Industry Training Systems and Education Conference, Orlando, Florida.

Kinsman B. (1965). Wind Waves. Prentice-Hall, Inc.

Martin, G. (1995 May). Multiresolution surfaces for use in representing terrain. Master's Thesis, Department of Computer Science, University of Central Florida.

Patel M.H. (1989). Dynamics of offshore structures. Butterworth & Co. Ltd.

Peachey D.R. (1986). Modeling waves and surf. Computer Graphics (Proc. of SIGGRAPH). 20 (4).

Ts'o P.Y., and Barsky B.A. (1987 July). Modeling and rendering waves: wave-tracing using beta-splines and reflective and refractive texture mapping. ACM Transactions on Graphics. 6. (3).

Worboys, M. (1994). A Unified model for spatial and temporal information. The Computer Journal. 37. (1). 26-34.

ACKNOWLEDGMENTS

This work is possible due to internal R&D. The authors wish to acknowledge STRICOM for funding the original work on the Terrain Service, and Jim Chen for his original graphics work on the 2D Navier-Stokes method. We would like to thank Lamar Harrell and Ken Hardis for their help in CAD models. We also thank Art Cortes, David Russell and Dan Mullally for reviewing early versions of the paper.