A Curvilinear Cutting Tool for the Analysis of Oceanography Data

Nathalie Rey da Silva

PUCRS - Pontifícia Universidade Católica do Rio Grande do Sul, Faculdade de Informática

Av. Ipiranga, 6681 Prédio 30, Bloco C

90619-900, Porto Alegre/RS, Brazil

nrsilva@inf.pucrs.br

Isabel Harb Manssour

PUCRS - Pontifícia Universidade Católica do Rio Grande do Sul, Faculdade de Informática

Av. Ipiranga, 6681 Prédio 30, Bloco C

90619-900, Porto Alegre/RS, Brazil

manssour@inf.pucrs.br

ABSTRACT

The use of new technologies for the ocean data acquisition produced a considerable increase of information. Thus, scientific visualization tools become essential to aid the oceanographers during the analysis and interpretation of ocean behavior. The main goal of this paper is to present the design and development of a set of tools for 3D visualization of simulated oceanic data, including a new approach for the selection of regions of interest using a curvilinear clipping tool. The implementation is based on a study about the tasks developed by the oceanographers of the Research Center of Petrobras.

Keywords

Visualization, Oceanography, Curvilinear Clipping Tool, Region of Interest.

1. INTRODUCTION

The ocean is a 3D space that changes as the time passes and is inhabited by living beings of all scales (from micro-organisms to whales), being subject to a great variety of force functions, including seismic, thermodynamic, atmospheric, inertial and maritime [Ros89a]. The study of the ocean is important for the climatic and meteorological aspects. There are also great solid mineral deposits, as diamonds, and soluble minerals, as oil and gas, deep in the ocean.

In the Physical Oceanography, many oceanographic parameters are involved, such as salinity, density, temperature, speeds of chains, pressure and pH. These data can be obtained through: simulation, where numerical models are processed to generate information about the evolution of the oceanic features as the time passes; or using data collection, where buoys and drills, and even satellites, catch real data from the ocean in definite intervals of time.

Tools and techniques of scientific visualization are being used for the analysis of the oceanic data. Through the visualization, the scientists can observe the changes of the ocean and aid in the forecasts about its behavior. Visualization techniques are used for physical oceanic data taking into account the kind of data, since it can be scalar, vector or even tensor. Polito [Pol04a] says that we seldom need to describe oceanographic variables as tensors. The geometric disposal and the space-weather scale of the geophysical flow also simplify the treatment of tensors variable. Directions, surfaces and symmetries, due to the macroscopic treatment, reduce the number of interesting components of the tensors. Thus, the majority of the cases deal with vectors and scalars.

With the intention of developing a scientific visualization tool, the Research Center of Petrobras (CENPES) was contacted. The oceanographers of CENPES had shown interest in contributing to the visualization techniques research for oceanographic data generated by simulation. The visualization of simulated data could help these professionals in the understanding of the ocean circulation, such as running and temperature, and of the licensing and ambient impact in the oil extraction. Moreover, these ambient data can be applied for the analysis of the oil spilling and sediments transport.

The main goal of this paper is to present the design and development of a prototype that provides a set of tools for 3D visualization of simulated oceanic data, including a new approach for the selection of regions-of-interest (ROI) using a curvilinear clipping tool. All the implementation is based on a study about the tasks developed by the oceanographers of CENPES. As a case study to experiment the implemented tools, we have used a simulated data obtained at CENPES.

The paper is organized as follows: Section 2 introduce some visualization techniques. The

developed prototype is presented in Section 3. Section 4 describes the curvilinear clipping tool. Conclusions and future work are presented in Section 5.

2. VISUALIZATION TECHNIQUES

There are several 2D and 3D visualization techniques for scalar and vector ocean data. According to Head [Hea97a] 2D techniques are enough for a lot of tasks, revision and planning in data analysis. However, a better understanding of the dataset and a reduction of the analysis time can be obtained including 3D visualization techniques.

A number of powerful techniques were already developed for the visualization of vector data. Techniques like Streamlines and Stream Tubes show the way that a particle follows in a fluid. Methods based on icons, like Hedgehogs [Gai02a], show the direction and the magnitude of a vector field in selected points. Methods as spot noise and line integral convolution (LIC) use texture to show the fluid direction on surfaces [Cro01a]. For scalar data, techniques as color mapping [Ebe06a] and isosurfaces are commonly used.

3. PROTOTYPE DESCRIPTION

Firstly we got in touch with the oceanographers of CENPES. Based in their work and necessities, we have made a research and developed a tool for the analysis of the oceanography data.

The prototype developed in this work allows the 3D visualization of the ocean, with the possibility to load data files in the NetCDF format [Rew90a] generated by the Princeton Ocean Model (POM). The data, as well as system functionalities, are described in the next Sections.

Oceanography Data

The simulation generates a great quantity of information about the ocean, allowing to identify and diagnose problems and to learn more about its behavior. Data models implemented in some programming languages as Fortran or C, are used to do the simulation. After processing, the results are sent to output files, using an appropriate data format, for analysis and visualization. Several data formats are used to store the ocean data. The oceanography applications usually work with HDF [Nat06a], NetCDF [Rew90a], CDF [Cdf05a], among others.

In this work we are using the data generated by the POM model, in the NetCDF format. These data, which had been supplied by CENPES, correspond to a ten days simulation done through Bacia de Campos, in Rio de Janeiro. Its size is 609 MB taking

into account a volumetric grid. For 3D grid generation 630000 cells are used.

POM is a sigma coordinated model where the vertical coordinate represents the depth of the water column. This coordinate system is an important attribute to deal with the existent topographical variability [Mel03a]. The coordinate sigma is defined in equation 1, where z represents the Cartesian coordinate and d represents the depth of the water column, with σ ranging from 0 to -1 (with 21 sigma levels). As a result, the water column always has 21 levels, as illustrated in Figure 1, where η is the surface elevation, h is the height of the water column and σ corresponds to the sigma level.
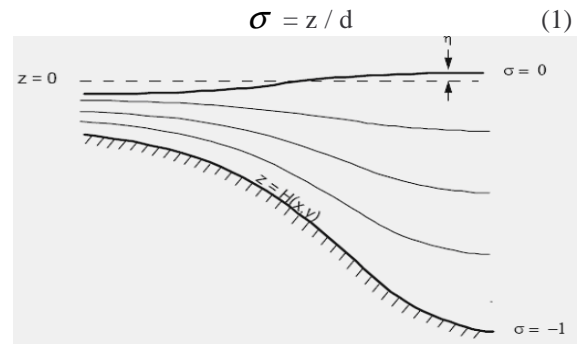
$$\sigma = z / d \tag{1}$$

Figure 1. Sigma Coordinate system [Mel03a].

Implementation and Functionalities

Our prototype was developed for Windows platform using C++ programming language and the Visualization ToolKit (VTK) version 5.0. A library developed by Unidata [Uni05a] was used to read the NetCDF data. The graphical user interface was implemented with the Microsoft Foundation Classes.

Among the implemented functionalities, the application provides vector and scalar visualization techniques. Figure 2 shows an example of the temperature variable visualization using RGB color mapping. In this volume the temperature values are mapped in five ranges between 0 and 30 Celsius degrees, with a color corresponding to each temperature range.
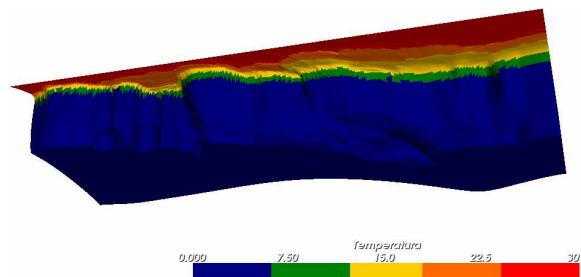

Figure 2. Temperature visualization.

For vector visualization we applied the *Hedgehogs* technique. In this case, the 3D arrows point to vector direction and its magnitude is represented through arrow size and color mapping. Figure 3 shows an example of the image generated by our prototype using the *Hedgehogs* technique. In this case, the speed magnitude is given by the length of the arrow and mapping of colors. The arrow is guided in agreement with the direction of the chains speed.

Interaction methods present in the application make possible to obtain a better understanding of volumetric data. For example, it is possible to translate, rotate and to apply zoom in and zoom out in the volume. The user can also select a ROI through splines manipulation [Bar87a] (Section 4).

Besides the volume visualization, our prototype also provides an animation tool. The simulated data are obtained in time steps that correspond to days of simulation. To analyze this temporary variation, we developed a technique to update the position and the state of each variable along the time. This can be done by reading the oceanic variable in all the time steps and applying the appropriate visualization technique for it. Thus, we provide a way to observe the changes in the variable behavior.
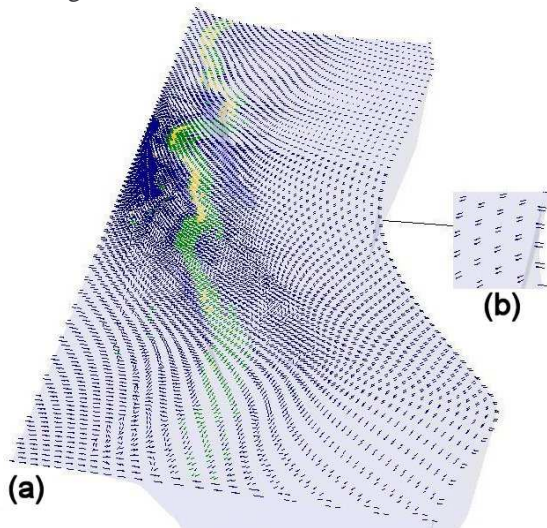

Figure 3. Example of vector visualization for the chains speed of the total volume (b) and applying zoom (a).

4. CURVILINEAR CLIPPING TOOL

The developed clipping technique allows to select a 3D ROI and to interact with the volume providing a better way to analyze the ocean in a 3D visualization. This clipping tool, as well as the other visualization tools, doesn't depend on the NetCDF format. Since a 3D grid is created, any other format could be used if an appropriate reader is used to generate the 3D grid, where the clipping is done.

The Curvilinear Clipping tool allows clipping the volume using splines. In order to facilitate ROI selection, several control points can be manipulated to specify the curve that will be used to clip. In the example presented in Figure 4, the spline is manipulated in a surface that corresponds to the *xy* plane. In this case, several control points of the segments of lines that compose the spline are manipulated by the user. After, the Kochanek interpolation [Koc84a] is used to specify the spline curve, allowing a better control of its shape. Figure 4a shows a spline with 9 control points positioned over the visualized volume where the colors represent the temperature.
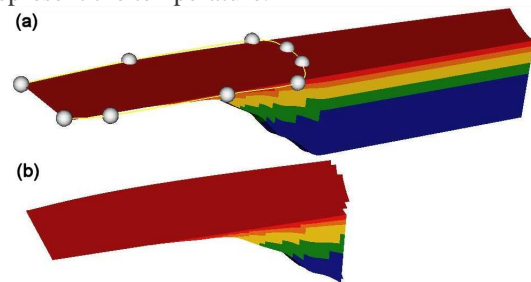

Figure 4. (a) Spline position in the volume after its manipulation; (b) ROI selected with the spline specified in (a).

After spline specification the sub-volume is generated, as shown in Figure 4b. The ROI is defined dragging the spline along the *z* axis, as if it was a knife. The implemented clipping algorithm selects the cells of the volume that collide with the spline and also the cells that are inside the selection area (along the *z* axis in the presented example).

The identification of the cells that collide with the defined points of the splines is a little complex. One of the alternatives that was implemented, is to seek in all cells and verify which one contains the searched for point *p*. For this, cell interpolation functions are resolved for the parametric coordinates. If these coordinates are inside the cell, then *p* is in the cell. To guarantee a better performance, data structures that support spatial search must be used (e.g. octrees and Kd-trees).

Thus, using the spatial search in octrees available in VTK, and then obtaining all the cells that collide with the splines, it is necessary to acquire the other cells that are part of the selection. The grid cells identifiers are increasing ordered from left to right, from bottom to top, and are attributed during the grid structure creation as natural numbers(Figure 5).

To do the clipping, the ends of the spline should be positioned in the left or right border of the surface, as indicated in Figure 4a. Initially, the identifiers of the surface cells that collide with the spline are

obtained, storing them in a list. This algorithm is executed until all cell identifiers of this list are read.

The clipping algorithm works in the following way:

1. Read a value of the cells list.
2. Taking into account the cell identifier from the list, firstly test if the cell right bellow (column) is part of the border and if it is smaller than the largest identifier value of the list. In a positive case, this identifier is part of the clipping. A new identifier is read from list (step 1). Otherwise, follows the step 3.
3. Starting from the cell identifier of the list, test if the cell right bellow (column) is part of the list, or if the neighbor of this cell right below (left neighbor in the case of the left border or right neighbor in the case of the right border) is part of the list. In case it is false, this cell is part of the clipping, if the value is bigger than the smallest value of the clipping list. A new value of the list is read (step 1).

Figure 5. Cells identifiers of the structured model.

Obtaining the cells that are part of the clipping surface, the sub-volume is generated, using the other cells of the volume that are in reason of depth in z. With the clipping, it is possible to analyze the oceanic information through interaction techniques and animation.

5. CONCLUSIONS AND FUTURE WORK

The curvilinear clipping tool allows selecting volume regions in a more accurate way through the manipulation of splines to generate the clipping curve. The developed prototype introduces a new approach to select a 3D ROI, provides an animation tool and an interactive scalar and vector field visualization for NetCDF data, very common in such kind of tools. All the tools were developed in order to support the work of the CENPES oceanographers. Now we are interested in the evaluation of the developed tools through the elaboration and application of tests with users. Our next topics of investigation are: development of new visualization techniques, as isosurface and streamlines, and new readers to other data files format.

6. REFERENCES

[Bar87a] Bartels, Richard H. An introduction to Splines for use in Computer Graphics and geometric modeling. Los Altos, CA : M. Kaufmann, 1987.

[Cdf05a] CDF. CDF Homepage, Available at: nssdc.gsfc.nasa.gov/cdf/cdf_home.html.(June, 2005).

[Cro01a] Crossno, P.; Angel, E.; Munich, D. Case Study: Visualizing Ocean Currents With Color and Dithering. In: IEEE symposium on parallel and large-data visualization and graphics, 2001. Proceedings… (2001).

[Ebe06a] D.S. Ebert, P. Rheingans. Visualization Techniques. Available at http://www.cs.umbc.edu/~rheingan/691V/aug99/index.html. (May, 2006).

[Gai02a] Gaither, K.; Moorhead, R. J. Visualizing Vector Information in Ocean Environments. In: IEEE Visualization, 2002. Proceedings... IEEE Computer Society, (2002). p. 529 – 532.

[Hea97a] Head, M. E. M. et al. Applications of 3D visualizations of oceanographic data bases. IEEE/ MTS OCEANS, New York, v. 2, n. 6-9, p. 1210 – 1215, (1997).

[Koc84a] Kochanek, D.; Bartels, R. Interpolating Splines with Local Tension, Continuity, and Bias Control. Computer Graphics, v. 18, n. 3, p. 33-41, (July 1984).

[Mel03a] Mellor, G. L. Users guide for a three-dimensional, primitive equation, numerical ocean model. 53 p. Relatório Técnico-Programa de Ciências Atmosféricas e Oceânicas, Universidade de Princeton, (June, 2003).

[Nat06a] The National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign. The NCSA HDF Home Page. Available at http://hdf.ncsa.uiuc.edu/. (May, 2006).

[Pol04a] Polito, S. Paulo. Oceanografia Dinâmica. 71 p. Apostila. Departamento de Oceanografia Física, Instituto Oceanográfico da Universidade de São Paulo, São Paulo, (2004).

[Rew90a] Rew, R.; Davis, G. NetCDF: an interface for scientific data access. IEEE Computer Graphics and Applications, Los Alamitos, v. 10 , n. 4 , p. 76–82, (July, 1990).

[Ros89a] Rosenblum, L.J. Visualization of Experimental Data at the Naval Research Laboratory. US Naval Res. Lab., Washington, DC, (1989).

[Uni05a] Unidata. Unidata Homepage. Available at: http://my.unidata.ucar.edu/content/software/netcdf/index.html. (June, 2005).