

ISTV: Interactive Structured Time-varying Visualizer

M.A. Chupa and R.J. Moorhead *

NSF Engineering Research Center, Mississippi State University,

S. Nations and A. Johannsen

Engineering Animation, Inc.,

K. Gaither and R. Vickery

NSF Engineering Research Center, Mississippi State University

I. Overview

The Interactive Structured Time-varying Visualizer (ISTV) visualization system, written at the NSF Engineering Research Center for Computation Field Simulation, Mississippi State University, was originally designed for high-performance interactive visualization of very large oceanographic models. However, its design is general enough for other scientific visualization work, and it has been used with structured datasets from a variety of domains including computational fluid dynamics, electromagnetic simulations and medical imaging. ISTV is written in C using OpenGL for graphics and X/Motif for the user interface. Initially written for SGI IRIX workstations, ISTV has also been ported to Sun Ultra workstations with OpenGL hardware support.

During the past two years, computational grid resolutions in global models we routinely work with have increased by a factor of 16; our visualization products have adapted to this explosive growth by adopting more efficient data access techniques while retaining the ability to work with model output in its native format. ISTV's data model and distributed visualization capabilities are discussed below; the distributed visualization capabilities include a data server using the Message-Passing Interface (MPI) to communicate with the visualization front-end, providing improved access to data using both multi-processor machines and networks of heterogeneous machines.

The simultaneous visualization of several datasets with varying temporal and spatial resolutions in a single view volume is also supported by ISTV; case studies are presented demonstrating this capability. Effective analysis of ocean flow requires the identification of vortical structures; ISTV provides a visualization technique using a two-dimensional mapping with a hue-saturation-value color wheel, giving researchers insight into ocean eddy structures while avoiding the display clutter associated with conventional vector field visualization techniques.

*contact address: Robert Moorhead, P.O. Box 9627, Mississippi State, MS 39762 rjm@erc.msstate.edu, 662-325-2850.

II. ISTV Data Model

ISTV treats each dataset as a logical four-dimensional grid (three spatial dimensions plus time) which can have an arbitrary number of scalar or vector values defined at every grid point. These values can be stored in a file or derived as needed by the software. The division of a dataset into multiple files is hidden from the user through the use of *Logical Grid Files* described below.

The inherent size of large-scale ocean models makes intelligent data handling a crucial part of the system. We can not expect to read the entire grid or solution set into main memory. Furthermore, since many ocean models are laid out on an implicit latitude-longitude grid, ISTV does not require an explicit auxiliary grid file in such cases, reducing the model's memory footprint. In the case of structured curvilinear grids, visualization in the physical space domain requires an explicit grid definition.

ISTV accesses data in small portions so that very large data sets can be processed without overwhelming the computer's memory. The visualization algorithms are implemented to exploit the use of a local memory cache, which also enhances the interactivity of the system.

Convenient controls are provided for selecting a subset of the 3-D space and animating this subset over a specified period of time. The animations may be captured to disk for later replaying.

Another important requirement of the software was the ability to work with data in its native file format. Converting to a new format is costly and problematic, so ISTV does not require this. The code was designed so that a reader for a new file format could be integrated with a minimal amount of effort.

A. Logical Grid Files

To simplify the visualization task for the end-user, ISTV employs *Logical Grid Files* (LGFs). These metadata files define all of the attributes necessary for ISTV to satisfy data requests during a visualization session, and thus provide a layer of abstraction for transparent access to disk-resident data as well as derived scalar and vector fields.

Table 1: LGF data directives

keyword	example usage
gridname	'NLOM Pacific Basin'
gridtype	'huv/add/up'
filetype	'Binary/Float32'
dimension	t0..tf z0..zf lat0..latf lon0..lonf
axis	'lat01' -72.0 0.125
pathname	'/path/to/data/dir/'
filename	'solution.dat' c='var1' 'var2'
components	'H' 'U' 'V'

Table 2: LGF: other directives

keyword	example usage
variable	'Luv' 'float' 'v2len(u,v)'
mask	'land' 1.0e10 0.0
value	'reference depth' 6500

Since they are written as ascii text, LGFs also provide a convenient location for other information pertinent to a dataset, since many of the file formats ISTV works with do not provide for self-contained data description. This has proven useful in documenting datasets that are composed of a large number of headerless binary files; this is particularly important when the user is not intimately acquainted with the data being visualized. The LGF approach provides a method for organizing the data that allows the user to work with the full spatial and temporal extent of the data without explicit reference to specific data files or to grid indices or ranges.

Keywords in LGFs pertaining to raw data description and handling are summarized in Table 1. The **gridname** directive provides a label for a dataset, while the **gridtype** and **filetype** directives allow ISTV to target a particular set of I/O and calculation routines tailored to a specific model. The **dimension** and **axis** directives specify the grid extents, initial values and step sizes for grids. The actual contents of disk files are described through the **pathname**, **filename** and **components** directives.

Other directives in LGFs are summarized in Table 2. The **variable** directive allows the user to specify the functional form for a derived data type. Since these are specified in the LGF, the use of derived data types is identical to that of actual components resident on disk (except that the derived data will be calculated on-the-fly). The **mask** directive allows the user to define a value read from a file as a mask value (e.g., corresponding to land), and the **value** directive allows for the definition of certain model specific constants.

B. Calculator Module

The **variable** directive in the LGF seen above provides the interface to ISTV's calculator module. Calculation func-

tions provide both basic algebraic manipulations of raw data fields, as well as model-specific functions to derive other quantities of interest.

Once defined in the LGF, a variable may be used in conjunction with the colormap tool, and can also be used as an axis in the 3D view window. Derived fields set up with the calculator can also be used to provide a variety of physical space mappings; for example, an ocean model calculated on a rectilinear latitude-longitude grid can be mapped onto a spherical surface for display.

C. Distributed Data Access

Normally ISTV runs as a single job on a single computer. Quite often, however, the data files reside on a central file server rather than on the visualization machine. In this case, ISTV has a data server / visualization client option in which each task executes on the appropriate machine and they communicate via a network. Starting up the software is slightly different for this configuration, but is otherwise transparent to the user.

The client / server version of ISTV is implemented using the MPICH message-passing library [1, 2]. Since the size of many datasets exceeds the local storage on a workstation, or in some cases may reside on a machine with a different floating point representation (e.g., non-IEEE Cray machine), the use of MPI allows for transparent access to the data over a network. MPI can also be used to start up distinct processes on a multi-processor workstation. In cases where the raw disk data has been compressed, the use of the data server can significantly accelerate the visualization process, since decoding can proceed while other visualization tasks are performed by the front-end client machine.

The current implementation uses the p4 communicator in MPI and has been used with the data server running on a variety of hardware, including SGI Power Challenge, SGI Origin 2000, Cray C90, and Sun HPC 10000 machines. By locating the data server process on a remote machine, ISTV has been successfully used to visualize large ocean models on modestly equipped desktop workstations. ISTV has been used to interactively work with and produce animation sequences from datasets consisting of 1560 timesteps of data on a $989 \times 657 \times 6$ computational grid.

III. Visualization Capabilities

A. Scalar Field Visualization

There are two basic geometric objects which can be displayed using ISTV: *layers* and *isosurfaces*. A layer is a surface defined along a constant grid index. An isosurface is defined by a constant value within a scalar field defined

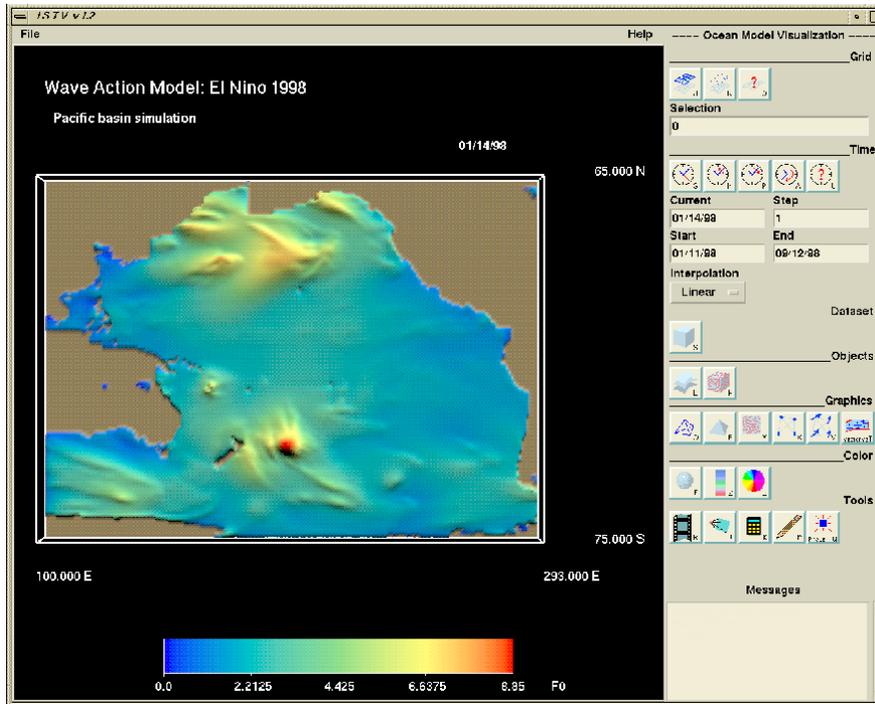


Figure 1: ISTV's Motif graphical user interface

by a data variable (i.e., vector magnitude = 0.75). Multiple layers and multiple isosurfaces can be created and displayed for each grid.

Graphical representations available for each geometric object include *wireframe*, *solid surface*, *contour lines*, and *mini-cubes*. For every layer and isosurface, all or none of these representations may be displayed at the same time. Controls are provided for adjusting the appearance of each representations, such as offsetting contour lines from a layer surface.

Three coloring schemes are available to paint each object: *solid color*, *color map*, and *color wheel*. Using solid color, a single color is given to the object. With color map the object is colored based on an independent variable. The color wheel described below uses the HSV color space to color objects based on two independent variables (e.g., 2D flow components). A lighting model may be applied to the color operation, and there are options for manipulating surface normals, etc. in the lighting model.

B. Vector Field Visualization

1) Standard Mappings

Vector visualization in ISTV begins with the definition of two or three variables to use as vector components; mappings provided include *tufts*, *streamlines*, *streaklines* and

pathlines. The tufts module displays geometry showing the vector magnitude and direction at grid locations. To avoid clutter, the field can be subsampled (independent of any subsampling done in reading the data), and a variety of methods are provided for scaling and displaying the tufts. The latter three modules, which can collectively be called flowline visualization, are initialized by selecting a set of seed points or rakes of seed points. Streamlines are lines tangent to the velocity field at a given timestep. Flow visualization of time-varying vector fields is supported by the pathline and streakline modules. Pathlines trace the trajectory of a single particle released from a fixed location. Streaklines join the positions of all particles that have been previously released from a fixed location [6].

2) Color Wheel Visualization

One of the challenges in rendering flow field information from high-resolution datasets is the tendency of tuft and flowline vector representations to clutter the display. An effective alternative mapping used in ISTV is the **color-wheel** mapping [5] of vector field information, as suggested by Hall [3]. To display the flow field on a particular layer, the colorwheel shown in Figure 2 maps the vector magnitude to saturation and intensity, while mapping the vector direction to hue in an HSV color space. Both linear and logarithmic scalings of the vector field magnitude

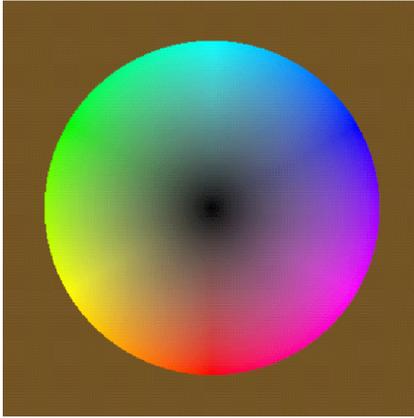


Figure 2: 2D flow color wheel

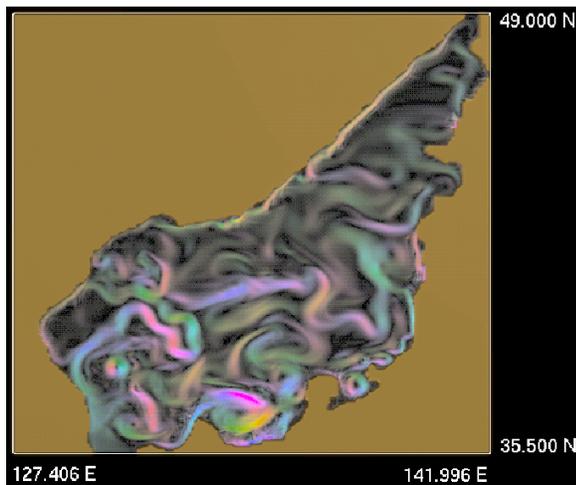


Figure 3: Sea of Japan (NLOM) color wheel mapping

are provided, allowing for effective visualizations of vector fields of varying dynamic ranges. Figure 3 shows the color wheel mapping applied to the flow field in an NLOM Sea of Japan model.

C. Image Annotation and Animation

Support for inserting text labels (including data-dependent labels such as time stamps) is provided by ISTV; these labels can be pinned to a specific screen location, or can be defined in the 3D space of the model so that it will move as the user navigates in the view window.

Still frames and sequences of frames can be saved to disk for postprocessing or display; a full-screen key toggle allows the user to temporarily hide the user interface panel from view while frames are being rendered.

IV. Case Studies

A. Navy Layered Ocean Model

We have used ISTV frequently with the Navy Layered Ocean Model (NLOM), both in regional simulations (e.g., Sea of Japan, Pacific Basin), as well as global ocean simulations [4, 8]. The current ($1/32^\circ$ resolution) global ocean model runs on a computational grid of 8192×4608 in the horizontal, and has six vertical layers. With three primitive components (H , the layer thickness deviation, U and V , the horizontal transport velocities) stored on disk as single-precision (32-bit) floating point values, a single timestep of data consumes more than 2.5GB of storage. Since working with a full timestep of data exceeds the 2GB file size limitation of 32-bit operating systems, we have recently modified ISTV to run in 64-bit mode on IRIX and Solaris.

The subsample/subset-on-read features of ISTV have been extremely useful in working with the NLOM global model. This allows us to select a smaller region of interest from the global dataset, or to "browse" the entire model at reduced resolution. Since the number of pixels available on a monitor is substantially smaller than the model resolution, these reduced resolution views provide interactive rates without sacrificing rendering detail. Another performance-enhancing feature is the generation of two resolutions when data is loaded; one will be displayed when the user is idling, while a coarse view is displayed when the keypad or mouse is being used to move the model about. The user has complete control over the resolution used in both the full- and coarse-resolution views. Using reduced resolution, the ISTV run that generated the image reproduced in Figure 4 had a total memory footprint of 200MB; only the top layer of six total layers was loaded, and this layer was subsampled by a factor of four.

In this view, the sea surface height field is used to produce a shaded relief, while the flow velocity magnitude is mapped to the scalar colorbar. ISTV allows any raw or derived variable to be used as one of the spatial display dimensions. By using another variable as the source for a scalar colormap, comparisons are easily made between the two fields.

B. Multiple-model Visualizations

Several distinct datasets can be simultaneously loaded into ISTV by opening multiple LGFs. This presents several opportunities to enrich the visualization process; for example, we can visualize ocean model output in a region and use terrain databases (e.g., GTOPO30 [7]) to render land regions that have been masked out of the ocean model data. This can provide a richer context for the visualization without requiring any modification of the two datasets. In Figure 5, we have rendered NLOM global $1/16^\circ$ data from the Gulf of Mexico and western Atlantic along with terrain data from the GTOPO30 dataset.

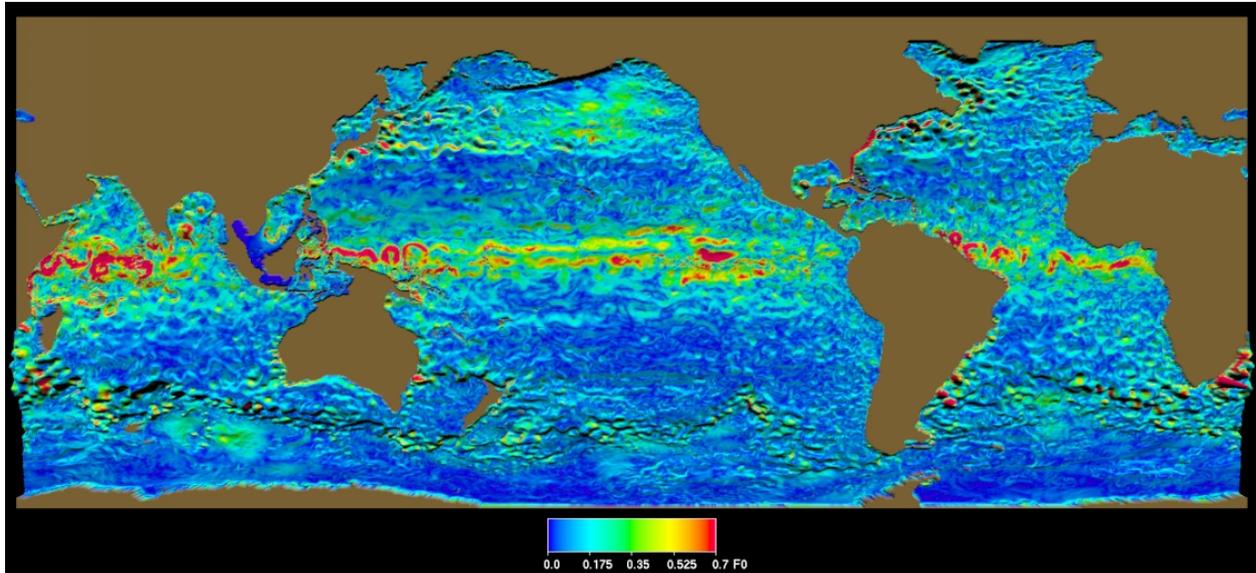


Figure 4: NLOM Global 32nd degree model; velocity magnitude on sea surface height.

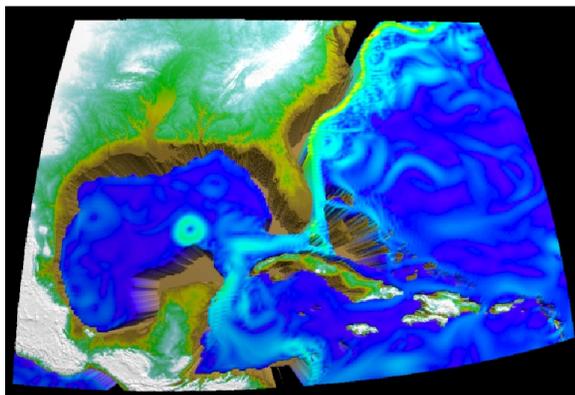


Figure 5: Multiple grids: NLOM and GTOPO30.

Note that in this image we have used the calculator functions that map the regular lat/lon grid onto a sphere, and have replaced the three spatial display variables with the sphere-mapped variables.

Another useful application for multigrid visualization is the examination of wind fields that are used as forcing functions in ocean models. Figure 6 shows NLOM output rendered with the colorwheel for the flow field, and the independent winds dataset rendered using vector tufts.

ISTV's multigrid capability has also been used to produce simultaneous visualizations of nested ocean models. An investigation conducted over the past year involved a Wave Action Model simulation of the January, 1998 El Niño event in the Pacific. Three nested compu-

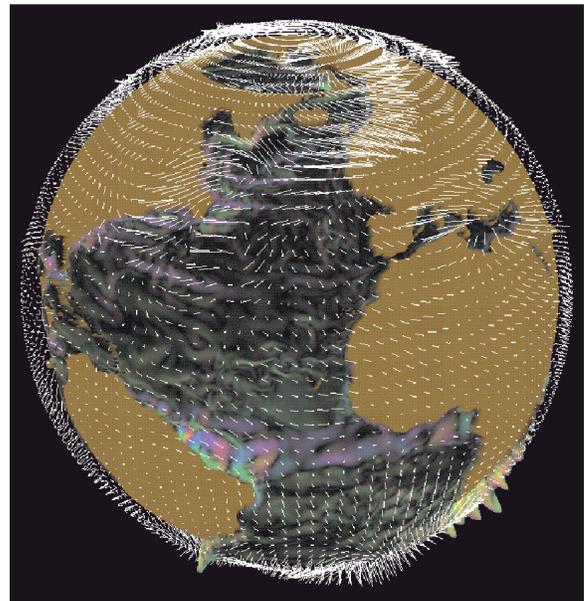


Figure 6: Multiple grids: NLOM and forcing winds

tational grids were set up for the Pacific basin, the north-east Pacific region, and the Los Angeles area subregion, with resolutions 194×141 at 11° , 57×77 at $1/4^\circ$, and 133×73 at $1/24^\circ$, respectively. The data were computed hourly for a total of 246 time steps; seven floating-point fields were written out at each grid location. ISTV was employed to generate animations showing the three nested grids mapped onto a sphere; these animations were useful in tracking features in the wave height field as they passed from one computational grid to another. A portion of the Pacific showing the three nested grids is shown in Figure 7; as in Figure 5, the grids have been mapped onto a sphere for display.

V. Acknowledgments

The work which lead to the original versions of ISTV was funded by the Strategic Environmental Research and Development Program and sponsored by the Advanced Research Projects Agency [5]. Additions and modifications have been funded by the ONR Naval Ocean Modeling Program and by the DoD HPC Modernization Program through the PET programs at the CEWES/MSRC and the NAVO/MSRC.

Other contributors to ISTV who previously worked in the Scientific Visualization Thrust at the Mississippi State University Engineering Research Center include: Jonathan Barlow, Wai-Tat Wong, Rachael Zhang, Sandeep Jolley, and Professor Bernd Hamann.

Navy Layered Ocean Model data provided courtesy of Harley Hurlburt, Alan Wallcraft, Joe Metzger, and Pat Hogan, Naval Research Laboratory, Stennis Space Center, Mississippi. Wave Action Model data provided courtesy of Dr. Robert Jensen, Corps of Engineers Waterways Experiment Station, Vicksburg, Mississippi.

The GTOPO30 data are distributed by the EROS Data Center Distributed Active Archive Center (EDC DAAC), located at the U.S. Geological Survey's EROS Data Center in Sioux Falls, South Dakota. Refer to <http://edcwww.cr.usgs.gov/landdaac/gtopo30/gtopo30.html>.

We would also like to recognize Peter Flynn of the visualization lab, Naval Research Laboratory, Stennis Space Center, Mississippi for his extensive testing and valuable feedback on ISTV's use with the Navy Layered Ocean Model. He has facilitated much of the interaction that produced the case studies reported herein.

Further information and downloading instructions for ISTV are available at <http://www.erc.msstate.edu/research/thrusts/scivi/>.

References

- [1] W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A high-performance, portable implementation of the MPI message passing interface standard. *Parallel Computing*, 22(6):789–828, September 1996.
- [2] W.D. Gropp and E. Lusk. *User's Guide for mpich, a Portable Implementation of MPI*. Mathematics and Computer Science Division, Argonne National Laboratory, 1996. ANL-96/6.
- [3] P. Hall. Volume Rendering of Vector Fields. *The Visual Computer*, 10(2):69–78, 1993.
- [4] H.E. Hurlburt and J.D. Thompson. A Numerical Study of Loop Current Intrusions and Eddy-shedding. *Journal of Physical Oceanography*, 10:1611–1651, 1980.
- [5] A. Johannsen and R. J. Moorhead II. AGP: Ocean Model Flow Visualization. *IEEE Computer Graphics and Applications*, 15(4):28–33, July 1995.
- [6] D.A. Lane. Scientific Visualization of Large-Scale Unsteady Fluid Flows. In G.M. Nielson, H. Hagen, and H. Müller, editors, *Scientific Visualization: Overviews, Methodologies, and Techniques*, pages 125–145. IEEE Computer Society, Los Alamitos, CA, 1997.
- [7] K.L. Verdin and S.K. Jenson. Development of Continental Scale DEMs and Extraction of Hydrographic Features. In *International Conference/Workshop on Integrating GIS and Environmental Modeling*, volume 3, Santa Fe, New Mexico, January 21–25, 1996. National Center for Geographic Information and Analysis (NCGIA).
- [8] A.J. Wallcraft. The Navy Layered Ocean Model User's Guide. NOARL Report 35, Naval Research Laboratory, Stennis Space Center, MS, 1991.

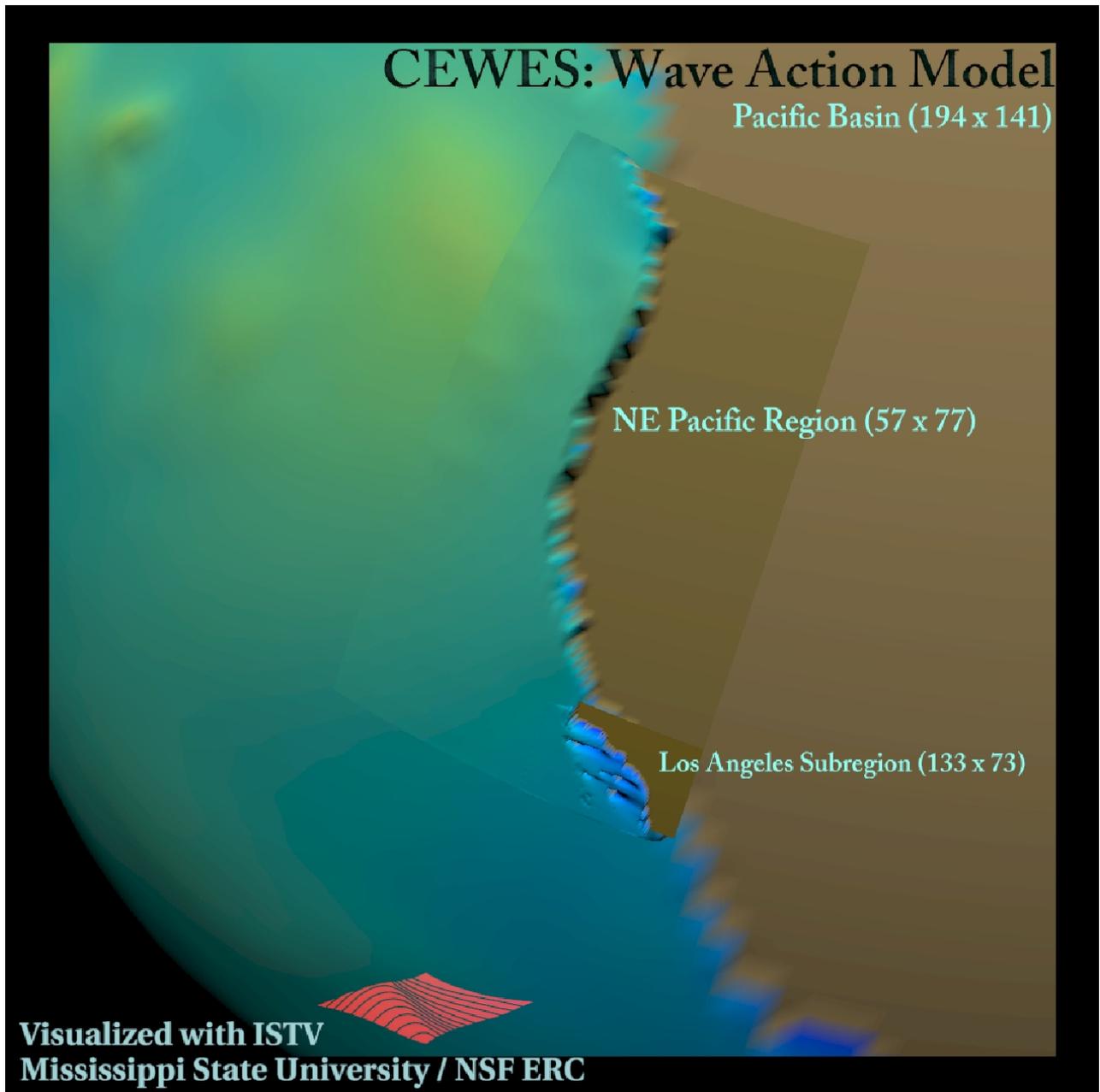


Figure 7: Multiple grids: WAM nested grids