Interactive Correlation Analysis and Visualization(ICAV) of Large Scale Climate Data

Jinrong Xie

University of California, Davis

A) Project Overview

- One component of Parvis
- Provide new ways to use 3D images in climate analysis.
- Uses GPU's in parallel to provide interactivity.

Participants:

Argonne NATIONAL LABORATORY

NCAR

Sandia National Laboratories

Pacific Northwest NATIONAL LABORATORY

UCDAVIS UNIVERSITY OF CALIFORNIA

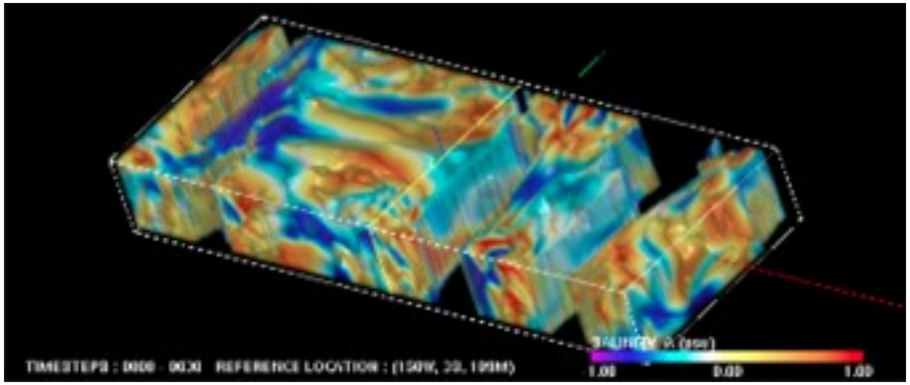
Sponsor:

U.S. DEPARTMENT OF ENERGY | Office of Science

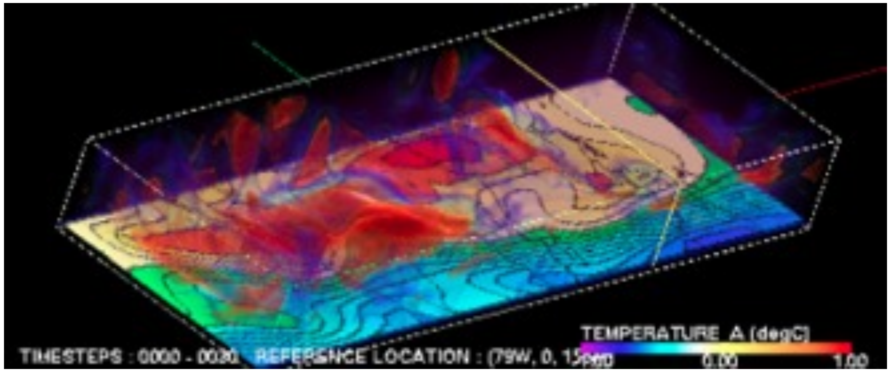
B) Science Lesson
F) Visualization and Analysis

- Volume Rendering.
- Animation of analyzed results such as self/cross correlation of salinity and temperature.

correlation of ocean temperature and salinity visualization


SALINITY A (psu)
TIMESTEPS : 0000 - 0030 REFERENCE LOCATION : (150W, 30, 100M)
-1.00
0.00
1.00

volume rendering



NCL image overlaid in 3D

C) Parallel Programming Model D) Computational Methods

- On GPUs: C/C++, OpenGL, GLSL, CUDA
- On CPUs: C/C++
- The application runs on Linux/Mac/Win.

- Now on a PC.
- In the future on small cluster and parallel supercomputers, because of model sophistication and the scale of the dataset.

E) I/O Patterns and Strategy

- **The Input I/O**
  - The main bottleneck
  - program needs to read large scale climate simulation data (netcdf file of gigabytes to petabytes, would use pnetcdf, MOAB to handle structured and unstructured grids cross processors in the future). They are time-varying multivariate volume data.
- **The Output I/O**
  - mainly rasterized images
  - neglectable as the data size is relatively limited.

G) Performance

- We usually measure the scalability (efficiency, speedup…).

- Performance measurements tools:
  - CPU side: Intel Parallel Studio
  - GPU side: NVIDIA Parallel Nsight 2.0, NVIDIA Compute Visual Profiler

- The bottleneck could be I/O, communication, load balancing, etc.

H) Tools

- How do you debug your code?
- What other tools do you use?

→ Use preliminary methods, such as print, to debug the code.

I) Status and Scalability

- How does your application scale now?
  - The visualization kernel can interactively render the dataset around 6GB on a single PC with newest GPU.

- What are your top 5 pains? (be specific)
  - 1.Parallel programming techniques(Reduce communication times, resolve memory conflicts…)
  - 2. Parallel I/O
  - 3. Load balancing (data partition…)
  - 4. Efficient rendering algorithm.
  - 5. Debug techniques.

J) Roadmap

- We will study the visualization requirements of the climate simulations, and the limitations of the computing facilities.

- Due to the scale of the problem, we will have to integrate the visualization support into the petascale simulations to see the modeled phenomena.