Scientific Visualization with ParaView

Sept 5, 2018

The Visualization Process

ParaView’s role in the Visualization Process

But first … Using the IU Research Desktop

IU HPC Desktop System(s)

- Research Desktop (aka RED) on Carbonate
  - Requires Carbonate Account
  - https://reg.iu.edu/Accounts
  - Requires “ThinkPad” software
  - https://reg.iu.edu/id/top
  - Available for:
    - MS Windows
    - Apple OS X
    - Linux
  - Provides:
    - Ready-to-use software
    - Desktop interface to HPC system
Scientific Visualization Workshop Series
Scientific Visualization with ParaView
William Sherman

Start ThinLinc and login

- Logging onto RED:
  - Start ThinLinc
  - Server: red.uits.iu.edu
  - Username: <your IU ADS name>
  - Password: <Your passphrase>

ThinLinc connects to Duo login

- RED uses the Duo two-factor login
  - Provide one of the confirmation options

Now connected to RED (IU Research Desktop)

Two ways to start ParaView on RED

- 1) Using the Applications menu:
  - Applications -> Visualization -> paraview
- 2) Using the command line terminal shell:
  - load module paraview
  - paraview

ParaView running

A Word about ParaView
Scientific Visualization Workshop Series
Scientific Visualization with ParaView
William Sherman

What is ParaView?
- An open-source, scalable, multi-platform visualization application.
- Support for distributed computation models to process large data sets.
- An open, flexible, and intuitive user interface.
- An extendable, modular architecture based on open standards.
- A flexible BSD-2 license.
- Commercial maintenance and support.

ParaView Application Architecture

ParaView & VTK
- General Purpose visualization tools from Kitware:
  - **VTK**:
    - Program your own tool with standard techniques
    - Can be scripted
      - Python, TCL, Java
    - Can be integrated into web notebooks
  - **ParaView**:
    - Typical Desktop User Interface
    - Can be integrated into Web interfaces
    - Extensible
    - Use the VTK library
    - Can be scripted
    - C, C++, Python
    - MPI & OpenMP available

What Can ParaView Do?
- Interactive Desktop Exploration
- Parallel computation using HPC systems
- Batch BigIron (HPC) rendering
- Interactive & Immersive Interfaces
  - AVL large format displays
  - Consumer-grade HMD displays
- Customizable
  - C++ extensions
  - Python scripting
  - Web interfaces

ParaView Development
- Started in 2000 as collaborative effort between Los Alamos National Laboratories and Kitware Inc. (led by James Ahrens).
- ParaView 0.8 released October 2002.
- ParaView 1.0 released in November 2003.
- ParaView 2.0 released in March 2005.
- September 2005: collaborative effort between Sandia National Laboratories, Kitware Inc. and Clarity, Inc. to write user interface to be more user-friendly and developquadricale analysis framework.
- ParaView 3.0 released in May 2007.
- ParaView 4.0 released in June 2013.
- ParaView 5.0 released in January 2016.
- ParaView 5.5 released in April 2018.

Basic Usage
Creating a Cylinder Source

1. Go to the Source menu and select Cylinder.
2. Click the Apply button to accept the default parameters.
3. Increase the Resolution parameter.
4. Click the Apply button again.

Simple Camera Manipulation

- Drag left, middle, right buttons for rotate, pan, zoom.
- Also use Shift, Ctrl modifiers.

Pipeline Object Controls

- Apply
- Reset
- Delete
Creating a Cylinder Source

1. Go to the Source menu and select Cylinder.
2. Click the Apply button to accept the default parameters.
3. Increase the Resolution parameter.
4. Click the Apply button again.
5. Delete the Cylinder.
Real Data

Data Types

- vtkImageData
- vtkRectilinearGrid
- vtkStructuredGrid
- vtkUnstructuredGrid
- Real Data

Geometry Representations
- Surface
- Surface with Edges
- Volume
- Wireframe

Pseudo Data

- Wavelets:
  - Another "dataset" that doesn't require a file
  - Good for quickly testing volume data features of ParaView
  - Size/Center can be adjusted (we'll stick with the defaults)

Supported Data Types

- vtkPolyData
- vtkUnstructuredGrid
- Real Data

Data Types (not limited to 2D)

- vtkPolyData
- vtkUnstructuredGrid

Geometry Representations

- Outline
- Point Gaussian
- Points
- Slice
Apply a Filter

1. Make sure that Wavelet1 is selected in the pipeline browser.

2. Select the contour filter.

3. Parameters (default)
   - Data: RTData
   - Value: 157.09 (i.e. middle of the Range)
   - Enable “Compute Scalars”
   - Scalars, yes
   - Isovalue

Apply a Filter

1. Make sure that disk_out_ref.ex2 is selected in the pipeline browser.

2. Select the contour filter.

3. Change parameters to create an isosurface at Temp = 400K.

4. Click

5. Color by “Pres”

Apply a Filter

• Why no color?
  • A: because ‘white’ is at the center of this color map, (and we contoured in the center)

Apply a Filter

• Why no color?
Apply a Filter: Clip

- Now clip away to see inside:
  - Select Wavelet1
  - Click on "Clip" icon
  - When "Clip1" is selected, can move the clipping plane
  - Click after making adjustments

Apply a Filter: Clip

- Now clip away to see inside:
  - Disable the "eye" icon on "Contour1" to see only the clip
  - To change the clip to another orthogonal, use:
    - ctrl/cmd-X
    - ctrl/cmd-Y
    - ctrl/cmd-Z

Reset ParaView

Real World Examples
(from Indiana University researcher)

AVL Visualization: Nuclear Pasta

- Condensed Matter Astrophysics (Nuclear Pasta)
- For Charles Horowitz, Cyclotron Physics
- Task: Produce analysis and videos for public

Nuclear Pasta: The Data

- Data provided as locations of Neutrons
  - Points can be loaded directly into ParaView, but...
  - A volume is better for the desired representation
  - Visualization expert transforms the data from points to density volume
  - Using "Gaussian Splat" to create neutron density
  - Simulation evolves over time:
    - Temporal data
    - each time is separately processed
    - ParaView understands time-varying data
    - Single file
    - Multiple files with numeric filenames
Nuclear Pasta: The Data

- Load Nuclear Pasta data
  - File ➔ Open
  - Browse to (or type):
    - /lustre/2018/jrhel7/paraview/Data/Horowitz
    - 500_40...
    - select .vtk
    - NOTE: the .. is the time sequence
  - OK

Basic representations

- Basic visualization techniques:
  - Points
    - NOT: only shows the outer surface
  - Contour (Gaussian)
    - New features of ParaView 5.0
    - NOT the same as slicing – more like sized points
  - Slice
  - Wireframe
  - Surface
  - Surface w/ Edges
  - Volume
    - Adjust color maps (opacity)

Animating through time

- Simple animation:
  - Press the "play arrow" ➔ !

Regional Connectivity

- Enhancing the visualization
  - Contour Filter (0.035)
  - Color:
    - Discretize to 12
    - Choose a varying colormap
    - Set range to 0-40
    - See "Information" tab
  - Play to follow changes over time

Reset ParaView

- ParaView provides the ability to save the state of your visualization:
  - File ➔ Save State...
  - I've already saved the final state, so restore to where we were:
    - File ➔ Load State
    - Select "Horowitz_..0000*.pvs" ➔ OK
    - Use "File Name From State" ➔ OK

But wait!

- ParaView provides the ability to save the state of your visualization:
  - File ➔ Save State...
  - I've already saved the final state, so restore to where we were:
    - File ➔ Load State
    - Select "Horowitz_..0000*.pvs" ➔ OK
    - Use "File Name From State" ➔ OK
Globular Clusters: The Data

- Vesperini Particle location data
  - Original data
    - Columns of ASCII encoded numbers
    - Output from Starlab N-Body simulation
  - Data set: (BCollision_05/data)
    - Point data
    - with identification ("id")
    - Multi-time step
    - JH (Jongsuk Hong data)
  - Binary/Trinary star collisions

Particles in Motion

- Where are they?
  - Center the data with the "Camera Reset" button
  - (or "Zoom to Data")
  - Animate!

- What happened?
  - How did the stars interact?
  - Trace their paths:
    - Add: Temporal Particles To Pathlines
  - NOTE: creates two new entries in the pipeline
  - Paricles:
    - Mask Points: off
    - Max Track Length: 150
    - Rep: 3D Glyphs
    - Glyph Type: Sphere
    - Coloring: id

AVL Visualizations: Globular Clusters

- Globular Cluster Evolution
  - For: Enrico Vesperini
  - Task: Produce analysis and videos for public
Particles/Paths in Motion

- Particles:
  - Make them visible
  - Add a "Tube"
  - Vary Radius: off (default)
- Paths:
  - Select "Tube"
  - Add filter: "Tube"
  - Radius: 0.15
- Animate!

- Animating Paths:
  - Jump in time?
  - No path
  - Path is calculated on-the-fly
  - Long path?
  - Set Max Track Length: 1000
  - Reanimate from the beginning

ParaView Sample Data: Bluntfin

- File: Open "Data/ParaviewData/Bluntfin.vts"
- Data:
  - Density
  - Momentum
  - Inelastic Energy
- Comparing variables

• Comparing variables?
  - Multi-view layout:
    - Split Horizontal
    - Choose new "Render View"
  - Can independently adjust parameters and visibility
  - Can connect the camera between views

NOTE: mouse required for right-click on "ThinLine" on Mac
ParaView Sample Data: Bluntfin

- Comparing variables?
- Multi-view layout:
  - Split horizontal
  - Choose new “Render View”
  - Can independently adjust parameters and view
  - Can connect the camera between views
  - NOTE: mouse required for right-click over thickness

More Information

- Help Menu
- The ParaView User’s Guide
  - http://paraview.org/Wiki/ParaView/Users_Guide/Table_Of_Contents
- The ParaView web page
  - www.paraview.org
- ParaView mailing list
  - paraview@paraview.org

Future Steps

- Color Map opacity:
  - Transfer functions (we will talk about this later)
- Enhanced animation:
  - Camera moves
  - Parameter controls
  - Each rendering
- Python scripting:
  - Creating objects
  - Manipulating data
- Data queries & selection:
  - Select data based on parameters
  - Manipulate/inspect particular data

Questions?

(or perhaps a video)

Fin