Data Analysis and Visualization Computing Requirements
A Case Study

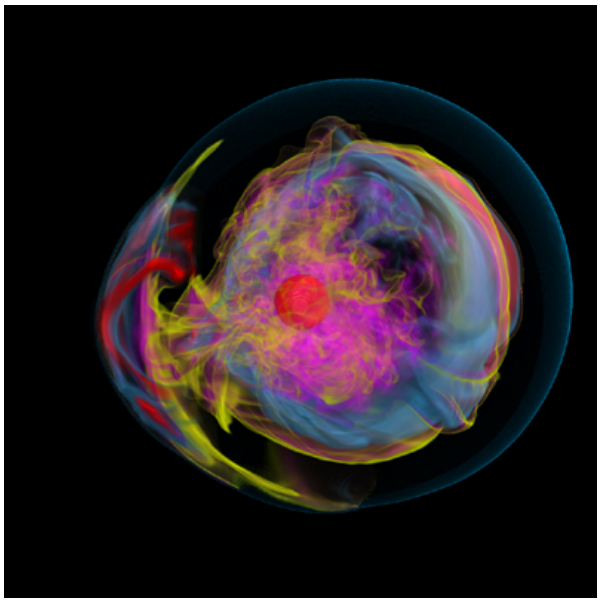
Kwan-Liu Ma
University of California, Davis
SciDAC Institute for Ultrascale Visualization

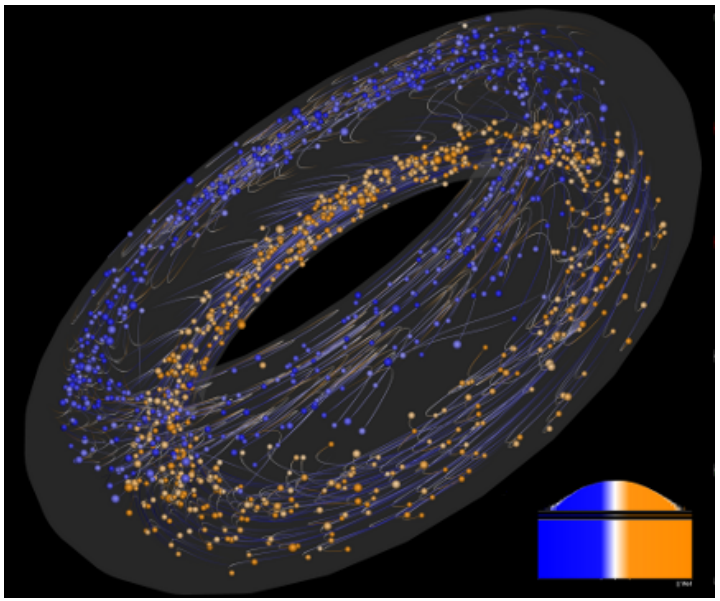
Outline

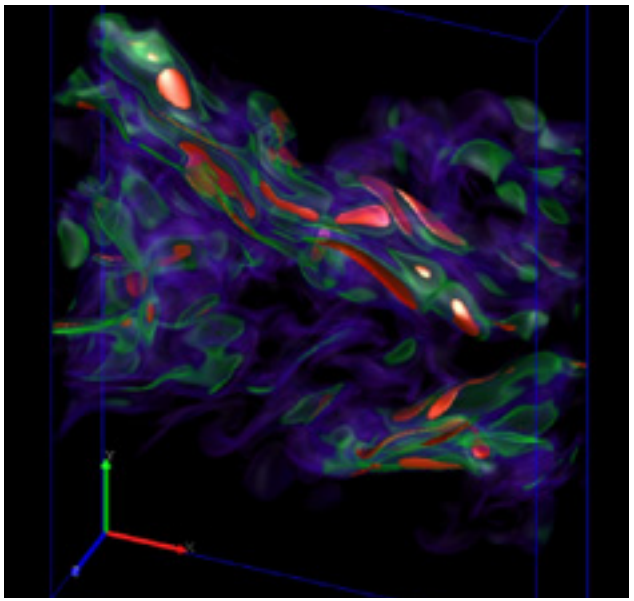
- SciDAC Institute for Ultrascale Visualization
- Visualization Solutions for Turbulent Combustion Simulations
- A Case Study on Particle Trajectories Data Visualization

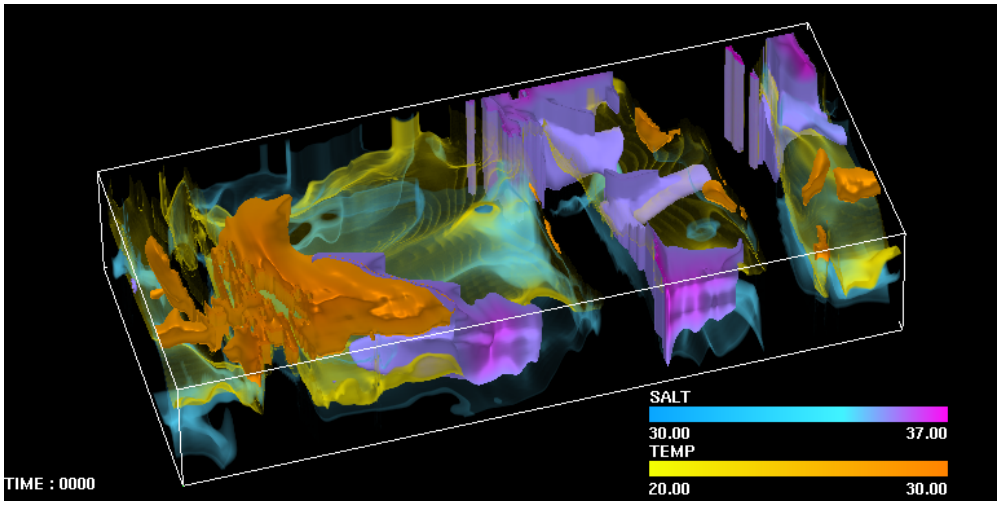
UltraVis

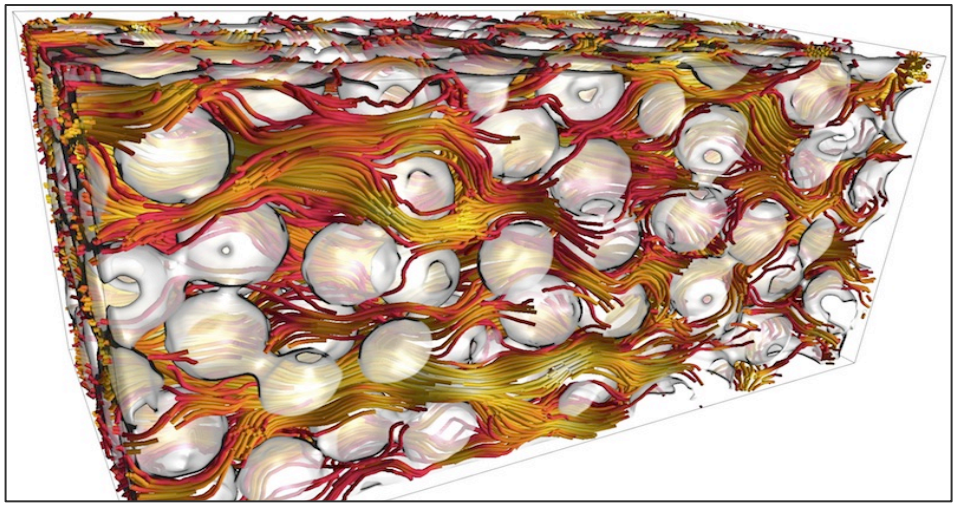
SciDAC Institute for Ultrascale Visualization





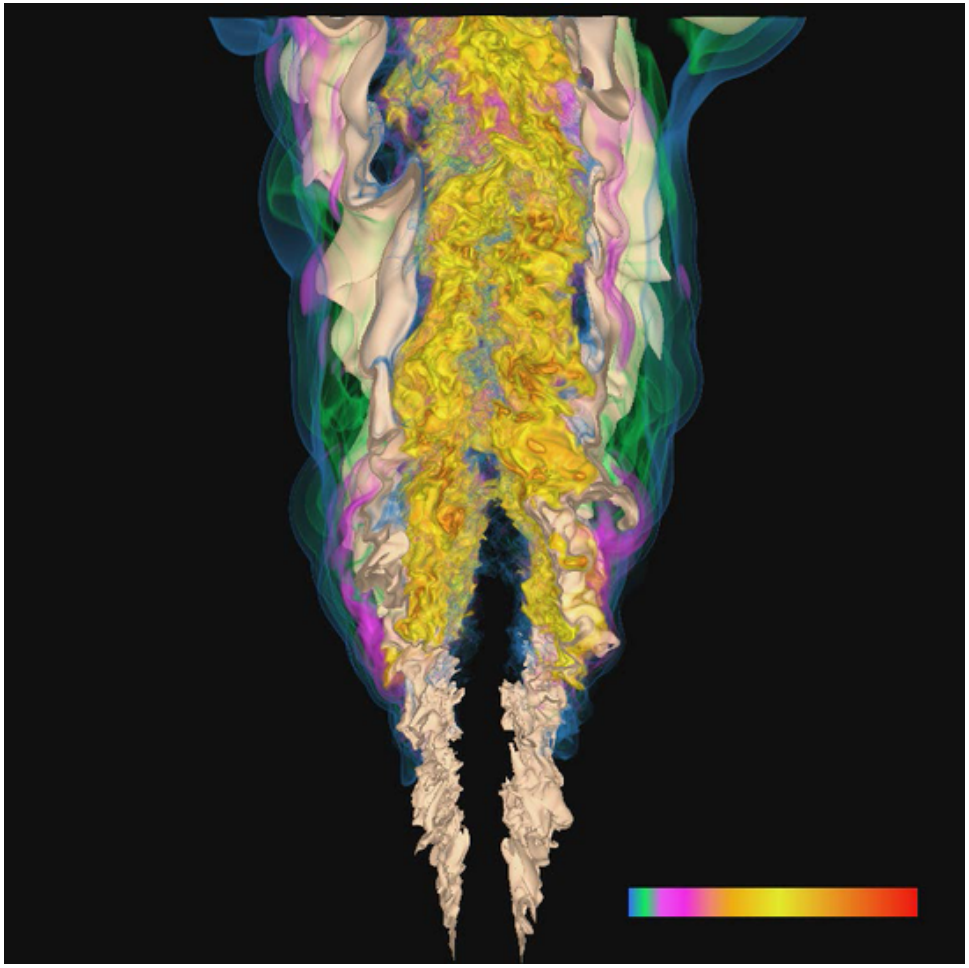


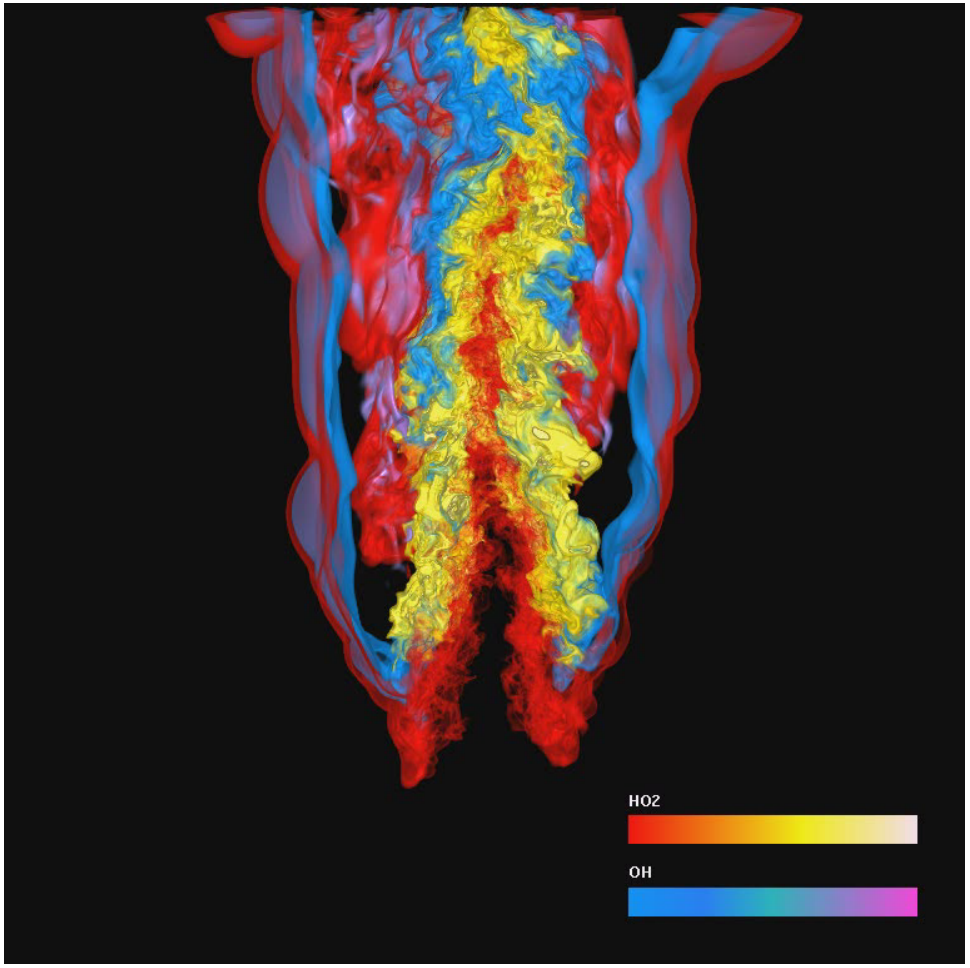




Turbulent Combustion Simulations

- Jackie Chen, Sandia National Laboratory
- Direct numerical simulation tools are being developed





Turbulent Combustion Simulations

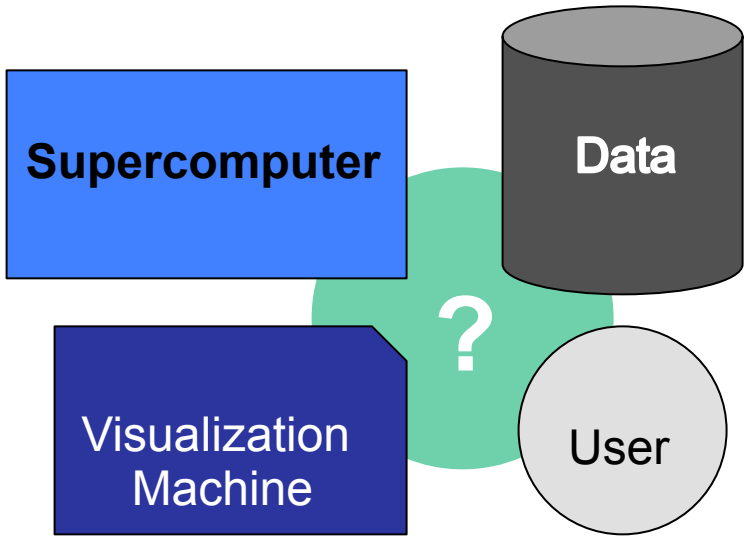
- High fidelity modeling is required to reliably predict efficiency and pollutant emission for new engines and new fuels
- The current simulation code running on a supercomputer like the Cray XT5 uses up to 48,000 cores and over 14 million CPU hours per run
- One question remains to be answered is how to extract and study important information from the hundreds of terabytes to petabytes of data that the simulation capable of outputting “today” and exabytes “tomorrow”.

Objective and Obstacles

- The objective is to develop scalable data analysis and visualization solutions to
  - Extract meaningful information from detailed combustion simulations
  - Lead to new understanding of important combustion processes
  - Help realize future efficiency improvements in energy conversion
- Evolving hardware architectures, bottlenecks in the data handling pipeline, and unsettled agreement on postprocessing strategies and their hardware requirements

Strategies

- In Situ Data Reduction and Visualization
  - Feature extraction
  - Encoding and packing
  - Rendering and animation
- Remote Visualization
  - Incremental visualization
  - Streaming
  - Provenance
- Postprocessing Data Analysis and Visualization

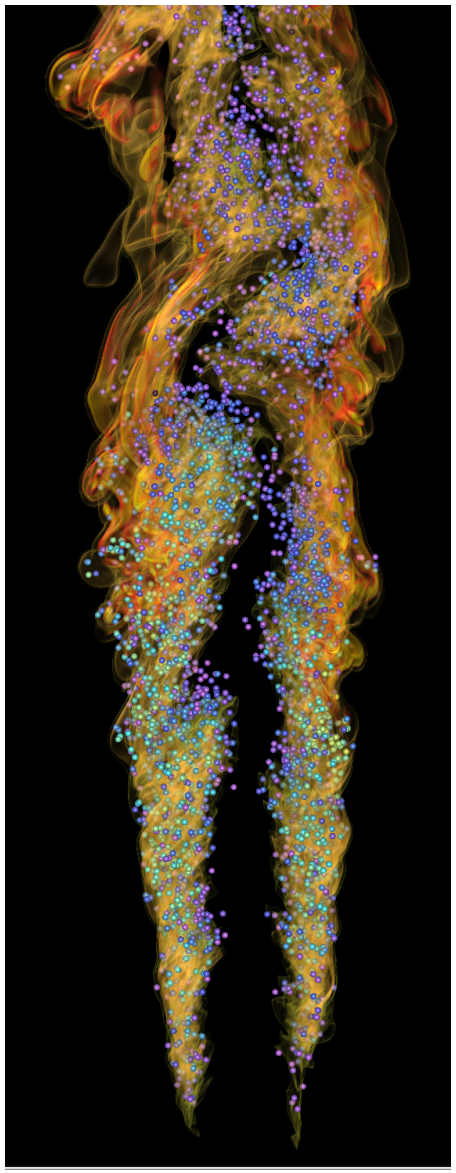


In Situ Visualization

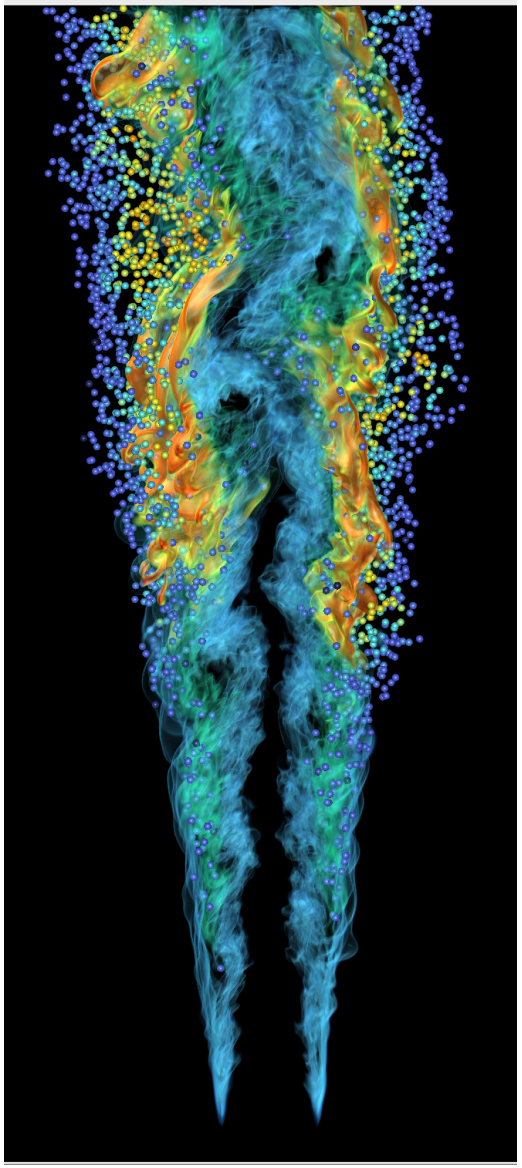
- Direct numerical simulation of turbulent combustion (up to 2025x1600x400)
- Visualizing both particle data and volume data
- Using a highly scalable parallel renderer
- Visualization takes under 1% of overall time
- Using up to 6,480 processors of the Cray XT5 at NCCS/ORNL
- The largest, most scalable in situ visualization ever achieved (2009-2010)

In Situ Visualization

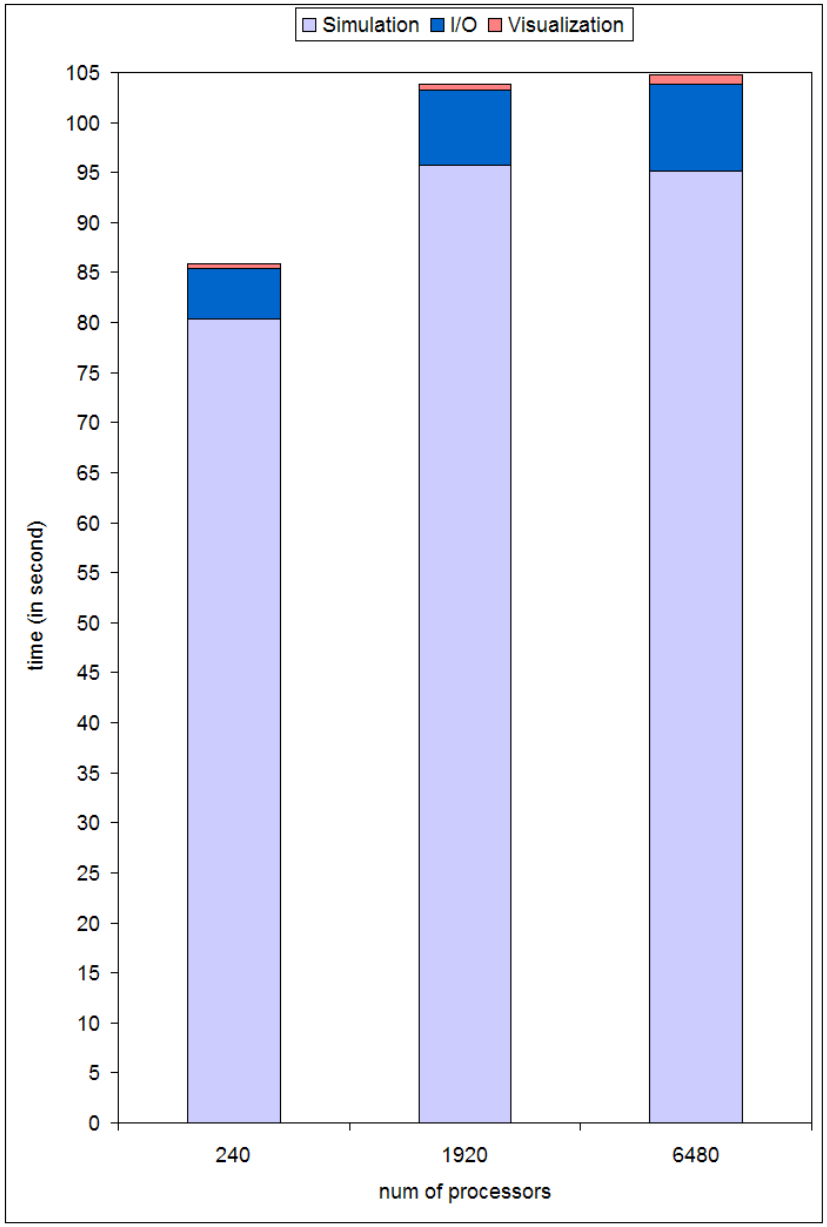
1215x960x240



CH2O/HO2



CH3/OH

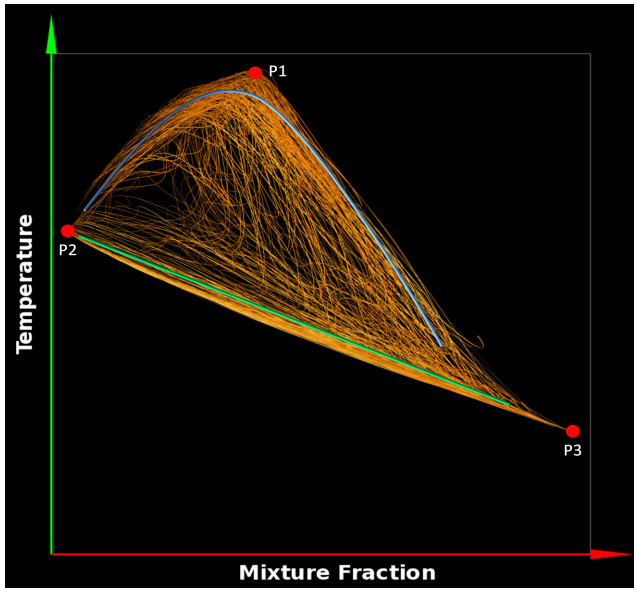


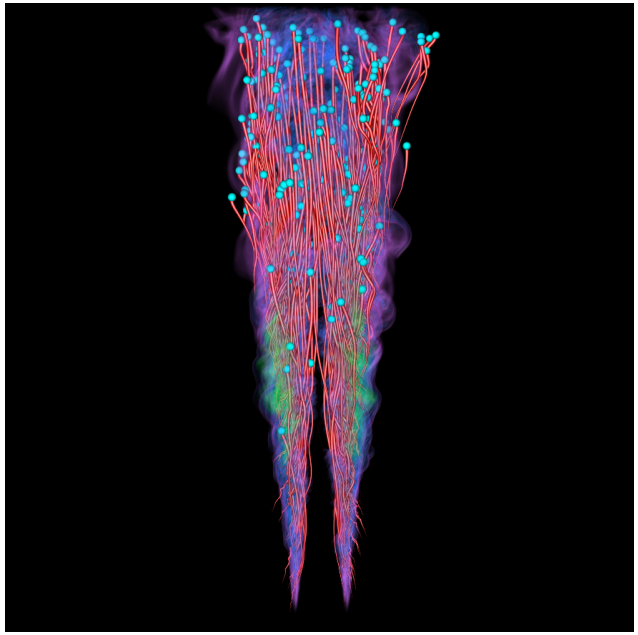
Particle Trajectories Study

- Lagrangian description of the combustion environment
  - Use from millions to billions of particles to capture the dynamic behavior of flames
- Categorization and understanding of the large amount of particle histories
  - Test several hypothesis about the nature of the particle trajectories
  - Present the particle data coherently to the combustion community
- Key task to hypothesis verification
  - Design a clustering system for grouping trajectories both automatically and in a user directed way

Considerations

- A DNS of a 3D turbulent ethylene jet flame
  - Takes14 million CPU-hours for 20 days using 30,000 cores
  - Generates 1.29 billion grid points and 27 principal variables
  - Outputs 6TB particle data saved for analysis
- Analysis tasks usually need the same leadership class resources used for simulations, but most of current clustering algorithms are not suitable for future exascale heterogeneous systems





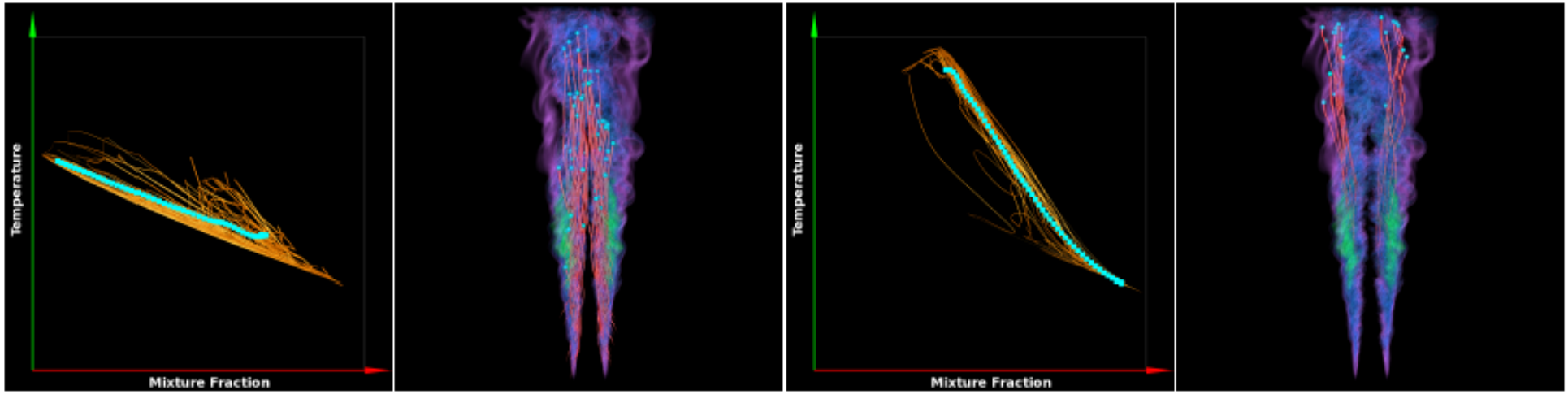
Our Tasks

Design and develop a clustering system to

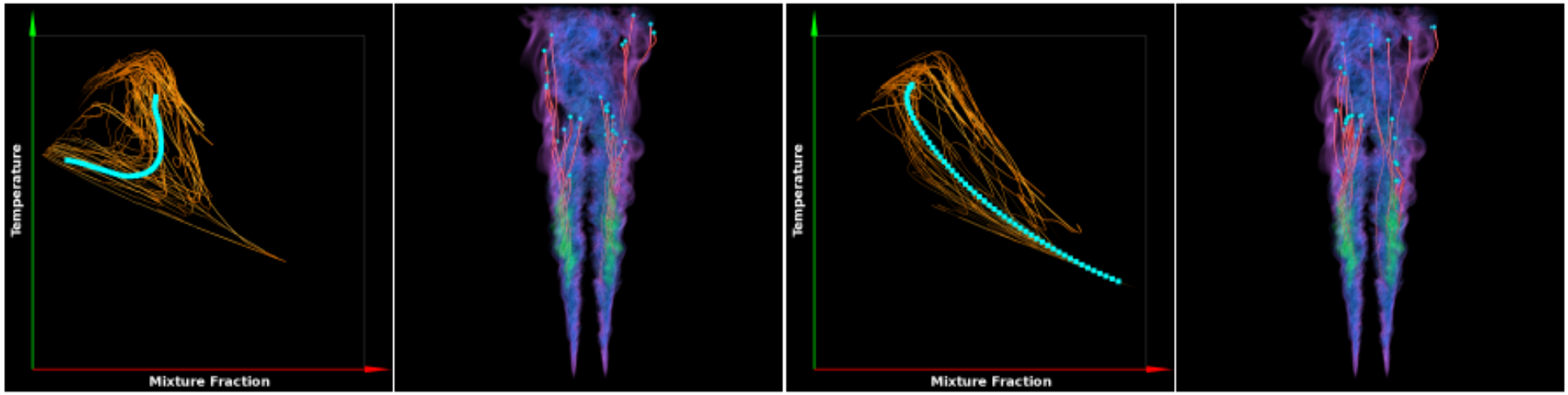
- Interactively categorize large particle data from detailed combustion simulations
  - Cluster particle trajectories based on appropriate similarity metrics of the selected variables
  - Use quantitative metrics for clustering quality
- Study scalable clustering algorithms for future exascale systems that have:
  - Heterogeneous architecture
  - Deep memory hierarchies
  - High levels of concurrency

Preliminary Work

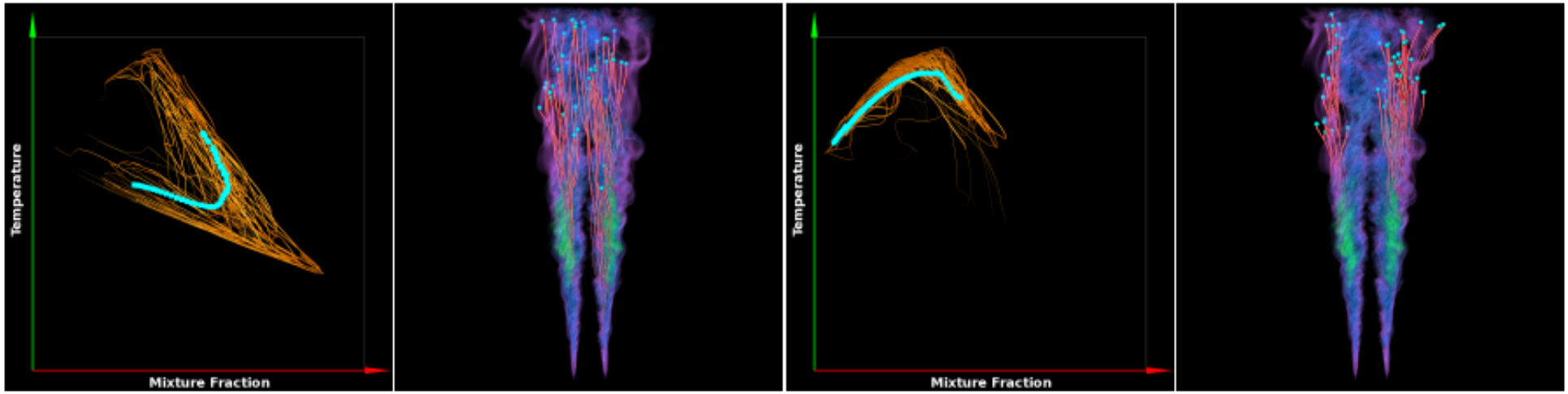
- Clustering analysis on a single GPU
  - Trajectory pre-processing: fitting and sampling
  - Clustering: EM (expectation-maximization) algorithm
- Leveraging high concurrency of GPU
  - CUDA implementation of per-trajectory calculations: fitting and sampling
  - CUDA implementation of matrix operations in EM algorithm



(a1) (a2) (b1) (b2)

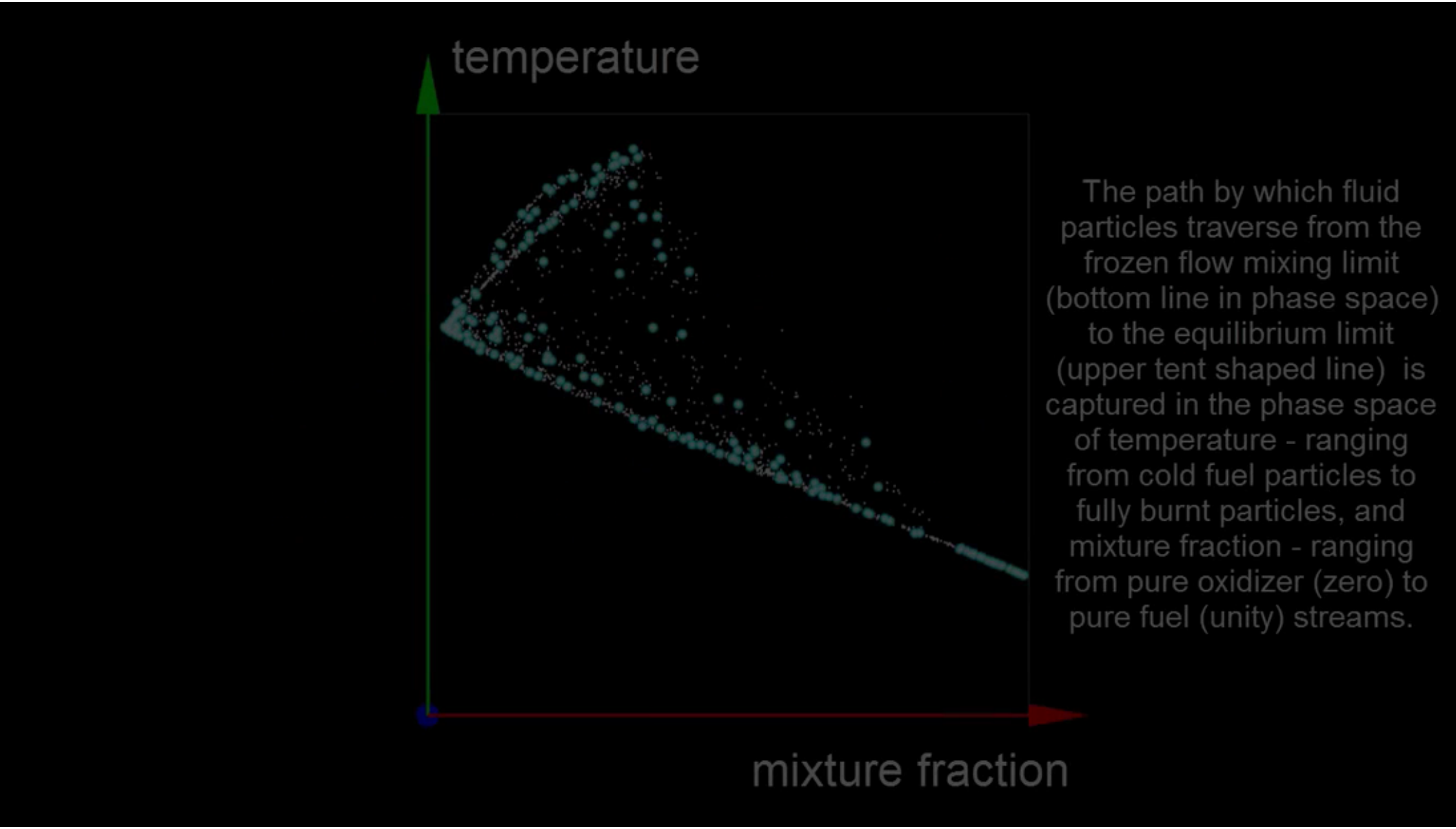


(c1) (c2) (d1) (d2)



(e1) (e2) (f1) (f2)

Video


temperature
mixture fraction
The path by which fluid particles traverse from the frozen flow mixing limit (bottom line in phase space) to the equilibrium limit (upper tent shaped line) is captured in the phase space of temperature - ranging from cold fuel particles to fully burnt particles, and mixture fraction - ranging from pure oxidizer (zero) to pure fuel (unity) streams.

Preliminary Results

- Performance gains with GPUs
  - For 10,000 trajectories, compared to conventional CPU implementation
    - Pre-processing: 20X speed up
    - Clustering: 5X speed up
  - GPU performances

#trajectories:	100	1000	10000
fitting time:	0.648s	2.63s	22.38
sampling time:	56.69s	116.51s	800.6s
clustering time:	2.9s	8.1s	74.1s

- Constrains imposed by GPUs
  - Limit memory size
  - High host/device data movement cost
  - Difficult GPU memory access optimization

Next

- Investigate an hybrid approach using both CPUs and GPUs
- Study data partitioning and distribution schemes
  - Balance the clustering and visualization workload across different types of processors and different nodes
- Study communication patterns
  - Exploit intra-node parallelism
  - Minimize inter-node communication
- Refine our strategies for a truly scalable solution on next generation exascale systems