Visual Analysis of Lagrangian Particle Data from Combustion Simulations

Hongfeng Yu

Sandia National Laboratories, CA

Ultrascale Visualization Workshop, SC11

Nov 13 2011, Seattle, WA

Joint work with

Jishang Wei and Kwan-Liu Ma (UC Davis), Ray Grout (NREL), and Jackie Chen (SNL)

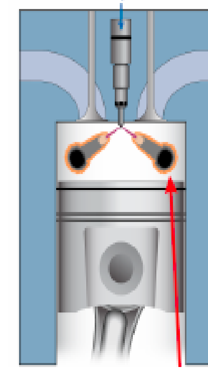
Direct Numerical Simulations of Combustion

- Energy Efficiency
  - 83% of U.S. energy comes from combustion of fossil fuels
  - Reduce greenhouse gas emissions by 80% by 2050
  - Reduce petroleum usage by 25% by 2020
- Large Combustion Simulations
  - High-fidelity
  - Critical for new engine designs
- Data Analysis Tools
  - Suitable for large data
    - Eulerian field data
    - Lagrangian particle data

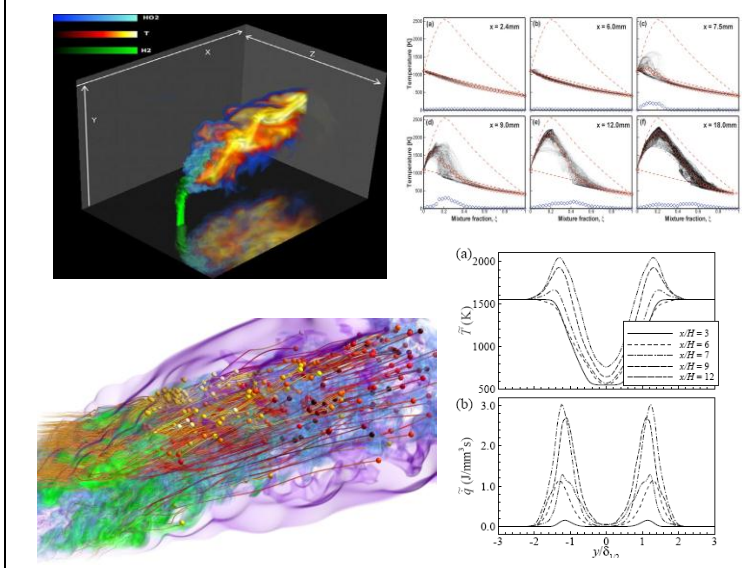
    (14 million CPU-hours running for 20 days on 30,000 cores; 1.3 billion grid points, 22 species; > 40 million particles per time step)

Large Combustion Simulations

New Designs



Detailed Analysis and Modeling



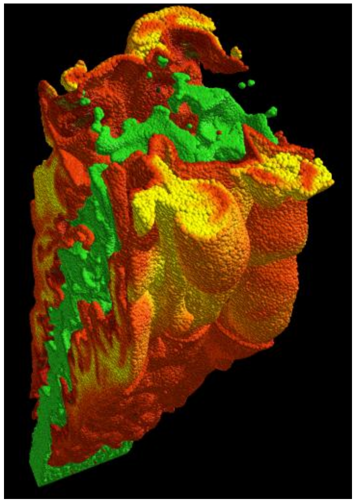
Background

- Particle Analysis Tasks
  - Select particle trajectories of interest
  - Collect statistical information
  - Assemble particles into time series

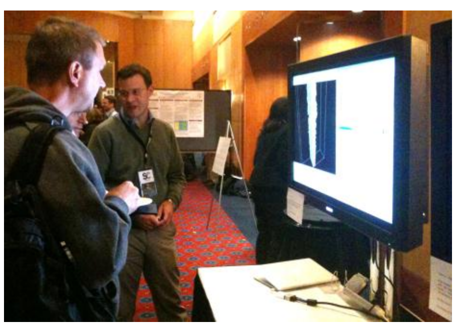
- COMPARED System
  - **Com**bined **p**article **a**nalysis, **r**eduction, **e**xploration, and **d**isplay
  - Leverage large heterogeneous systems
    - Interactive evaluation, query, analysis, and visualization
    - Full resolution particle data
  - Run-time calculation for advanced queries
    - Complex derived variables and flow topology classification (that are a priori unknown and cannot be indexed)
  - Performance optimization
    - Store results from individual GPUs in collision-free hash table
    - Explicitly cache primary and computed variables at multiple levels

COMPARED System

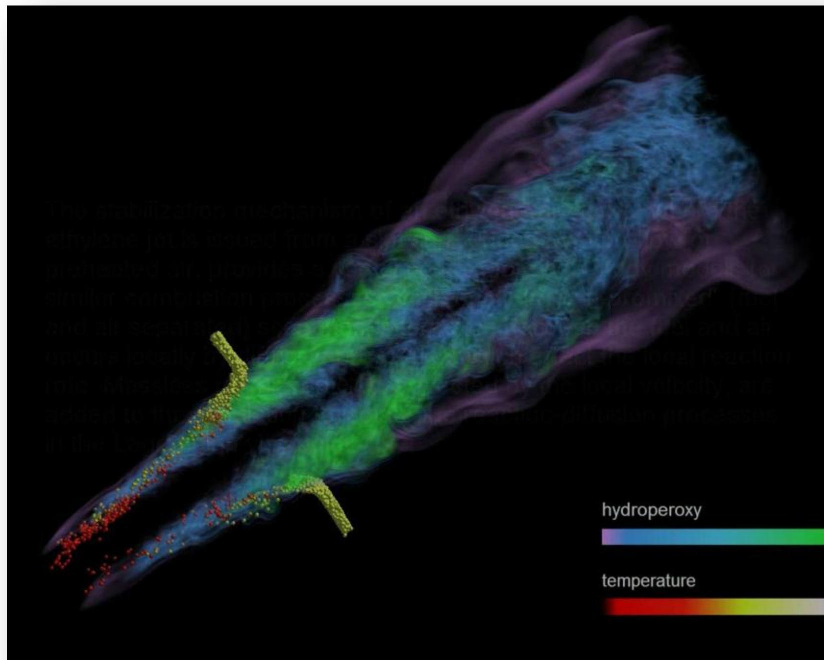
Combined particle analysis, reduction, exploration, and display



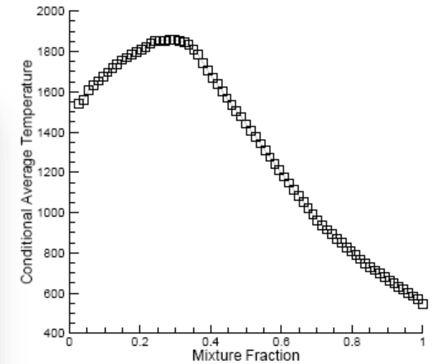
The core fuel jet (Y_{N2} >0.815) and the region where the flame reaction zone is located (Y_{N2} <=0.815 & Y_{OH} >0.0005)



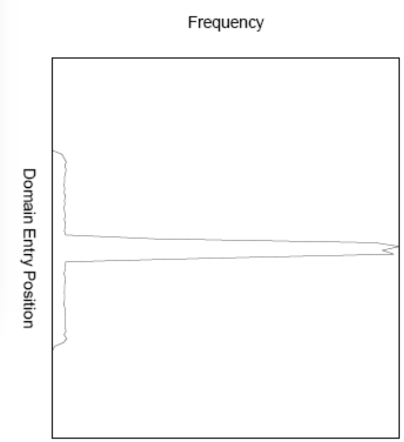
Interactive demo at SC09



A lifted ethylene-air jet flame stabilized by the interaction between a fuel jet and the surrounding preheated air



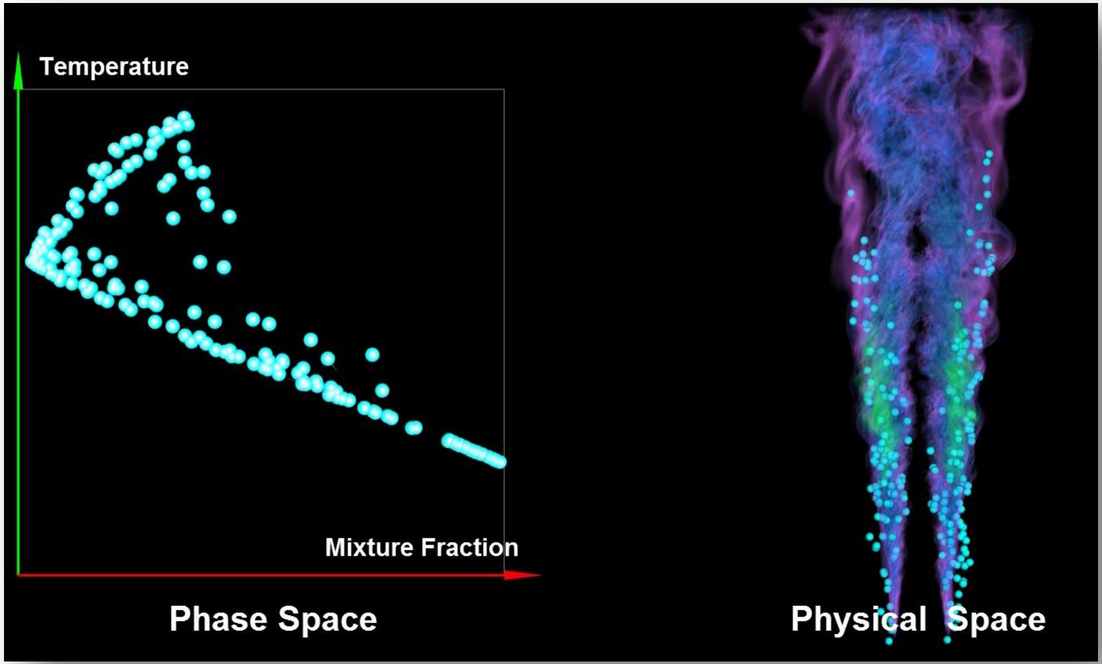
Conditional mean of temperature conditional on mixture fraction for the particles where Y_{N2} > 0.768



Histogram of particle y-position where AGE < 1μs, output between t=1.4710ms and t=1.4950ms

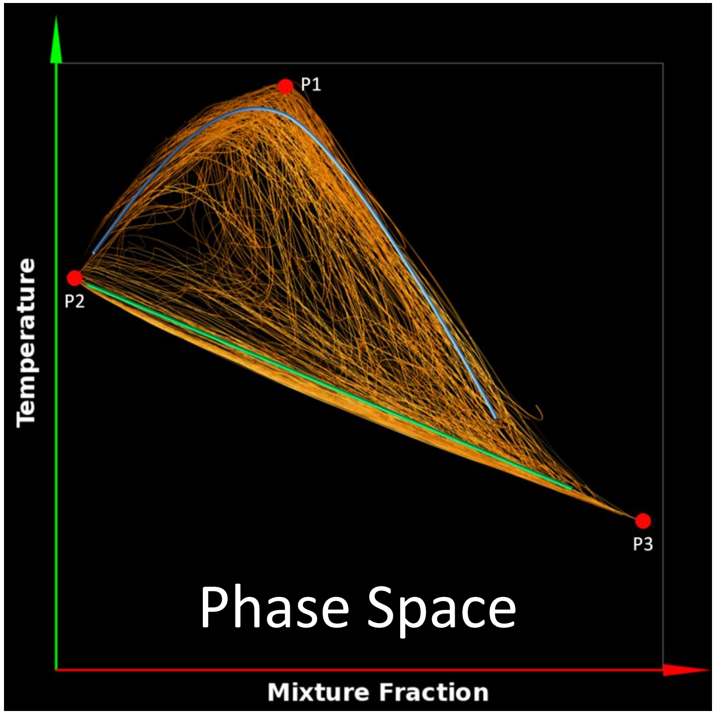
Motivation

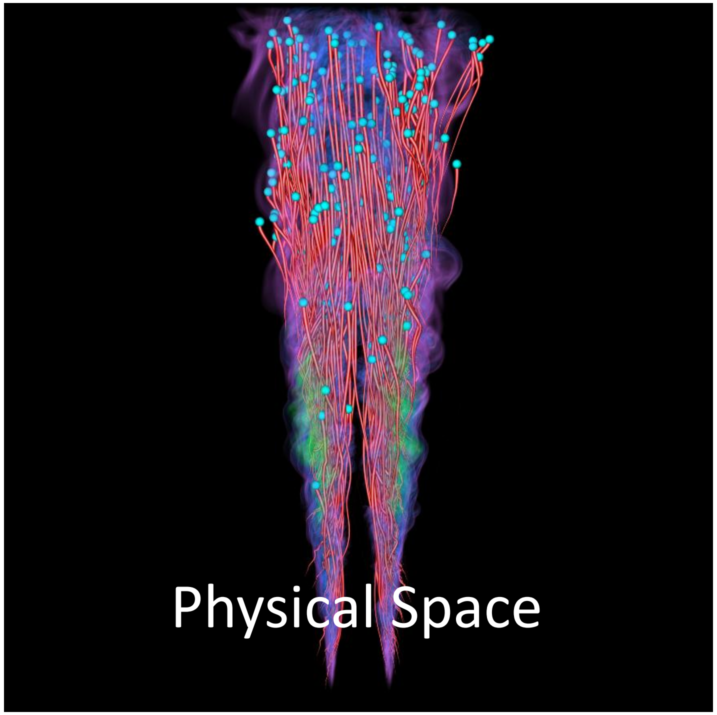
- Dual Space Analysis
  - Categorize particle time series curves in phase space
  - Explore corresponding particle trajectories in physical space



Motivation

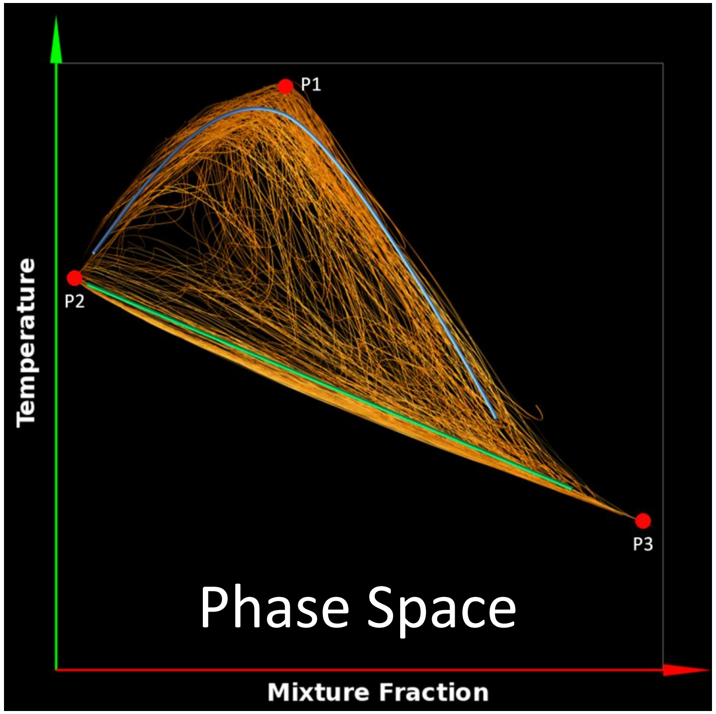
- Dual Space Analysis
  - Categorize particle time series curves in phase space
  - Explore corresponding particle trajectories in physical space

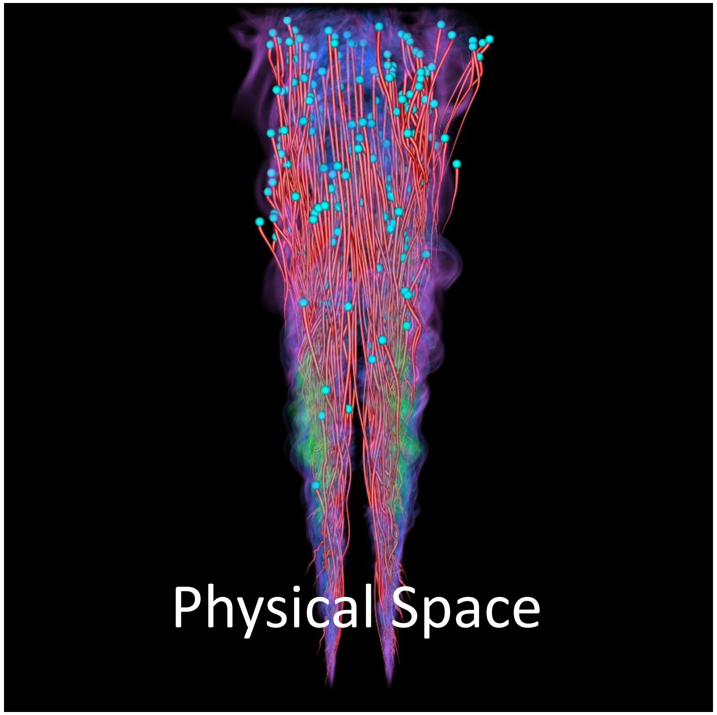




Motivation

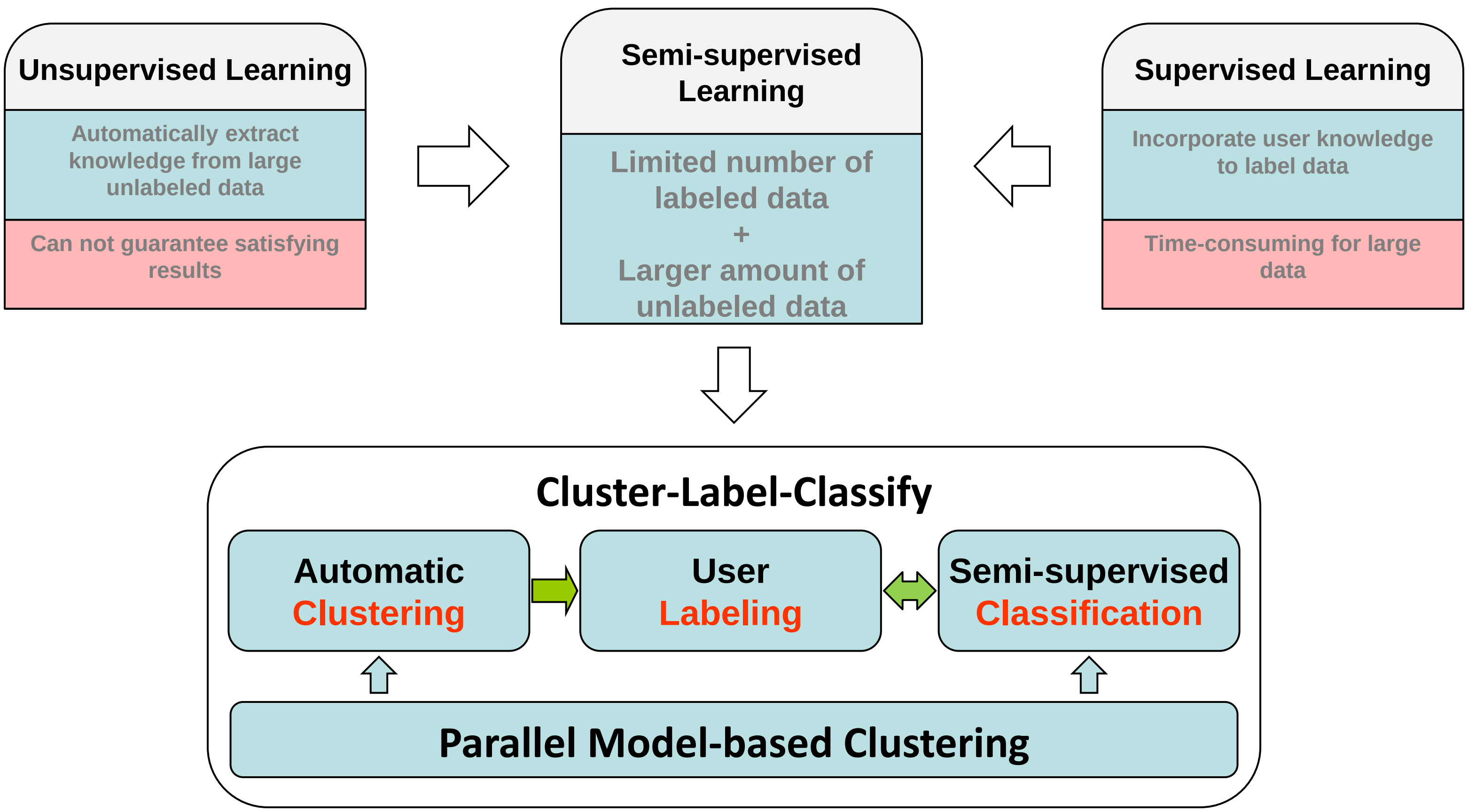
- Challenges
  - Analysis based on geometric properties of curves
  - Visual clutter
  - Large data





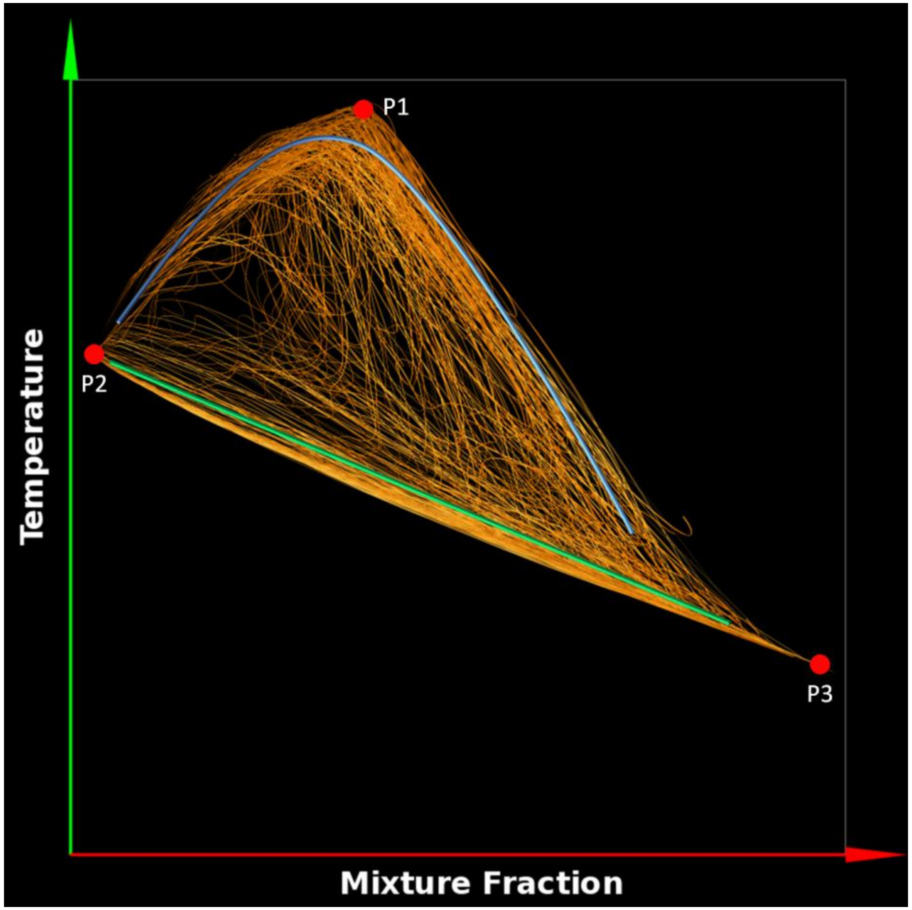
Our Solution

- Cluster-Label-Classify



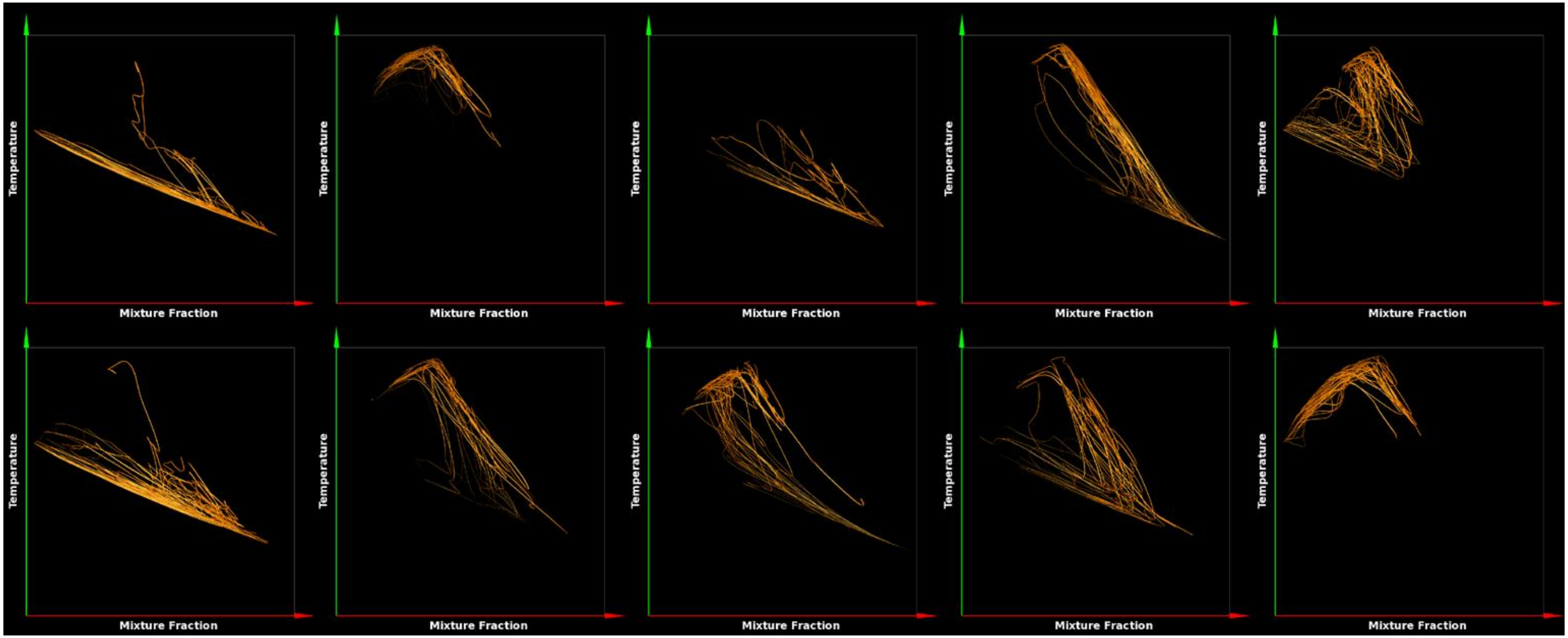
Cluster-Label-Classify

Initial Curves → Automatic Clustering → User Labeling ↔ Semi-supervised Classification → Visualization And Analysis



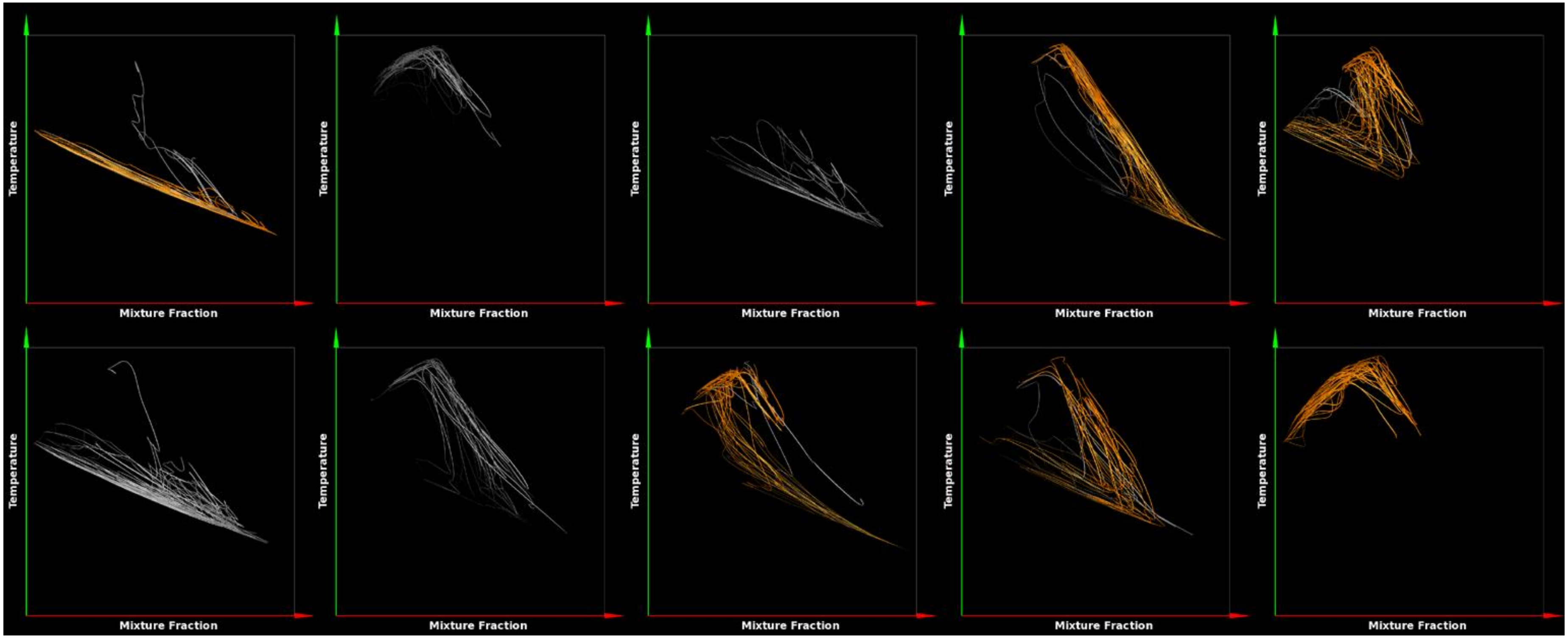
Cluster-Label-Classify

Initial Curves → **Automatic Clustering** → User Labeling ↔ Semi-supervised Classification → Visualization And Analysis



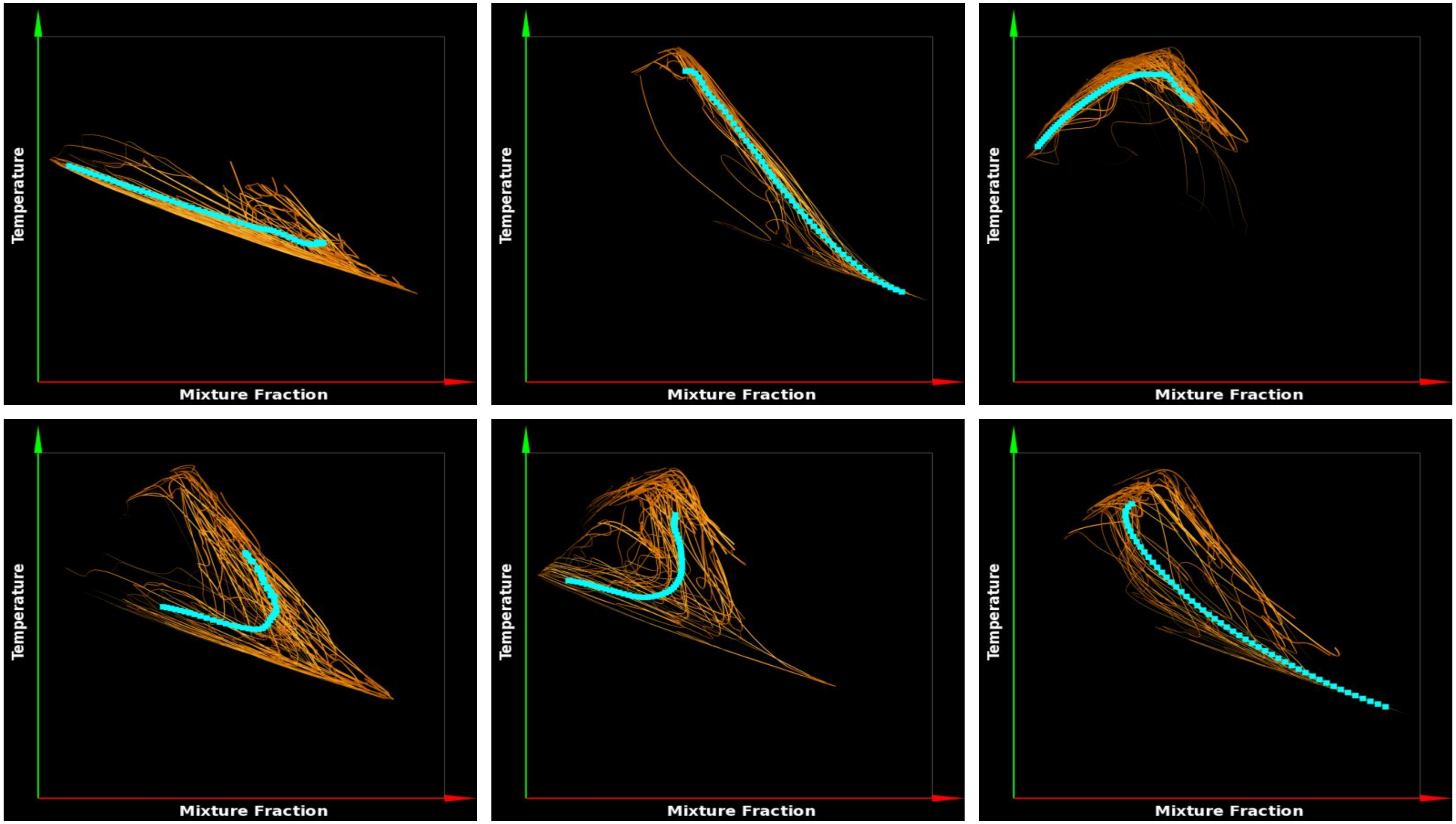
Cluster-Label-Classify

Initial Curves → Automatic Clustering → **User Labeling** ↔ Semi-supervised Classification → Visualization And Analysis



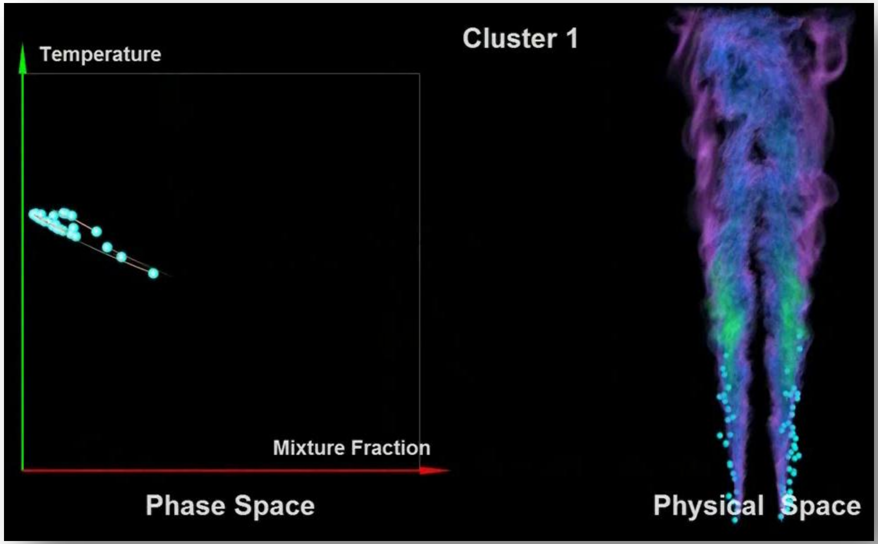
Cluster-Label-Classify

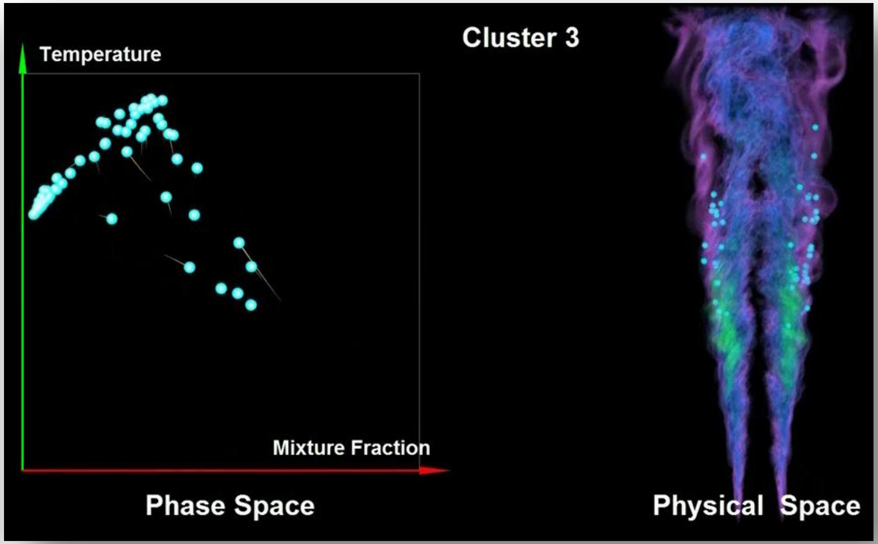
Initial Curves → Automatic Clustering → User Labeling ↔ **Semi-supervised Classification** → Visualization And Analysis



Cluster-Label-Classify

Initial Curves → Automatic Clustering → User Labeling ↔ Semi-supervised Classification → **Visualization And Analysis**





Model-based Clustering

- What is Model-based Clustering
  - Assume that data can be divided into K groups, and each has a probabilistic model to describe the data within it
  - Recover model parameters from data
  - Assign a data object to a cluster with highest probability
- Why is Model-based Clustering
  - Cluster lines of different lengths
  - Process large data efficiently
- How to Perform Model-based Clustering
  - Polynomial regression model
  - Recover model parameters using Expectation-Maximization algorithm

Parallel Model-based Clustering

- Distribute Line Data to Multiple Compute Nodes
  - Keep workload balanced and minimize communication costs between compute nodes
  - Use a sorted balancing algorithm to ensure the total number of data points on each compute node roughly the same

- Preprocess Line Data on Each Compute Node
  - Smooth and sample local lines on each compute node
  - Use **GPUs** to accelerate the preprocessing

Parallel Model-based Clustering

- Cluster Lines Using Multiple **CPUs**
  - On each compute node, initialize K component model parameters
  - Iterate between two steps
    - Expectation step: on each compute node, estimate local lines' probabilistic membership in different clusters
    - Maximization step: on each compute node, calculate the K model parameters globally
  - Assign each local line to a cluster with highest membership probability on each CPU node

Experiment Settings

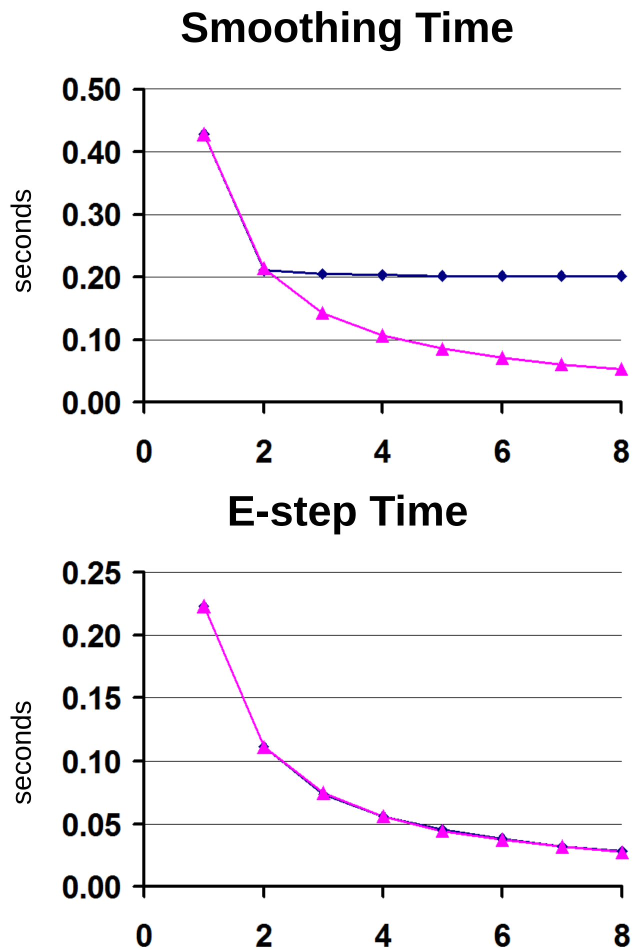
- Cluster: 8 computer nodes, each node contains
  - One Intel quad-core 3.00GHz CPU with 4GB of memory
  - One NVIDIA GeForce GTX 285 GPU
- Dataset
  - 1,000,000 time series curves correlating multiple variables generated from a combustion simulation

case	Number of lines	Number of computer nodes 1	2	3	4	5	6	7	8
1	10,000	X	X	X	X	X	X	X	X
2	100,000	X	X	X	X	X	X	X	X
3	1,000,000				X	X	X	X	X

Entries marked with "x" represent experiment runs.

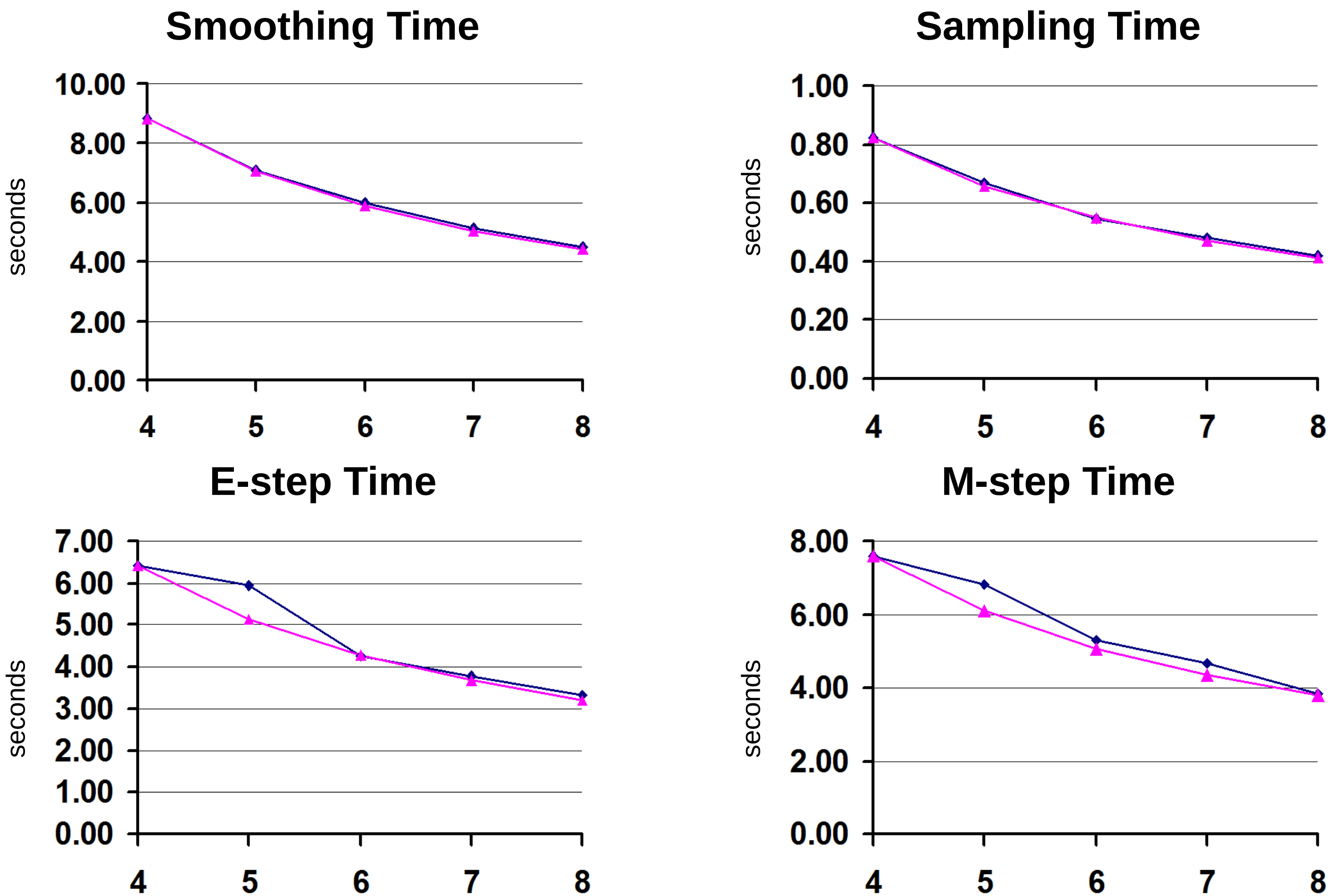
Performance Results

- 10 Thousand Time Series Curves (Speedup)



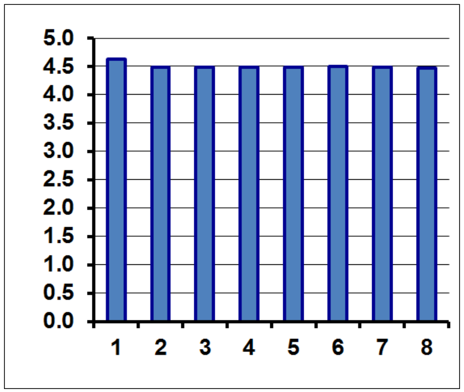
Performance Results

- 1 Million Time Series Curves (Speedup)

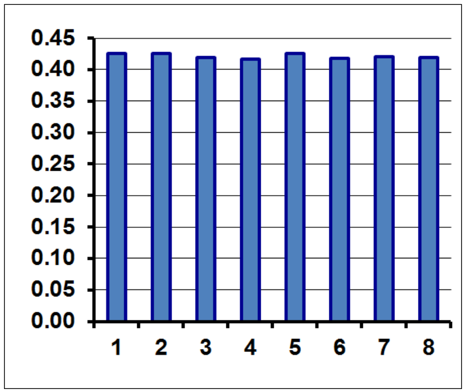


Performance Results

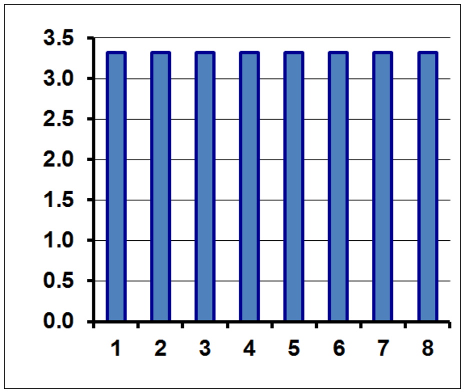
- 1 Million Time Series Curves (Workload)



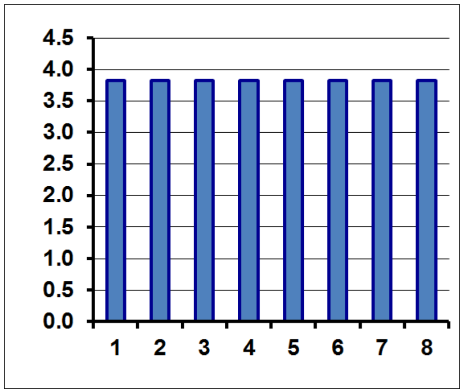
Smoothing time(3.46%)



Sampling time(2.09%)



E-Step time(0.16%)



M-Step time(0.01%)

Workloads among 8 nodes. In each plot, the horizontal axis represents the node ID, and the vertical axis represents the running time in second. The percentage number associated with each plot is the difference ratio ($dr = (max\_time - min\_time)/max\_time$) between the maximum and minimum times among the nodes.

Conclusion and Future Work

- Cluster-Label-Classify
  - Incorporate expert domain knowledge
  - Effectively and efficiently process large line data
  - Parallel implementation with multiple CPUs and GPUs
    - Distribute line data for balanced workload
    - Efficiently preprocess line data in CUDA
    - Devise and implement the regression model-based clustering in MPI
  - Support dual space particle analysis
- Future Work
  - Conduct particle data analysis in situ and compress lines as much as possible
  - Explore high dimensional lines

Acknowledgement

- This work has been sponsored in part by
  - The U.S. Department of Energy through the SciDAC program with Agreement No. DE-FC02-06ER25777
  - The U.S. National Science Foundation through grants OCI-0749217, CCF-0811422, CCF-0850566, OCI-0749227, and OCI-0950008
- Sandia National Laboratories is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the U.S. Department of Energy under contract DE-AC04-94-AL85000.

Thank You