Parallelizing Data Analytics

INTERNATIONAL ADVANCED RESEARCH WORKSHOP ON HIGH PERFORMANCE COMPUTING

From Clouds and Big Data to Exascale and Beyond

Cetraro (Italy)

July 10 2014

Geoffrey Fox

gcf@indiana.edu

http://www.infomall.org

School of Informatics and Computing

Digital Science Center

Indiana University Bloomington

Abstract

- We discuss a variety of large scale optimization/data analytics including deep learning, clustering, image processing, information retrieval, collaborative filtering and dimension reduction.
- We describe parallelization challenges and nature of kernel operations.
- We cover both batch and streaming operations and give some measured performance on both MPI and MapReduce frameworks.
- Use context of SPIDAL (Scalable Parallel Interoperable Data Analytics Library)

Machine Learning in Network Science, Imaging in Computer Vision, Pathology, Polar Science

Algorithm	Applications	Features	Status	Parallelism
Graph Analytics				
Community detection	Social networks, webgraph	Graph .	P-DM	GML-GrC
Subgraph/motif finding	Webgraph, biological/social networks		P-DM	GML-GrB
Finding diameter	Social networks, webgraph		P-DM	GML-GrB
Clustering coefficient	Social networks		P-DM	GML-GrC
Page rank	Webgraph		P-DM	GML-GrC
Maximal cliques	Social networks, webgraph		P-DM	GML-GrB
Connected component	Social networks, webgraph		P-DM	GML-GrB
Betweenness centrality	Social networks	Graph, Non-metric, static	P-Shm	GML-GRA
Shortest path	Social networks, webgraph		P-Shm	
Spatial Queries and Analytics				
Spatial relationship based queries	GIS/social networks/pathology informatics	Geometric	P-DM	PP
Distance based queries			P-DM	PP
Spatial clustering			Seq	GML
Spatial modeling			Seq	PP

PP Pleasingly Parallel (Local ML)

GRA or P-Shm Shared memory

GML Global (parallel) ML

GrA Static GrB Runtime partitioning

Some Core Machine Learning Building Blocks

DA Vector Clustering	Accurate Clusters	Vectors	P-DM	GML
DA Non metric Clustering	Accurate Clusters, Biology, Web	Non metric, O(N^2)	P-DM	GML
Kmeans; Basic, Fuzzy and Elkan	Fast Clustering	Vectors	P-DM	GML
Levenberg-Marquardt Optimization	Non-linear Gauss-Newton, use in MDS	Least Squares	P-DM	GML
SMACOF Dimension Reduction	DA- MDS with general weights	Least Squares, O(N^2)	P-DM	GML
Vector Dimension Reduction	DA-GTM and Others	Vectors	P-DM	GML
TFIDF Search	Find nearest neighbors in document corpus	Bag of “words” (image features)	P-DM	PP
All-pairs similarity search	Find pairs of documents with TFIDF distance below a threshold		Todo	GML
Support Vector Machine SVM	Learn and Classify	Vectors	Seq	GML
Random Forest	Learn and Classify	Vectors	P-DM	PP
Gibbs sampling (MCMC)	Solve global inference problems	Graph	Todo	GML
Latent Dirichlet Allocation LDA with Gibbs sampling or Var. Bayes	Topic models (Latent factors)	Bag of “words”	P-DM	GML
Singular Value Decomposition SVD	Dimension Reduction and PCA	Vectors	Seq	GML
Hidden Markov Models (HMM)	Global inference on sequence models	Vectors	Seq	PP & GML

Introduction

- Also will need many local machine learning algorithms for image processing (such as OpenCV, Matlab, CImg, VLFeat, and ImageJ)
- Here discuss Global Machine Learning as part of **SPIDAL (Scalable Parallel Interoperable Data Analytics Library)**
- Focus on 4 big data analytics
  - Dimension Reduction (Multi Dimensional Scaling)
  - Levenberg-Marquardt Optimization
  - Clustering: similar to Gaussian Mixture Models, PLSI (probabilistic latent semantic indexing), LDA (Latent Dirichlet Allocation)
  - Deep Learning
- Surprisingly little packaged scalable GML; Mahout low performance and R largely sequential (LML); MLlib just starting

Parallelism

- All use parallelism over data points
  - Entities to cluster or map to Euclidean space
- Except deep learning which has parallelism over pixel plane in neurons
  - as need to look at small numbers of data items at a time in Stochastic Gradient Descent
- Maximum Likelihood or χ^2 both lead to structure like
- **Minimize sum $\sum_{items=1}^{N}$ (Positive nonlinear function of unknown parameters for item *i*)**
- All solved iteratively with (clever) first or second order approximation to shift in objective function
  - Sometimes steepest descent direction; sometimes Newton
  - Have classic Expectation Maximization structure

Parameter "Server"

- Note learning networks have huge number of parameters (11 billion in Stanford work) so that inconceivable to look at second derivative
- Clustering and MDS have lots of parameters but can be practical to look at second derivative and use Newton's method to minimize
- Parameters are determined in distributed fashion but are typically needed globally
  - MPI use broadcast and "AllCollectives"
  - AI community: use parameter server and access as needed

Some Important Cases

- Need to cover non **vector semimetric** and **vector spaces** for clustering and dimension reduction (N points in space)
- **Vector spaces** have Euclidean distance and scalar products
  - Algorithms can be O(N) and these are best for clustering but for MDS O(N) methods may not be best as obvious objective function $O(N^2)$
- MDS Minimizes Stress

$$\sigma(\underline{X}) = \Sigma_{i<j=1}^{N} \text{weight}(i,j)\ (\delta(i, j) - d(\underline{X}_i , \underline{X}_j))^2$$

- **Semimetric spaces** just have pairwise distances defined between points in space $\delta(i, j)$
- Note matrix solvers all use conjugate gradient – converges in 5-100 iterations – a big gain for matrix with a million rows. This removes factor of N in time complexity
- Ratio of #clusters to #points important; new ideas if ratio >~ 0.1

Deterministic Annealing Algorithms

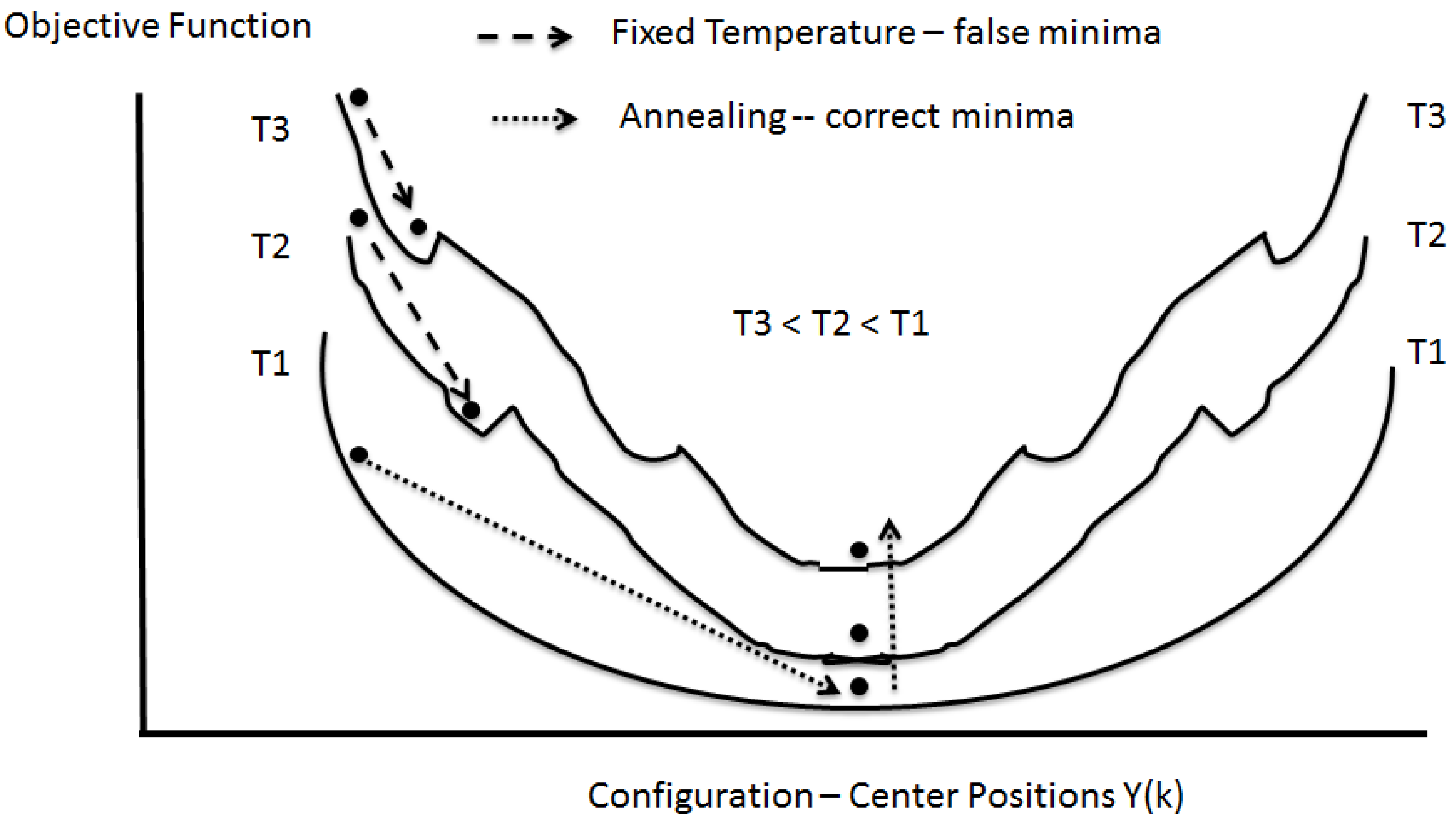
Some Motivation

- Big Data requires high performance – achieve with parallel computing
- Big Data sometimes requires robust algorithms as more opportunity to make mistakes
- **Deterministic annealing (DA)** is one of better approaches to robust optimization
  - Started as “Elastic Net” by Durbin for Travelling Salesman Problem TSP
  - Tends to remove local optima
  - Addresses overfitting
  - Much Faster than simulated annealing

- Physics systems find true lowest energy state if you anneal i.e. you equilibrate at each temperature as you cool
- Uses mean field approximation, which is also used in “Variational Bayes” and “Variational inference”

(Deterministic) Annealing

- Find minimum at high temperature when trivial
- Small change avoiding local minima as lower temperature
- Typically gets better answers than standard libraries- R and Mahout
- And can be parallelized and put on GPU's etc.


Objective Function
Fixed Temperature – false minima
Annealing -- correct minima
T3
T2
T1
T3 < T2 < T1
T3
T2
T1
Configuration – Center Positions Y(k)

General Features of DA

- In many problems, decreasing temperature is classic multiscale – finer resolution (√T is “just” distance scale)
- In clustering √T is distance in space of points (and centroids), for MDS scale in mapped Euclidean space
- **T =** ∞, all points are in same place – the center of universe
- For MDS all Euclidean points are at center and distances are zero. For clustering, there is one cluster
- As Temperature lowered there are phase transitions in clustering cases where clusters split
  - Algorithm determines whether split needed as second derivative matrix singular
- Note DA has similar features to hierarchical methods and you do not have to specify a number of clusters; you need to specify a distance scale

Basic Deterministic Annealing

- **H(χ) is objective function** to be minimized as a function of parameters χ
- Gibbs Distribution at Temperature T
  $P(\chi) = \exp(- H(\chi)/T) / \int d\chi \exp(- H(\chi)/T)$
- Or $P(\chi) = \exp(- H(\chi)/T + F/T)$
- Minimize Free Energy combining Objective Function and Entropy
  $F = < H - T S(P) > = \int d\chi \{P(\chi)H + T P(\chi) \ln P(\chi)\}$
- Simulated annealing performs these integrals by Monte Carlo
- Deterministic annealing corresponds to doing integrals analytically (by mean field approximation) and is much much faster
- In each case temperature is lowered slowly – say by a factor 0.95 to 0.9999 at each iteration

Some Uses of Deterministic Annealing

- Clustering
  - Vectors: Rose (Gurewitz and Fox)
  - Clusters with fixed sizes and no tails (Proteomics team at Broad)
  - No Vectors: Hofmann and Buhmann (Just use pairwise distances)
- Dimension Reduction for visualization and analysis
  - Vectors: GTM Generative Topographic Mapping
  - No vectors SMACOF: Multidimensional Scaling) MDS (Just use pairwise distances)
- Can apply to HMM & general mixture models (less study)
  - Gaussian Mixture Models
  - Probabilistic Latent Semantic Analysis with Deterministic Annealing DA-PLSA as alternative to Latent Dirichlet Allocation for finding “hidden factors”

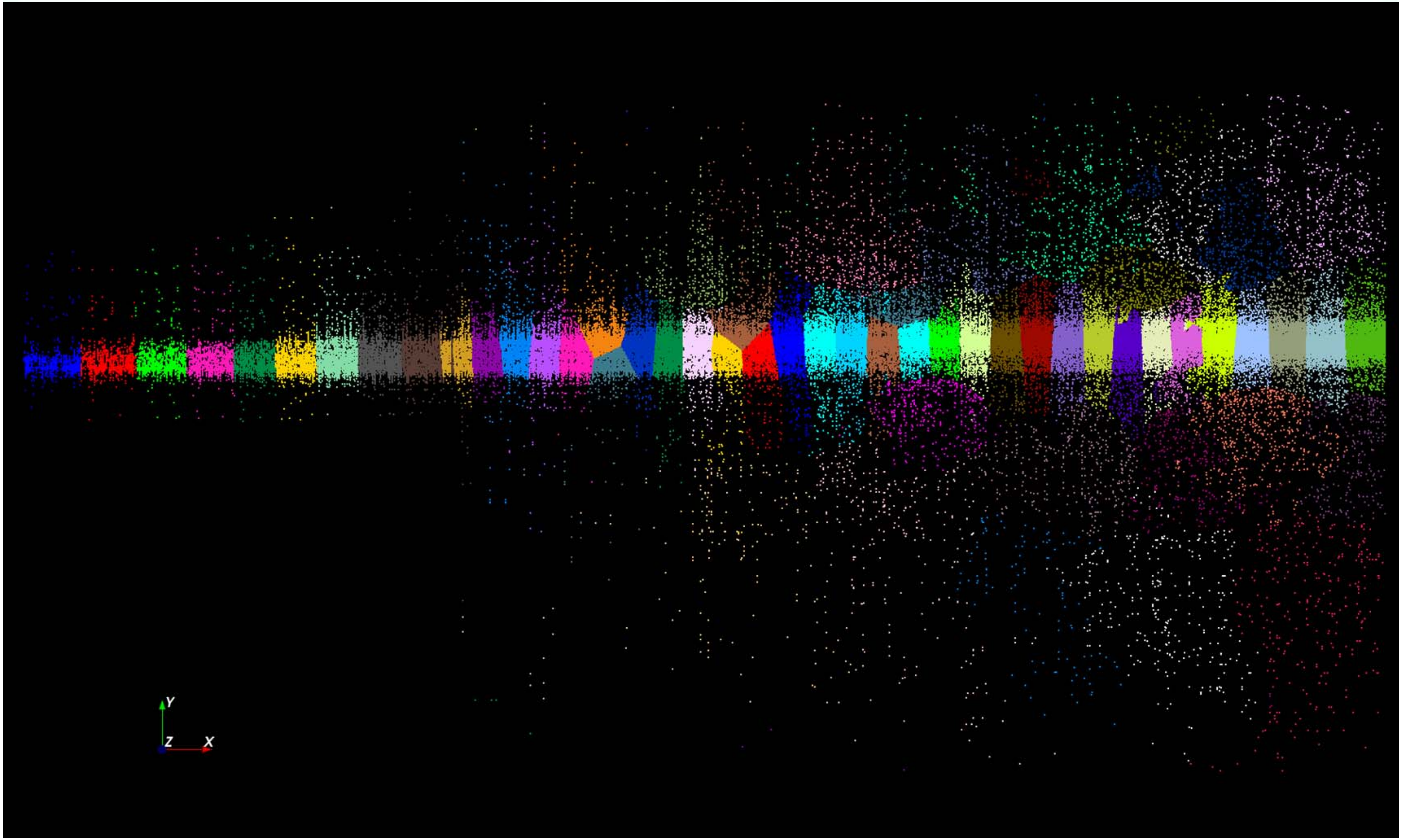
Some Clustering Problems

- Analysis of Mass Spectrometry data to find peptides by clustering peaks (Broad Institute)
  - ~0.5 million points in 2 dimensions (one experiment) -- ~ 50,000 clusters summed over charges
- Metagenomics – 0.5 million (increasing rapidly) points NOT in a vector space – hundreds of clusters per sample
- Pathology Images >50 Dimensions
- Social image analysis is in a highish dimension vector space
  - 10-50 million images; 1000 features per image; million clusters
- Finding communities from network graphs coming from Social media contacts etc.
  - No vector space; can be huge in all ways

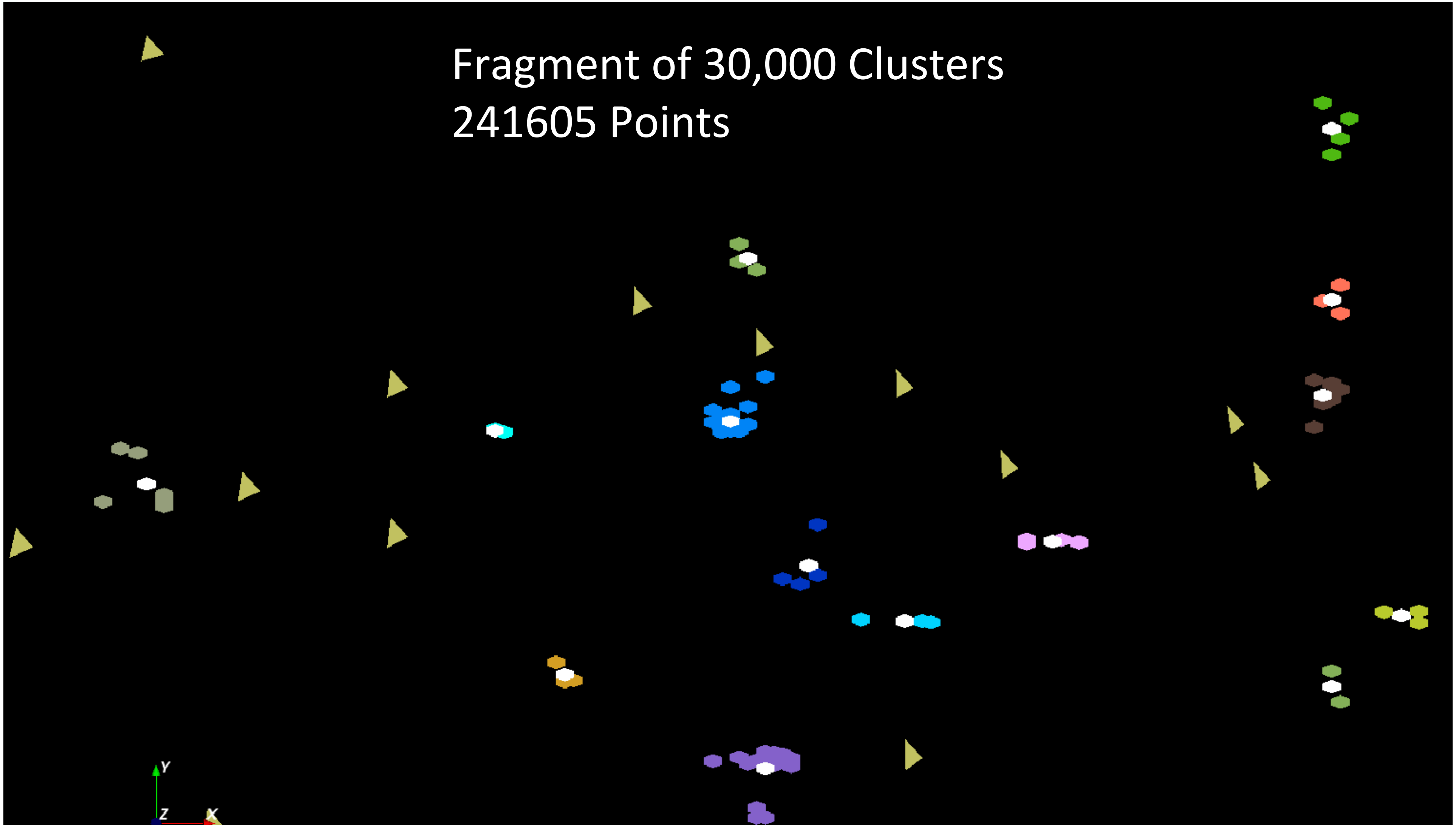
Background on LC-MS

- Remarks of collaborators – Broad Institute
- Abundance of peaks in “label-free” LC-MS enables large-scale comparison of peptides among groups of samples.
- In fact when a group of samples in a cohort is analyzed together, not only is it possible to “align” robustly or cluster the corresponding peaks across samples, but it is also possible to search for patterns or fingerprints of disease states which may not be detectable in individual samples.
- This property of the data lends itself naturally to big data analytics for biomarker discovery and is especially useful for population-level studies with large cohorts, as in the case of infectious diseases and epidemics.
- With increasingly large-scale studies, the need for fast yet precise cohort-wide clustering of large numbers of peaks assumes technical importance.
- In particular, a scalable parallel implementation of a cohort-wide peak clustering algorithm for LC-MS-based proteomic data can prove to be a critically important tool in clinical pipelines for responding to global epidemics of infectious diseases like tuberculosis, influenza, etc.

Proteomics 2D DA Clustering T= 25000 with 60 Clusters (will be 30,000 at T=0.025)

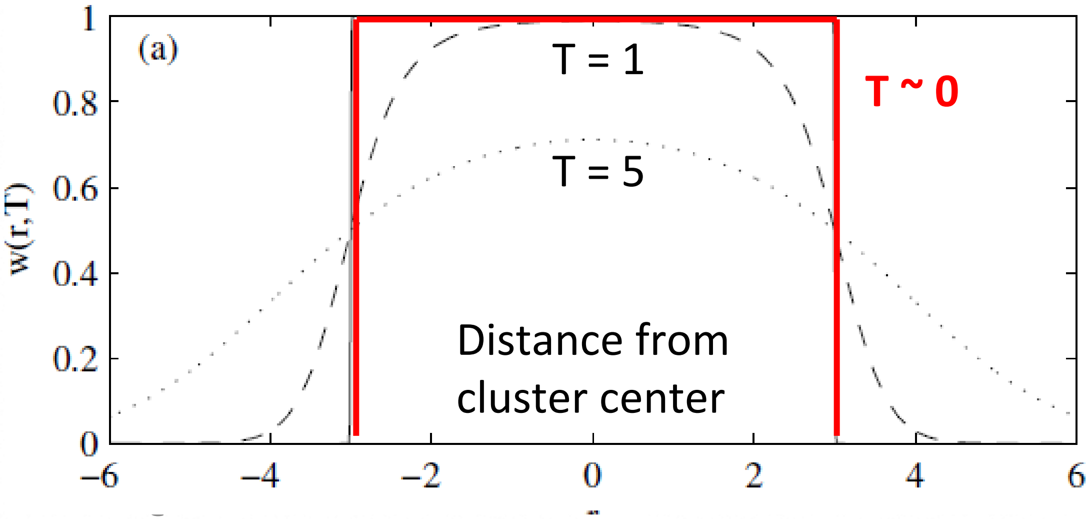


The brownish triangles are sponge peaks outside any cluster.
The colored hexagons are peaks inside clusters with the white hexagons being determined cluster center

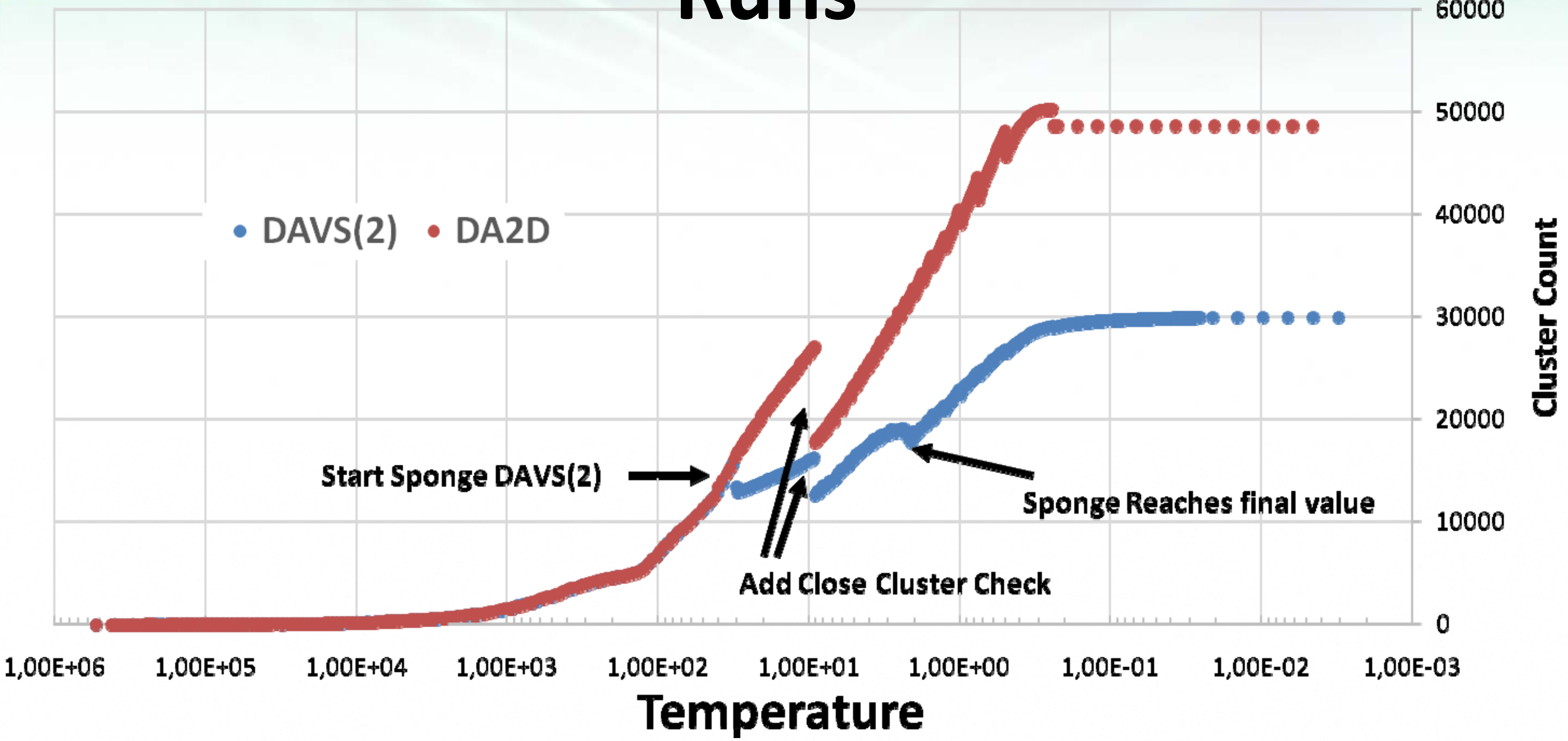

Fragment of 30,000 Clusters
241605 Points

Trimmed Clustering

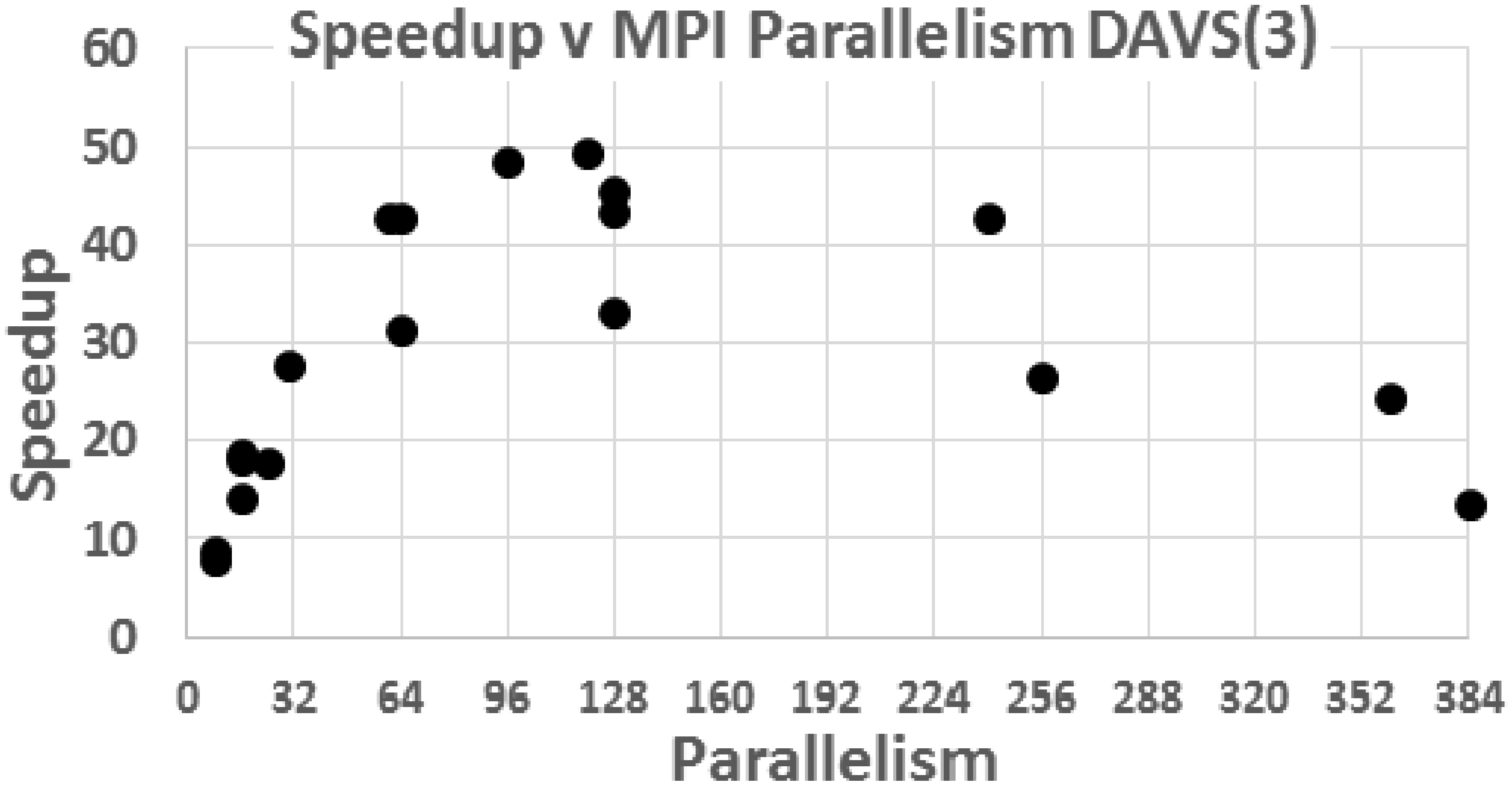
- *Clustering with position-specific constraints on variance: Applying redescending M-estimators to label-free LC-MS data analysis* (Rudolf Frühwirth , D R Mani and Saumyadipta Pyne) *BMC Bioinformatics* 2011, **12:**358
- $H_{TCC} = \sum_{k=0}^{K} \sum_{i=1}^{N} M_i(k)\ f(i,k)$
  - $f(i,k) = (\underline{X}(i) - \underline{Y}(k))^2/2\sigma(k)^2 \quad k > 0$
  - $f(i,0) = c^2 / 2 \quad k = 0$
- The 0'th cluster captures (at zero temperature) all points outside clusters (background)
- Clusters are trimmed $(\underline{X}(i) - \underline{Y}(k))^2/2\sigma(k)^2 < c^2 / 2$
- Relevant when well defined errors



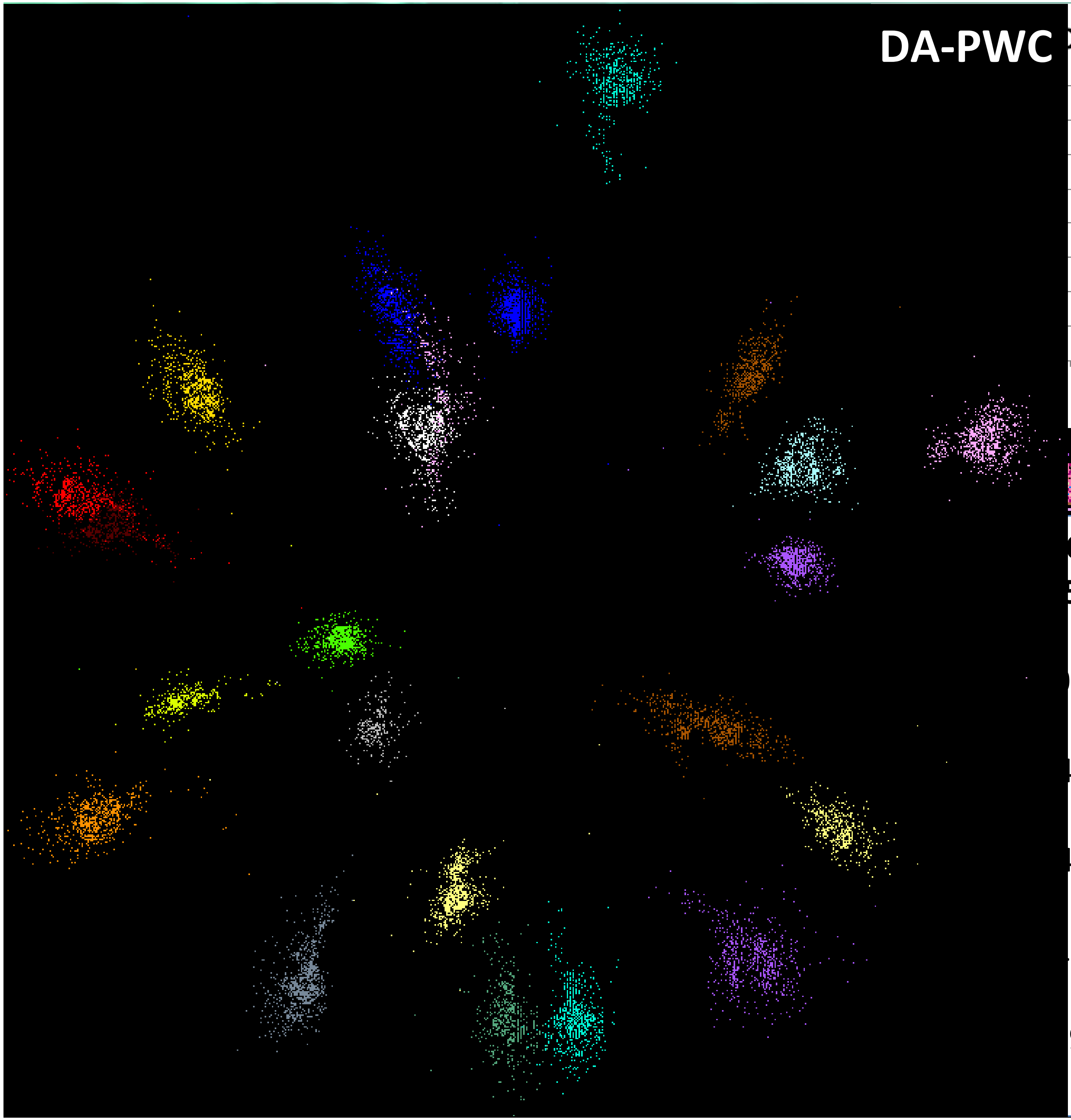
Cluster Count v. Temperature for 2 Runs



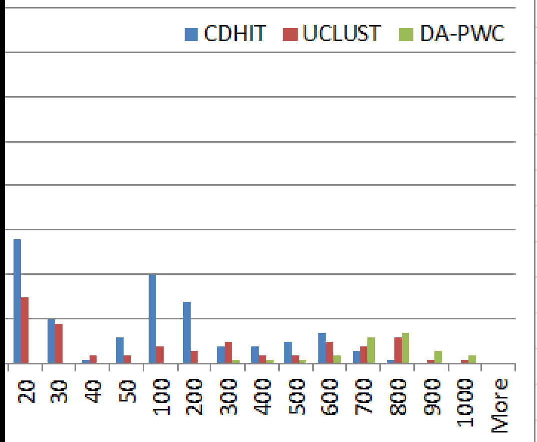
- All start with one cluster at far left
- T=1 special as measurement errors divided out
- DA2D counts clusters with 1 member as clusters. DAVS(2) does not

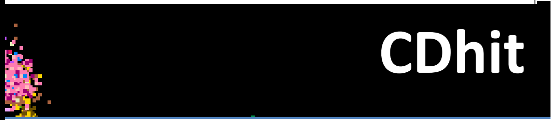


Speedups for several runs on Tempest from 8-way through 384 way MPI parallelism with one thread per process. We look at different choices for MPI processes which are either inside nodes or on separate nodes



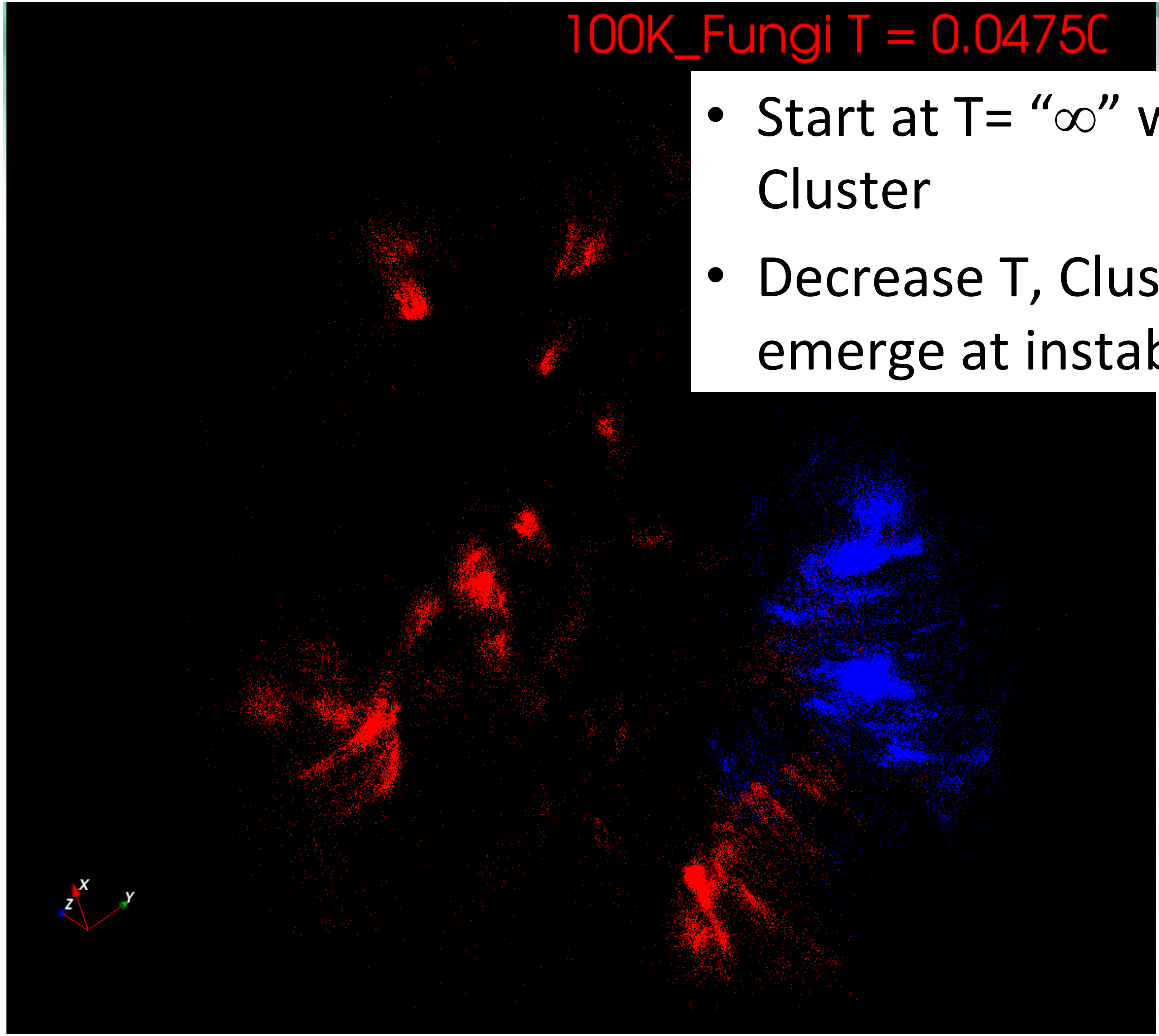
Divergent 15761





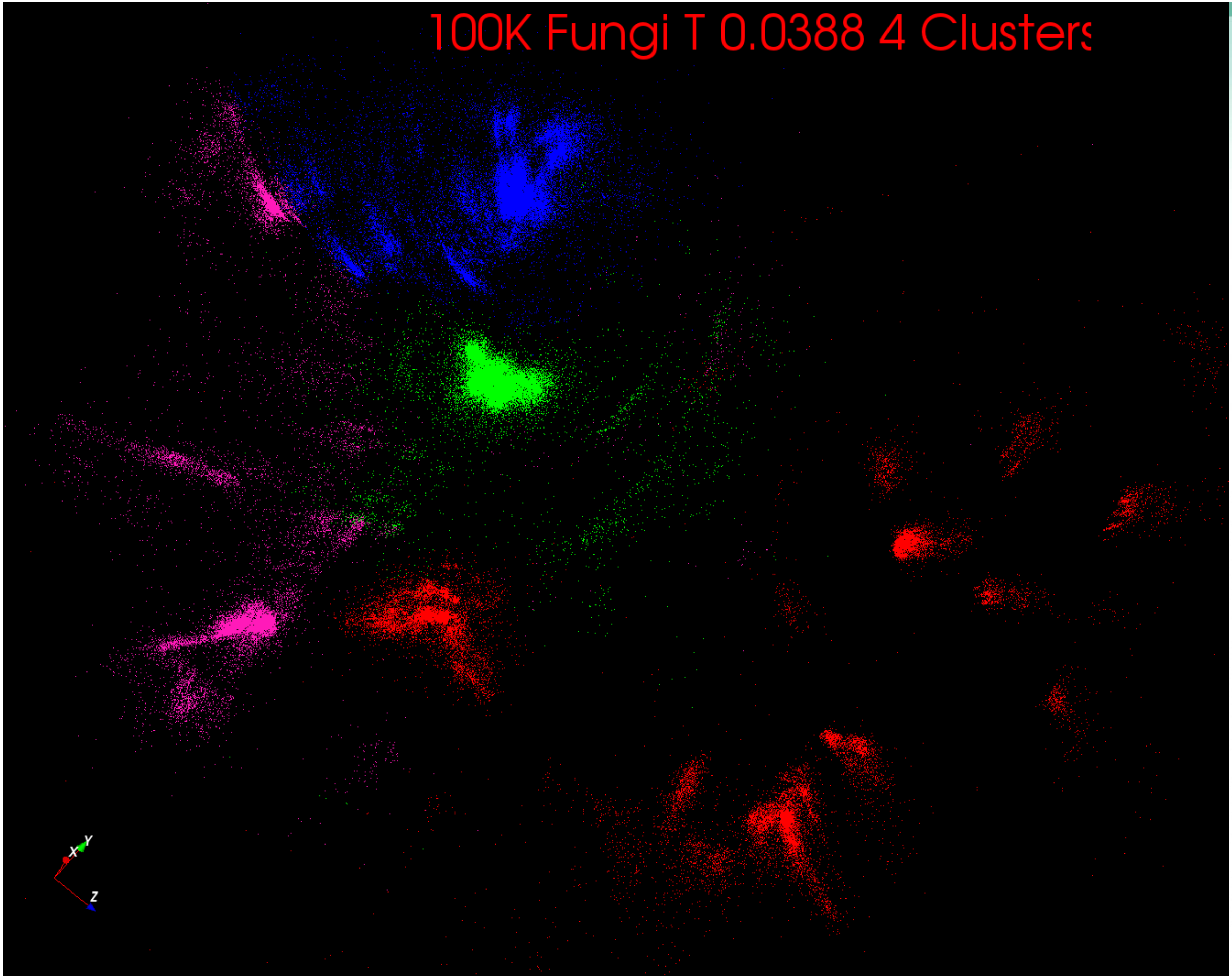
Clust (Cuts 0.65 to 0.95)

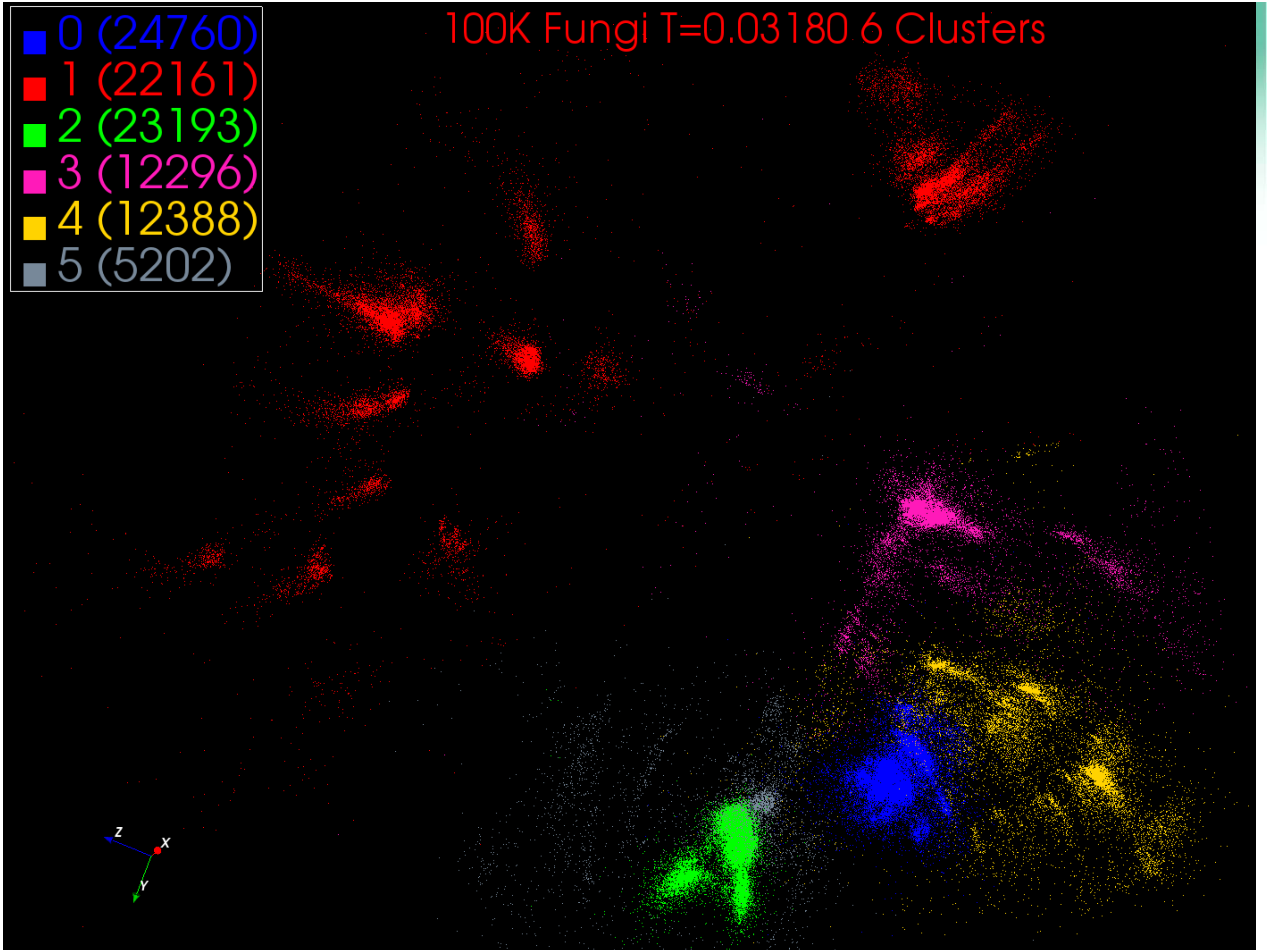
5	0.75	0.85	0.95
	10	36	91
)	0	13	16
4	10	5	0
4	10	17(11)	72(62)
4	9	5	0
9	14	5	7



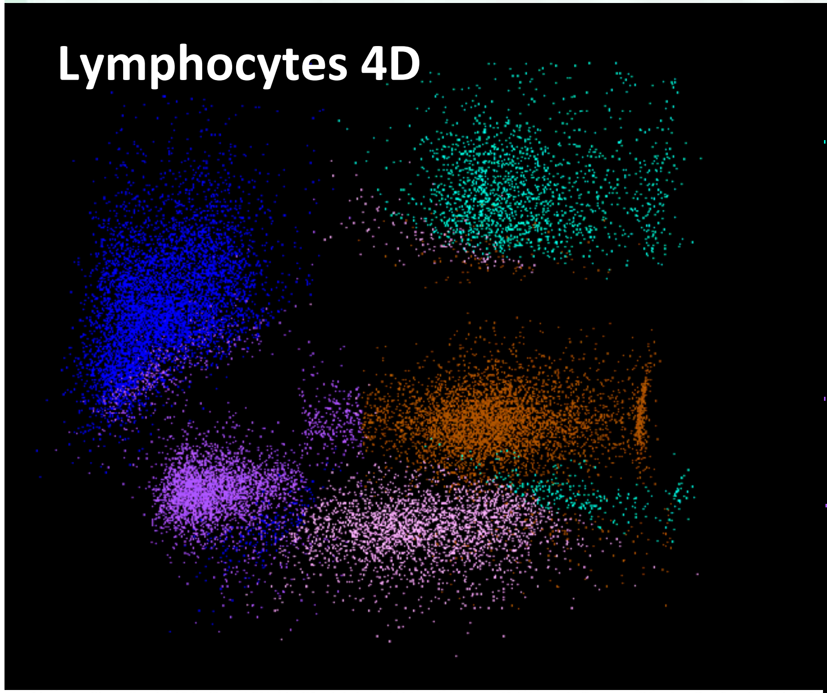
- Start at T= “∞” with 1 Cluster
- Decrease T, Clusters emerge at instabilities

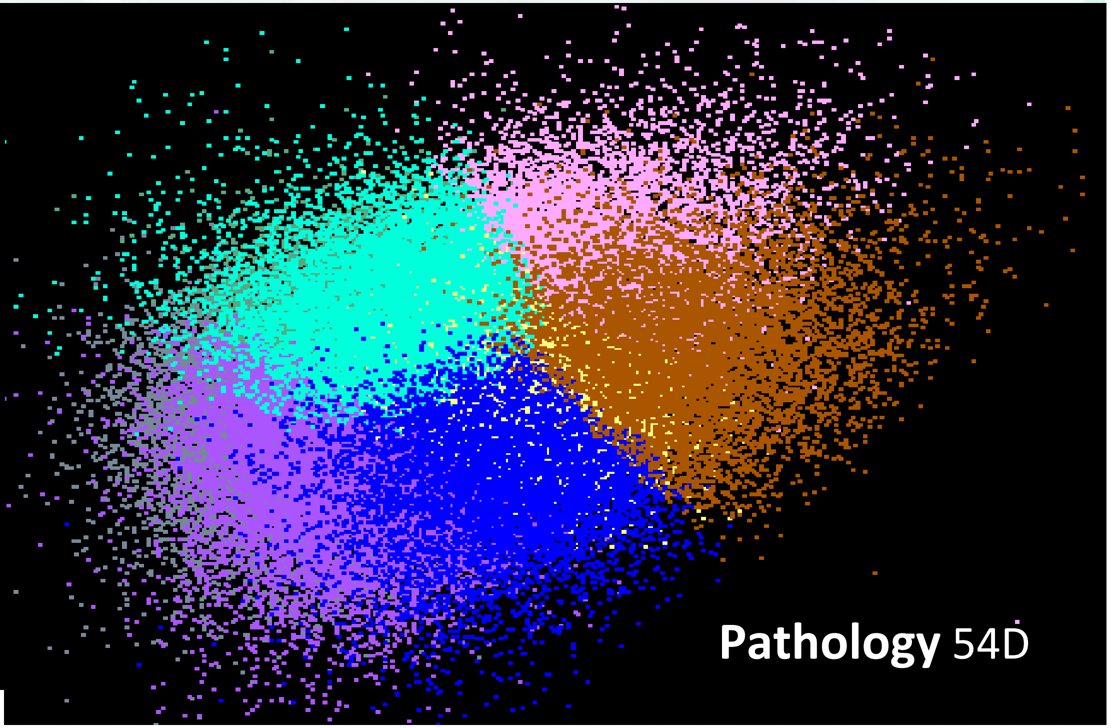
100K Fungi T 0.0388 4 Clusters




100K Fungi T=0.03180 6 Clusters
0 (24760)
1 (22161)
2 (23193)
3 (12296)
4 (12388)
5 (5202)
Z
X
Y

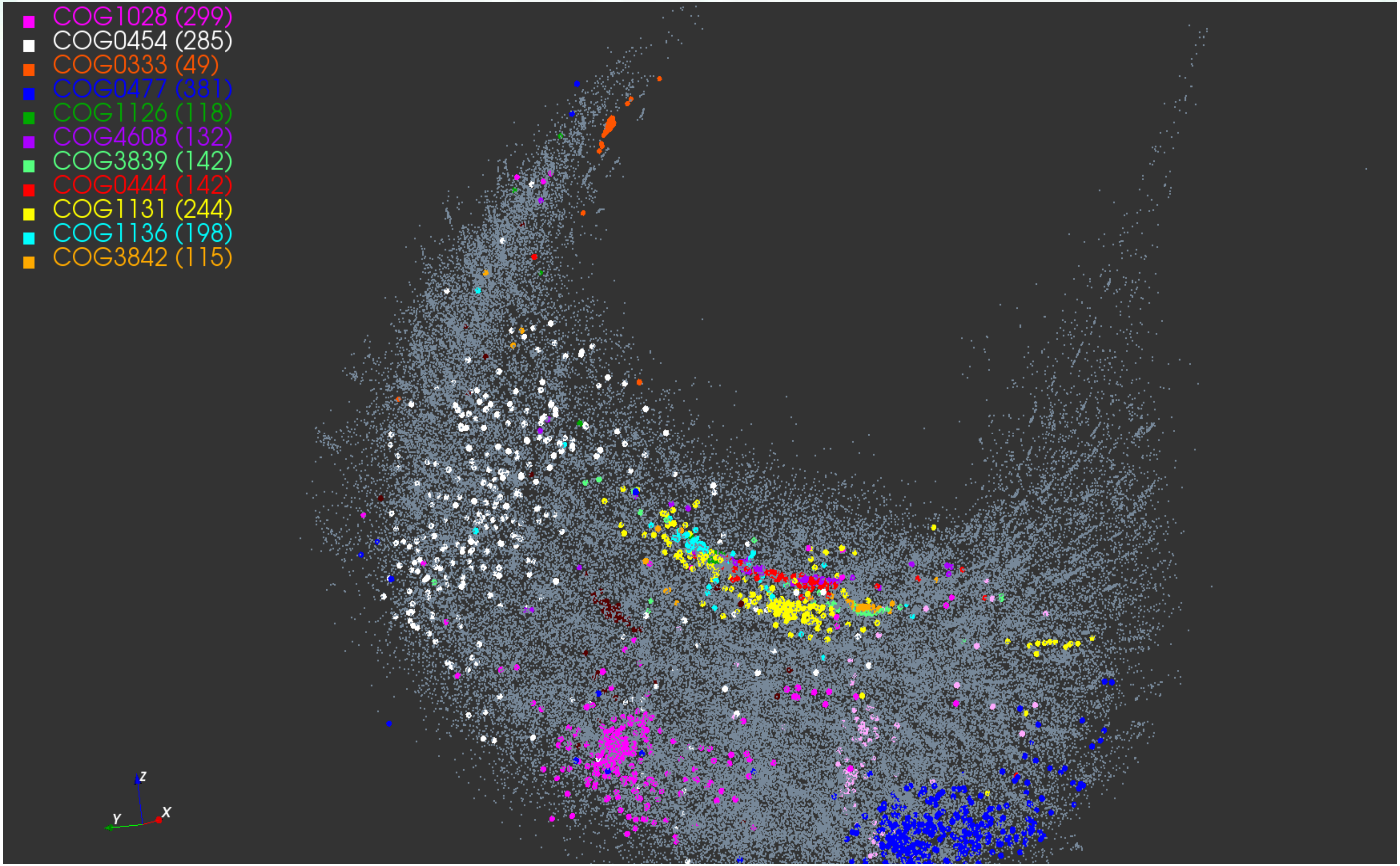
Clusters v. Regions



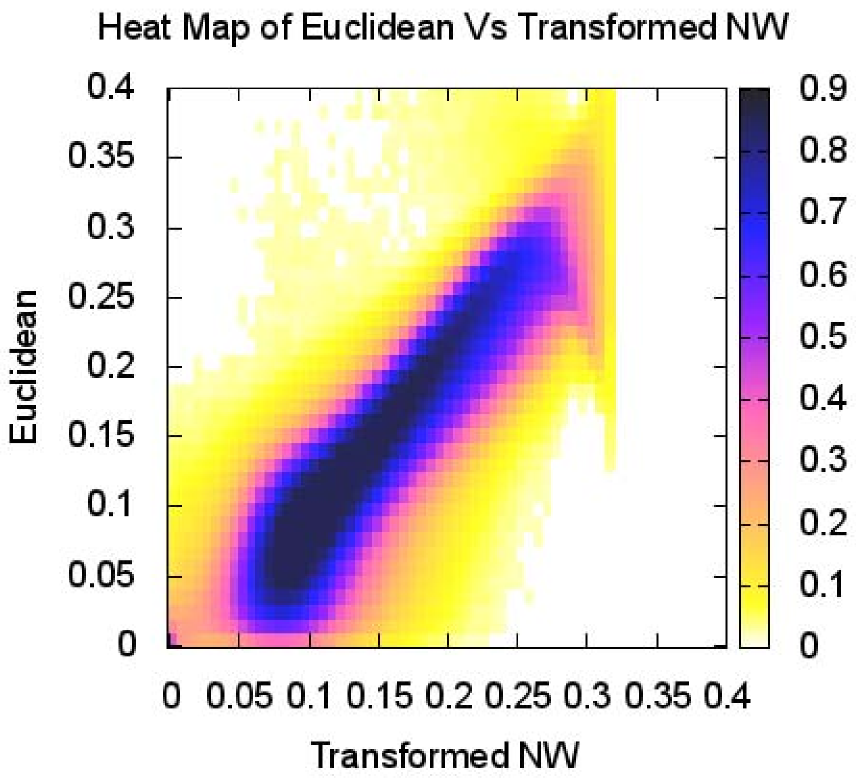


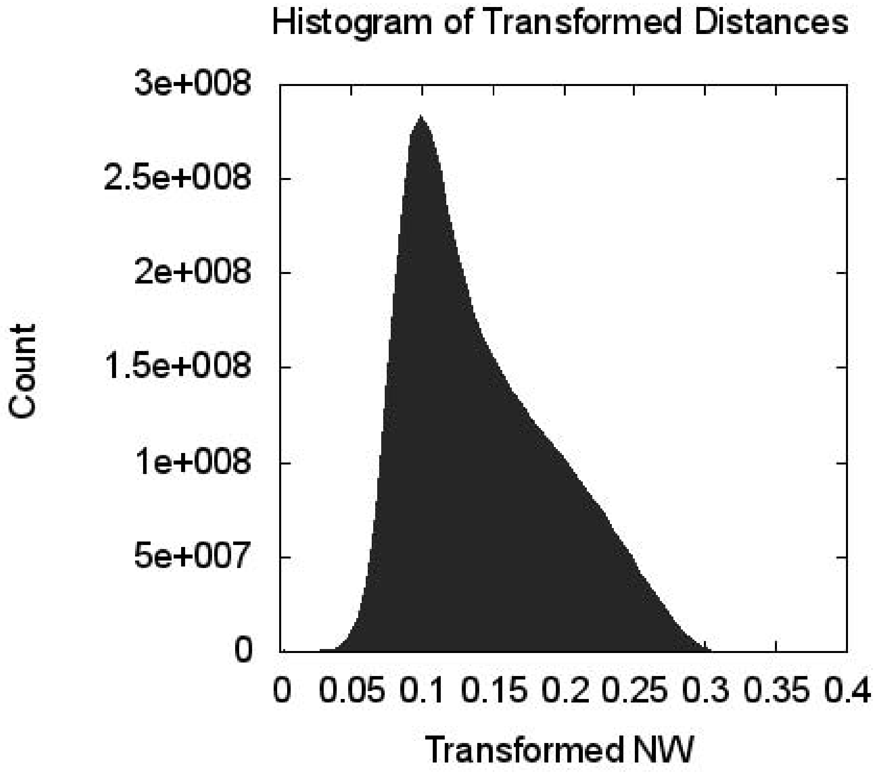
- In Lymphocytes clusters are distinct
- In Pathology, clusters divide space into regions and sophisticated methods like deterministic annealing are probably unnecessary

Protein Universe Browser for COG Sequences with a few illustrative biologically identified clusters



Heatmap of biology distance (Needleman-Wunsch) vs 3D Euclidean Distances

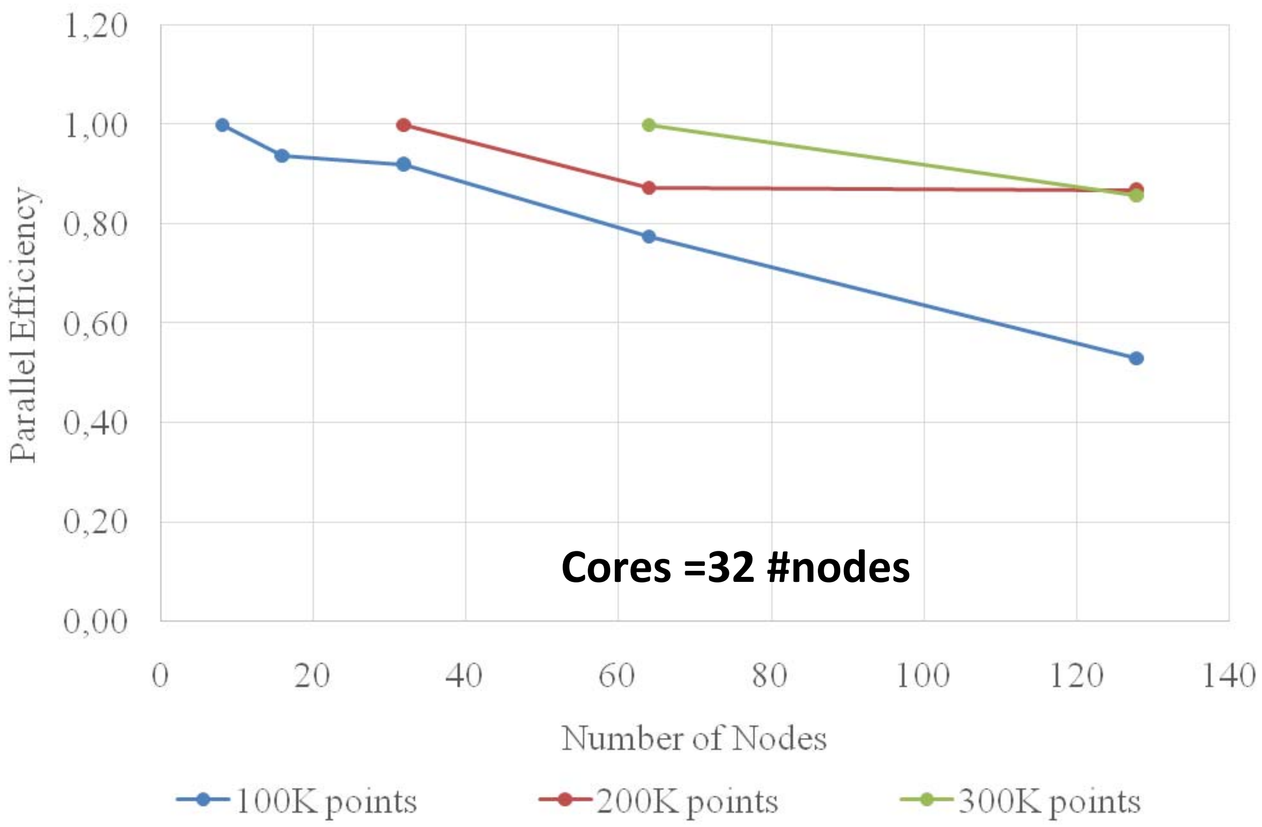




If d a distance, so is f(d) for any monotonic f. Optimize choice of f

WDA SMACOF MDS (Multidimensional Scaling) using Harp on IU Big Red 2

Parallel Efficiency: on 100-300K sequences



Cores =32 #nodes

Best available MDS (much better than that in R)

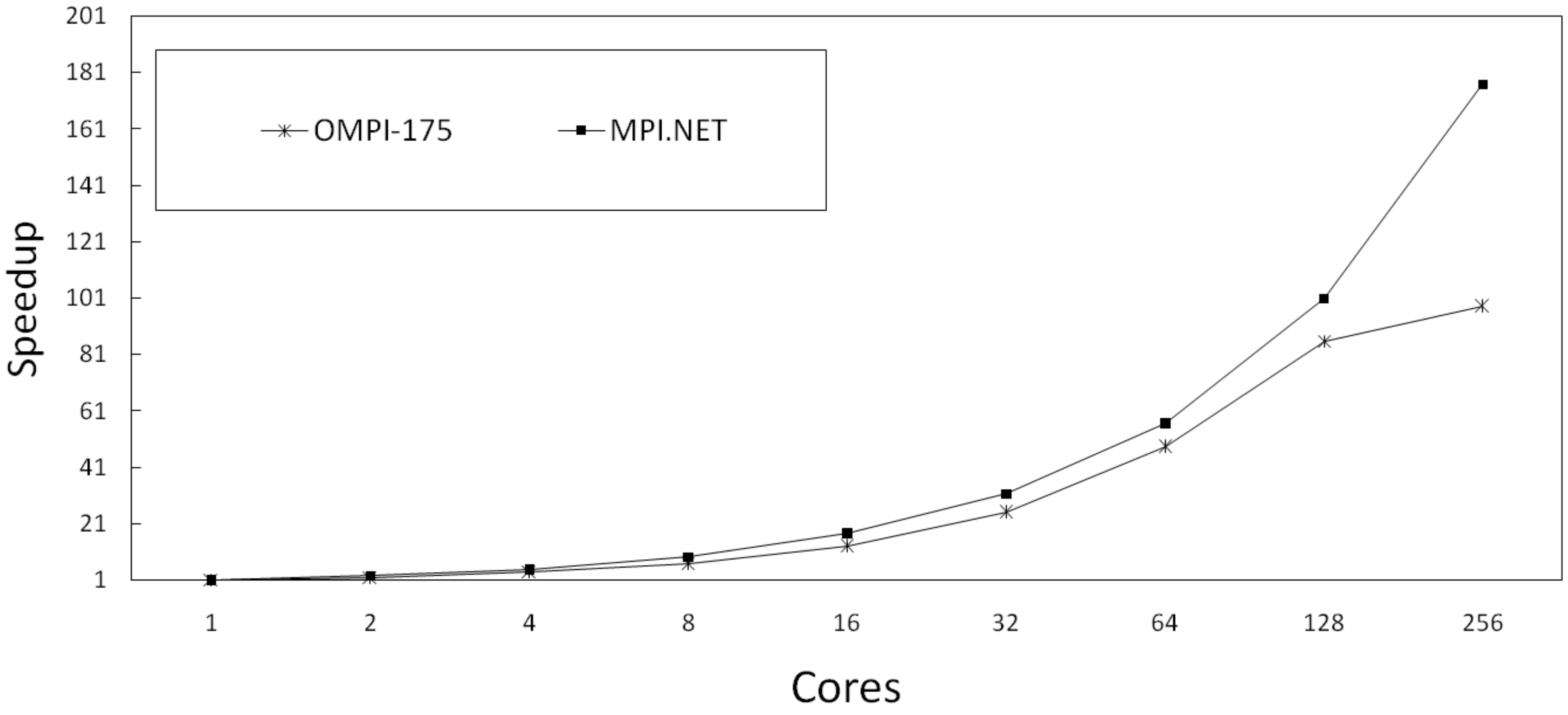
Java

Harp (Hadoop plugin) described by Qiu earlier

Conjugate Gradient (dominant time) and Matrix Multiplication

Non metric Clustering Speed up

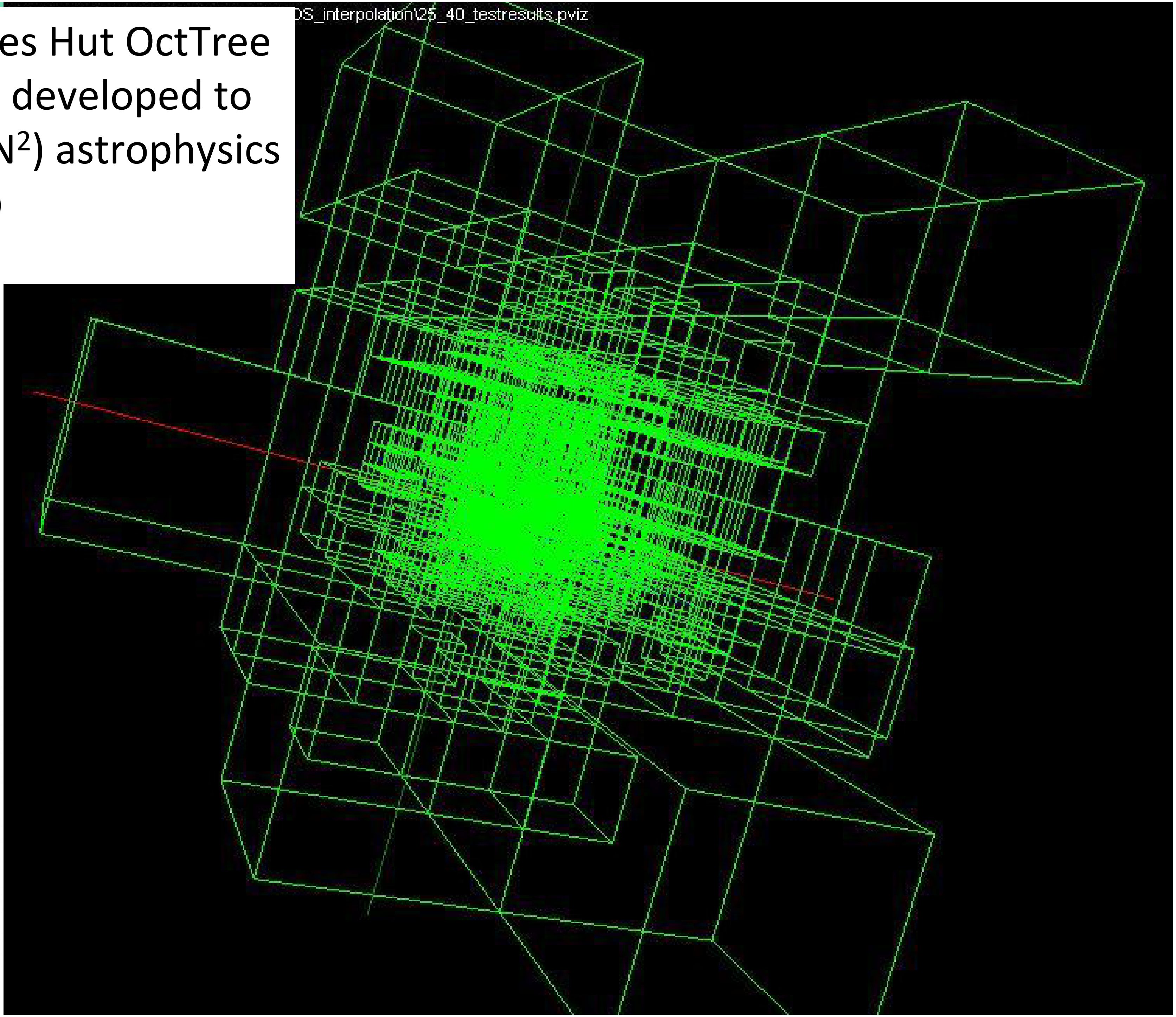
- Small 12000 point clustering
- Note code(s) converted from C# to Java



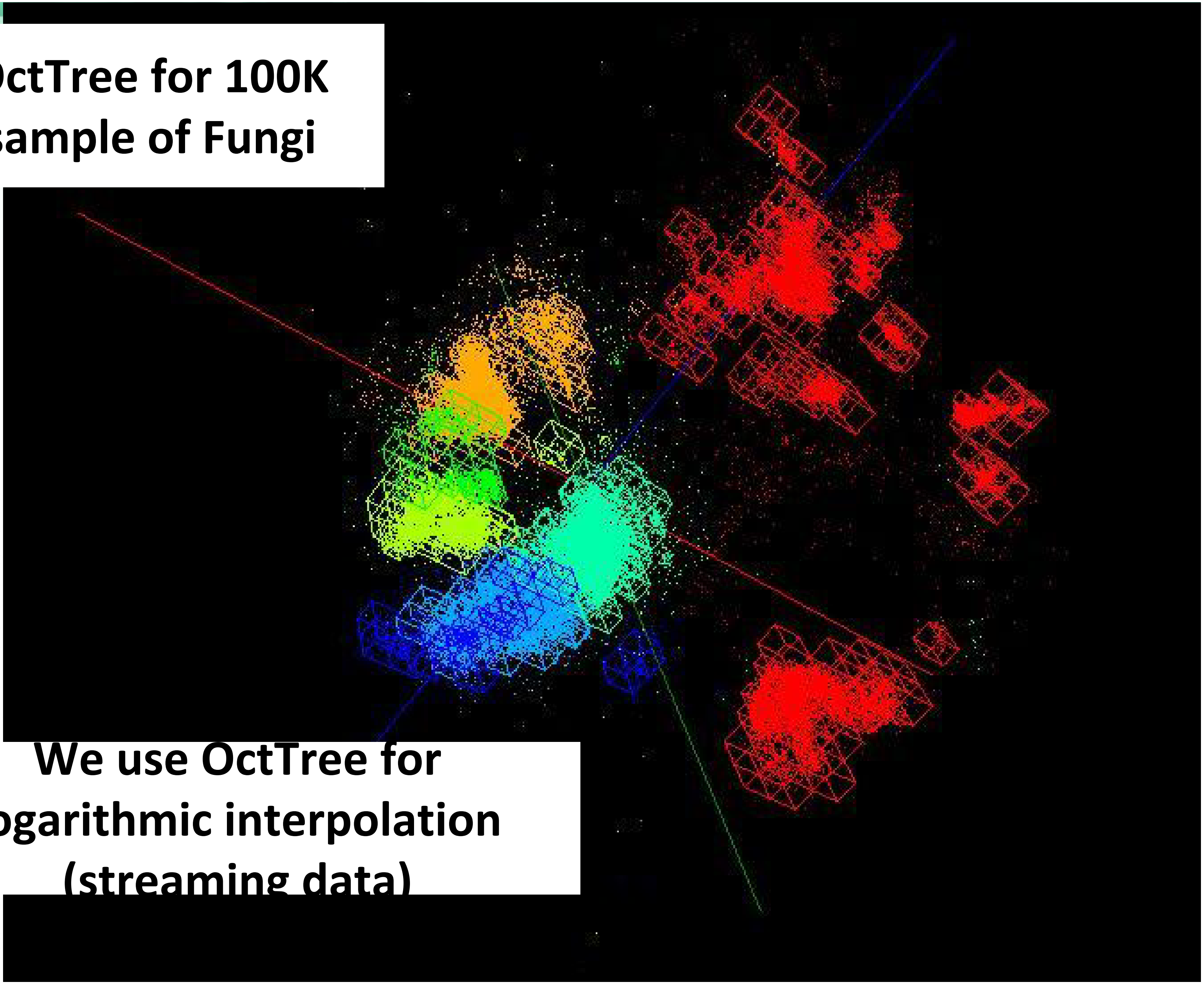
More Efficient Parallelism

- The canonical model is correct at start but each point does not really contribute to each cluster as damped exponentially by exp(- ($\underline{X}_i$- $\underline{Y}(k)$)^2 /T)
- For Proteomics problem, on average only 6.45 clusters needed per point if require ($\underline{X}_i$- $\underline{Y}(k)$)^2 /T ≤ ~40 (as exp(-40) small)
- So only need to keep nearby clusters for each point
- As average number of Clusters ~ 20,000, this gives a factor of ~3000 improvement
- Further communication is no longer all global; it has nearest neighbor components and calculated by parallelism over clusters

Use Barnes Hut OctTree originally developed to make $O(N^2)$ astrophysics $O(N\log N)$



OctTree for 100K sample of Fungi



We use OctTree for logarithmic interpolation (streaming data)

Futures

- Always run MDS. Gives insight into data
- Claim is algorithm change gave as much performance increase as hardware change in simulations. Will this happen in analytics?
- Need to start developing the libraries that support Big Data
  - Understand architectures issues
  - Develop much better algorithms
- Please join **SPIDAL (Scalable Parallel Interoperable Data Analytics Library**) community