

# Deterministic Annealing

Networks and Complex Systems Talk

6pm, Wells Library 001

Indiana University

November 21 2011

Geoffrey Fox

[gcf@indiana.edu](mailto:gcf@indiana.edu)

<http://www.infomall.org> <http://www.futuregrid.org>

Director, Digital Science Center, Pervasive Technology Institute

Associate Dean for Research and Graduate Studies, School of Informatics and Computing

Indiana University Bloomington



<https://portal.futuregrid.org>



# References

- **Ken Rose**, Deterministic Annealing for Clustering, Compression, Classification, Regression, and Related Optimization Problems. Proceedings of the IEEE, 1998. 86: p. 2210--2239.
  - References earlier papers including his Caltech Elec. Eng. PhD 1990
- T Hofmann, JM Buhmann, “Pairwise data clustering by deterministic annealing”, IEEE Transactions on Pattern Analysis and Machine Intelligence 19, pp1-13 1997.
- Hansjörg Klock and Joachim M. Buhmann, “Data visualization by multidimensional scaling: a deterministic annealing approach”, Pattern Recognition, Volume 33, Issue 4, April 2000, Pages 651-669.
- Recent algorithm work by **Seung-Hee Bae, Jong Youl Choi** (Indiana CS PhD's)
- <http://grids.ucs.indiana.edu/ptliupages/publications/CetraroWriteupJune11-09.pdf>
- [http://grids.ucs.indiana.edu/ptliupages/publications/hpdc2010\\_submission\\_57.pdf](http://grids.ucs.indiana.edu/ptliupages/publications/hpdc2010_submission_57.pdf)



<https://portal.futuregrid.org>



# Some Goals

- We are building a library of parallel data mining tools that have best known (to me) robustness and performance characteristics
  - Big data needs super algorithms?
- A lot of statistics tools (e.g. in R) are not the best algorithm and not always well parallelized
- **Deterministic annealing (DA)** is one of better approaches to optimization
  - Tends to remove local optima
  - Addresses overfitting
  - Faster than simulated annealing
- Return to my heritage (physics) with an approach I called **Physical Computation** (23 years ago) -- methods based on analogies to nature
- Physics systems find true lowest energy state if you anneal i.e. you equilibrate at each temperature as you cool



<https://portal.futuregrid.org>





# Some Ideas I

- **Deterministic annealing** is better than many well-used optimization problems
  - Started as “Elastic Net” by Durbin for Travelling Salesman Problem TSP
- Basic idea behind deterministic annealing is **mean field** approximation, which is also used in “Variational Bayes” and many “neural network approaches”
- Markov chain Monte Carlo (MCMC) methods are roughly single temperature **simulated annealing**

- Less sensitive to initial conditions
- Avoid local optima
- Not equivalent to trying random initial starts

## Why Do I Need to Anneal Beads?





# Some non-DA Ideas II

- Dimension reduction gives **Low dimension mappings** of data to both visualize and apply geometric hashing
- **No-vector** (can't define metric space) problems are  $O(N^2)$
- For no-vector case, one can develop  $O(N)$  or  **$O(N \log N)$  methods** as in “Fast Multipole and OctTree methods”
  - Map high dimensional data to 3D and use classic methods developed originally to speed up  $O(N^2)$  3D particle dynamics problems



<https://portal.futuregrid.org>



# 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
  - **No vectors**: MDS (Just use pairwise distances)
- Can apply to **general mixture models** (but less study)
  - **Gaussian Mixture Models**
  - **Probabilistic Latent Semantic Analysis** with Deterministic Annealing DA-PLSA as alternative to **Latent Dirichlet Allocation** (typical informational retrieval/global inference topic model)



<https://portal.futuregrid.org>



# Deterministic Annealing I

- **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
$$F = \langle H - T S(P) \rangle = \int d\chi \{P(\chi)H + T P(\chi) \ln P(\chi)\}$$
- Where  $\chi$  are (a subset of) parameters to be minimized
- **Simulated annealing** corresponds to doing these integrals by Monte Carlo
- **Deterministic annealing** corresponds to doing integrals analytically (by mean field approximation) and is naturally much faster than Monte Carlo
- In each case temperature is lowered slowly – say by a factor 0.95 to 0.99 at each iteration



<https://portal.futuregrid.org>

**SALSA** *HPC*

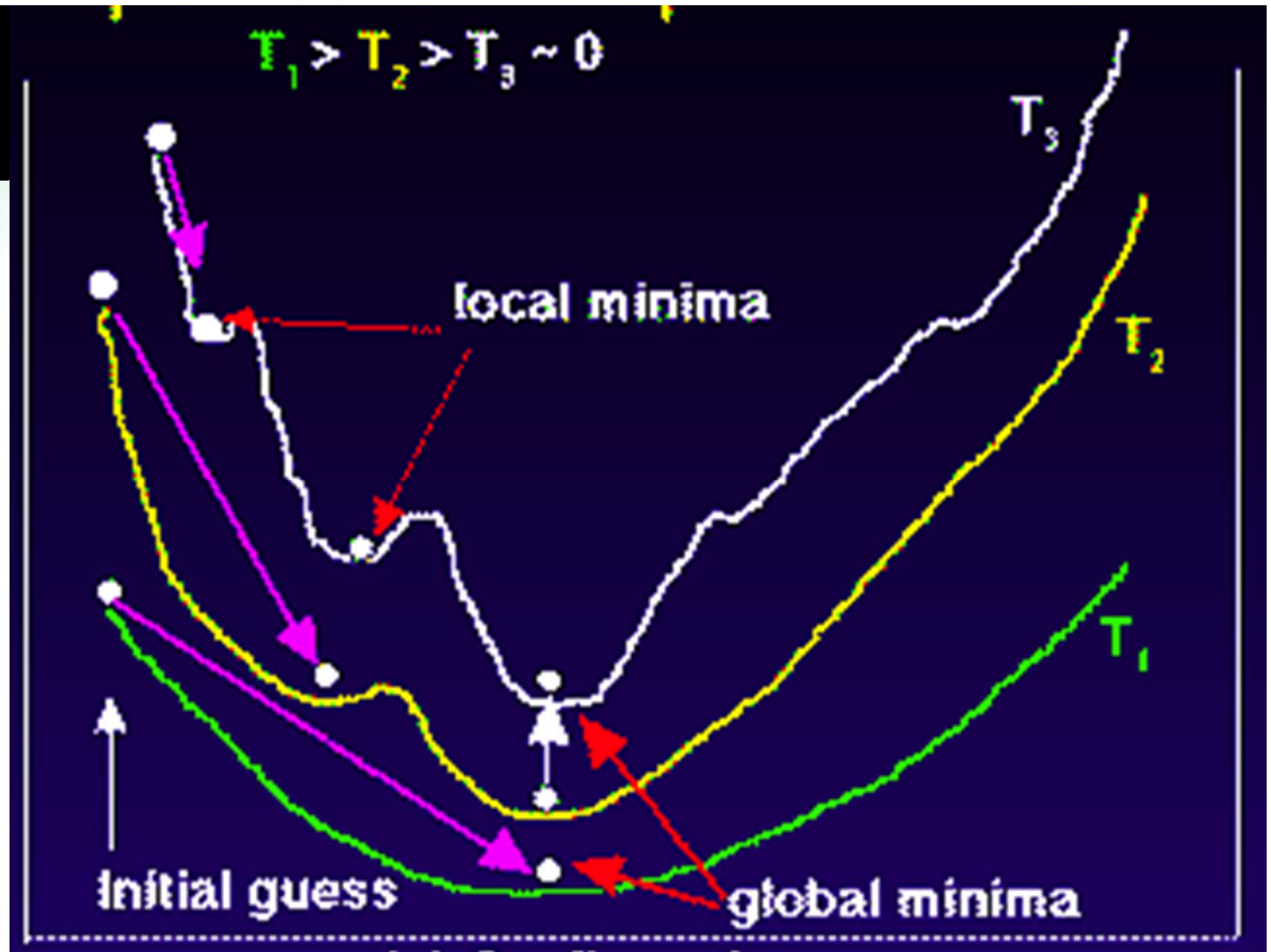


# Deterministic Annealing



$$F(\{y\}, T)$$

Solve Linear Equations for each temperature

Nonlinear effects mitigated by initializing with solution at previous higher temperature



**Configuration {y}**

-  Minimum evolving as temperature decreases
-  Movement at fixed temperature going to local minima if not initialized "correctly"

# Deterministic Annealing II

For some cases such as vector clustering and Mixture

Note 3 types of variables

$\underline{\varepsilon}$  used to approximate real Hamiltonian

$\underline{\chi}$  subject to annealing

The rest – optimized by traditional methods

tractable integrals

- $P_0(\underline{\chi}) = \exp( - H_0(\underline{\chi})/T + F_0/T )$  approximate Gibbs for H
- $F_R(P_0) = \langle H_R - T S_0(P_0) \rangle|_0 = \langle H_R - H_0 \rangle|_0 + F_0(P_0)$
- Where  $\langle \dots \rangle|_0$  denotes  $\int d\underline{\chi} P_0(\underline{\chi})$
- Easy to show that real Free Energy (the Gibb's inequality)  
 $F_R(P_R) \leq F_R(P_0)$  (Kullback-Leibler divergence)
- Expectation step E is find  $\underline{\varepsilon}$  minimizing  $F_R(P_0)$  and
- Follow with M step (of EM) setting  $\underline{\chi} = \langle \underline{\chi} \rangle|_0 = \int d\underline{\chi} \underline{\chi} P_0(\underline{\chi})$   
(mean field) and one follows with a traditional minimization of remaining parameters



<https://portal.futuregrid.org>



# Implementation of DA Central Clustering

- Clustering variables are  $M_i(k)$  (these are  $\chi$  in general approach) where this is probability point  $i$  belongs to cluster  $k$
- In **Central** or **PW** Clustering, take  $H_0 = \sum_{i=1}^N \sum_{k=1}^K M_i(k) \varepsilon_i(k)$ 
  - Linear form allows DA integrals to be done analytically
- **Central clustering** has  $\varepsilon_i(k) = (\underline{X}(i) - \underline{Y}(k))^2$  and  $M_i(k)$  determined by Expectation step
  - $H_{\text{Central}} = \sum_{i=1}^N \sum_{k=1}^K M_i(k) (\underline{X}(i) - \underline{Y}(k))^2$
  - $H_{\text{central}}$  and  $H_0$  are identical
- $\langle M_i(k) \rangle = \exp(-\varepsilon_i(k)/T) / \sum_{k=1}^K \exp(-\varepsilon_i(k)/T)$
- Centers  $\underline{Y}(k)$  are determined in M step



# Implementation of DA-PWC

- Clustering variables are again  $M_i(k)$  (these are  $x$  in general approach) where this is probability point  $i$  belongs to cluster  $k$
- **Pairwise Clustering** Hamiltonian given by nonlinear form
- $H_{PWC} = 0.5 \sum_{i=1}^N \sum_{j=1}^N \delta(i, j) \sum_{k=1}^K M_i(k) M_j(k) / C(k)$
- $\delta(i, j)$  is pairwise distance between points  $i$  and  $j$
- with  $C(k) = \sum_{i=1}^N M_i(k)$  as number of points in Cluster  $k$
- Take same form  $H_0 = \sum_{i=1}^N \sum_{k=1}^K M_i(k) \varepsilon_i(k)$  as for central clustering
- $\varepsilon_i(k)$  determined to minimize  $F_{PWC}(P_0) = \langle H_{PWC} - T S_0(P_0) \rangle|_0$  where integrals can be easily done
- And now linear (in  $M_i(k)$ )  $H_0$  and quadratic  $H_{PC}$  are different
- Again  $\langle M_i(k) \rangle = \exp(-\varepsilon_i(k)/T) / \sum_{k=1}^K \exp(-\varepsilon_i(k)/T)$

# General Features of DA

- Deterministic Annealing DA is related to Variational Inference or Variational Bayes methods
- In many problems, decreasing temperature is classic **multiscale** – finer resolution ( $\sqrt{T}$  is “just” distance scale)
  - We have factors like  $(\underline{X}(i) - \underline{Y}(k))^2 / T$
- In clustering, one then looks at **second derivative matrix** of  $F_R(P_0)$  wrt  $\underline{\varepsilon}$  and as temperature is lowered this develops **negative eigenvalue** corresponding to instability
  - Or have multiple clusters at each center and perturb
- This is a **phase transition** and one splits cluster into two and continues EM iteration
- One can start with just one cluster

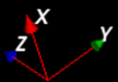


<https://portal.futuregrid.org>



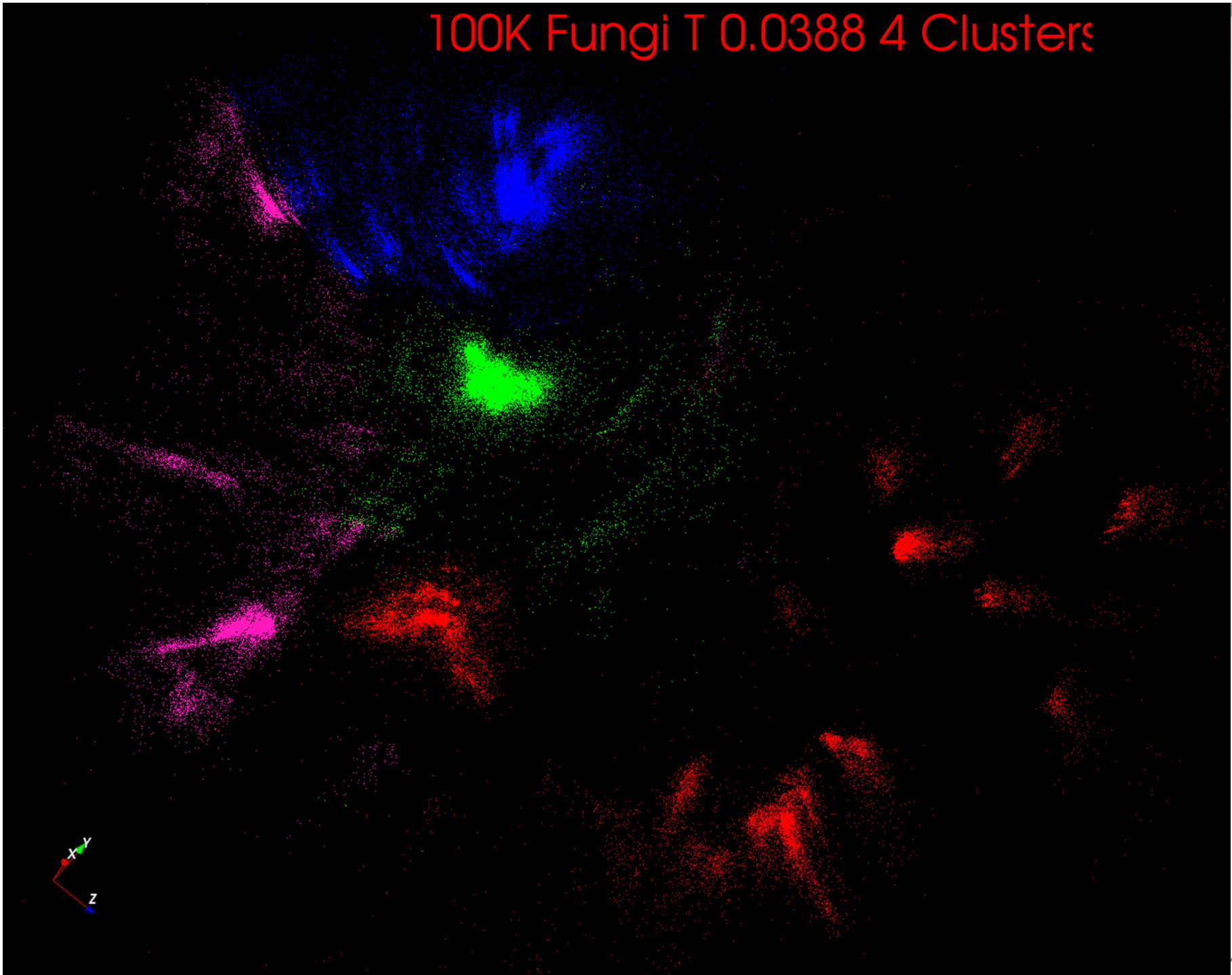
100K\_Fungi T = 0.0475C

- Start at  $T = \infty$  with 1 Cluster
- Decrease T, Clusters emerge at instabilities





# 100K Fungi T 0.0388 4 Clusters

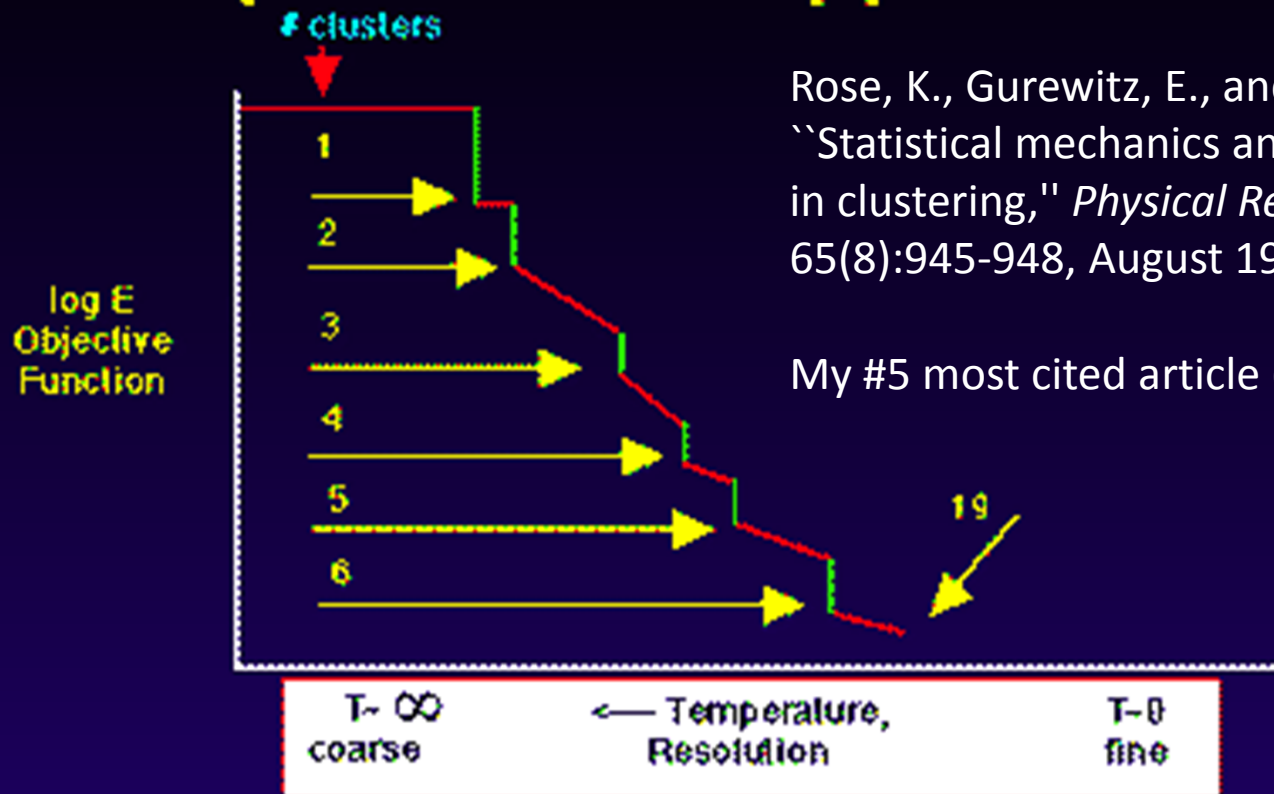




100K Fungi T=0.03180 6 Clusters



# Phase Transitions in Physical Optimization Approach



Rose, K., Gurewitz, E., and Fox, G. C.  
"Statistical mechanics and phase transitions  
in clustering," *Physical Review Letters*,  
65(8):945-948, August 1990.

My #5 most cited article (387 cites)

- The clustering problem - like any good physical system - exhibits phase transitions as one lowers the temperature



# DA-PWC EM Steps (**E is red**, M Black)

k runs over clusters; i,j points

$$1) A(k) = - 0.5 \sum_{i=1}^N \sum_{j=1}^N \delta(i, j) \langle M_i(k) \rangle \langle M_j(k) \rangle / \langle C(k) \rangle^2$$

$$2) B_j(k) = \sum_{i=1}^N \delta(i, j) \langle M_i(k) \rangle / \langle C(k) \rangle$$

$$3) \varepsilon_i(k) = (B_i(k) + A(k))$$

$$4) \langle M_i(k) \rangle = p(k) \exp(-\varepsilon_i(k)/T) / \sum_{k=1}^K p(k) \exp(-\varepsilon_i(k)/T)$$

$$5) C(k) = \sum_{i=1}^N \langle M_i(k) \rangle$$

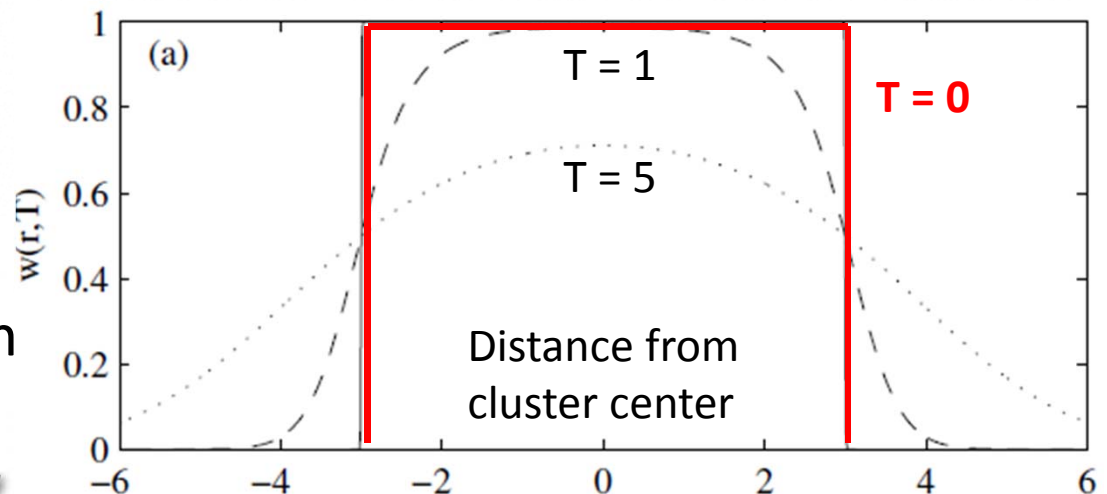
$$6) p(k) = C(k) / N$$

Steps 1 global sum (reduction)  
Step 1, 2, 5 local sum if  $\langle M_i(k) \rangle$   
broadcast

- Loop to converge variables; decrease T from  $\infty$ ;  
split centers by halving p(k)

# 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  
 $(\underline{X}(i) - \underline{Y}(k))^2 / 2\sigma(k)^2 < c^2 / 2$
- Another case when  $H_0$  is same as target Hamiltonian
- Proteomics Mass Spectrometry



# High Performance Dimension Reduction and Visualization

- Need is pervasive
  - Large and high dimensional data are everywhere: biology, physics, Internet, ...
  - Visualization can help data analysis
- Visualization of large datasets with high performance
  - Map high-dimensional data into low dimensions (2D or 3D).
  - Need Parallel programming for processing large data sets
  - Developing high performance dimension reduction algorithms:
    - MDS(Multi-dimensional Scaling)
    - GTM(Generative Topographic Mapping)
    - DA-MDS(Deterministic Annealing MDS)
    - DA-GTM(Deterministic Annealing GTM)
  - Interactive visualization tool **PlotViz**



<https://portal.futuregrid.org>



# Multidimensional Scaling MDS

- Map points in high dimension to lower dimensions
- Many such dimension reduction algorithms (PCA Principal component analysis easiest); simplest but perhaps best at times is MDS
- Minimize Stress
$$\sigma(\underline{X}) = \sum_{i < j=1}^n \text{weight}(i,j) (\delta(i,j) - d(\underline{X}_i, \underline{X}_j))^2$$
- $\delta(i,j)$  are input dissimilarities and  $d(\underline{X}_i, \underline{X}_j)$  the Euclidean distance squared in embedding space (3D usually)
- SMACOF or Scaling by minimizing a complicated function is clever steepest descent (expectation maximization EM) algorithm
- Computational complexity goes like  $N^2 * \text{Reduced Dimension}$
- We describe Deterministic annealed version of it which is much better
- Could just view as non linear  $\chi^2$  problem (Tapia et al. Rice)
  - Slower but more general
- All parallelize with high efficiency



<https://portal.futuregrid.org>





# Implementation of MDS

- $H_{MDS} = \sum_{i < j=1}^n \text{weight}(i,j) (\delta(i, j) - d(\underline{X}(i), \underline{X}(j)))^2$
- Where  $\delta(i, j)$  are observed dissimilarities and we want to represent as Euclidean distance between points  $\underline{X}(i)$  and  $\underline{X}(j)$
- $H_{MDS}$  is quartic or involves square roots, so we need the idea of an approximate Hamiltonian  $H_0$
- One tractable integral form for  $H_0$  was linear Hamiltonians
- Another is Gaussian  $H_0 = \sum_{i=1}^n (\underline{X}(i) - \underline{\mu}(i))^2 / 2$
- Where  $\underline{X}(i)$  are vectors to be determined as in formula for Multidimensional scaling
- The E step is minimize  $\sum_{i < j=1}^n \text{weight}(i,j) (\delta(i, j) - \text{constant.T} - (\underline{\mu}(i) - \underline{\mu}(j))^2)^2$
- with solution  $\underline{\mu}(i) = 0$  at large T
- Points pop out from origin as Temperature lowered

# Pairwise Clustering and MDS are $O(N^2)$ Problems

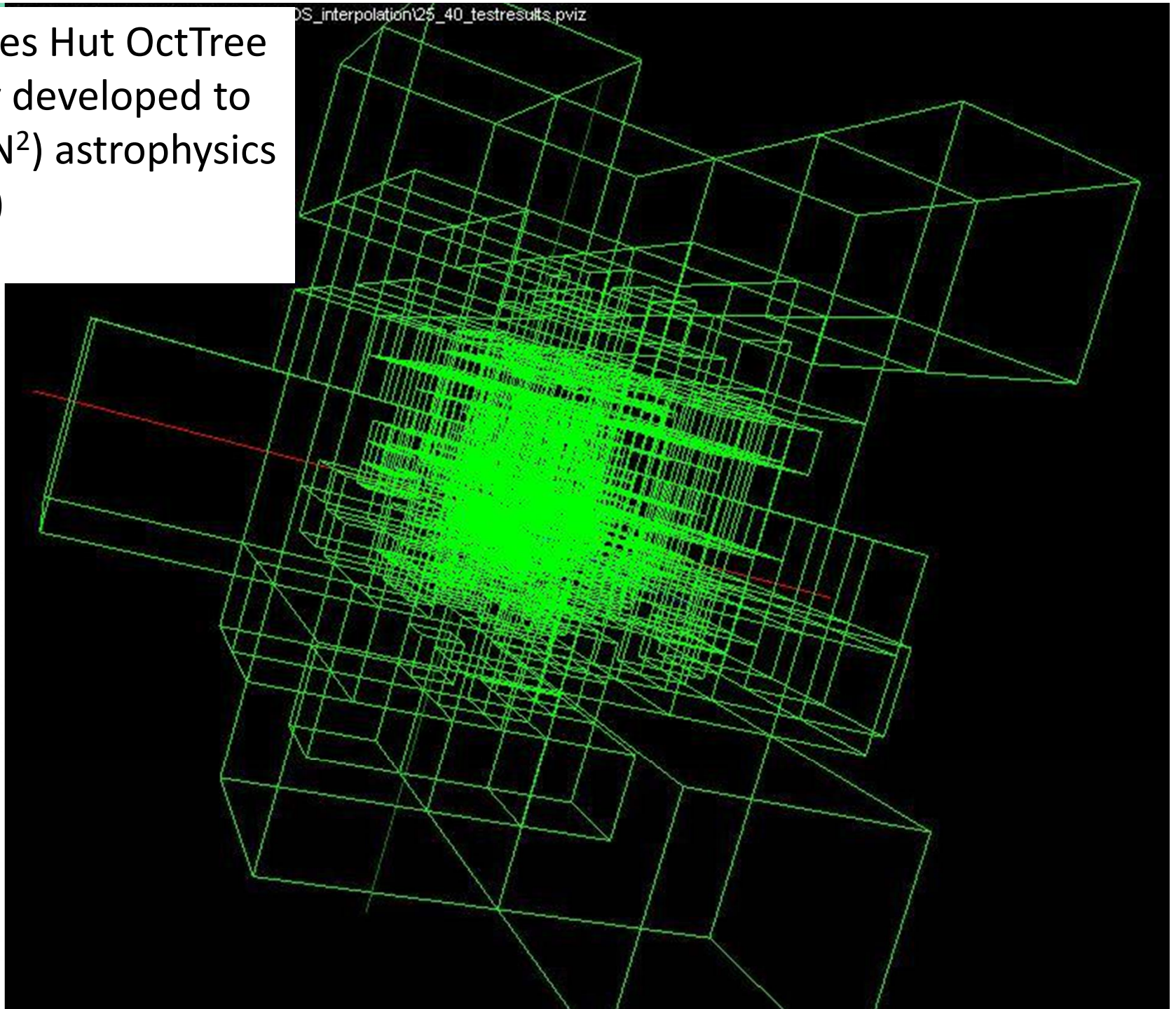
- 100,000 sequences takes a few days on 768 cores  
32 nodes Windows Cluster Tempest
- Could just run 440K on 4.4<sup>2</sup> larger machine but lets try to be “cleverer” and use hierarchical methods
- Start with 100K sample run fully
- Divide into “megaregions” using 3D projection
- Interpolate full sample into megaregions and analyze latter separately
- See [http://salsahpc.org/millionseq/16SrRNA\\_index.html](http://salsahpc.org/millionseq/16SrRNA_index.html)



<https://portal.futuregrid.org>

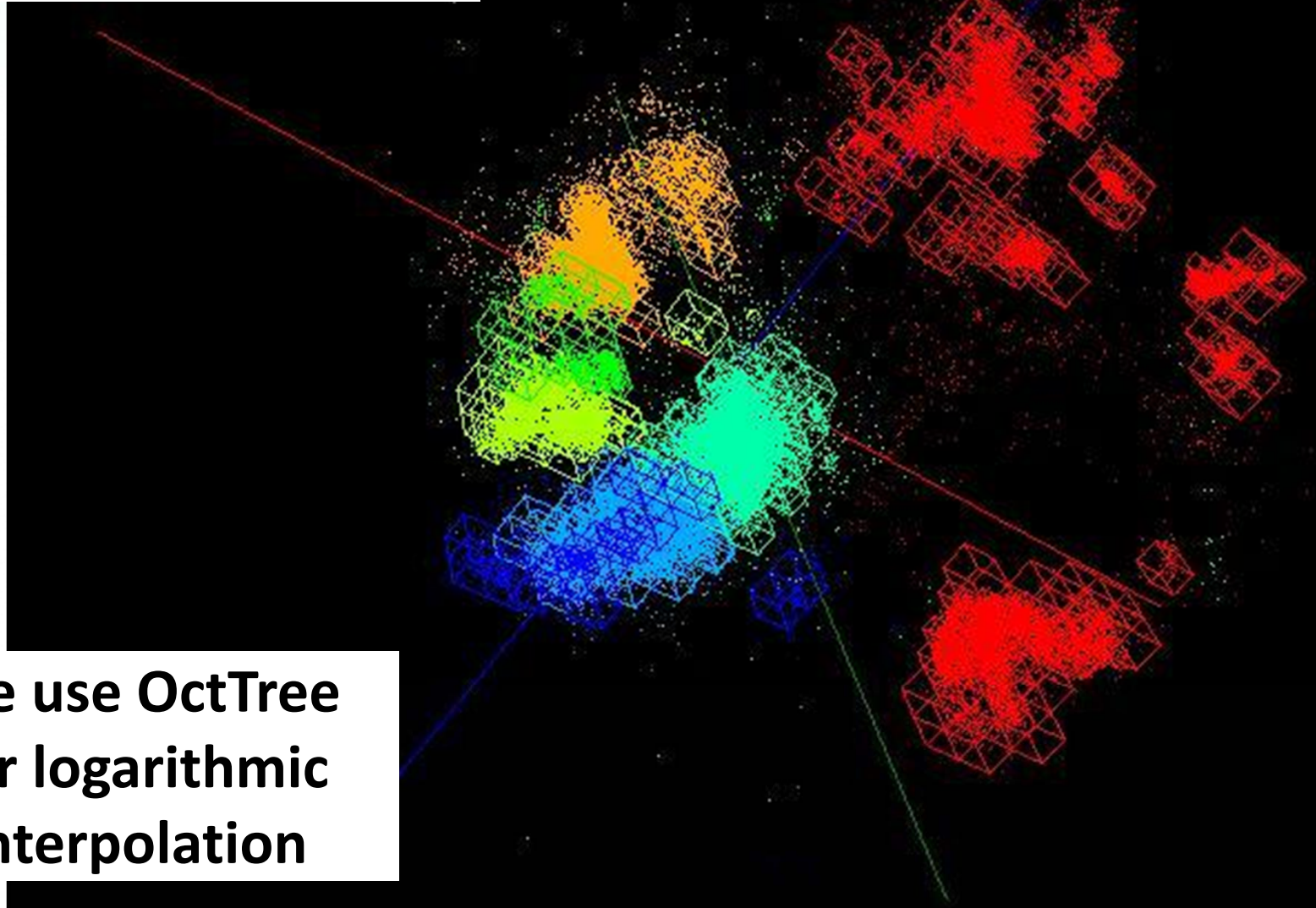


Use Barnes Hut OctTree  
originally developed to  
make  $O(N^2)$  astrophysics  
 $O(N\log N)$





## OctTree for 100K sample of Fungi



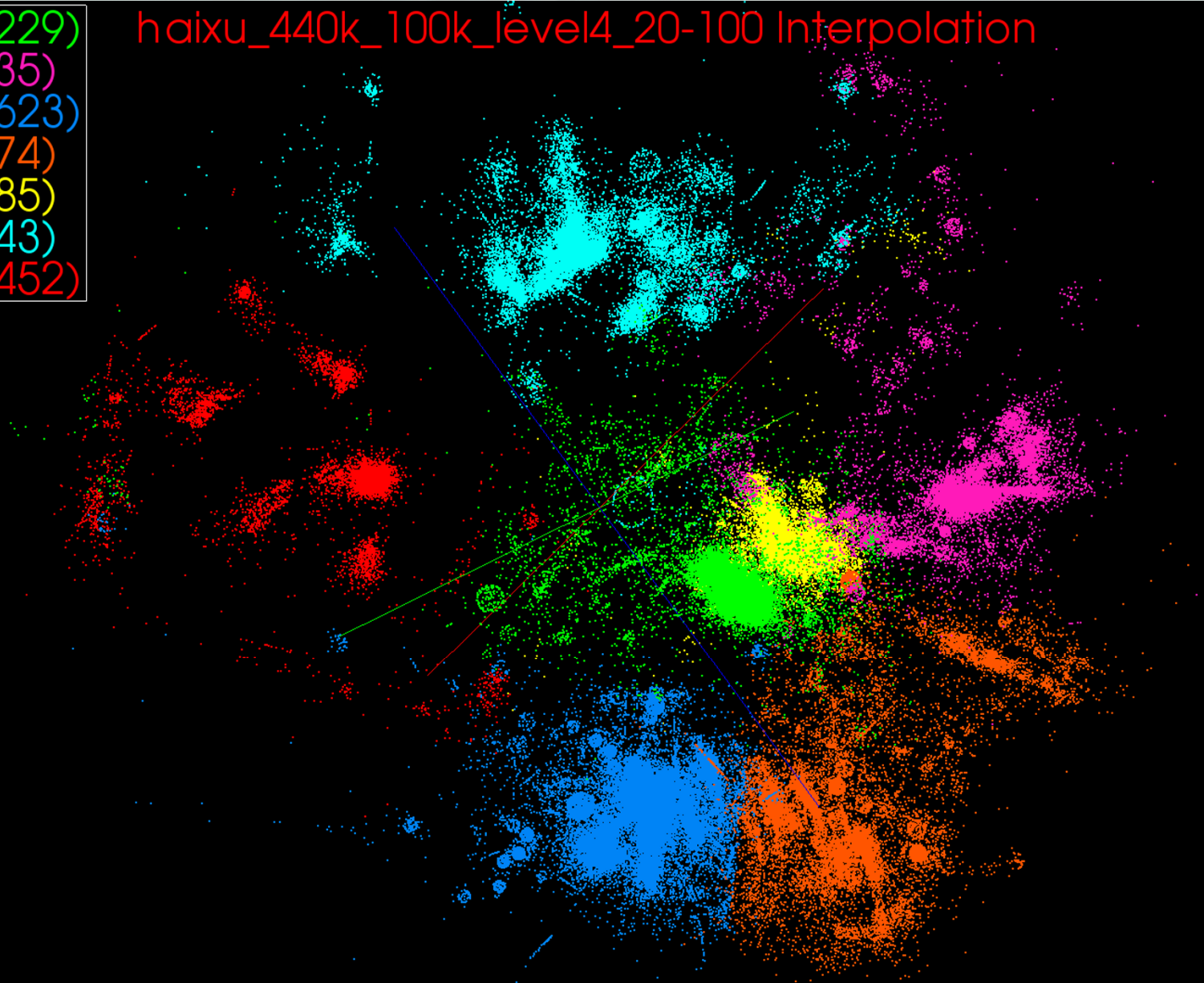
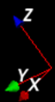
We use OctTree  
for logarithmic  
interpolation



# 440K Interpolated

■ 2 (111229)  
■ 3 (50235)  
■ 5 (114623)  
■ 6 (42174)  
■ 8 (73885)  
■ 9 (38443)  
■ 10 (15452)

haixu\_440k\_100k\_level4\_20-100 Interpolation



# A large cluster in Region 0

■	0	(63959)
■	1	(1777)
■	2	(476)
■	3	(1255)
■	4	(885)
■	5	(3554)
■	6	(2946)
■	7	(10590)
■	8	(25787)

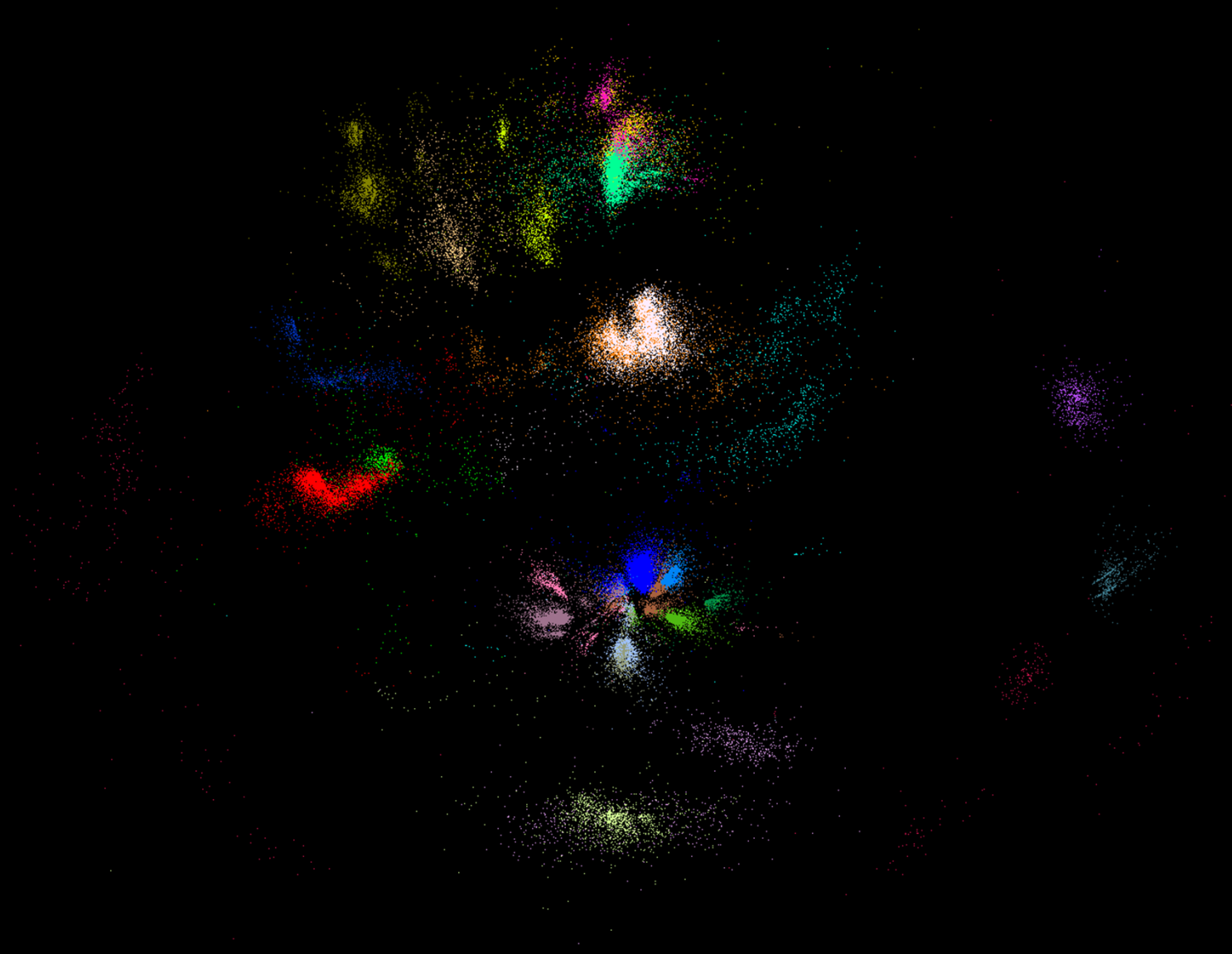
Haixu Metaregion 0 111229 Seqs 9 Clusters



# 26 Clusters in Region 4

Haixu Megaregion 4 73885 Seqs 26 Clusters

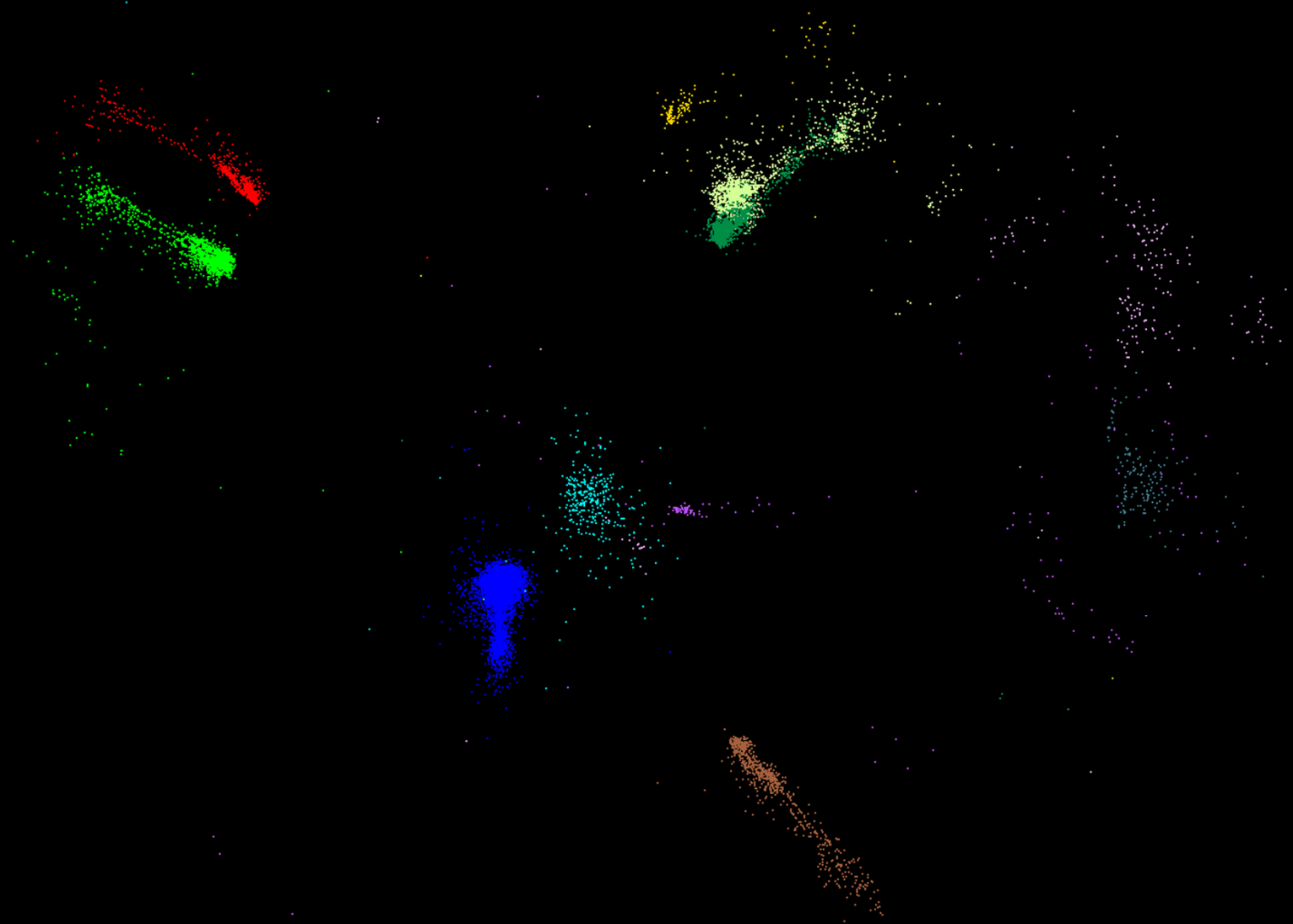
- 0 (15930)
- 1 (4569)
- 2 (673)
- 3 (2514)
- 4 (2089)
- 5 (2156)
- 6 (688)
- 7 (2833)
- 8 (906)
- 9 (569)
- 10 (852)
- 11 (1426)
- 12 (587)
- 13 (472)
- 14 (2219)
- 15 (5496)
- 16 (8557)
- 17 (5394)
- 18 (1337)
- 19 (786)
- 20 (1184)
- 21 (4137)
- 22 (2767)
- 23 (1926)
- 24 (2855)
- 25 (963)



# 13 Clusters in Region 6

Haixu Megaregion 6 15452 Seqs 13 Clusters

- 0 (7384)
- 1 (744)
- 2 (2246)
- 4 (137)
- 6 (1507)
- 7 (753)
- 8 (336)
- 9 (159)
- 10 (210)
- 11 (1797)
- 12 (179)

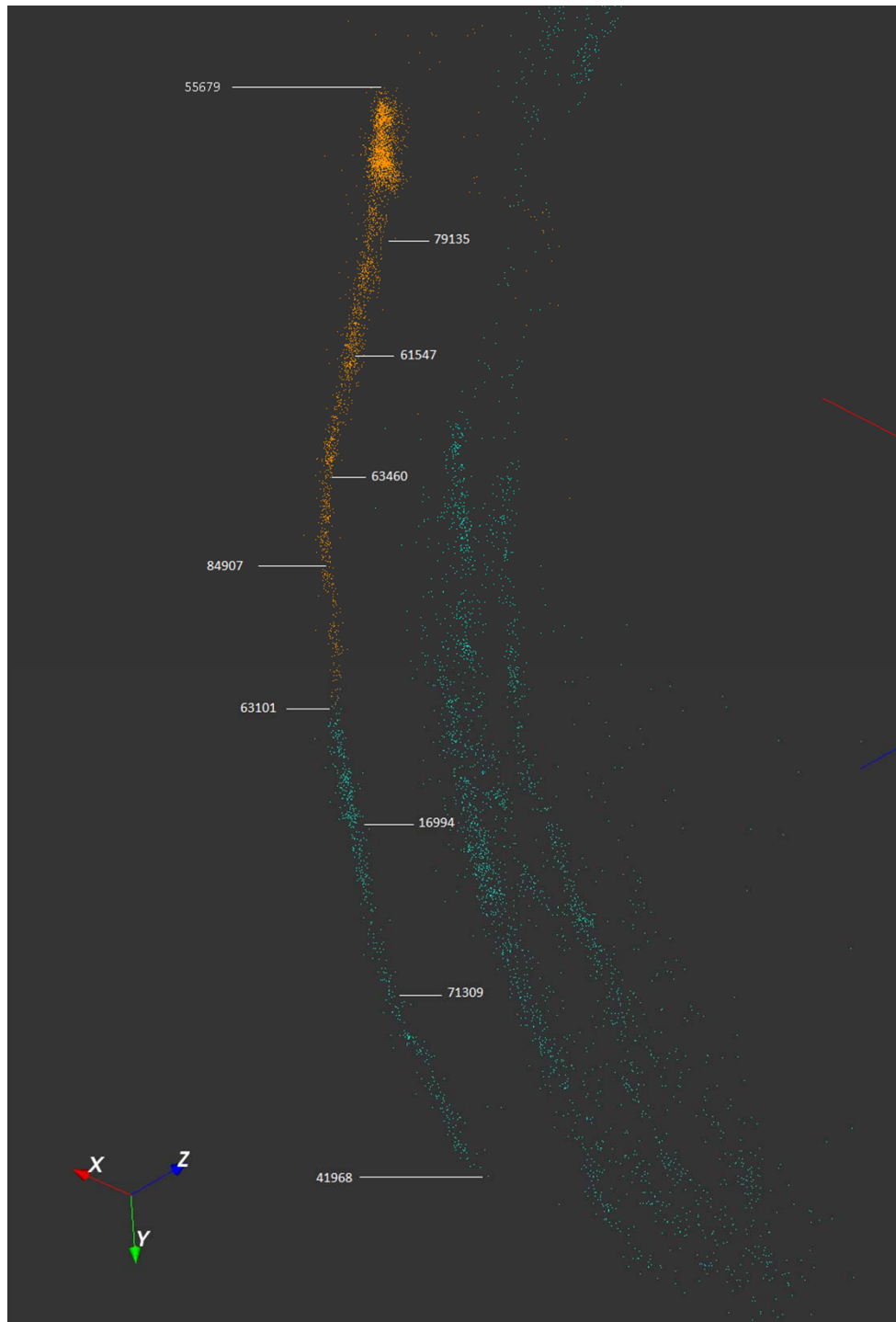




16sRNA 100K Needleman Wunsch Distances \_

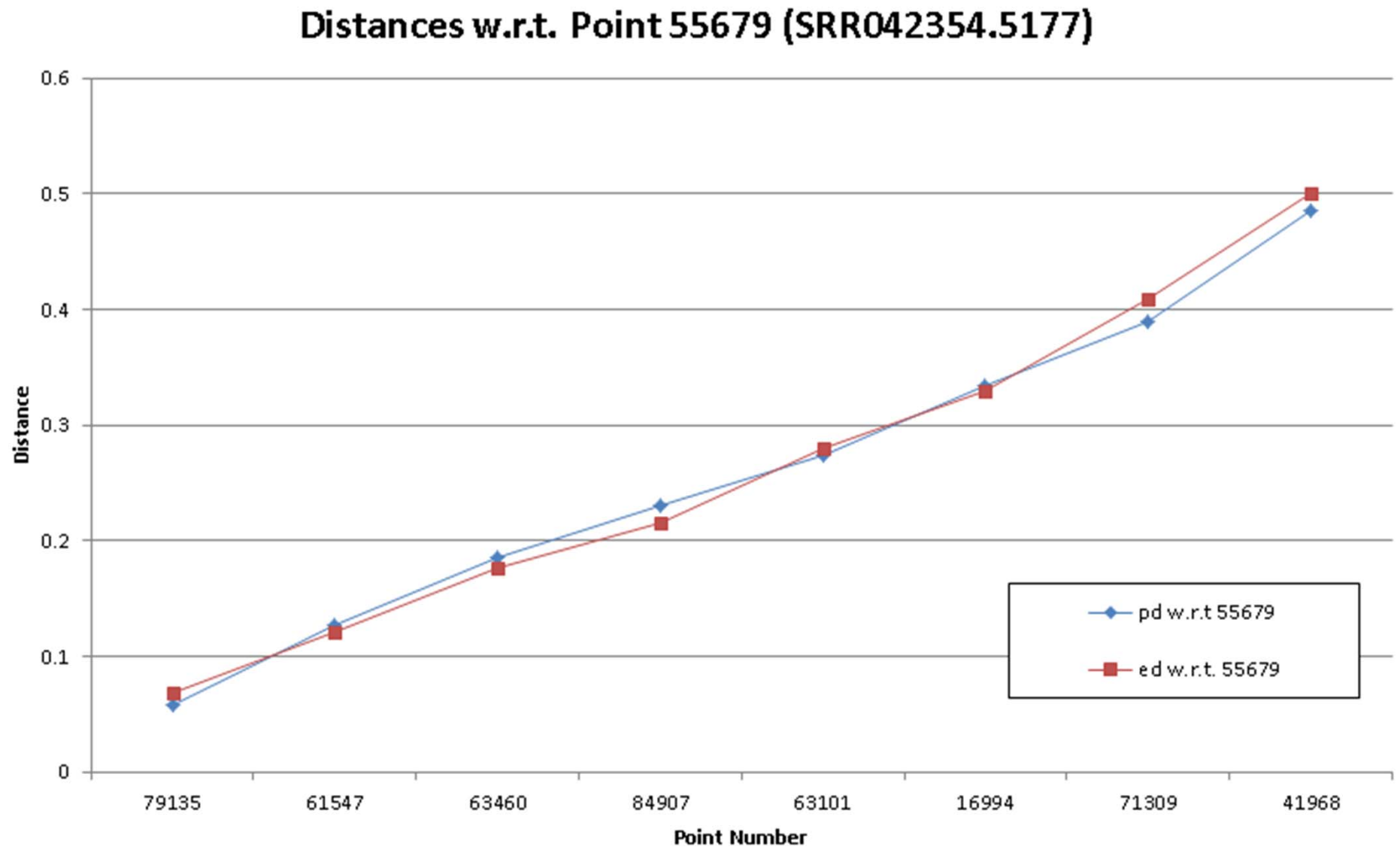
# Understanding the Octopi





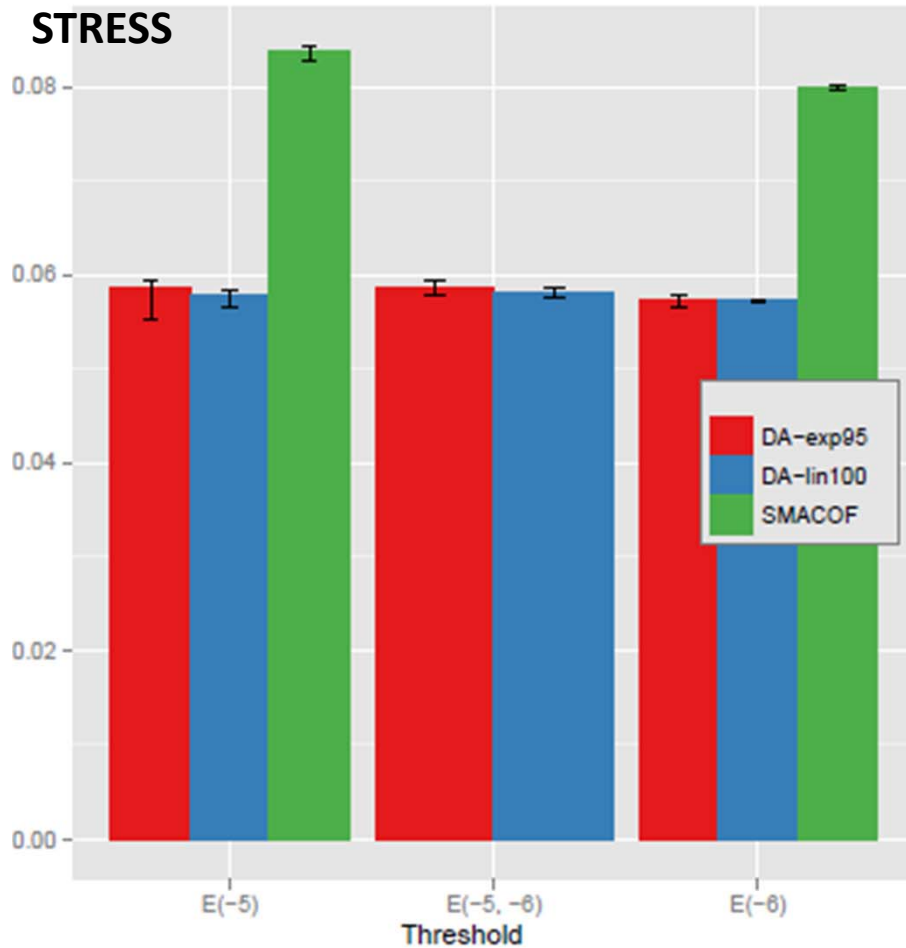
- The octopi are globular clusters distorted by length dependence of dissimilarity measure
- Sequences are 200 to 500 base pairs long
- We **restarted** project using local (SWG) not global (NW) alignment

- Note mapped (Euclidean 3D shown as red) and abstract dissimilarity (blue) are very similar



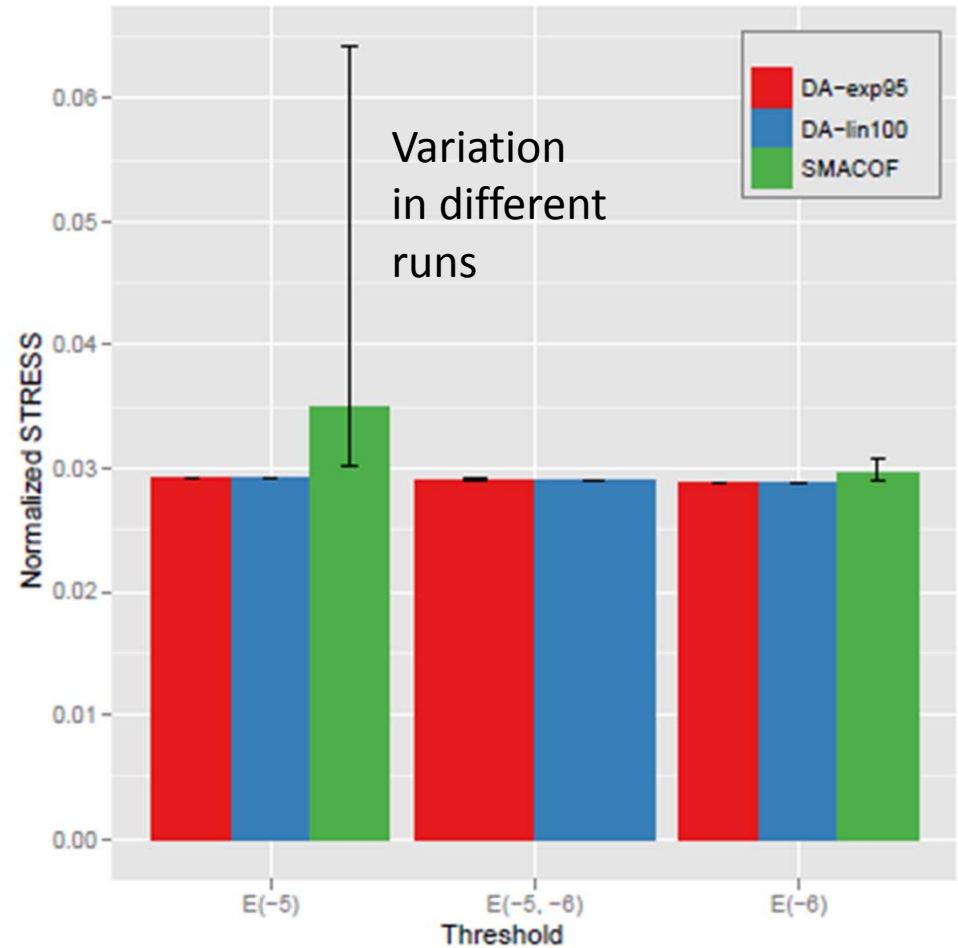
# Quality of DA versus EM MDS

Normalized  
STRESS



Map to 2D

100K Metagenomics



Map to 3D



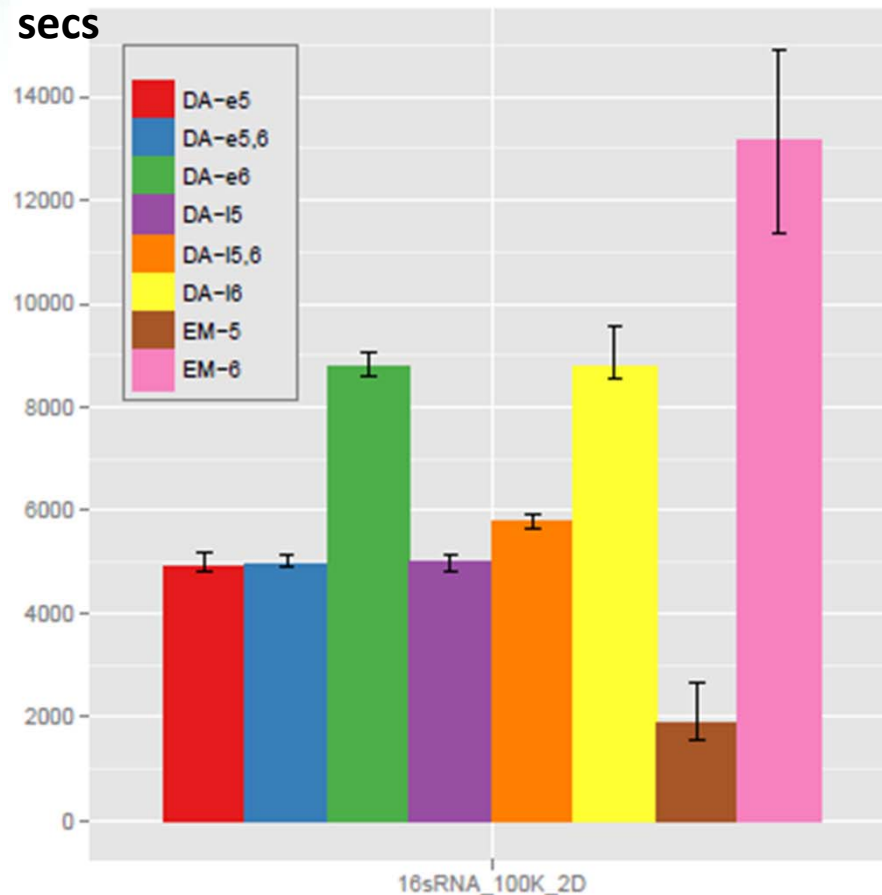
<https://portal.futuregrid.org>

**SALSA**<sub>32</sub> HPC



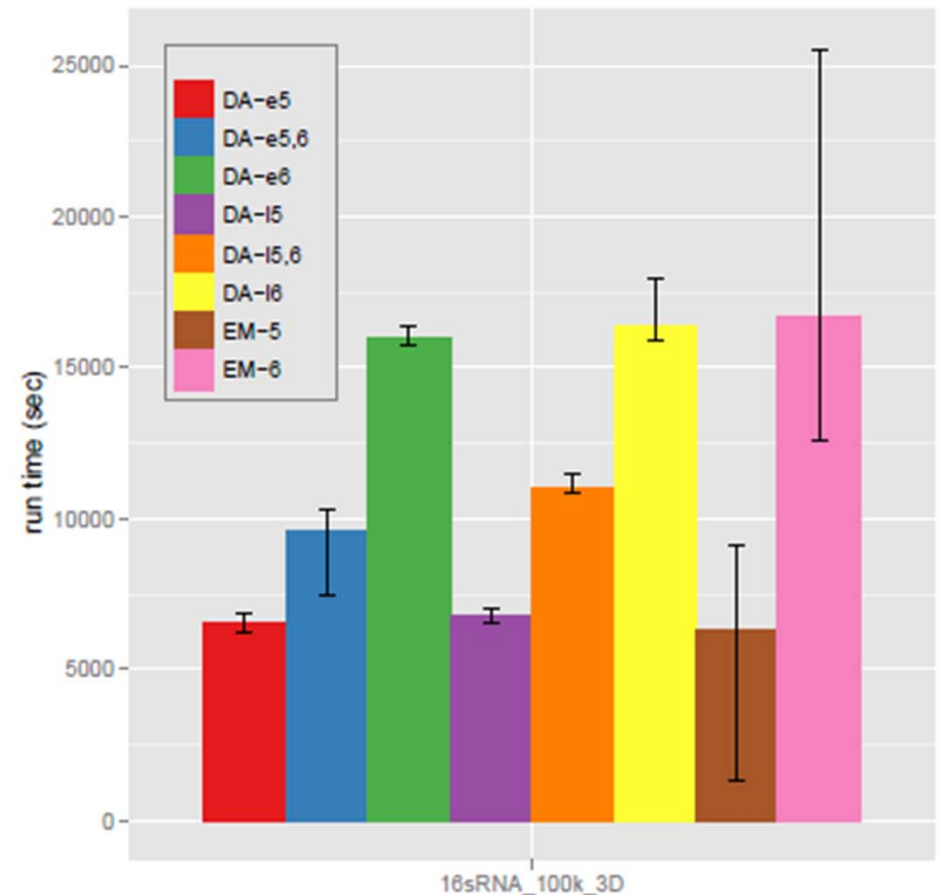
# Run Time of DA versus EM MDS

Run time  
secs



Map to 2D

100K Metagenomics



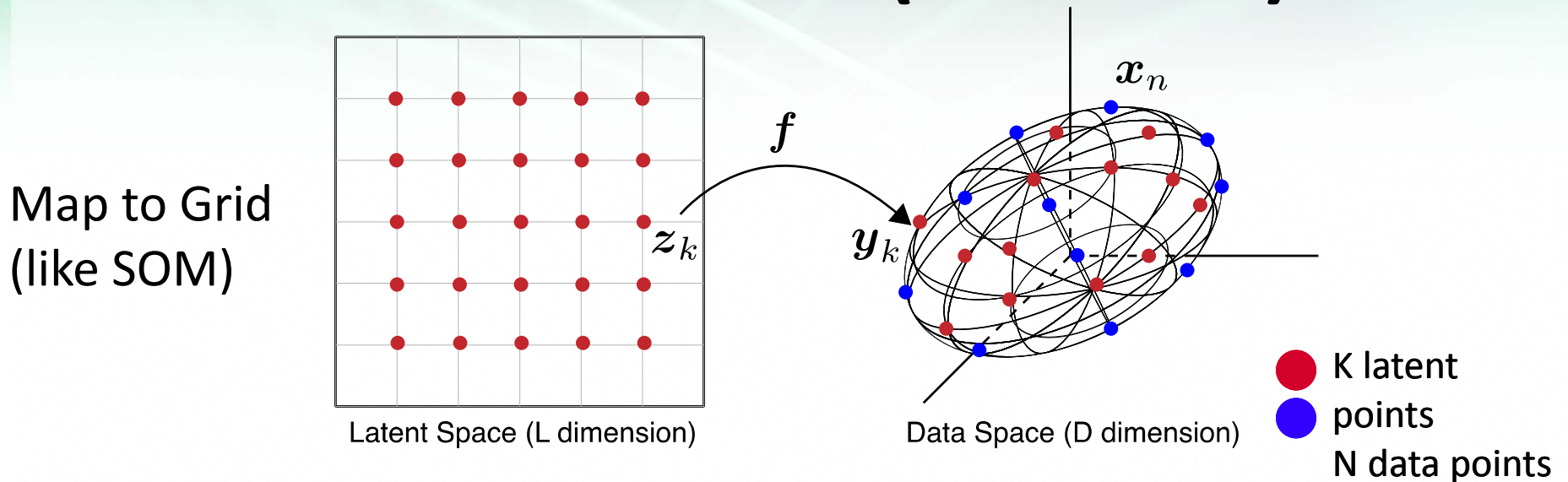
Map to 3D



<https://portal.futuregrid.org>



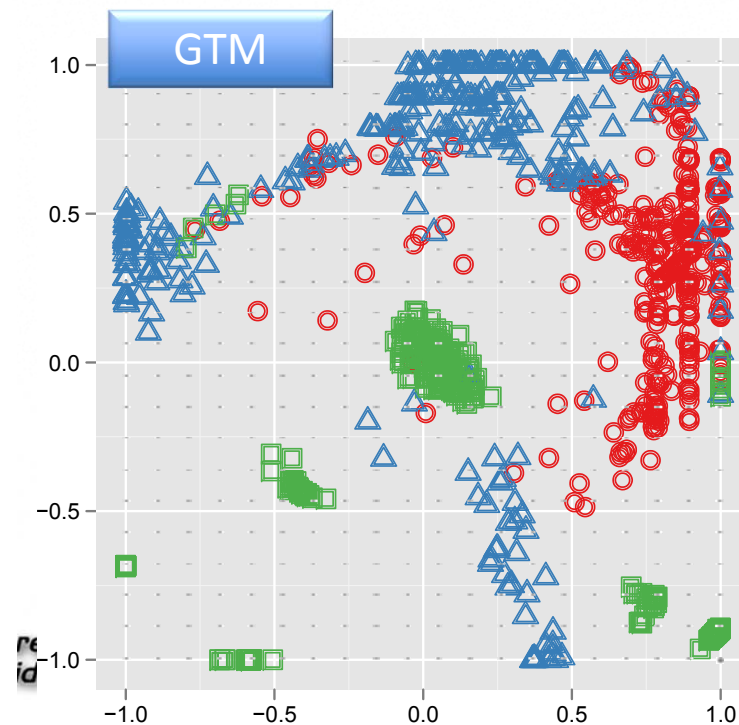
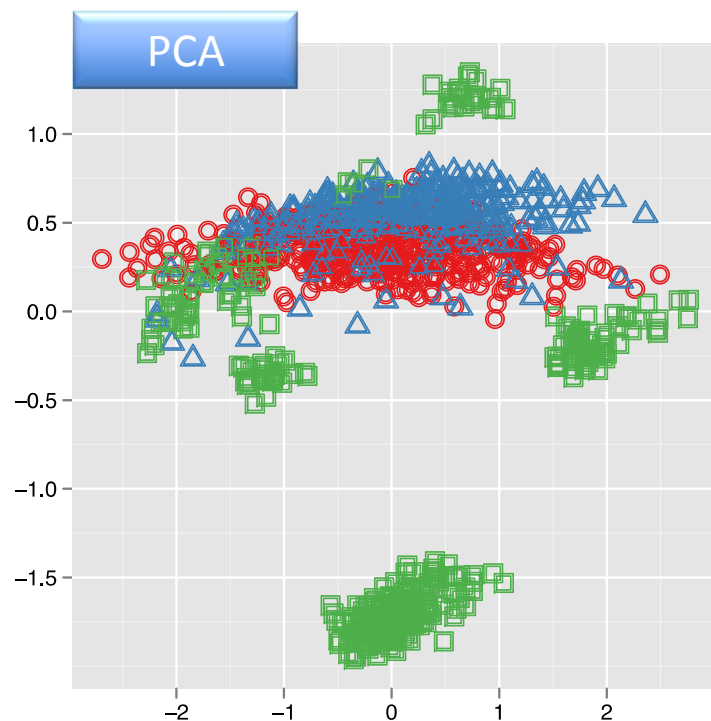
# GTM with DA (DA-GTM)



- GTM is an algorithm for dimension reduction
  - Find optimal K latent variables in Latent Space
  - $f$  is a non-linear mapping function
  - Traditional algorithm use EM for model fitting
- DA optimization can improve the fitting process

# Advantages of GTM

- Computational complexity is  $O(KN)$ , where
  - $N$  is the number of data points
  - $K$  is the number of latent variables or *clusters*.  $K \ll N$
- Efficient, compared with MDS which is  $O(N^2)$
- Produce more separable map (right) than PCA (left)



**Oil flow data**  
1000 points  
12 Dimensions  
3 Clusters

# Free Energy for DA-GTM

- Free Energy

$$F = D - TH = -T \sum_{n=1}^N \ln Z_n$$

- D : expected distortion
- H : Shannon entropy
- T : computational temperature
- $Z_n$  : partitioning function

- Partition Function for GTM

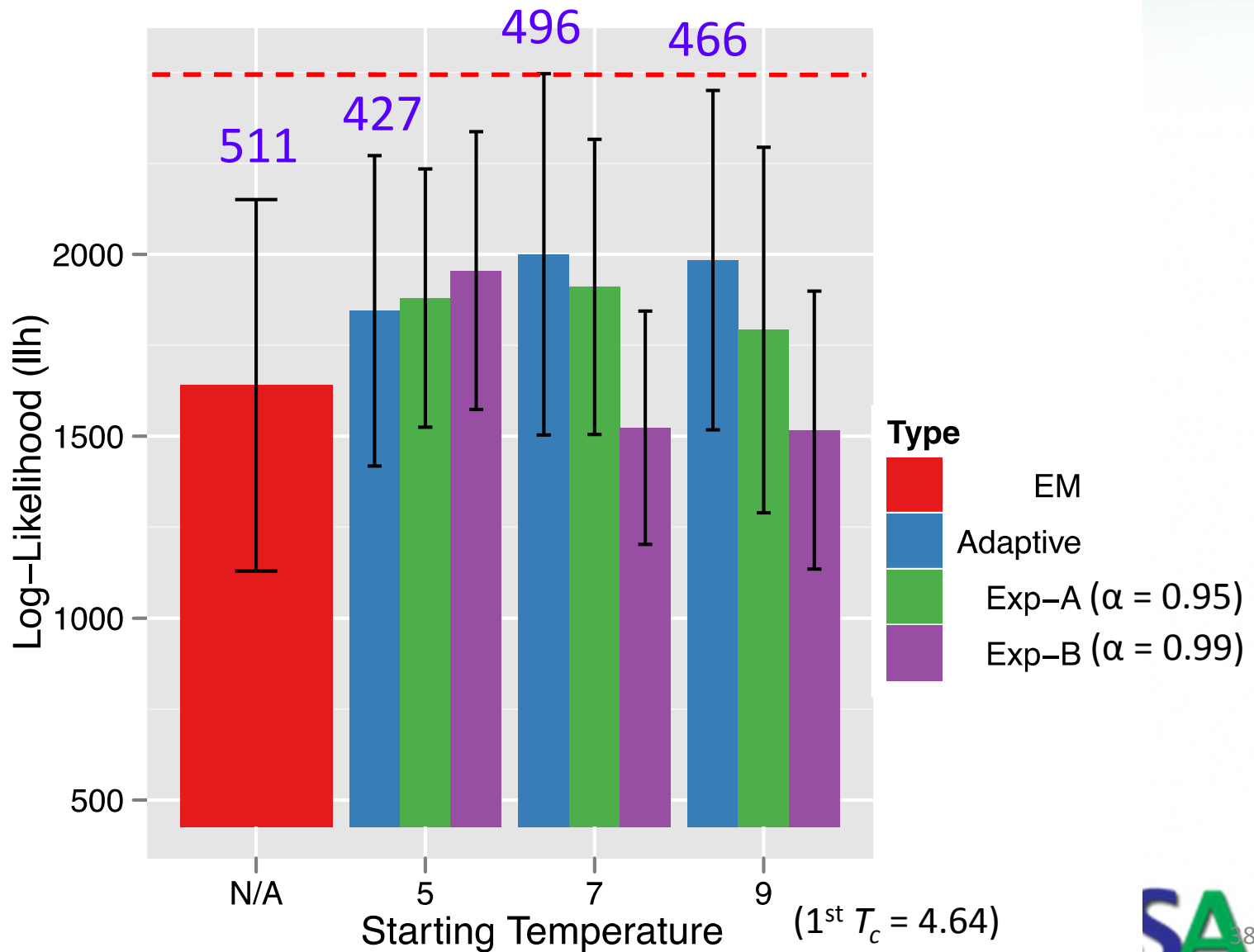
$$Z_n = \sum_{k=1}^K \exp \left( \frac{-d_{nk}}{T} \right) \quad d_{nk} = -\log \left( \frac{\mathcal{N}(\mathbf{x}_n, \mathbf{y}_k)}{T} \right)$$



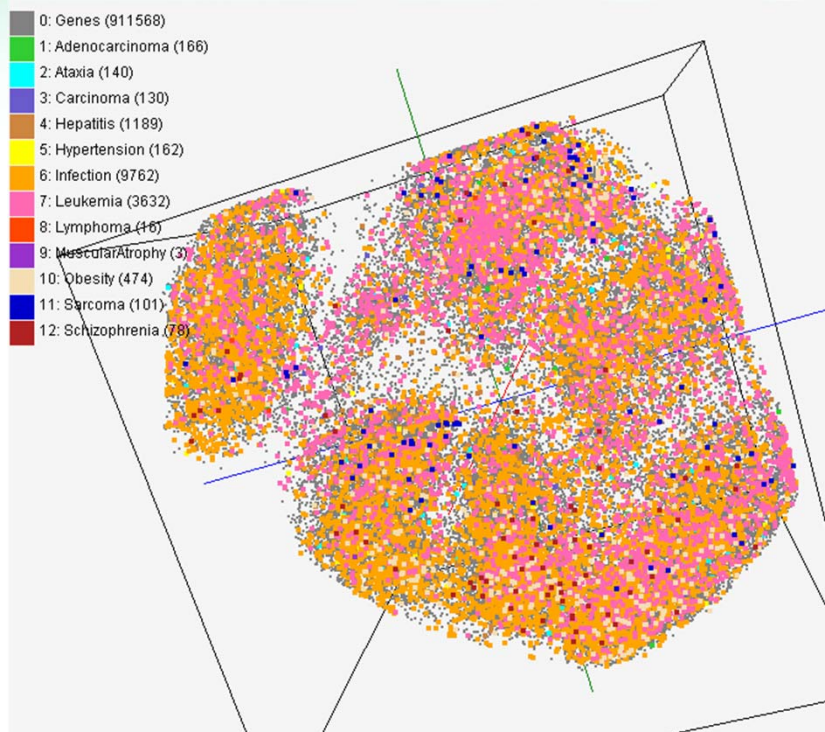
# DA-GTM vs. EM-GTM

	EM-GTM	DA-GTM
Optimization	Maximize log-likelihood $L$	Minimize free energy $F$
Objective Function	$\sum_{n=1}^N \ln \left\{ \frac{1}{K} \sum_{k=1}^K p(\mathbf{x}_n   \mathbf{y}_k) \right\}$	$-T \sum_{n=1}^N \ln \left\{ \left( \frac{1}{K} \right)^{\frac{1}{T}} \sum_{k=1}^K p(\mathbf{x}_n   \mathbf{y}_k)^{\frac{1}{T}} \right\}$
	When $T = 1$ , $L = -F$ .	
Pros & Cons	<ul style="list-style-type: none"> <li>▪ Very sensitive</li> <li>▪ Trapped in local optima</li> <li>▪ Faster</li> <li>▪ Large deviation</li> </ul>	<ul style="list-style-type: none"> <li>▪ Less sensitive to an initial condition</li> <li>▪ Find global optimum</li> <li>▪ Require more computational time</li> <li>▪ Smaller standard deviation</li> </ul>

# DA-GTM Result

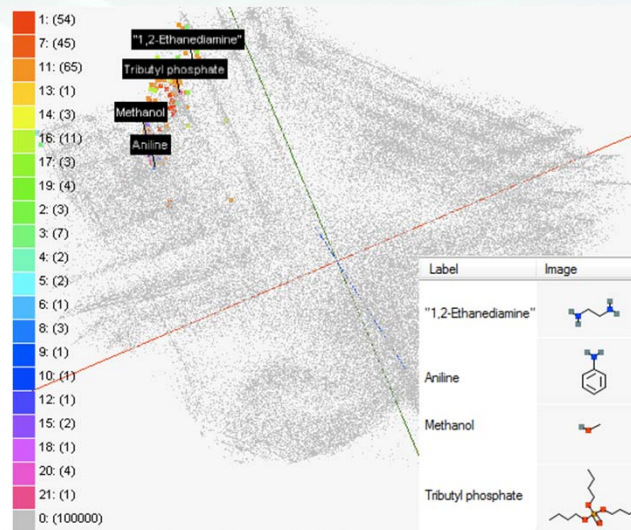


# Data Mining Projects using GTM



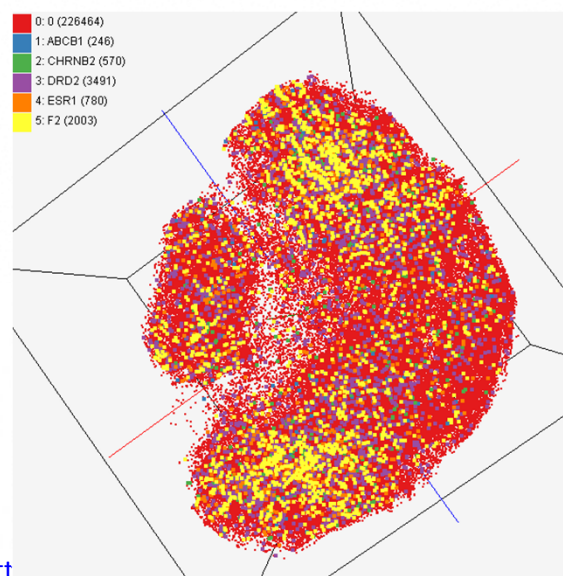
## **PubChem data with CTD visualization**

About 930,000 chemical compounds are visualized in a 3D space, annotated by the related genes in Comparative Toxicogenomics Database (CTD)



## **Visualizing 215 solvents by GTM-Interpolation**

215 solvents (colored and labeled) are embedded with 100,000 chemical compounds (colored in grey) in PubChem database

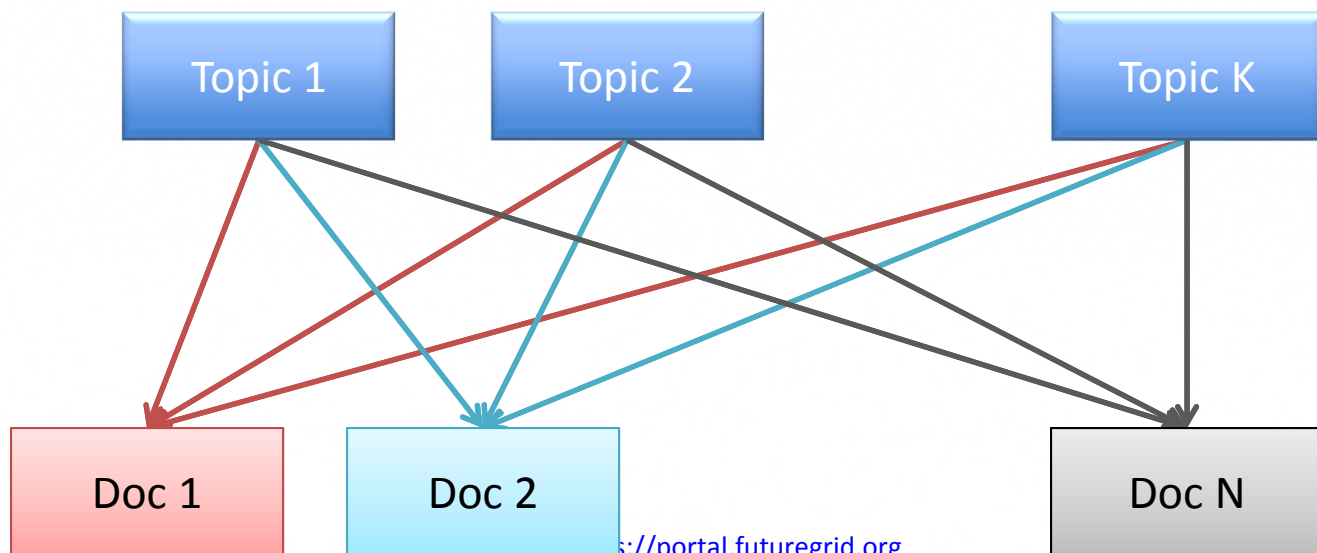


## **Chemical compounds reported in literatures**

Visualized 234,000 chemical compounds which may be related with a set of 5 genes of interest (ABCB1, CHRN2, DRD2, ESR1, and F2) based on the dataset collected from major journal literatures

# Probabilistic Latent Semantic Analysis (PLSA)

- Topic model (or latent model)
  - Assume generative K topics (document generator)
  - Each document is a mixture of K topics
  - The original proposal used EM for model fitting



<https://portal.futuregrid.org>



# DA-Mixture Models

- Mixture models take general form

$$H = - \sum_{x=1}^n \sum_{k=1}^K M_n(k) \ln L(n | k)$$

$$\sum_{k=1}^K M_n(k) = 1 \text{ for each } n$$

$n$  runs over things being decomposed (documents in this case)

$k$  runs over component things— Grid points for GTM, Gaussians for Gaussian mixtures, topics for PLSA

- Anneal on “spins”  $M_n(k)$  so  $H$  is linear and do not need another Hamiltonian as  $H = H_0$
- Note  $L(n | k)$  is function of “interesting” parameters and these are found as in non annealed case by a separate optimization in the M step



<https://portal.futuregrid.org>



# EM vs. DA-{GTM, PLSA}

		EM	DA
Optimization		Maximize log-likelihood $L$	Minimize free energy $F$
Objective Functions	GTM	$\sum_{n=1}^N \ln \left\{ \frac{1}{K} \sum_{k=1}^K p(\mathbf{x}_n   \mathbf{y}_k) \right\}$	$-T \sum_{n=1}^N \ln \left\{ \left( \frac{1}{K} \right)^{\frac{1}{T}} \sum_{k=1}^K p(\mathbf{x}_n   \mathbf{y}_k)^{\frac{1}{T}} \right\}$
	PLSA	$\sum_{n=1}^N \ln \sum_{k=1}^K \{ \psi_{nk} \text{Multi}(\mathbf{x}_n   \mathbf{y}_k) \}$	$-T \sum_{n=1}^N \ln \sum_{k=1}^K \{ \psi_{nk} \text{Multi}(\mathbf{x}_n   \mathbf{y}_k) \}^{\frac{1}{T}}$
		<p>Note: When <math>T = 1</math>, <math>L = -F</math>.</p> <p><i>This implies EM can be treated as a special case in DA</i></p>	
Pros & Cons		<ul style="list-style-type: none"> <li>Very sensitive</li> <li>Trapped in local optima</li> <li>Faster</li> <li>Large deviation</li> </ul>	<ul style="list-style-type: none"> <li>Less sensitive to an initial condition</li> <li>Find global optimum</li> <li>Require more computational time</li> <li>Small deviation</li> </ul>

# DA-PLSA Features

- DA is good at both of the following:
  - To improve model fitting quality compared to EM
  - To avoid over-fitting and hence increase predicting power (generalization)
    - Find better relaxed model than EM by stopping  $T > 1$
    - Note Tempered-EM, proposed by Hofmann (the original author of PLSA) is similar to DA but annealing is done in reversed way
- LDA uses prior distribution to get effects similar to annealed smoothing

# An example of DA-PLSA

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
percent	stock	soviet	bush	percent
million	market	gorbachev	dukakis	computer
year	index	party	percent	aids
sales	million	i	i	year
billion	percent	president	jackson	new
new	stocks	union	campaign	drug
company	trading	gorbachevs	poll	virus
last	shares	government	president	futures
corp	new	new	new	people
share	exchange	news	israel	two

Top 10 popular words of the AP news dataset for 30 topics.  
Processed by DA-PLSI and showing only 5 topics among 30 topics

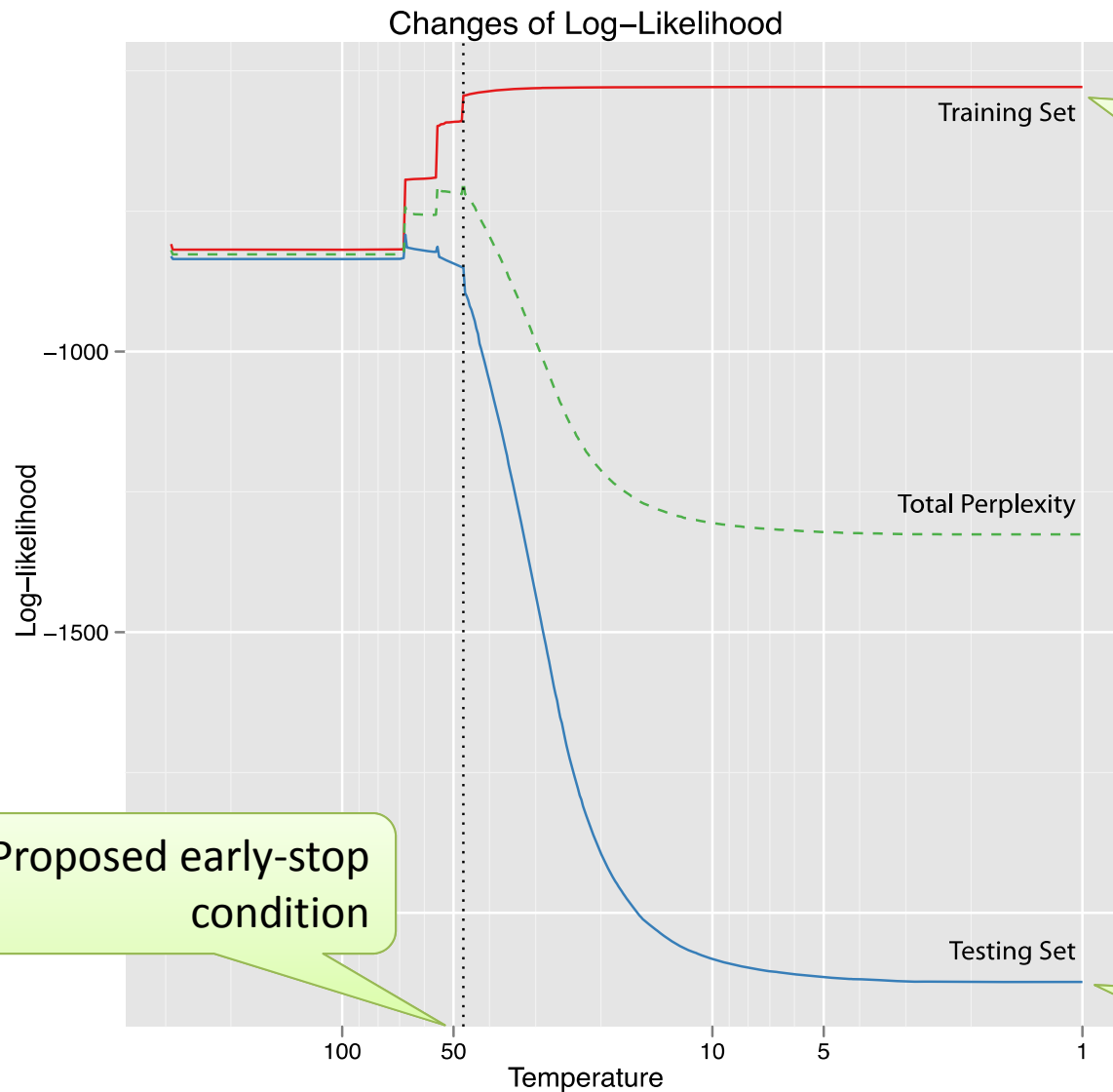


<https://portal.futuregrid.org>





# Annealing in DA-PLSA



Improved fitting quality of training set during annealing

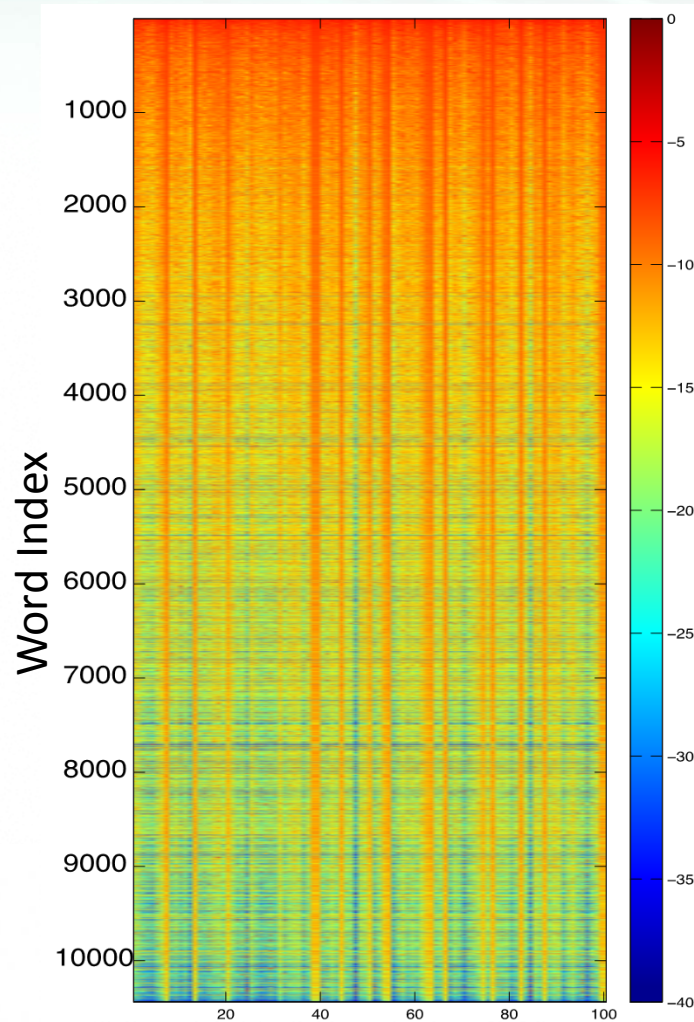
Annealing progresses from *high* temp to *low* temp

Proposed early-stop condition

Over-fitting at Temp=1

# Predicting Power in DA-PLSA

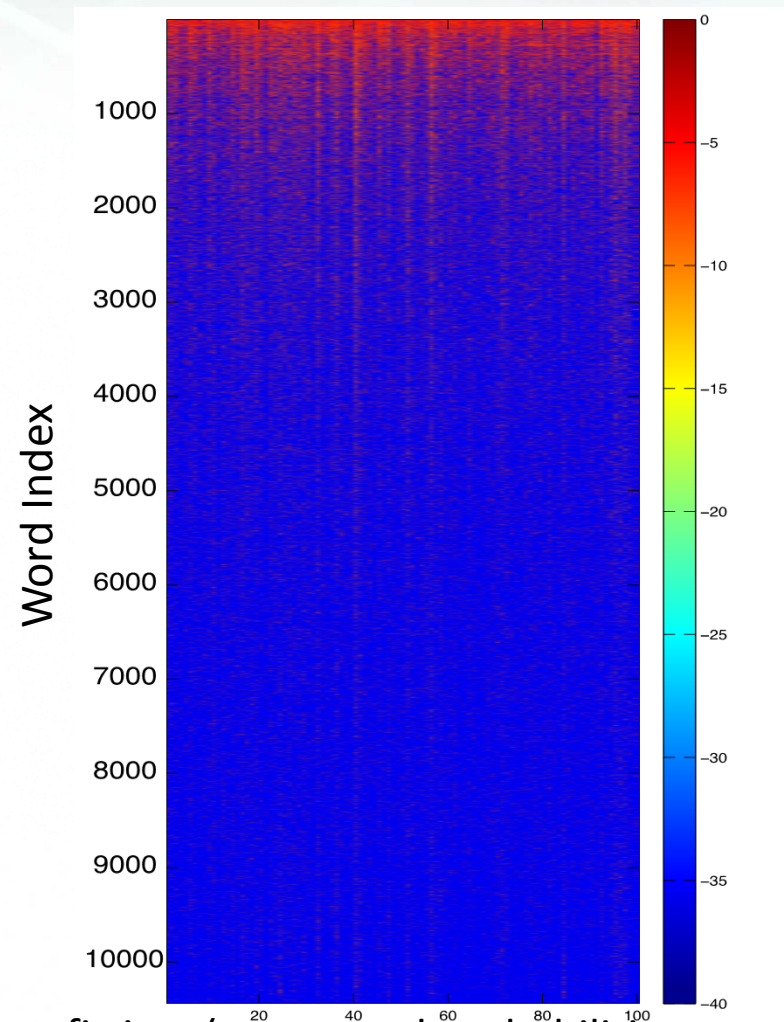
AP Word Probabilities (100 topics for 10473 words)



optimized stop  
(Temp = 49.98)



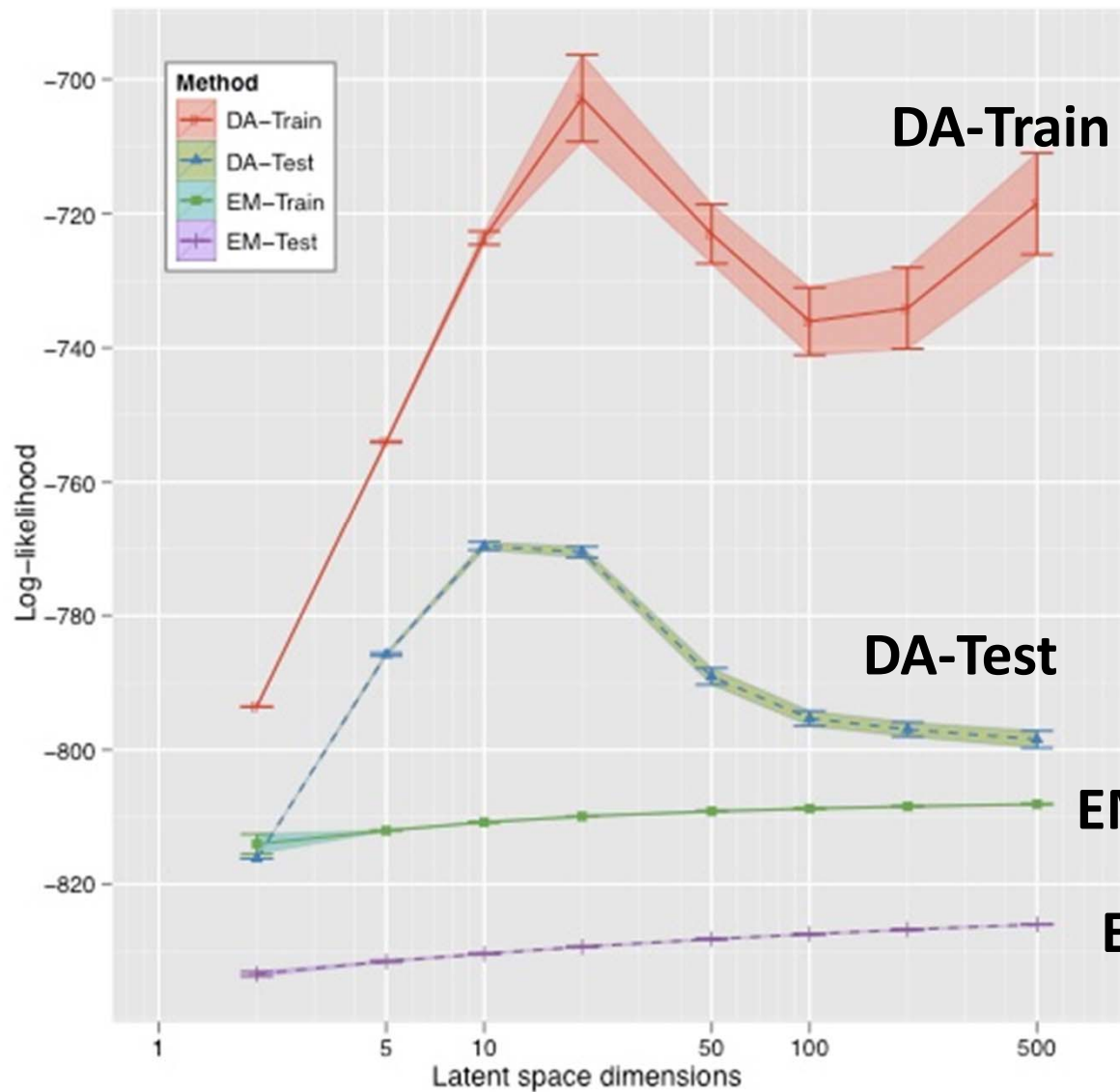
<https://portal.futuregrid.org>



Over-fitting (most word probabilities are zero)  
at  $T = 1$

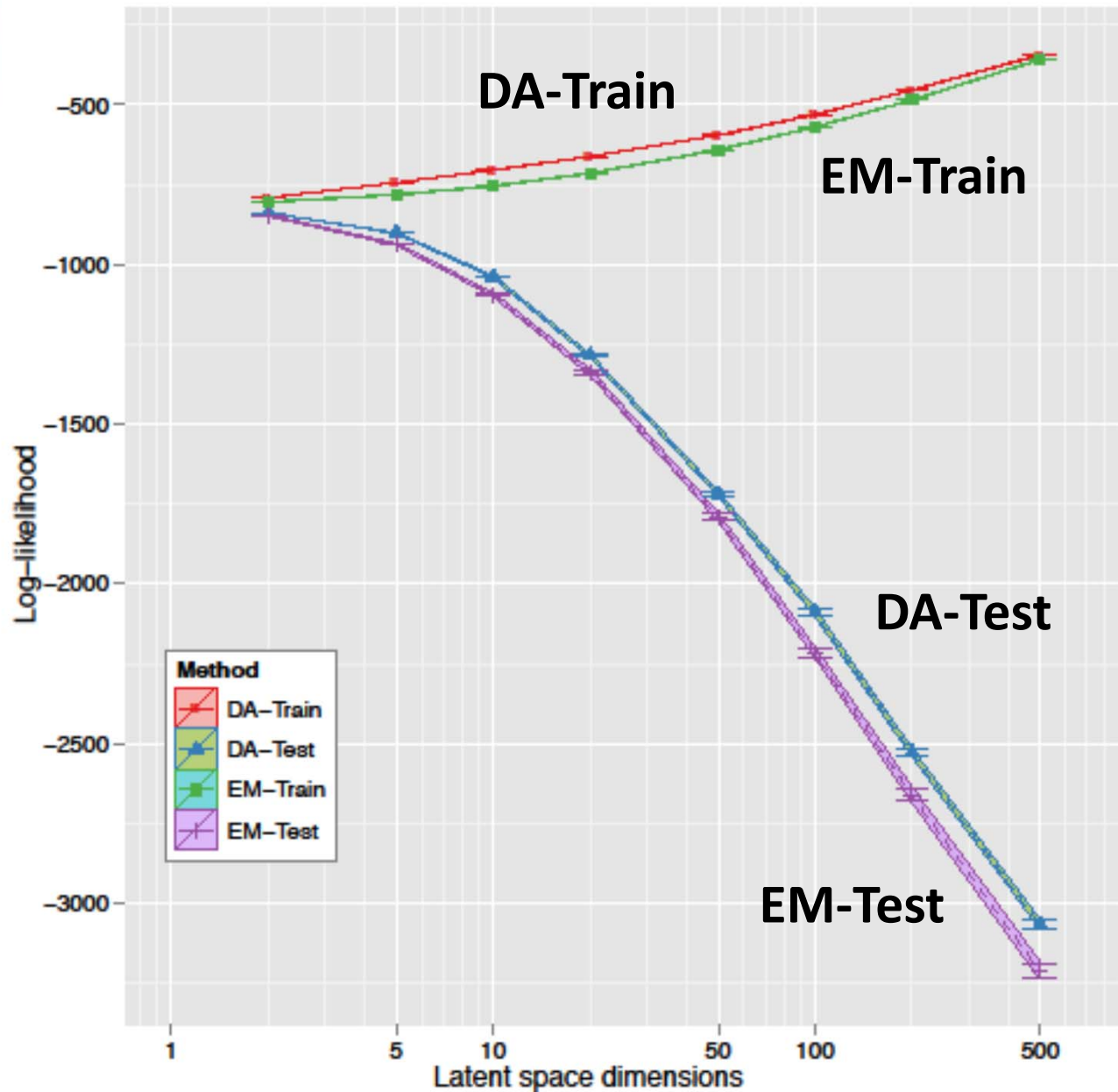


# Training & Testing in DA-PLSA I



- Here terminate on maximum of testing set
- DA outperform EM
- Improvements in training set matched by improvement in testing results

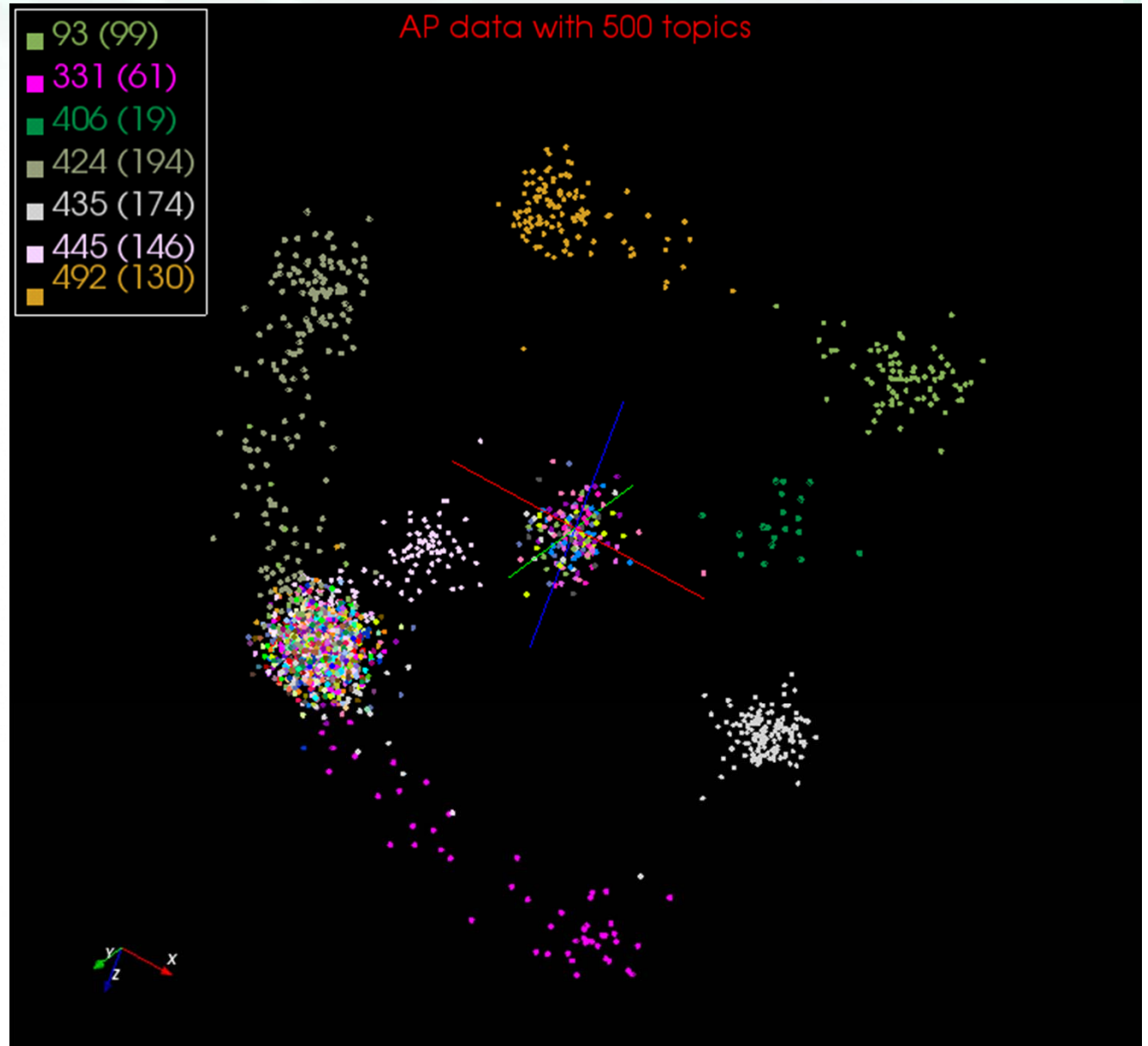
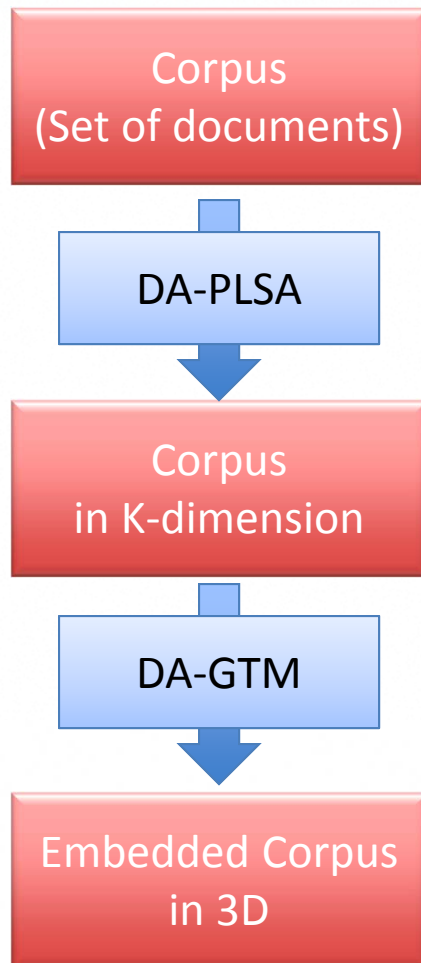
# Training & Testing in DA-PLSA II



- Here terminate on maximum of training set
- Improvements in training set NOT matched by improvement in testing results



# DA-PLSA with DA-GTM



# AP Data Top Topic Words

- In the previous picture, we found among 500 topics:

Topic 331	Topic 435	Topic 424	Topic 492	Topic 445	Topic 406
lately oferrell mandate ACK fcc cardboard commuter exam kuwaits fabrics	lately oferrell ACK fcc mandate cardboard exam commuter fabrics corroon	mandate kuwaits cardboard commuter ACK fcc lately exam fabrics oferrell	mandate kuwaits cardboard commuter lately ACK exam fcc oferrell fabrics	mandate lately ACK cardboard fcc commuter oferrell exam kuwaits fabrics	plunging referred informal Anticommu. origin details relieve psychologist lately thatcher

ACK : acknowledges

Anticommu. : anticommunist

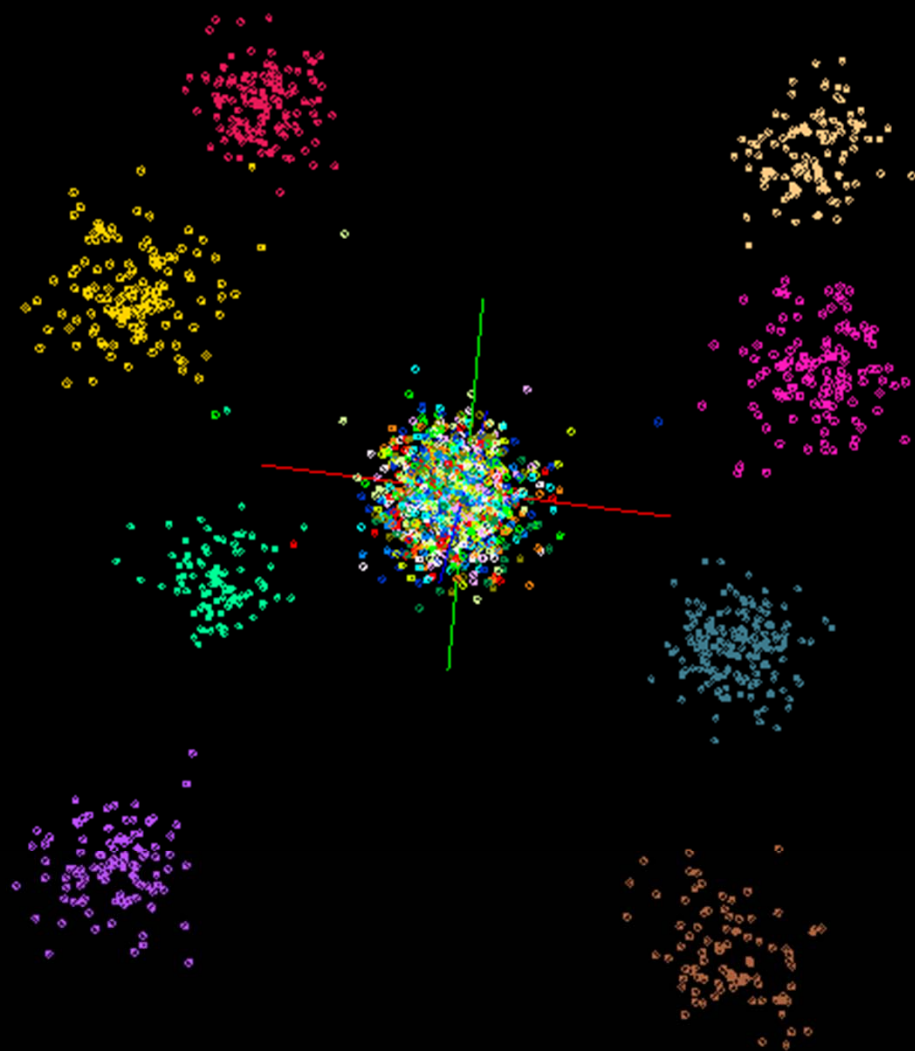


<https://portal.futuregrid.org>



# AP data with 20 topics

- 3 (123)
- 4 (135)
- 7 (89)
- 9 (173)
- 12 (105)
- 13 (132)
- 15 (92)
- 20 (115)



# AP Data Top Topic Words

- With 20 topics

#3	#4	#7	#9	#12	#13	#15	#20
marriage kuwaits algerias commuter exam cardboard accuse exceed	mandate kuwaits cardboard commuter fabrics minnick glow theyd	mandate resolve fabrics kuwaits cardboard fcc commuter oferrell	lately informal PSY referred oferrell ACK Anitcomm clearly	lately overdue ACK fcc oferrell corroon resolve van	mandate fcc fabrics ACK campbell cardboard solis sikhs	mandate commuter kuwaits cardboard fcc turbulence fabrics exam	oferrell van fcc attorneys Anticomm lately formation ACK

ACK : acknowledges

Anticomm : anticomunist

PSY : psychologist



<https://portal.futuregrid.org>





# What was/can be done where?

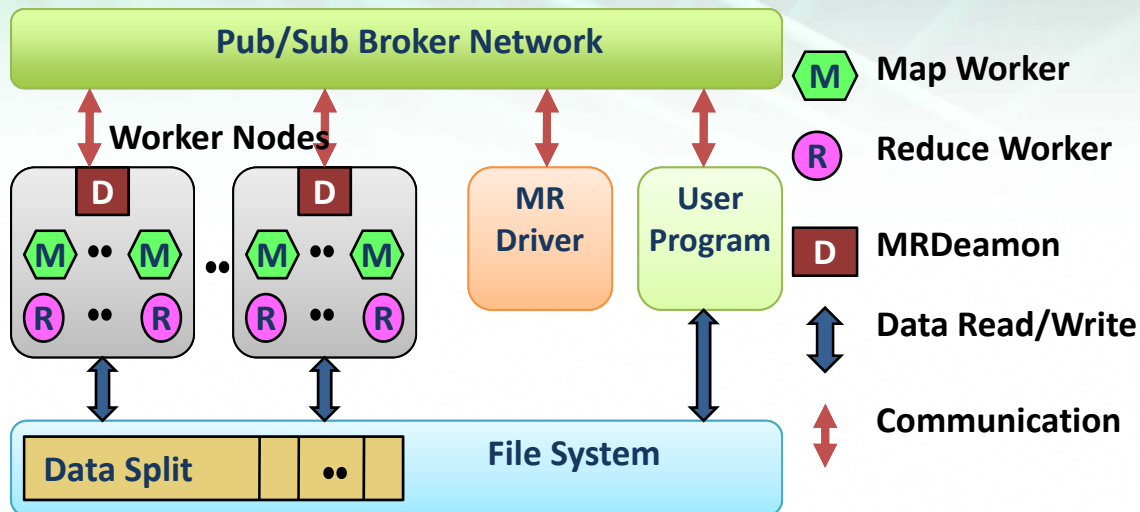
- **Dissimilarity Computation** (largest time)
  - Done using Twister on HPC
  - Have running on Azure and Dryad
  - Used Tempest (24 cores per node, 32 nodes) with MPI as well (MPI.NET failed(!), Twister didn't)
- **Full MDS**
  - Done using MPI on Tempest
  - Have running well using Twister on HPC clusters and Azure
- **Pairwise Clustering**
  - Done using MPI on Tempest
  - Probably need to change algorithm to get good efficiency on cloud but HPC parallel efficiency high
- **Interpolation** (smallest time)
  - Done using Twister on HPC
  - Running on Azure



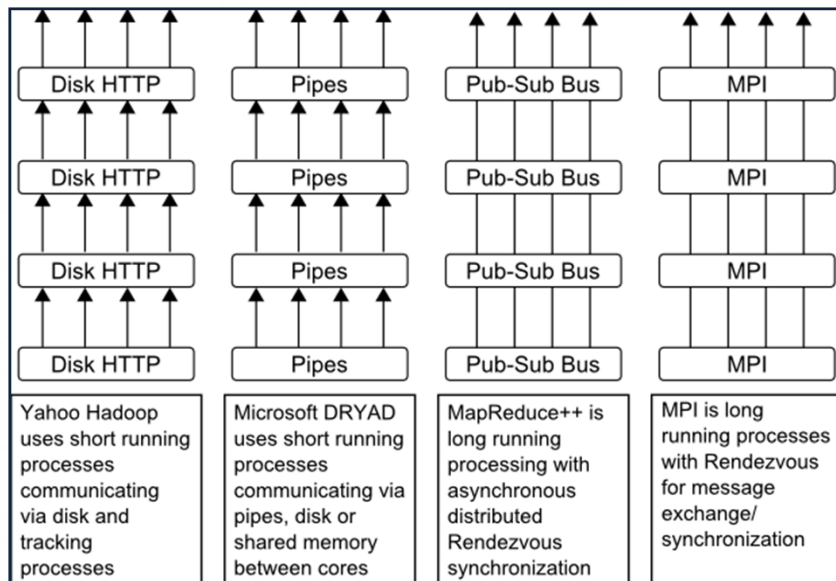
<https://portal.futuregrid.org>



# Twister

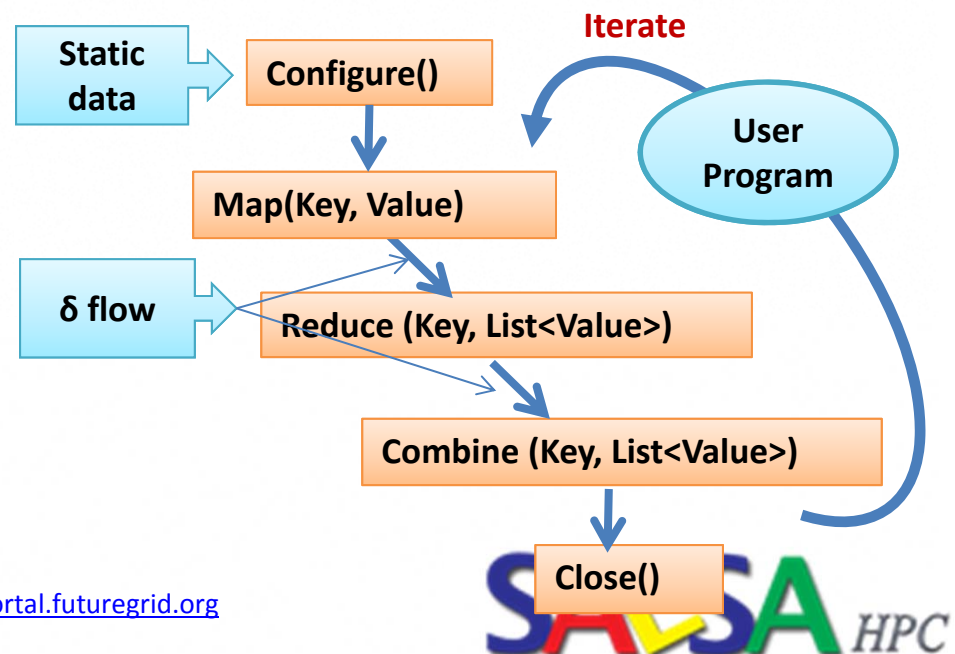


- **Streaming based** communication
- Intermediate results are directly transferred from the map tasks to the reduce tasks – **eliminates local files**
- **Cacheable** map/reduce tasks
  - Static data remains in memory
- **Combine** phase to combine reductions
- User Program is the **composer** of MapReduce computations
- **Extends the MapReduce model to iterative computations**



Different synchronization and intercommunication mechanisms used by the parallel runtimes

<https://portal.futuregrid.org>



SAHSA HPC

# Expectation Maximization and Iterative MapReduce

- **Clustering** and **Multidimensional Scaling** are both EM (**expectation maximization**) using deterministic annealing for improved performance
- **EM** tends to be **good** for **clouds** and **Iterative MapReduce**
  - Quite **complicated computations** (so compute largish compared to communicate)
  - Communication is **Reduction** operations (global sums in our case)
  - See also **Latent Dirichlet Allocation** and related Information Retrieval algorithms similar EM structure

# May Need New Algorithms

- **DA-PWC** (Deterministically Annealed Pairwise Clustering) splits clusters automatically as temperature lowers and reveals clusters of size  $O(\sqrt{T})$
- Two approaches to splitting
  1. Look at correlation matrix and see when becomes singular which is a separate parallel step
  2. Formulate problem with multiple centers for each cluster and perturb ever so often spitting centers into 2 groups; unstable clusters separate
- Current MPI code uses first method which will run on Twister as matrix singularity analysis is the usual “power eigenvalue method” (as is page rank)
  - However not super good compute/communicate ratio
- Experiment with second method which “just” EM with better compute/communicate ratio (simpler code as well)



# Next Steps

- Finalize MPI and Twister versions of Deterministically Annealed Expectation Maximization for
  - Vector Clustering
  - Vector Clustering with trimmed clusters
  - Pairwise non vector Clustering
  - MDS SMACOF
- Extend  $O(N \log N)$  Barnes Hut methods to all codes
- Allow missing distances in MDS (Blast gives this) and allow arbitrary weightings (Sammon's method)
  - Have done for  $\chi^2$  approach to MDS
- Explore DA-PLSA as alternative to LDA
- Exploit better Twister and Twister4Azure runtimes



<https://portal.futuregrid.org>

