Pattern Decomposition of Inorganic Materials: Optimizing Computational Algorithm

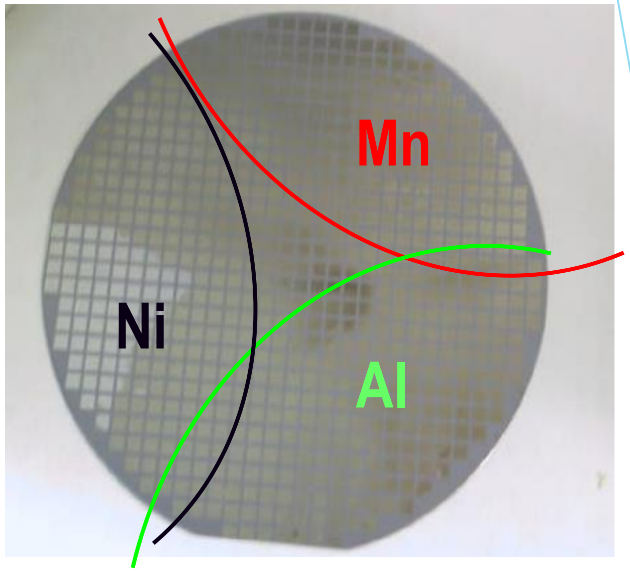
Graham Antoszewski
ganto@math.umd.edu

Advisor: Dr. Hector Corrada-Bravo
Center for Bioinformatics and Computational Biology
University of Maryland, Department of Computer Science
hcorrada@umiacs.umd.edu

December 12, 2016

Background Information – Materials Sciences

- Mixtures of metal alloys – ternary systems
  - Composition of metals varies through material
  - Different composition = unique crystalline structure
  - Different chemical properties

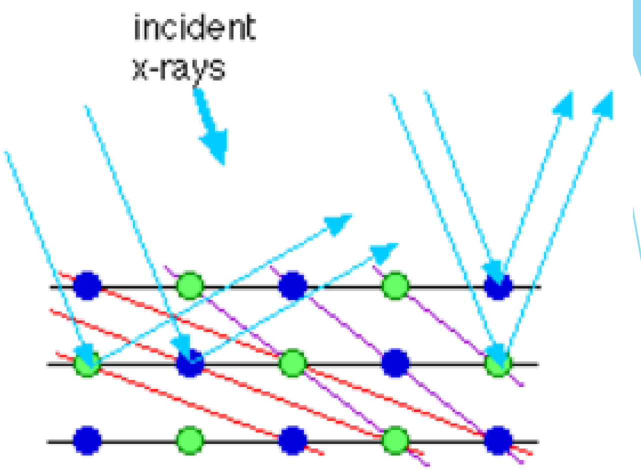


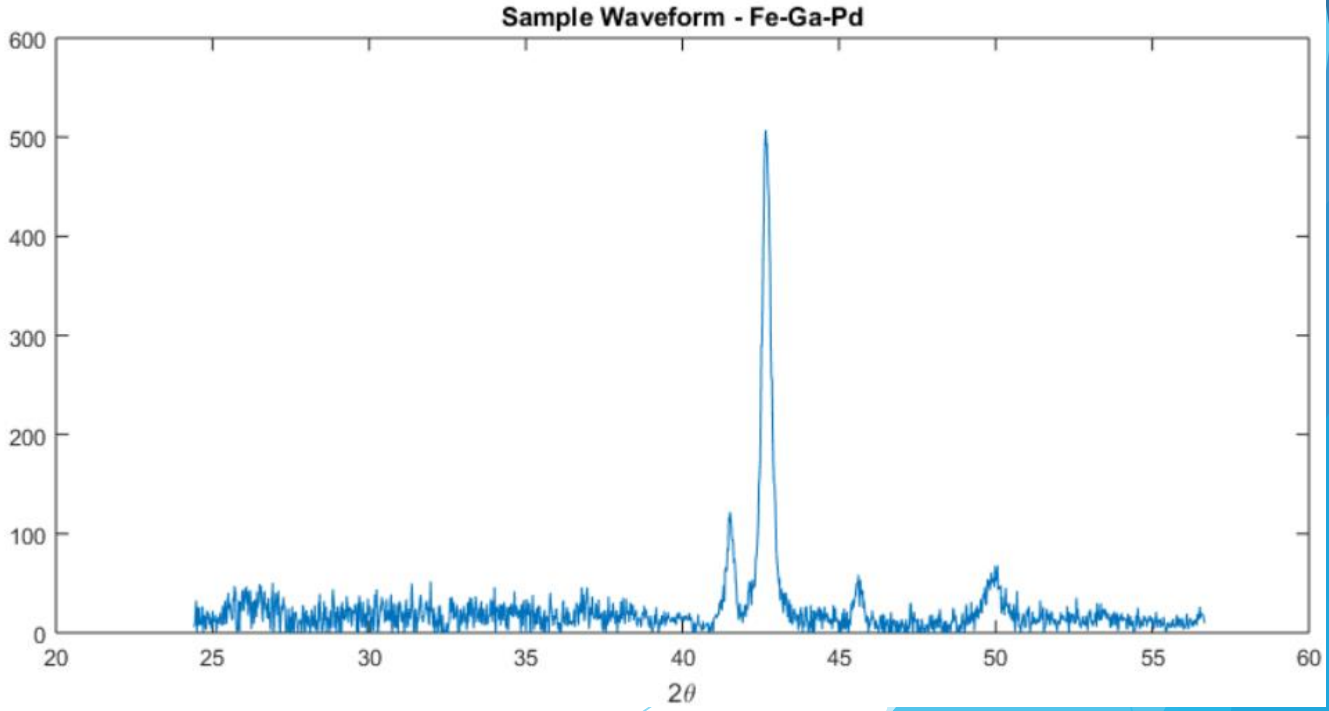
- Pattern Decomposition
  - Given a system of N sample points of numeric data (Ex: light intensity)
  - Want to find K basis “phase patterns” that describe data at all points
  - Like finding basis of a vector space
  - Phases tell us about the chemical properties of the material

Takeuchi I. (2016) MRS Meeting

Background Information – Pattern Decomposition

- Given material is sampled using electron probe
  - X-ray light is diffracted back at a certain angle
  - Based on lattice spacing
- Output is a continuous waveform
  - X - Scattering angle
  - Y - Intensity of diffracted light
- Determine composition via waveform
  - Like human fingerprint
  - Combination of basis waveforms

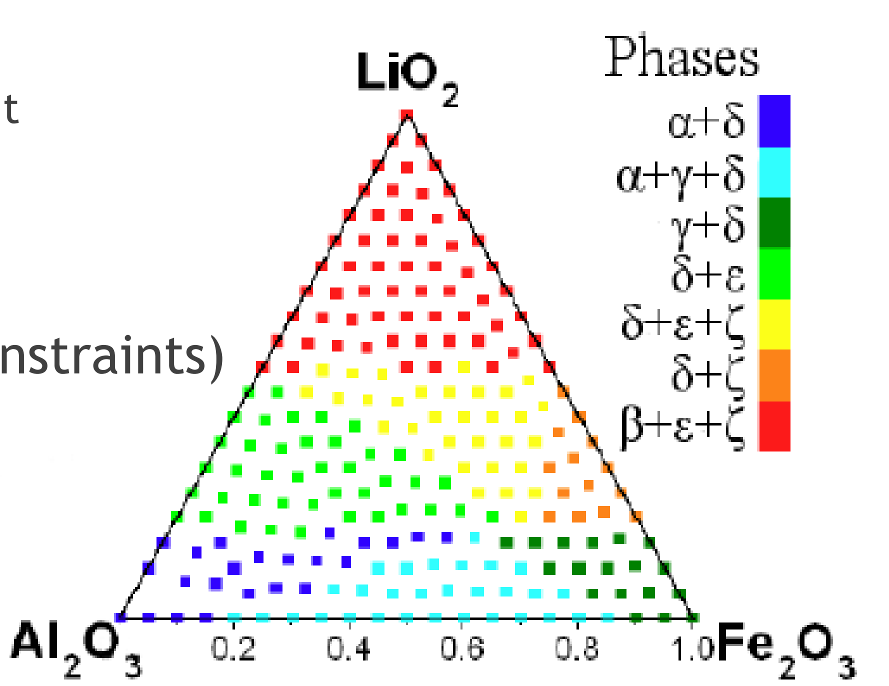




Top figure: http://physics.bu.edu/py106/notes/Resolution.html

Background Information – Phase diagrams

- After probing all sample points of a material, a simplex can be created
  - Illustration of phase composition at a given point
  - Colors = clusters (similar phase structure)
- Results must uphold to laws of physics (constraints)
  - Gibbs phase rule
  - Connectivity (continuity of phases in space)
  - Peak Shifting (effect of alloying process)



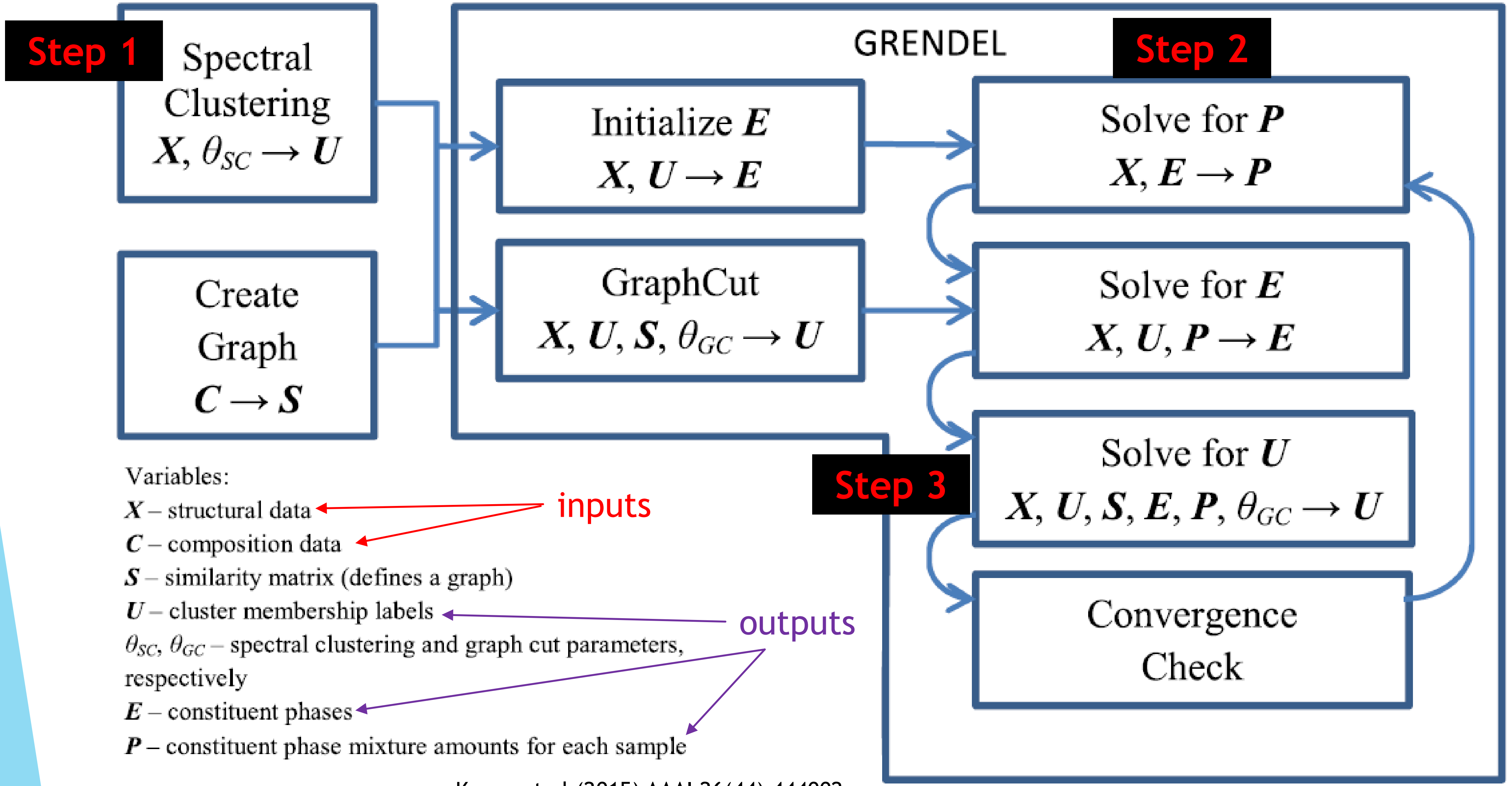
LeBras et al (2011) AAAI CP'11 508-522

Project Goal Part 1

- White House Materials Genome Initiative
  - Develop algorithm to take in diffraction/composition data, output phase structure of materials

- Algorithm must:
  - Obey physical constraints (laws of physics)
  - Identify regions/clusters of similar phase composition within material
  - Identify basis phases accurately (≤3 per cluster)
  - Be efficient – short run times so more materials can be analyzed

GRENDEL Algorithm



Step 1

Spectral Clustering
$X, \theta_{SC} \rightarrow U$

Create Graph
$C \rightarrow S$

GRENDEL

Initialize E
$X, U \rightarrow E$

GraphCut
$X, U, S, \theta_{GC} \rightarrow U$

Step 2

Solve for P
$X, E \rightarrow P$

Solve for E
$X, U, P \rightarrow E$

Step 3

Solve for U
$X, U, S, E, P, \theta_{GC} \rightarrow U$

Convergence Check

Variables:

X – structural data

C – composition data

S – similarity matrix (defines a graph)

U – cluster membership labels

θ_{SC}, θ_{GC} – spectral clustering and graph cut parameters, respectively

E – constituent phases

P – constituent phase mixture amounts for each sample

inputs

outputs

Kusne et al (2015) AAAI 26(44) 444002

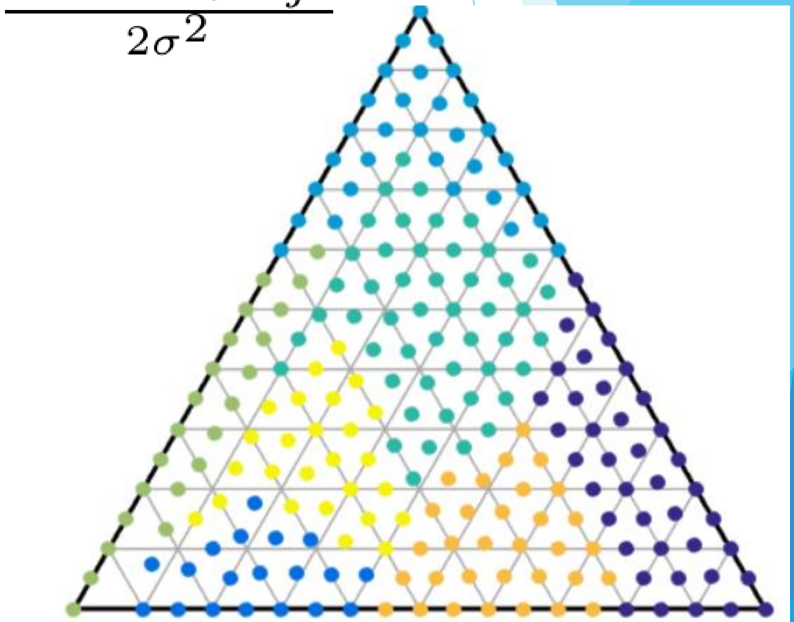
Project Goal Part 2 – Extending GRENDEL

- Take existing GRENDEL code, apply strategies to make the algorithm better
- Increase accuracy of clustering and basis phase detection results by incorporating constraints
  - Laws of physics
  - “Expert” prior knowledge of material
  - Affects cluster analysis and overall phase composition
- Decrease time needed to probe given material in the lab
  - Minimize data points needed to resolve constituent phases

Algorithm – GRENDEL
Step 1 – Spectral Clustering

$$W_{ij} = e^{\frac{-\delta_{\cos}(X_i, X_j)}{2\sigma^2}}$$

- Input diffraction data – **X**, NxD matrix
  - N = # of data points
  - D = # of scattering angles sampled (length of waveform)
- Takes in diffraction data, creates a similarity matrix **W**
  - i,j – sample points
  - $\delta_{cos}(X_i, X_j)$ – cosine distance (1 – cosine of waveform vectors)
  - σ – spectral clustering bandwidth parameter (θ_{sc})
- Spectral Clustering Algorithm:
  - **G** = diagonal matrix summing rows of **W**
  - Find *k* smallest nontrivial eigenvectors of Graph Laplacian, **L = G^{-1}W**
  - use MATLAB k-means function to group points into clusters
  - **U** (kxN) – cluster membership matrix, **U**(c,i) = 1 if point i is in cluster c



Algorithm – GRENDEL
Step 2 – Nonnegative Matrix Factorization

- The goal of GRENDEL is to minimize the ‘Objective Function’ each iteration:

$$J(E,P,U)=\sum_{i=1}^{K}\left(\sum_{j=1}^{N}u_{ij}(X_j-p_{ij}E_i)^T(X_j-p_{ij}E_i)+\alpha\sum_{h=1}^{M-1}\sum_{l=h+1}^{M}(e_{ih}-e_{il})^T(e_{ih}-e_{il})\right)$$

- Assume **X** can be approximated by **P*E**
- $\mathbf{E}_i$ (MxD) – basis phases of ith cluster (unknown), e_{ij} is jth row of $\mathbf{E}_i$
- $\mathbf{p}_{ij}$ (1xM) – phase proportions of ith cluster for jth sample point (unknown)
- **U** (KxN) – cluster membership
- Peak shifting physical constraints added here

Algorithm – GRENDEL
Step 2 – NMF updates

- Minimize Objective Function to update **E, P** matrices
- Set derivatives with respect to **E,P** equal to zero to obtain:

$$E_k = \left(\sum_i u_{ki} p_{ki}^T p_{ki} + 2\alpha (M I_{M\times M} - 1_{M\times M}) \right)^{-1} \left(\sum_i u_{ki} p_{ki}^T X_i \right)$$

$$p_{ki} = \max\left((E_k E_k^T)^{-1} (E_k X_i^T - \lambda_k 1_{M\times 1}), 0 \right)$$

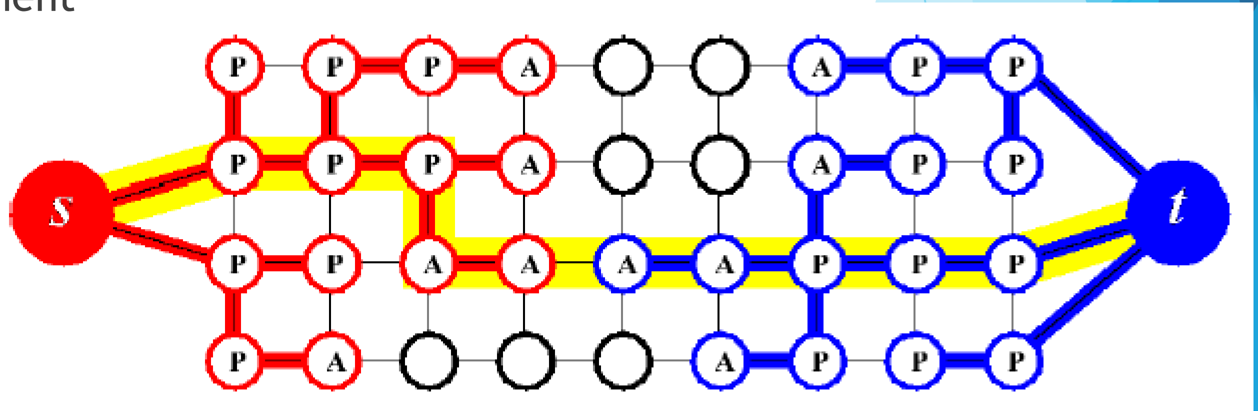
$$\lambda_k = \frac{1_{1\times M} (E_k E_k^T)^{-1} E_k X_i^T - 1}{1_{1\times M} (E_k E_k^T)^{-1} 1_{M\times 1}}$$

Algorithm – GRENDEL
Step 3 – Graph Cut

- General “cost” equation to minimize: $$V = \lambda_d \sum_i \overset{(1)}{V^i(L_i)} + \lambda_s \sum_{i,j \in N} \overset{(2)}{V^{i,j}(L_i, L_j)}$$
- Smoothness cost (2) is 0 if cluster labels match, 1 otherwise, Data cost matrix (1):

$$V^j(L_j = i) = \frac{3}{4}\delta_{\cos}(X_j, \bar{X}_i) + \frac{1}{4}\frac{||X_j - p_{ij}E_i||_2}{\sum_i ||X_j - p_{ij}E_i||_2}$$

- Minimize **V** through Max Flow Algorithm (see figure)
  - Uses residuals of **V** to find best cluster assignment for whole connected material
  - One cluster at a time
  - Iterates over all cluster assignments to find minimized configuration, update **U**
- Connectivity constraints added here



Boykov et al (2004) PAMI 26(9) 1124-1137

Implementation

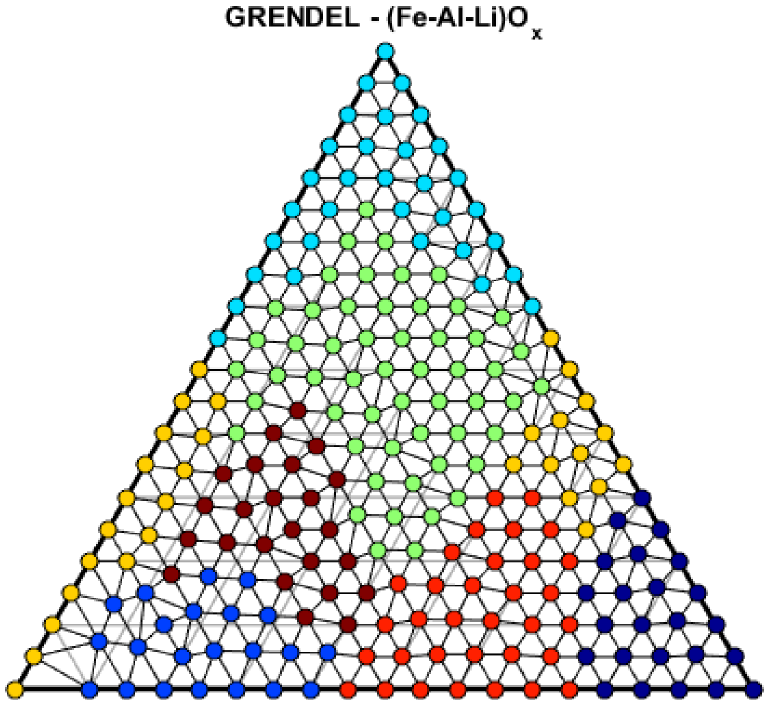
- Language - MATLAB R2015a
  - Graph Cut written in C++
- Hardware - personal computer
  - ASUS, 8 GB RAM
- Data sets – Inorganic Crystal Structure Database (Fe-Ga-Pd)
  - Synthetic spectral and structural data from previous research efforts ((Fe-Al-Li)O_x)
  - X – input spectral waveform data (NxD, N ≈ 200, D≈500-2000)
  - C – input composition data (spatial coordinates on ternary diagram)

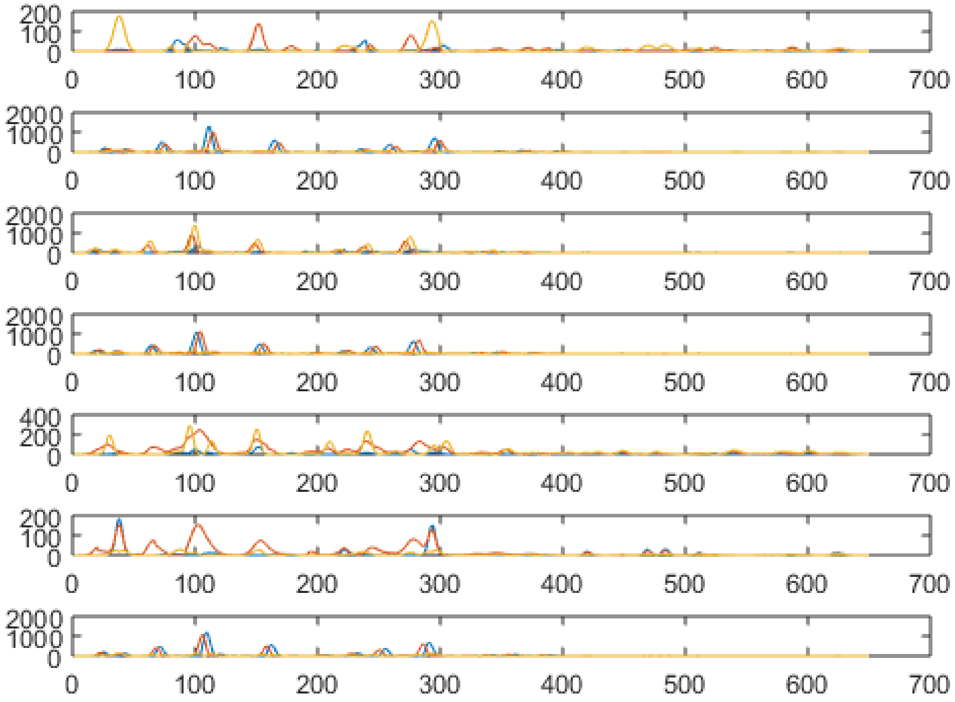
Unexpected Events

- Bugs in GRENDEL code
- Learning C++
- Synthetic data sets not originally compatible with our code
  - Peak shifting
  - Expected phases/clusters are extremely similar
- Gibb’s phase rule already enforced in GRENDEL

Results – Original GRENDEL

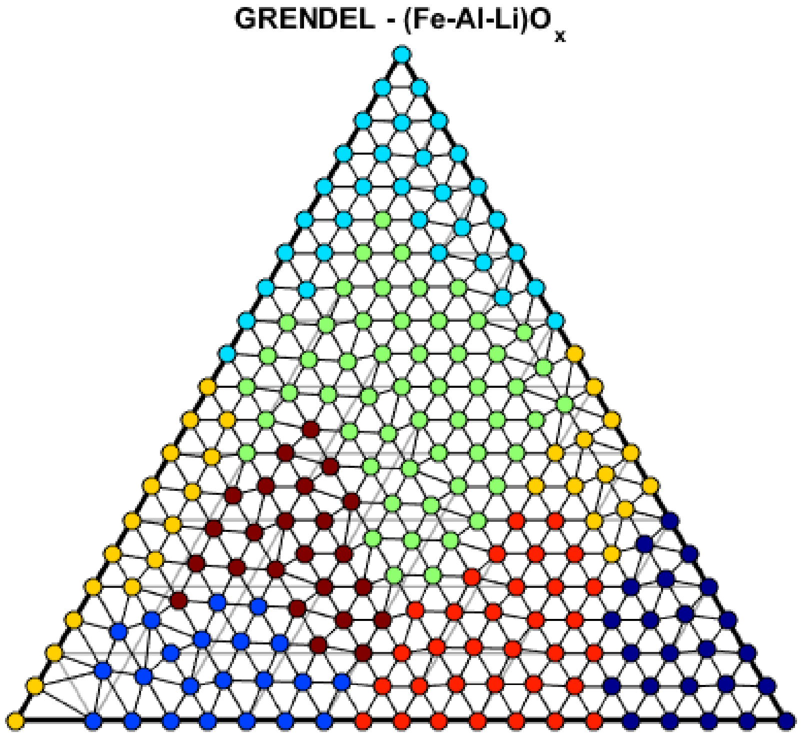
- Plot to the left is ternary diagram (showing the 7 different clusters/colors)
- Plot to the right are the spectral (waveform) plots of the constituent phases for each cluster

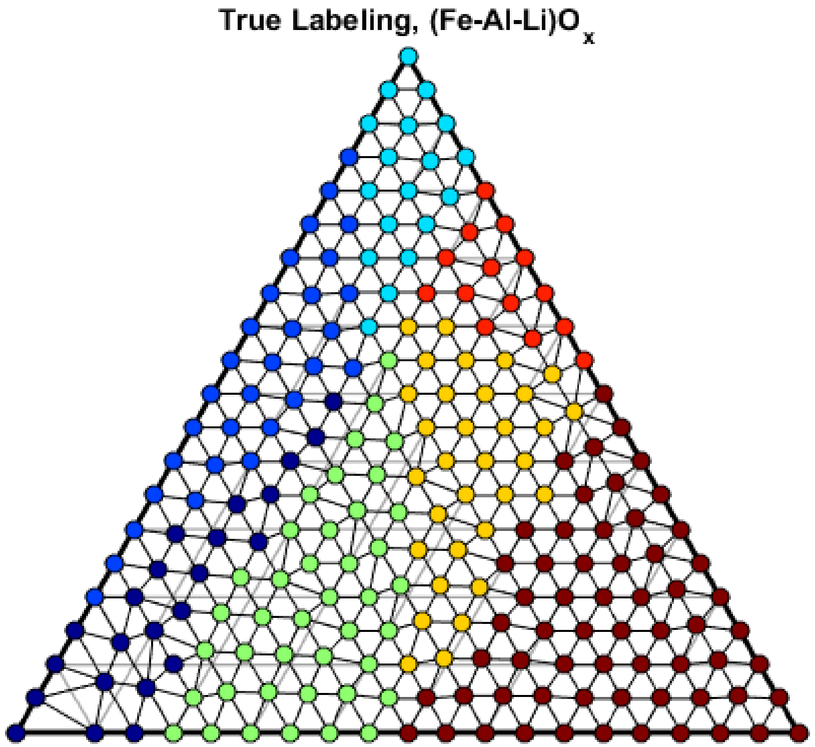




Comparison to True Values

- We know the clustering and basis phases for this synthetic data set
- Colors/clusters in the diagrams don’t completely correspond



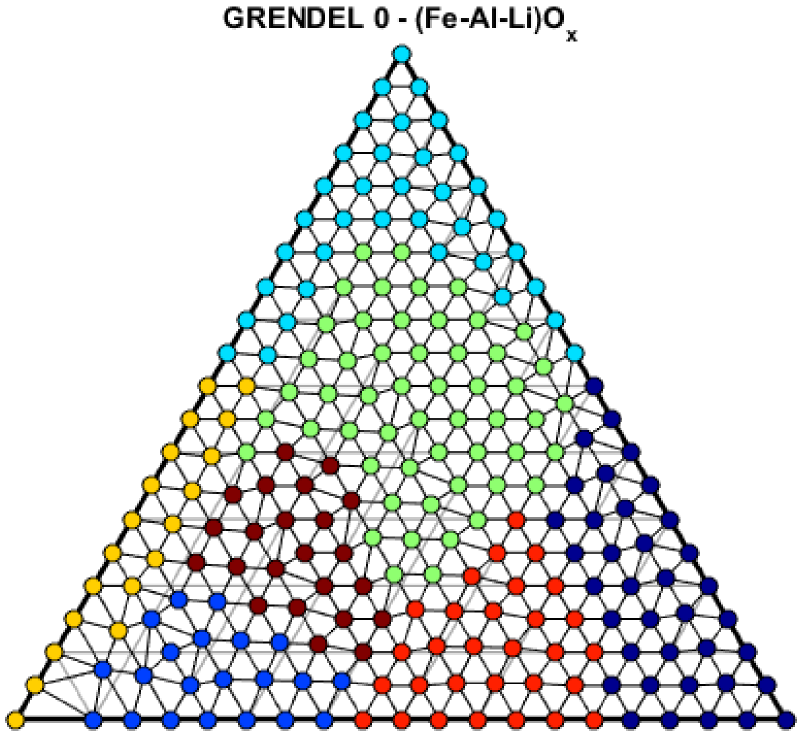


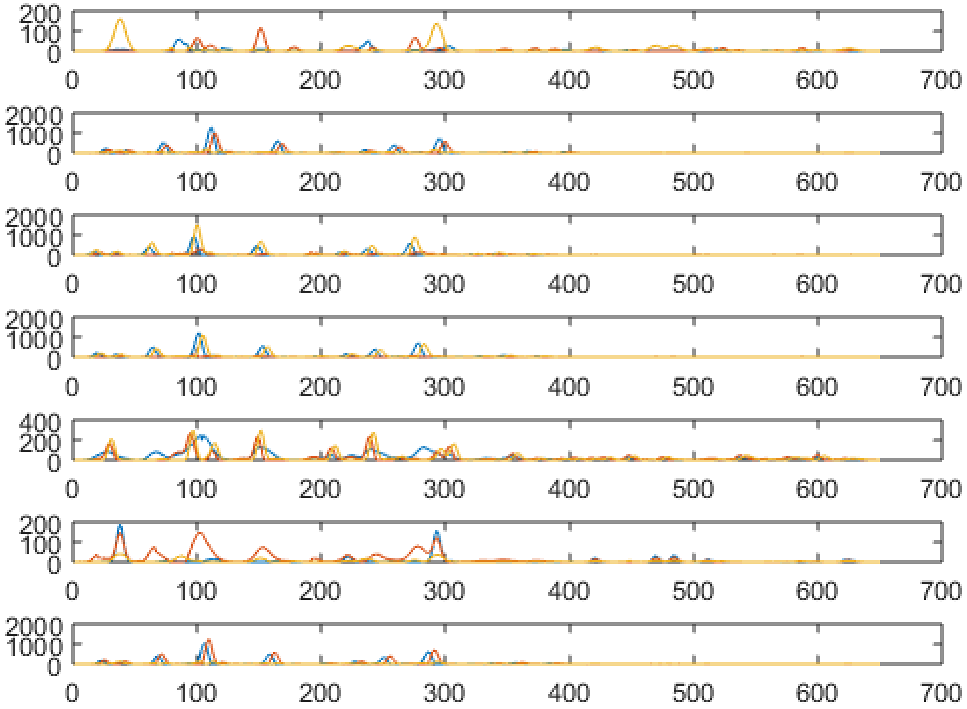
Results – Algorithm 0
‘Cannot Link’ Expert Constraints

- Expert knowledge – certain pairs of sample points in material ‘Cannot Link’
- Algorithm:

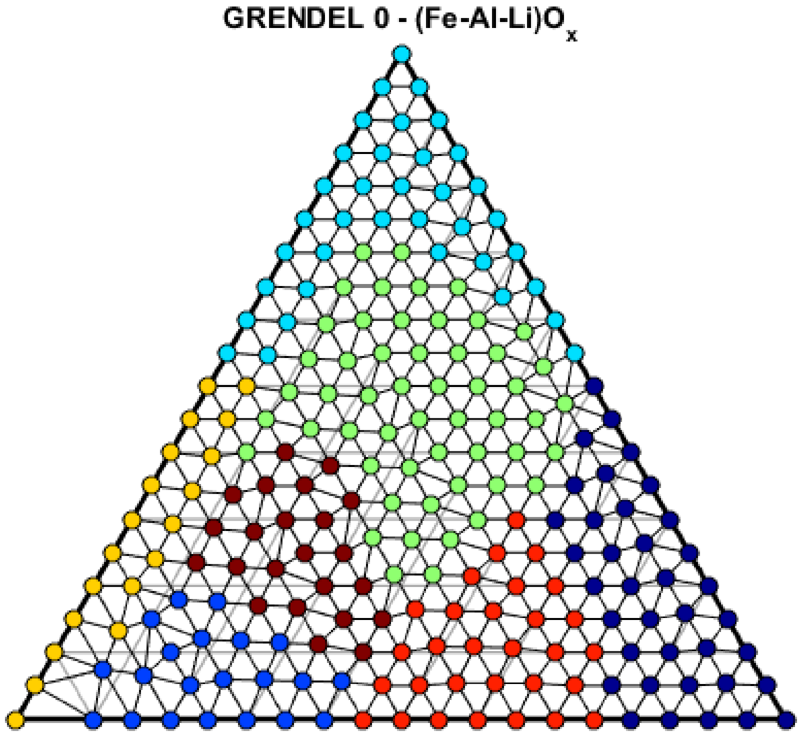
```
Compute cosine distance between all pairs
Assign top p% dissimilar pairs to ‘Cannot Link’ array
After Graph Cut:
    Loop through all CL pairs
    If pair in same cluster
        If 1st point changed cluster
            Revert cluster assignment of 1st point to old cluster
        Else
            Revert 2nd point’s cluster assignment
        end
    end
```

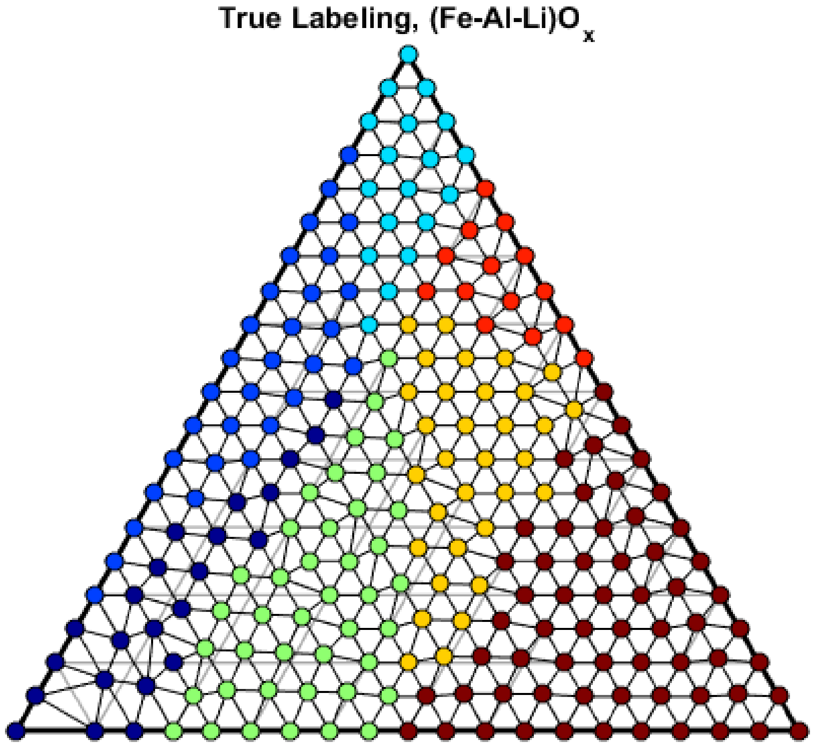
Results – Algorithm 0
‘Cannot Link’ Expert Constraints





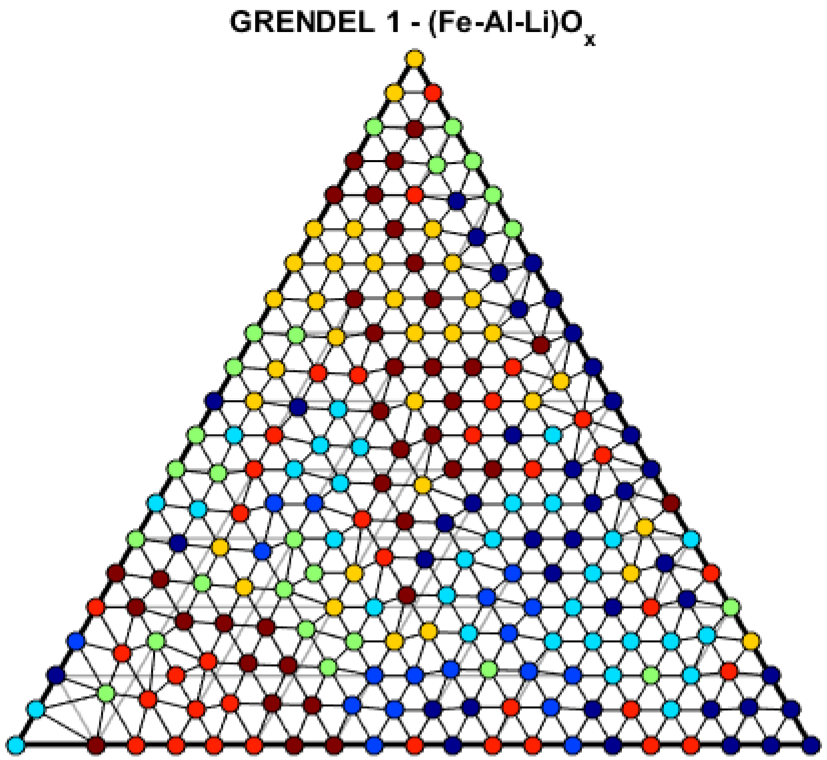
Results – Algorithm 0
‘Cannot Link’ Expert Constraints





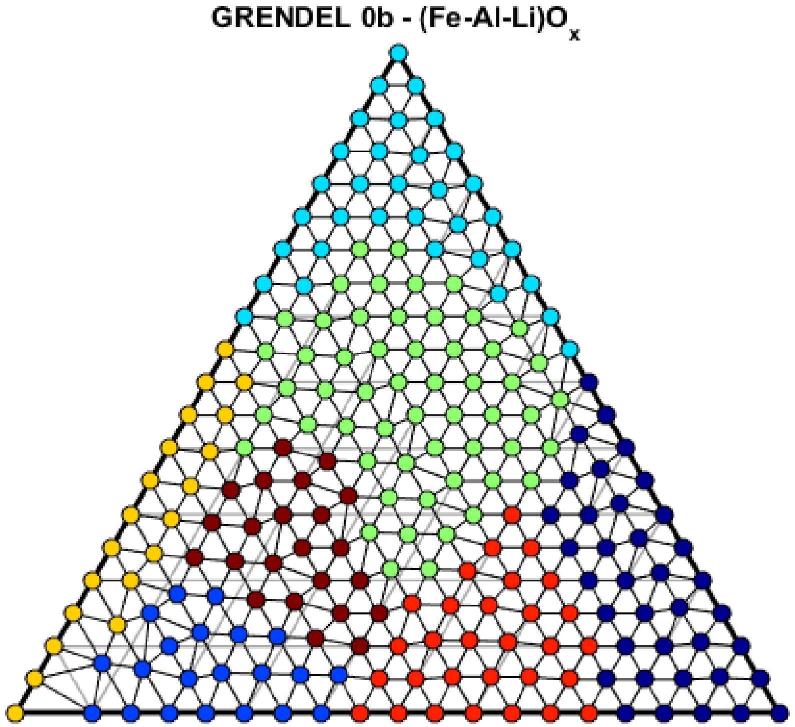
Results – Algorithm 1
‘Cannot Link’ Approach Using Cost Matrix

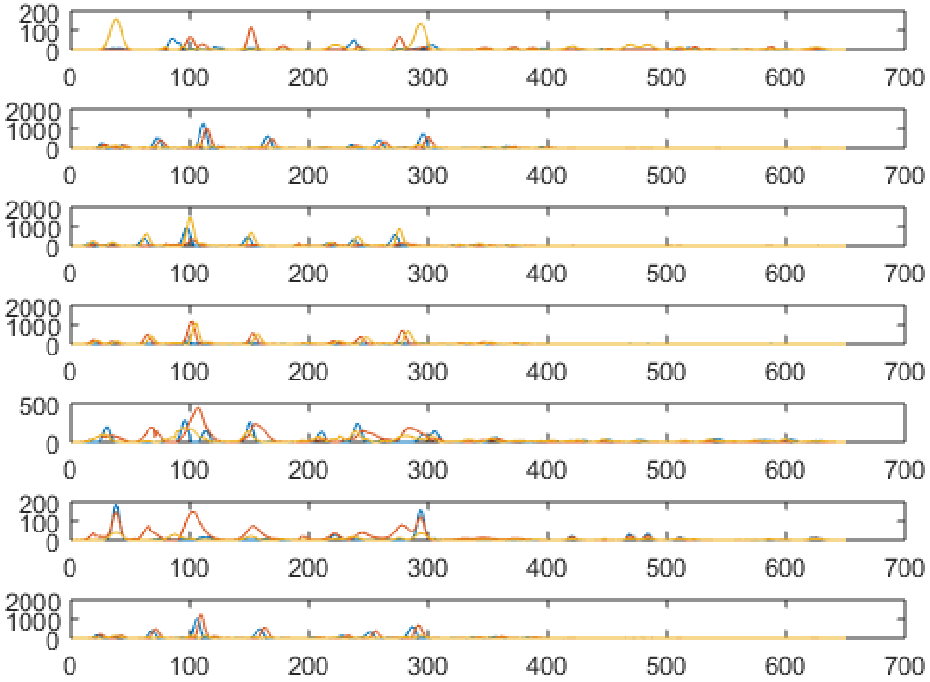
- Tried to use Cost matrix of Graph Cut to enforce this constraint
- If a CL pair is in the same cluster, look at Data cost of other cluster assignments
  - Switch the point has minimum cost to move into another cluster to that cluster
  - Set old cluster to have data cost ~ Infinity
- Violates connectivity due to not trusting spectral clustering



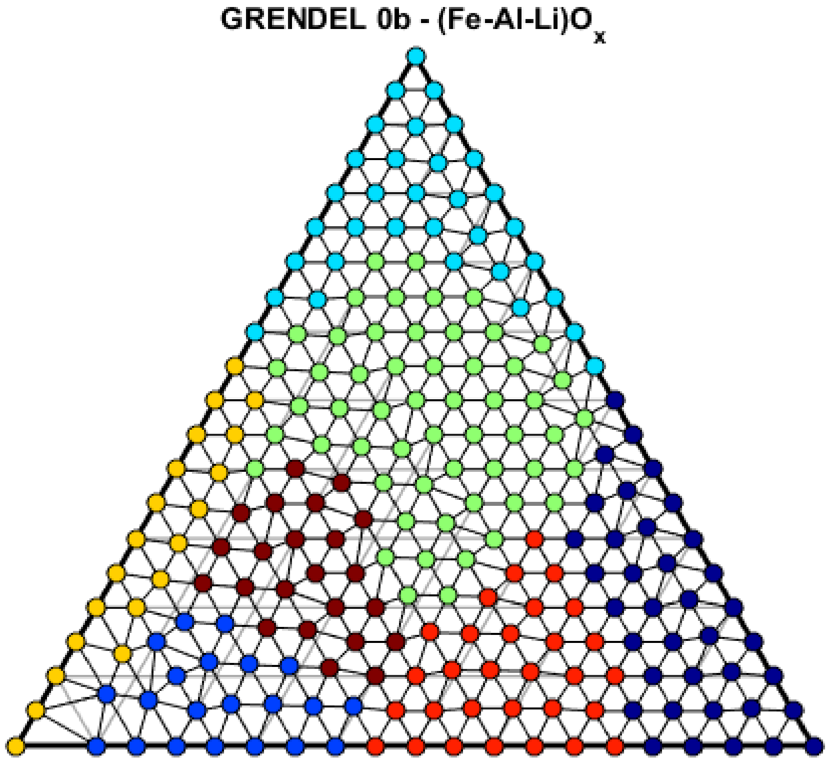
Results – Algorithm 0b Optimizing Cannot Link

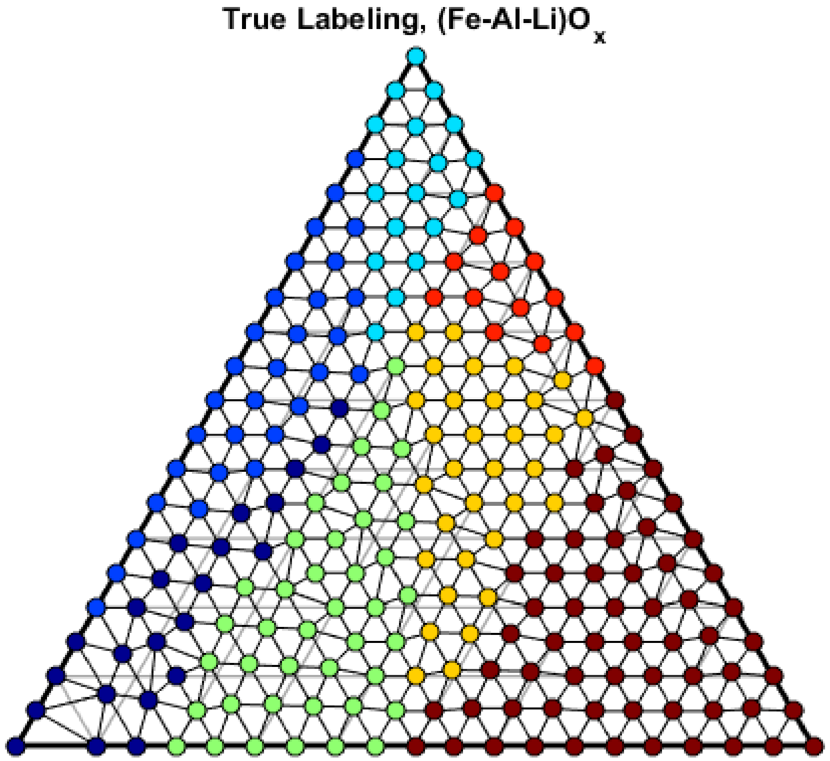
- Analysis of algorithm – NMF updates of **E** and **P** are what violate connectivity
- To become more efficient, eliminated CL pairs from array that are paired in initial Graph Cut (we assume this to be correct)



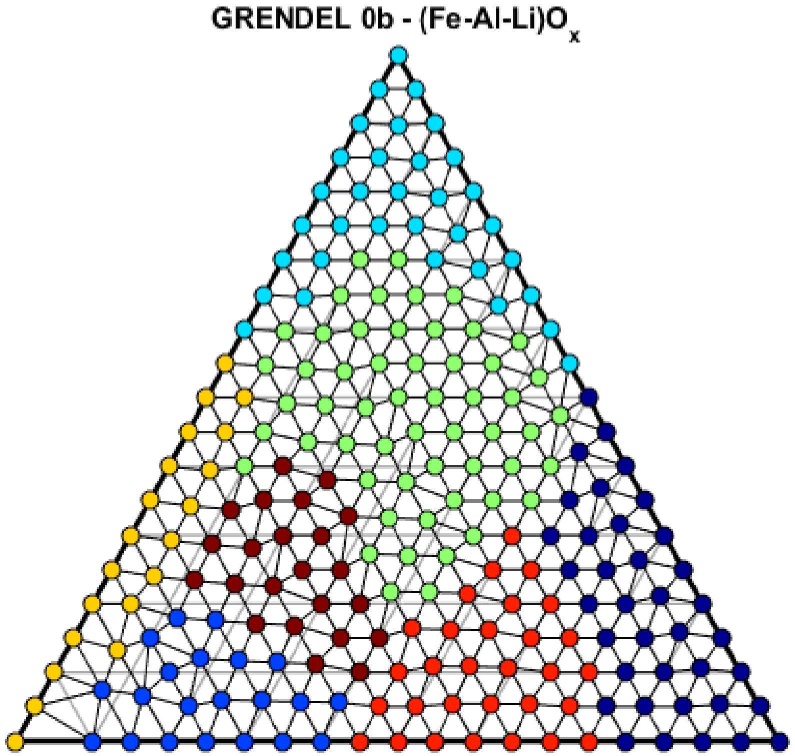


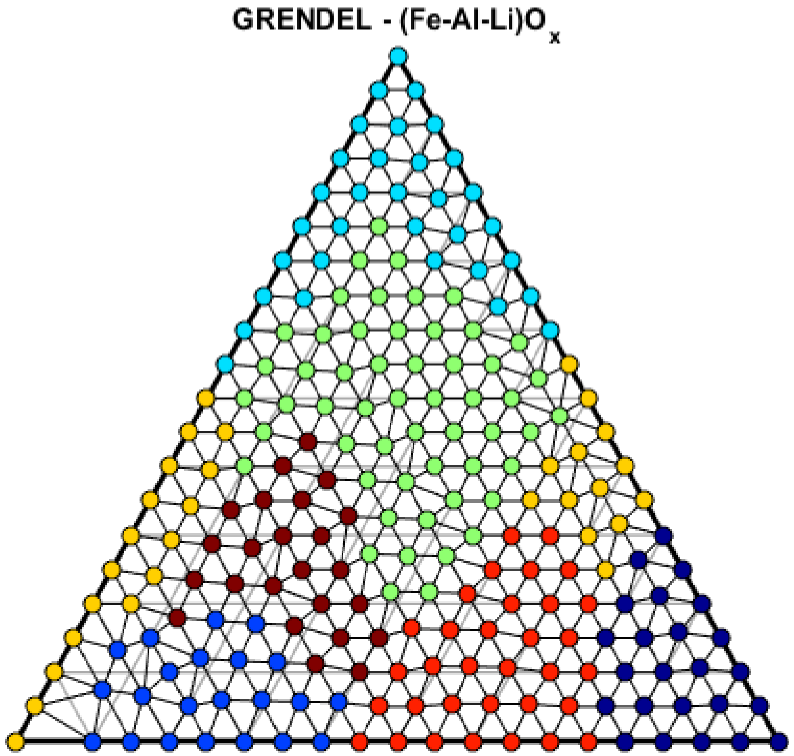
Comparison to True Values





Comparison to Original GRENDEL





Results – Algorithm 0b_connect
Adding Harder Connectivity Constraint

- Every iteration of Graph Cut, use NotConnected function:

  NotLinked = **U*(S - I).*U**

  % nonzero values in ith row are the neighbors of ith point in the same cluster

  Sum rows of NotLinked

  % value of 1 corresponds to point with no neighbors in the same cluster

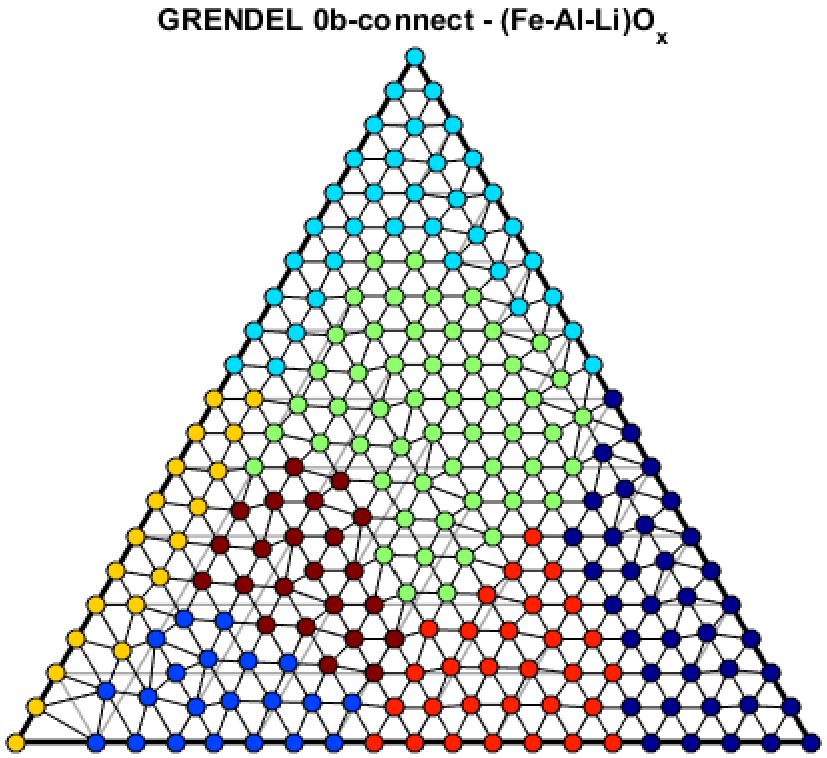
  Find that cluster assignment of NotLink data points

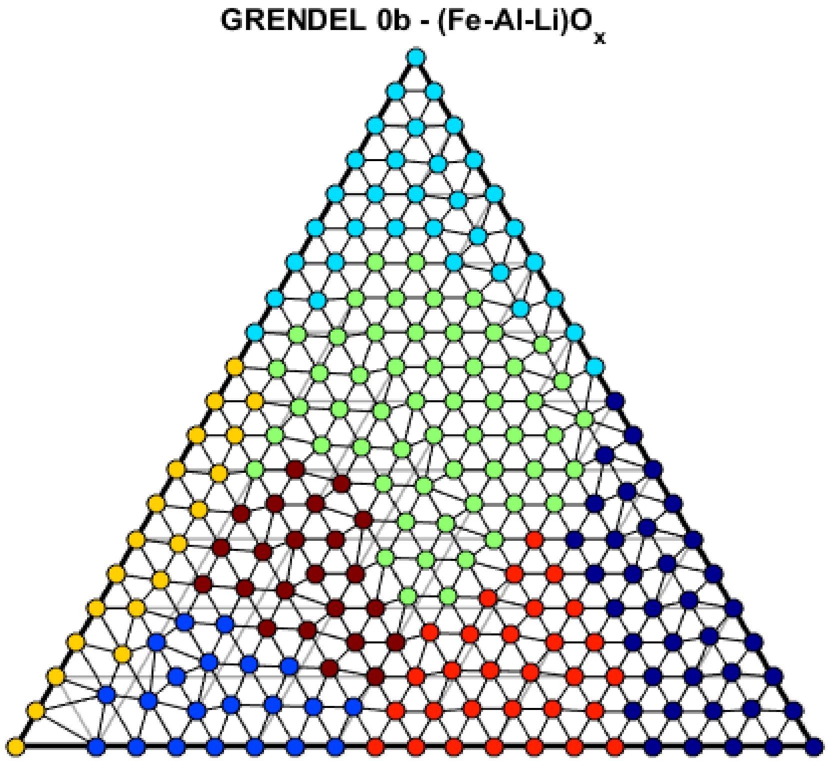
  Switch it to another cluster with smallest Data cost

  Set old cluster Data cost to ~ Infinity

Results – Algorithm 0b_connect
Adding Harder Connectivity Constraint

- Similar results to 0b algorithm, but condition number (change in results between iterations) much higher → slower convergence, longer CPU time





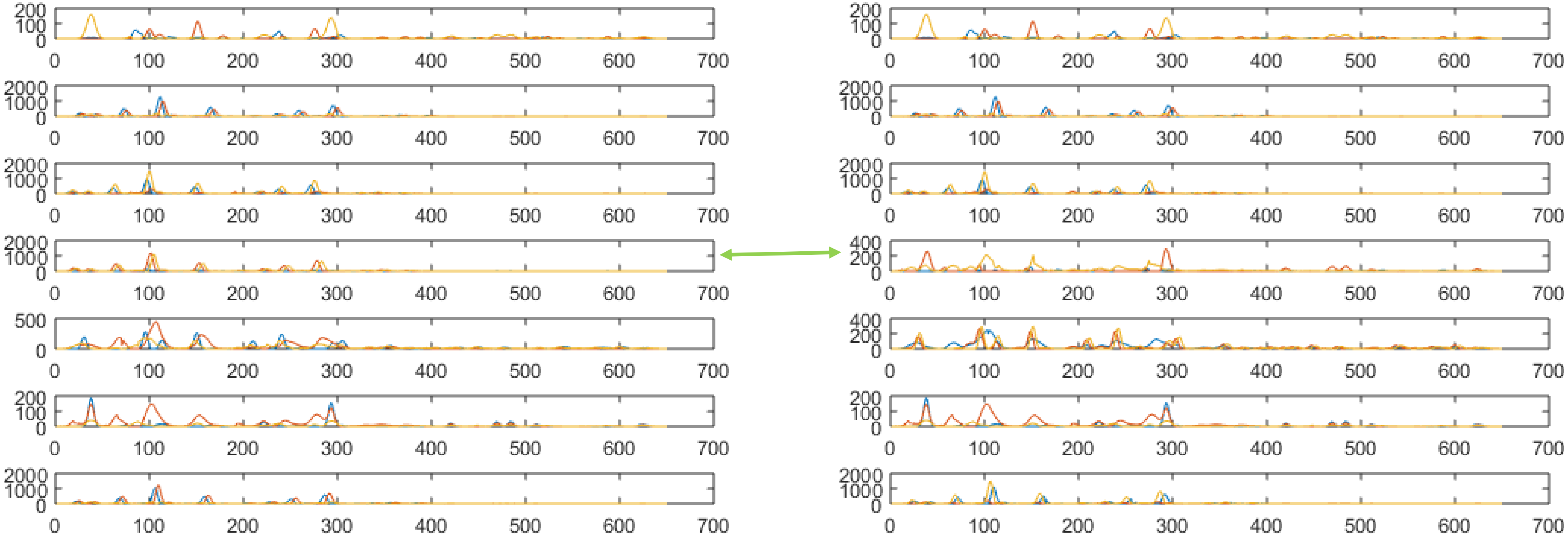
Current Work – Physical Constraints
Peak Shifting

- Seen in algorithm 0b, peak shifting is an issue for the synthetic data
- Topological Data Analysis – Mode Clustering
  - Goal: Create confidence intervals of peaks in basis phases, use these to identify phases who are the same shifted waveform
  - Not fully implemented yet
- PeakShift function:

```
Iterate over phases of all cluster, compare each of them:
Use MATLAB findpeaks() to find peak location, width, birth time (amplitude), and death time
Only look at peaks with amplitude at least $\beta$% of max peak
Cluster Interval = location ± 0.5*width
If overlap of cluster intervals is below ratio $\gamma$ AND (birthtime – deathtime) agree within $\zeta$%
    Average waveforms of both peaks (make them the same)
end
```

Current Work – Physical Constraints Peak Shifting

- Only running this every 25 iterations, get same ternary/cluster diagram
- Unshifted Spectral Plots vs. Shifted Plots:



Optimizing Parameters and Validation Statistics

- Tested parameter values for every algorithm
  - How often to run CannotLink function (every iteration)
  - Spectral Bandwidth (σ = 1e5)
  - Data cost and smoothness cost weight (Dc = 1e5, Sc = 10)
  - CannotLink Percentage (75%)
- Cutting off portions of waveform, adding noise to waveform (no help)
- Also tried Must Link algorithm, terrible results
- Use confusionmatStats MATLAB package for validation statistics, takes into account cluster-splitting
  - (Weighted) Accuracy – % of correctly assigned labels (weighted by size of cluster)
  - ‘F-score’ - % of points in correct cluster (no cluster-splitting)

Validation Statistics of Each Algorithm

Algorithm	$\|\|PE_{GRENDEL} - PE_{true}\|\|/ \|\|PE_{true}\|\|$	Accuracy	Weighted Accuracy	F-score	CPU-time (s)
Original GRENDEL	.8462	0.8552	0.8519	0.3406	37.05
0 (Cannot Link)	.8460	0.8774	0.8751	0.4339	59.74
1 (Data Cost)	1.0447	0.8082	0.7964	0.2179	50.35
0b (Optimizing CL)	.8434	0.8774	0.8758	0.4562	57.46
0b_connect	.8398	0.8774	0.8751	0.4339	61.14
0b_shift	.7786	0.8774	0.8751	0.4339	71.40

Timeline/Milestones (OLD)

- Fully understand, replicate previous code/results – mid/late October
- Phase 1 – Constraint Programming
  - Add constraints/prior knowledge, increase accuracy of results for one sample material – mid November
  - Generalize constraints, increase accuracy for all data sets given – early/mid December
- Phase 2 – Active Learning
  - Have algorithm to predict next best point to sample – early/mid February
  - Optimize the sampling algorithm for one material – early/mid March
  - Optimize algorithm for all material data given – mid/late April

Timeline/Milestones (NEW)

- Fully understand, replicate previous code/results – mid/late October
- Phase 1 – Constraint Programming
  - Add connectivity constraints, expert prior knowledge for given samples - November
  - Add constraints for peak shifting - January
  - Potential addition of other physical laws, Mixed Integer Programming - February
- Phase 2 – Active Learning
  - Have algorithm to predict next best point to sample – March
  - Optimize the sampling algorithm for one material – mid April
  - Optimize algorithm for all material data given – late April

Deliverables

- Final code/algorithm
- Results for given materials
  - Phase diagrams
  - Spectral graphs
  - Constituent phase compositions
- End of the year report and presentation

Bibliography

- LeBras R., Damoulas T., Gregoire J.M., Sabharwal A., Gomes C.P., and van Dover R.B., 2011. Constraint reasoning and kernel clustering for pattern decomposition with scaling. AAAI. CP'11: pp.508-522.
- Kusne A.G., Keller D., Anderson A., Zaban A., and Takeuchi I., 2015. High-throughput determination of structural phase diagram and constituent phases using GRENDEL. Nanotechnology. 26(44): pp. 444002.
- Ermon S., LeBras R., Suram S., Gregoire J.M., Gomes C.P., Selman B., and van Dover R.B., 2015. Pattern decomposition with complex combinatorial constraints: application to materials discovery. AAAI Conference on Artificial Intelligence. Available at http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/10020
- Hastie T., Tibshirani R., and Friedman J., 2013. *The Elements of Statistical Learning - Data Mining, Interference, and Prediction*. ed. 2 (Berlin: Springer).
- Settles B., 2012. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning #18 (Morgan & Claypool).
- Kan D., Suchoski R. Fujino S., Takeuchi I., 2009. Combinatorial investigation of structural and ferroelectric properties of A- and B- site co-doped BiFeO3 thin films. Integrated Ferroelectrics. 111: pp. 116-124.
- Takeuchi I., 2016. Data Driven Approaches to Combinatorial Materials Science. Materials Research Society Spring Meeting (presentation).
- Zare A., Gader P., Bchir O., and Frigui H., *Piecewise Convex Multiple-Model Endemember Detection and Spectral Unmixing*, IEEE Transactions on Geoscience and Remote Sensing **51** (2013), no. 5, 2853-2862.
- Boykov Y. and Kologorov V., *An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision*, IEEE Transactions on PAMI, **26** (2004), no. 9, 1124-1137.
- Xue Y., Bai J., Le Bras R., Rappazzo B., Bernstein R., Bjork J., Longpre L., Suram S., van Dover R., Gregoire J., and Gomes C., *Phase-Mapper: An AI Platform to Accelerate High Throughput Material Discovery*, CoRR, **1610** (2016).
- Suram S., Xue Y., Bai J., Le Bras R., Rappazzo B., Bernstein R., Bjorck J., Zhou L., van Dover R., Gomes C., and Gregoire J., *Automated Phase Mapping with AgileFD and its Application to Light Absorber Discovery in the V-Mn-Nb Oxide System*, arXiv:1610.02005 (2016).
- Wasserman L., *Topological Data Analysis*, arXiv:1609.08227 (2016) (Submitted to Annual Reviews in Statistics).
- Information about White House Genome Initiative courtesy of https://www.whitehouse.gov/mgi