

UCSF

UC San Francisco Electronic Theses and Dissertations

Title

Exploring the Protein Universe from General Principles

Permalink

<https://escholarship.org/uc/item/600245hv>

Author

Apeltsin, Leonard

Publication Date

2011

Peer reviewed|Thesis/dissertation

EXPLORING THE PROTEIN UNIVERSE

FROM GENERAL PRINCIPLES

by

Leonard Apeltsin

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Biological and Medical Informatics

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, SAN FRANCISCO

Approved:

Handwritten signatures in blue ink, appearing to be 'L. Apeltsin' and another signature.

Copyright ©2011

by

Leonard Apeltsin

Acknowledgments

My thesis committee, consisting of Thomas Ferrin (chair), Matthew Jacobson, and Andrej Sali, deserves special thanks for reviewing this dissertation. The text of Chapter 2 largely contains material published in the *Oxford Journal of Bioinformatics*. The text of Chapter 3 partially contains material published in *BMC Bioinformatics*. The text of Chapter 4 contains material to be submitted for publication at a later date. In these chapters, co-authors made the following contributions: John Morris aided the development of the theory in Chapters 2 and 4, and provided code for the implementation of clusterMaker discussed in Chapter 3. He cowrote portions of Chapter 3 and helped edit the manuscripts for Chapters 2 and 4. Patricia Babbitt participated in the discussions pertaining to Chapters 2 and 4. She focused on keeping the content of both these Chapters biologically relevant, while providing suggestions related to analysis and dataset selection. She participated in editing both manuscripts. Thomas Ferrin edited all manuscripts and also supervised the research that produced these chapters.

Abstract

This dissertation is concerned with the construction and validation of an organizational framework for processing large protein sequence datasets. The framework relies on the accurate clustering of input sequences into functionally similar families. We demonstrate how the quality of output for existing protein clustering techniques may be improved by running a simple edge weight selection heuristic prior to clustering. Once clustering is completed, we are able to topologically organize the data by treating each cluster as a node in a network and searching for the union of minimum spanning trees that reconnects the clusters to each other. When thusly organized, the topological relationships between neighboring clusters exhibit properties similar to evolutionary relationships computed from phylogenetic models. We demonstrate how these topological relationships may be used to algorithmically identify the functionally significant residues within the sequences in the organized dataset. This predictive capacity of the organizational framework serves as a quantitative metric for validating the framework's biological significance.

Contents

List of Figures	ix
List of Tables	x
1 Introduction	1
1.1 Perspective	1
1.2 Contribution	8
1.3 Synopsis	9
1.4 References	9
2 Clustering Protein Similarity Networks into Tables	11
2.1 Introduction	12
2.2 Methods	15
2.2.1 Dataset Selection	15
2.2.2 Computing the Similarity Network	16
2.2.3 Evaluating Clustering Performance across an Edge Weight Threshold	17
2.2.4 Analyzing the Edge Weight Distribution	17
2.2.5 Designing and Testing an Automated Edge Weight Threshold Selection Heuristic	18
2.3 Results	20
2.3.1 Edge Weight Distribution Shape	20
2.3.2 MCL Performance over Threshold Range	21
2.3.3 Force Performance over Threshold Range	22
2.3.4 Developing an Automated Threshold Heuristic from the	

Edge Weight Distribution	23
2.3.5 Automated Threshold Selection Performance on	
Additional Clustering Algorithms	30
2.4 Discussion	32
2.5 Conclusions	35
2.6 Supplementary Figures and Tables	35
2.7 References	43
3 Using Aggregated Network Clustering Techniques to Hypothesize the	
 Functions of Unknown Proteins	47
3.1 Introduction	48
3.2 Methods	50
3.2.1 Implementation	50
3.2.2 Data Sources	52
3.2.3 Protocol	53
3.3 Results	54
3.4 Discussion	58
3.5 Conclusions	59
3.6 Supplementary Figures and Tables	59
3.7 References	60
4 A Network Filtration Protocol for Elucidating Relationships between	
 Families in a Protein Similarity Network	62
4.1 Introduction	63
4.2 Methods	67
4.2.1 Outline of the Network Filtration Protocol	67

4.2.2 Data Set Selection	70
4.3 Results	71
4.3.1 Visualizing the Topology of the Enolase Superfamily Network	71
4.3.2 Structure, Function, Topology and Evolution in the Muconate Lactonizing Enzyme Subgroup	73
4.3.3 Examining the Protein Kinase Network Topology	78
4.4 Discussion	83
4.4.1 Network Topology as a Metric for Functional Similarity	84
4.4.2 Contrasting Network Analysis with Phylogenetic Analysis	85
4.4.3 Caveats	87
4.5 Conclusions	87
4.6 References	88

5 Validating Filtered Network Topologies Using a Functional

Residue Prediction Algorithm	92
5.1 Introduction	92
5.2 Predicting Protein Properties from a Filtered Network Topology	95
5.3 Developing and Implementing Protein Space Trace	100
5.3.1 Defining Invariance and Class-Specificity in PST	100
5.3.2 Propagation of Class-Specificity	100
5.3.3 Defining Functional Significance in PST	101
5.3.4 Defining Column Conservation	102
5.3.5 Ranking Functional Significance with PST	104
5.4 Testing Protein Space Trace on a Transmembrane Superfamily	105
5.5 Results	107

5.5.1 The SLC Network Topology	107
5.5.2 Top Ranking Functional Residues in OAT3	107
5.6 Discussion	111
5.6.1 Validating SLC Network Topology and the OAT3	
Functionally Critical Sites	111
5.6.2 Caveats and Future Directions	113
5.7 Conclusions	113
5.8 References	114
6 Conclusion	117
7 Appendix	120

List of Figures

2.1	Clustering performance and edge weight distributions	21
2.2	Visualizing MCL Clusters for the SLC Superfamily	26
2.3	Visualizing MCL Clusters for the Kinase Superfamily	28
S2.1	Network summary value (Nsv) from our threshold selection heuristic	36
S2.2	Visualizing MCL clusters for the Amidohydrolase superfamily	37
S2.3	Visualizing MCL clusters for the Enolase superfamily	38
S2.4	Visualizing the thresholded networks prior to clustering	39
S2.5	Mapping of node colors to family assignments for SLC and Kinase	40
3.1	Clustering indicates possible family membership for uncharacterized proteins . .	53
S3.2	Clustering the VOC superfamily using Transitivity Cluster	60
4.1	Unfiltered Enolase similarity network	72
4.2	Enolase similarity network	75
4.3	Kinase similarity network	79
5.1	Flow diagram of the PST algorithm	105
5.2	Filtered SLC protein similarity network	108
5.3	MODBASE model structure for OAT3	110

List of Tables

2.1	F-measure scores of clustering algorithms	21
S2.1	Exploring inflation parameter for MCL clustering	41
S2.2	Average edge-weights, by category, within the superfamily networks	42
S2.3	Geometric separation scores across clustering algorithms	43
3.1	HMM alignments of clustered uncharacterized proteins	56
4.1	Comparing phylogenetic divergence to filtered network topology data	81
5.1	Top ranking residues overlaid with other available data	109

Chapter 1

Introduction

1.1 PERSPECTIVE

A few simple rules on how to better organize data may on occasion greatly impact the entire scientific community. At an 1869 presentation before the Russian Chemical Society, Dmitri Mendeleev stated that the elements, when arranged according to their atomic mass, cluster into groups exhibiting similar valencies and chemical properties. Ordering the elements in a two-dimensional table revealed periodically repeating patterns of chemical behavior. This periodic table allowed Mendeleev to organize all chemical properties associated with all known elements in a way that made the relationships between properties simple to visualize and easy to understand. Furthermore, by examining the position of gaps in the table, Mendeleev could predict the atomic weights and chemical properties of unknown elements that had yet to be discovered. As these predictions proved accurate, it became obvious that the table itself was more than a useful manmade abstraction for organizing the elements. Mendeleev's rules captured certain fundamental characteristics of how elements exist in relation to each other. Fifty years later, Niels Bohr explored these fundamental characteristics by integrating electron shells into his model of the atom. The basic rules put down by Mendeleev to better organize the elements eventually led to a revolution in science.

As this example shows, integrating available scientific information into a proper organizational framework can potentially lead to three significant results:

- 1) A cleaner emphasis on patterns across known data;
- 2) The prediction of new properties for unknown data;
- 3) A deeper understanding of the given system as whole.

This is why many researchers today are focused on building a framework to aid in one of the most significant scientific endeavors of our time: deciphering the proteome. Currently the Uniprot database contains approximately 5 million protein sequences, a number that is growing exponentially. Half these proteins have not been characterized in any way, leaving unsolved the primary problem of modern biology, which is the full characterization of every known protein. Full characterization for any given protein requires relating structure-based biochemical mechanisms to the observed phenotypic characteristics associated with the protein in question. Such understanding may be achieved through scientific analysis, in which a protein is probed by a series of rigorous techniques so that enough reliable data may be recorded to partially characterize it in some way. Partial characterization yields only a single piece of the puzzle, such as phenotype relationships or structure, but not the total range of features for a protein. Nonetheless, with persistent analysis, data continues to accumulate until a protein is characterized in full.

Characterizing a single protein is a difficult enterprise, which makes the ambitions of today's biologists that much more daunting. They aim to fully characterize the millions of unknown protein sequences by relying on a multitude of experimental and computational techniques currently available for proper analysis. Unfortunately, prioritizing the application of these techniques to the proper protein targets remains an unsolved optimization problem. A Brute-Force approach in which each protein is analyzed by a series of techniques that are selected

independently of all other proteins would clearly involve an inefficient use of both time and resources. Certain proteins may only be characterized through rigorous experimental analysis, while other proteins may more efficiently be characterized by guiding experimental hypothesis selection through computational analysis of available data. The remaining proteins cannot be effectively characterized using existing experimental techniques. This is particularly true of certain hypothetical proteins, which have been predicted from nucleic acid sequences, but have not been shown to exist by experimental evidence (Lubec, *et al.* 2005). For the time being, such proteins must be characterized using only computational analysis that generates predictions to a reasonable degree of accuracy. Selecting the appropriate combination of experimental and computational techniques is a difficult problem in and of itself, but the order in which unknown proteins are characterized must also be taken into account. Experimentally characterizing one well-chosen protein within a group of unknowns might provide enough information to computationally characterize all other proteins in that group. By contrast, experimentally characterizing most any other protein in that group may provide only limited information, requiring further experiments to be carried out on additional proteins. This reliance on order is frequently seen in homology modeling, where selection of a crystallization target with the greatest number of homologous relationships helps maximize the number of homology models obtained from minimal experimental effort (Marsden and Orengo, 2008). Proper target selection may sometimes lead to more immediate hypothesis generation while also minimizing laboratory resource expenditures and time. Thus, it is desirable to be able to infer from the relationships between proteins in any given dataset the most optimal ordering of experimental and computational techniques. An appropriate organizational framework is needed to capture these relationships. Over the past decade, biological theoreticians have been working on

developing such a framework, which is most frequently referred to in the literature as “The Protein Universe.”

The protein universe encompasses within itself all possible proteins, including those that have not yet been discovered and those that have not yet evolved. The concept of the protein universe first appeared in a 1996 paper by Liisa Holm and Chris Sander. Holm and Sander suggested the protein universe represents an abstract high dimensional structural space in which in each protein structure is represented by a single point (Holm and Sander, 1996). They showed how the spatial distribution of known structures in this abstract space is centered on a finite number of key structural folds. Three years later, Golan Yona took an alternate approach to exploring the protein universe, defining proteins as points in sequence space rather than structural space (Yona, *et al.* 1999). His paper demonstrated how groups of evolutionary related proteins with similar structural and functional properties, which are frequently referred to as protein families, cluster individually in sequence space. Additionally, his paper suggested that pairs of possibly related clusters could be used to plot local maps for neighborhoods of protein families. These maps would then depict the “geometry” of protein space in the vicinity of the included families. A “protein space geometry” (PSG), defined in the paper as the topology of neighboring and non-neighboring families, somehow reflects an observable protein universe shape. Implicitly, the protein universe is more than just a density distribution of protein similarity in an abstract space. Instead, the protein universe has a geometry that can be quantified with the appropriate data and algorithmic techniques.

Golan Yona and Michael Levitt continued to advance the concept of protein space by using both sequence and structural information to explore the geometry of that space (Yona and Levitt, 2000). Distances between points in large protein datasets were calculated using sequence

similarity as well as any available structural similarity data. The distances were then projected into two-dimensional Euclidian space in order to estimate spatial geometry. This work represented two key conceptual advances in exploring the protein universe. First, despite the fact that available protein structures number in the tens of thousands while known protein sequences number in the millions, the structural information that is available could still help guide the exploration of the PSG. Second, after this paper, our understanding of protein space was no longer restricted to either sequence or structural coordinates. Sequence space and structural space both encompass aspects of the protein universe, but protein space itself represents a more general set of coordinates. In protein space, distance is a measure of proximity between protein sequence, protein structure, and protein function. Boundaries in protein space delineate families, the members of which share similar sequences, contain the same structural folds, and perform the same measurable combination of functions. These functions are defined as entropically unlikely chemical interactions between each protein and all other molecules normally found in biological systems. Neighboring families in protein space share some sequence homology, and show similar structural and functional properties. Nonetheless, despite the overlap shared with its neighbors, each family performs a unique combination of functions. More explicitly, if we represent each measured interaction between a protein and other molecules using a vector, then the resulting cosine similarity across functional vectors in the same family will, on average, outrank the cosine similarity between vectors in neighboring families. By generalizing protein space, Yona and Levitt introduced a better organizational framework with which to examine the global proteome. Their depiction of the protein universe helps address the problems associated with characterizing a large protein dataset using an optimal configuration of computational and experimental approaches. In generalized protein space, the structural and functional properties of uncharacterized families

may be inferred directly from their characterized neighbors. Uncharacterized families with multiple characterized neighbors may be analyzed by computational means. When no characterized neighbors are present, an uncharacterized family must first be analyzed experimentally. Each such family may be ranked in importance by the number of neighbors that it borders, as well as its distance to these neighbors. Priority is given to experimentally characterizing those families with the most number of reasonably proximate neighbors in order to maximize the computational dissemination of experimentally observed information. As a result, generalized protein space may be used as an organizational framework for optimally ordering the application of computational and experimental techniques over all existing proteins.

Such promising theoretical implications have motivated numerous researchers to explore various properties of the protein universe in additional detail. Based on structural data, it has been estimated that the spatial distribution of protein clusters in the protein universe is scale-free (Dokholyan *et al.* 2002). When structural similarity is used to define connectedness across a network representation of protein space, the connected network components follow a scale-free distribution of sizes that is significantly different from what could be expected in randomly generated networks. Structural data was also used to estimate the global shape of the three-dimensional representation of protein space (Hou *et al.* 2003). This representation took place in linear, Euclidian space, though it has been suggested in another recent paper that the geometry of protein space is highly nonlinear (Farnum *et al.* 2003). Embedding proteins across the surface of a stochastically determined multidimensional manifold better preserves the sequence-defined distances between them than a dimensionally-equivalent embedding in linear Euclidian space.

These research efforts support the conjecture that the protein universe adheres to a definitive, geometric structure. Implicitly, the above mentioned publications describe certain definite, shared features of spatially organized protein datasets that would be lacking in a randomly generated protein distribution. Just as the periodic table is more scientifically significant than a random arrangement of elements, certain spatial arrangements of proteins must be more correct than others. While the precise nature of this “correctness” has not yet been explicitly defined, we may nevertheless infer of its existence based on protein patterns observed in published literature. It is therefore reasonable to hypothesize that, like the periodic table, the protein universe is not a mere interpretational abstraction but is representative of an actual underlying physical reality. If this hypothesis is true, then each proposed arrangement of the protein universe is only correct to the degree with which it captures that reality. Potentially, there is single physical interpretation of the protein universe, and if true, then the geometry of the proteome in protein space must not be subject to multiple constructions. Rather, there should exist a single valid PSG corresponding directly to tangible relationships between the proteins in that space. Thus, the logical extension of the aforementioned hypothesis is that all mappings of protein space which do not completely capture this one valid geometry are not correct.

The existing body of literature gives credence to the hypothesis regarding a single correct interpretation of protein space geometry. Unfortunately, despite the progress being made, there have as of yet been no serious efforts to completely map out the geometry of the protein universe for the purposes of optimally characterizing all unknown proteins. This is because any such effort would be for the most part subjective. Currently in the literature, there is no method of quantifying how well a proposed map of the protein universe matches the actual geometry of

protein space. If two labs publish two different mappings of the proteome in protein space, there is no objective way to determine which mapping is more correct.

What is missing is an agreed-upon metric of quality for distinguishing between two geometries that vary in accuracy. The lack of metric is not surprising, given that no consensus exists on how to define a valid protein space geometry. Adequately quantifying an input PSG is a very difficult problem. Current scientific literature shows no recorded attempts to determine if a solution is possible. The goal of this thesis is to take preliminary steps that may eventually yield a resolution. We lay the groundwork for developing a metric capable of quantifying the significance of data points distributed in protein space.

1.2 CONTRIBUTION

Certain complex scientific dilemmas first require solutions to simplified versions of the problem presented. Unraveling the micro-problem may lead to valuable insights into the greater dilemma as a whole. This is the approach we have chosen to take in our search for a quantitative metric. Rather than tackle the protein universe in its entirety, we have focused on constructing and understanding a simplified organizational framework for analyzing protein sequence data. We embedded in the framework two key protein universe properties: clustering and topology. Our framework takes as input protein sequences, which are then clustered into discrete families. Afterwards, the clusters are bounded topologically into sets of neighbors and non-neighbors. This straightforward organizational technique allows us to better explore a question fundamental to the protein universe itself: how does one evaluate the biological significance of spatially organized protein data?

For our simplified framework we are able to provide an answer. Our solution, like Mendeleev's, rests on the predictive properties of the organizational framework at hand. Mendeleev was able to justify his periodic table because the table accurately predicted the existence of unknown elements. We are likewise able to justify certain structured features of our organized protein sequence data based on the accuracy of predictions made from the topological relationships across the protein clusters in that data. We hypothesize that the true geometry of the protein universe may be validated using a similar prediction-based approach.

1.3 SYNOPSIS

This thesis is structured as follows: we build a protein sequence framework from the ground up, focusing first on clustering, then on topological relationships between clusters, and finally on justifying the topological relationships using a testable prediction technique. Chapter 2 of the thesis presents a modified technique for better clustering of protein sequences into families. Chapter 3 applies this technique to a protein sequence dataset in order to yield hypothetical functional classifications for unknown proteins within that data. Chapter 4 discusses a technique that yields topological relationships across the protein sequence clusters. These topological relationships are examined qualitatively using reliable phylogenetic information. Finally, Chapter 5 presents a prediction algorithm that takes as input an organized protein sequence topology and outputs testable functional predictions pertaining to the proteins in question. The quality of the predictions may be used to justify the protein sequence topology itself, indicating that it is possible to measure the accuracy of spatially structured protein data.

1.4 REFERENCES

- Dokholyan N.V. *et al.* (2003) Expanding the protein universe and its origin from the biological Big Bang. *Proc Natl Acad Sci.*, **29**, 14132-14136.
- Farnum, M.A. *et al.* (2003) Exploring the nonlinear geometry of protein homology. *Protein Sci.* **12**, 1604-1612.
- Hediger, M.A *et al.* (2004). The ABCs of solute carriers: physiological, pathological and therapeutic implications of human membrane transport proteins. *Pflugers Ar* **447** (5): 465–8.
- Hou, J. *et al.* (2003) A global representation of the protein fold space. *Proc Natl Acad Sci.* **4**, 2386-2390.
- Lubec, G. *et al.* (2005) Searching for hypothetical proteins: theory and practice based upon original data and literature. *Prog Neurobiol.* **77**, 90-127.
- Marsden, R.L. and Orengo, C.A. (2008) Target selection for structural genomics: an overview. *Methods Mol Biol.* **426**, 3-25.
- Yona, G. *et al.* (1999). ProtoMap: Automated classification of protein sequences, a hierarchy of protein families, and local maps of the protein space. *Proteins: Structure, Function and Genetics*, **37**, 360-378.
- Yona, G and Levitt M. (2000) Towards a complete map of the protein space based on a unified sequence and structure analysis of all known proteins. *In the proceedings of ISMB 2000*, 395-406.

Chapter 2

Clustering Protein Similarity Networks into Families

The material in this chapter has been published in the Oxford

Journal of Bioinformatics as:

Improving the Quality of Protein Similarity Network Clustering

Algorithms using the Network Edge Weight Distribution

Leonard Apeltsin¹, John H. Morris¹, Patricia C. Babbitt^{2,1}, Thomas E. Ferrin^{1,2}

¹Department of Pharmaceutical Chemistry, University of California, San Francisco, USA

²Department of Bioengineering and Therapeutic Sciences, University of California, San Francisco, USA

Abstract

Motivation: Clustering protein sequence data into functionally specific families is a difficult but important problem in biological research. One useful approach for tackling this problem involves representing the sequence dataset as a protein similarity network, and afterwards clustering the network using advanced graph analysis techniques. Although a multitude of such network clustering algorithms have been developed over the past few years, comparing algorithms is often difficult because performance is affected by the specifics of network construction. We investigate an important aspect of network construction used in analyzing protein superfamilies and present a heuristic approach for improving the performance of several algorithms.

Results: We analyzed how the performance of network clustering algorithms relates to thresholding the network prior to clustering. Our results, over four different datasets, show how for each input dataset there exists an optimal threshold range over which an algorithm generates its most accurate clustering output. Our results further show how the optimal threshold range correlates with the shape of the edge weight distribution for the input similarity network. We used this correlation to develop an automated threshold selection heuristic in order to most optimally filter a similarity network prior to clustering. This heuristic allows researchers to process their protein datasets with runtime efficient network clustering algorithms without sacrificing the clustering accuracy of the final results.

2.1 INTRODUCTION

In the last decade, there has been an explosion in the available protein sequence data. Currently, the Uniprot database contains approximately 11 million protein sequences and is growing exponentially (Apweiler et al., 2004); a very large proportion of these proteins have not been experimentally characterized. Computational clustering approaches can provide an important means to deciphering the functions of these uncharacterized proteins in an efficient way. Recent efforts in this area, discussed below, have focused on developing and testing algorithms for clustering proteins by functional similarity based only on sequence data. These algorithms go beyond traditional clustering approaches, such as hierarchical and k-means, which require advance knowledge approximating the number of functional groups present in order to either cluster effectively or to interpret clustering output correctly. Rather, these algorithms rely on the network properties of a protein sequence dataset to cluster the data into functional groups without any prior knowledge of the group identities (Schaeffer, 2007).

Network clustering algorithms take as input a protein similarity graph (Noble et al., 2005).

Vertices in the graph represent individual proteins, while edges represent the pairwise sequence similarities between the proteins. Often, BLAST (Altschul et al., 1997) scores are used as edge weights. Subsequent to input, the similarity graph is processed by the network clustering algorithm to identify distinct groups of nodes in the graph that in many cases correspond to groups of proteins that share the same function.

How the similarity graphs are processed varies with each clustering algorithm. In general, most network clustering approaches may be assigned to one of two categories; geometry-based and flow-based (Frivolt and Pok, 2006). Geometry-based approaches, such as Force (Wittkop et al., 2007), Regularized Kernel Estimation (Lu et al., 2005), spectral clustering (Paccanaro et al., 2006) and TransClust (Wittkop et al., 2010) embed the protein graph into high-dimensional space and then group the nodes into clusters based on spatial proximity. Flow-based approaches, such as the Markov Clustering Algorithm (MCL; Enright et al., 2002) and Affinity Propagation (Frey and Dueck, 2007) model the possible flow of information between nodes based on edge weight.

How the information congregates across groups of nodes then determines the final output of clusters.

The differences between these two categorizes of algorithms reflect a difference in performance. Geometry-based approaches such as Force rely on non-linear calculations between pairwise elements in the similarity graph, leading to potentially long execution times. Flow-based approaches such as MCL rely on simple matrix and vector multiplication, which leads to relatively short execution times. However, it has been shown that Force outperforms MCL for certain similarity graphs (Wittkop et al., 2007), making the hours to seconds difference in run times a worthwhile performance trade-off.

While comparative performance of various network clustering algorithms has been examined in great detail in the literature, there remains a property of network clustering that warrants additional investigation. The protein similarity graphs themselves are treated as static objects when used as input to the algorithms. These graphs, however, are not static but rather exhibit a variety of dynamic properties when studied over a series of different edge weights (Atkinson et al., 2009). As the threshold for allowing edge weight inclusion is adjusted, the similarity graphs break and regroup into varying representations of protein similarity when visualized using an edge weighted network layout algorithm (Fruchterman and Rheingold, 1991). By viewing the graph behavior over a range of thresholds with a network visualization tool such as Cytoscape (Shannon et al., 2003) or BioLayout (Enright and Ouzounis, 2001), a researcher may observe degrees of protein similarity that are not visible in the complete, unthresholded graph. In other words, given a graph, one particular threshold may be more optimal for analysis than another.

It was our goal to examine how dynamic graph thresholding relates to the various network clustering approaches. We set out to answer a number of important questions. Given a protein similarity graph and a network clustering algorithm, does a threshold exist at which network clustering performance is optimal? If so, how does the optimal threshold vary across different graphs and different categories of network clustering algorithms? Given an uncharacterized protein similarity graph, can we estimate the optimal edge weight threshold from the properties of the graph itself, prior to clustering?

We used two representative and well-studied network clustering algorithms for our analysis, Force and MCL. Our results are somewhat surprising. For any of our four datasets (see below) and the two network clustering algorithms, there is a range of thresholds over which algorithm performance will be near optimal. This threshold range does not necessarily include zero, the

threshold at which the graph remains completely unfiltered. More importantly, our research shows that the optimal threshold range for any given similarity graph is dependent on the edge weight distribution across that graph. These findings allowed us to test a heuristic for estimating thresholds within the optimal range using network properties pertaining to the edge weight distribution. Applying our automated threshold selection heuristic prior to clustering improves performance for both Force and MCL. In addition, automated threshold selection bridges the gap between Force and MCL in comparative accuracy analysis. We also tested the threshold heuristic on three other clustering algorithms. The use of a threshold yielded improvement, but MCL continued to outperform all other algorithms after a threshold was applied. As a result, we believe researchers may now more confidently use the time-efficient MCL clustering technique for most of their protein sequence analysis needs.

2.2 METHODS

2.2.1 Dataset Selection

Protein sequence datasets from four well-studied superfamilies were used in our study. Each superfamily is composed of individual families categorized by a distinct set of functions. This allowed us to test cluster performance based on how well individual clusters overlap with functionally characterized protein families. Two of the superfamilies, Enolase (Gerlt et al., 2005) and Amidohydrolase (Seibert and Raushel, 2005), represent enzymes that perform catalytic functions. These superfamilies are available as a 'gold standard' set of well-characterized mechanistically diverse enzyme superfamilies (Brown et al., 2006) in the Structure-Function Linkage Database (Pegg et al., 2006). A third dataset was composed of sequences from a recent study on the solute-carrier transferase (SLC) superfamily (Schlessinger et al., 2010). The final

dataset contained sequences from the extensively studied Kinase superfamily (Manning et al., 2002). A total of 1174 amidohydrolase sequences, 1308 enolase sequences, 696 SLC sequences and 527 kinase sequences were used in our study.

Of course, this data represents just a small sampling of each superfamily. For amidohydrolase alone, there are over 20 000 known members. Nonetheless, the families in each dataset represent a diverse sampling of sequence–structure–function relationships which are not trivial to distinguish from one another. Certain superfamily members in the dataset share nearly identical sequences, with a few amino acids accounting for the different functions they perform (Seffernick et al., 2001). More divergent families often share similar structural elements in which at least the active site residues associated with the superfamily-common partial reaction are conserved despite sharing a low level of sequence identity (Glasner et al., 2006). Thus, our sampling of superfamily data provides good test cases for measuring algorithm performance.

2.2.2 Computing the Similarity Network

For all four datasets, we carried out an all-by-all BLAST search using a custom database built from all sequences in the dataset. Four such runs were executed for each of the four families. The BLAST expectation value (E-value) cutoff for each search was set to one. Next, each protein was treated as a node in the similarity network. Whenever a BLAST alignment was returned between two proteins in the dataset, we connected these proteins with an edge. Each edge was given a weight equivalent to the $-\log$ of the BLAST e-value. Of course, using such a relaxed cutoff value produces a dense network where virtually every node is connected to every other node.

2.2.3 Evaluating Clustering Performance across an Edge Weight Threshold

Range

Each superfamily similarity network was filtered across a consecutive series of edge weight thresholds, ranging from zero to 100. At each threshold, all edges below the threshold were removed from the network. The filtered similarity network was then clustered using both Force and MCL. Clustering performance for each algorithm was quantified through F-measure, an evaluation criterion previously used both to study and compare multiple protein clustering techniques (Paccanaro et al., 2006; Wittkop et al., 2007) as well as in other areas of research where clustering is used (Chim and Dang, 2007). F-measure, ranging in value from zero to one, allowed us to compare the performance of both algorithms over the entire threshold range.

To compute the F-measure, we characterized all pairs of proteins classified as belonging to the same functional family as true positives and all pairs of proteins classified as belonging to different families as true negatives. Each clustering run estimated the identities of the families, with respect to the family assignments in each dataset, leading to a count of true positives, false positives, true negatives and false negatives in the clustered data. These four values were then used to compute precision (P) and recall (R), which were then used to generate the F-measure using the formula $2 * P * R / (P + R)$. An F-measure of 0.5 or less indicates clustering performance that was no better than random. An F-measure of 0.9 or more indicates very accurate clustering performance, because both high precision and high recall are desirable, and the F-measure reflects both these values as their arithmetic mean (Rodriguez-Esteban et al., 2009).

2.2.4 Analyzing the Edge Weight Distribution

In order to test how threshold selection relates to the similarity network edge weight distribution, we computed the edge weight histogram for each of the four superfamily networks. The number of edges in each network at each threshold value was counted and plotted. For binning purposes, we rounded the $-\log$ of the edge weights to the nearest integer. The edge weight histogram plot could then be overlaid with the thresholded clustering performance data to test for a relationship between performance and the shape of the distribution.

To better overlay distribution shape and clustering performance, we normalized each edge weight distribution. Normalization was carried out by first selecting the edge weight bin containing the greatest edge count. Next, the edge count in each edge weight bin was divided by this maximum value. This resulted in a distribution whose value at each edge weight ranged from zero to one. This range also corresponds to the range of F-measure, allowing us to view clustering performance and edge weight distribution shape using a single axis in our plots.

2.2.5 Designing and Testing an Automated Edge Weight Threshold

Selection Heuristic

Network-based clustering algorithms group protein sequences into clusters that ideally correspond to functional families by estimating the edges that most likely connect proteins belonging to the same cluster based on network topology. These techniques arise directly from graph theory, and therefore do not consider certain biological properties relevant to our networks of interest. In particular, a purely topological analysis does not explicitly take into account that proteins with very low sequence identity are less likely to perform the same function as proteins with greater similarity (Ponting, 2001). With this assumption in mind, we

aimed to design a simple threshold selection heuristic for automatically prefiltering a protein similarity network prior to clustering.

In order to do so, we first needed to study the properties of the similarity network edge weight distribution, and how these properties overlap with Force and MCL clustering performance.

These observations, discussed in Section 2.3, led to the direct development of a threshold selection heuristic. The details and logic behind our heuristic are discussed in Section 2.3.4.

Suspecting our heuristic would arise from specific details pertaining to the Force and MCL clustering algorithms, we expanded the scope of our evaluation beyond these two clustering approaches by choosing three additional biological network clustering algorithms for testing. Using each algorithm, we clustered all four networks, both in their unthresholded state as well as at the threshold determined using our heuristic. We used this comparison to evaluate whether the automatically selected threshold generally leads to better clustering performance.

The three additional algorithms selected for testing our heuristic were TransClust, Spectral Clustering of Protein Sequences (SCPS) (Paccanaro et al., 2006) and Affinity Propagation. The first two of these were designed to cluster protein similarity networks and the third is a general purpose clustering algorithm. TransClust is a geometrical layout-based clustering algorithm similar to Force, designed to cluster proteins into families directly. SCPS is a variation of spectral clustering. Unlike most spectral clustering algorithms, SCPS does not require the number of clusters to be known in advance. SCPS was designed with the purpose of clustering protein sequences into superfamilies, but its capacity to cluster proteins into families has not yet been explored. Affinity Propagation has been suggested as an alternative to MCL for protein interaction networks (Vlasblom et al., 2009). The diversity of purpose behind these approaches

gave us additional reason to measure their ability to cluster proteins into families, relative to Force, MCL, and each other, under both thresholded and unthresholded conditions.

2.3 RESULTS

2.3.1 Edge Weight Distribution Shape

Figure 2.1 shows the clustering performance for all four datasets over the threshold range. Clustering performance has been overlaid with the normalized edge weight distribution associated with each dataset.

The edge weight distributions for all four datasets share similar characteristics. The maximum point in each distribution is located at a very low edge weight value, at or near zero. As the edge weight value increases, the normalized edge count begins to decay. It descends from the maximum value of one towards a small value between 0.1 and zero. In three of the four distributions (Fig. 2.1A, B and D), a second, much smaller peak is also present. The smaller local maximum is located further along each distribution, at a larger edge weight value. Eventually, as the edge weight increases, each edge weight distribution drops to a value of zero and does not rise again.

The four edge weight distributions may be further subdivided into two broad categories based on the rate with which each distribution descends from the maximum toward the local minimum. In the Amidohydrolase and SLC distributions (Fig. 2.1A and B, respectively), the descent is immediate, occurring over a range of less than five edge weight bins. In the Enolase and Kinase distributions (Fig. 2.1C and D, respectively), the descent is more gradual, occurring over a range of 20 or more edge weight bins. We refer to the former as rapid-descent histograms, and the later as gradual-descent histograms.

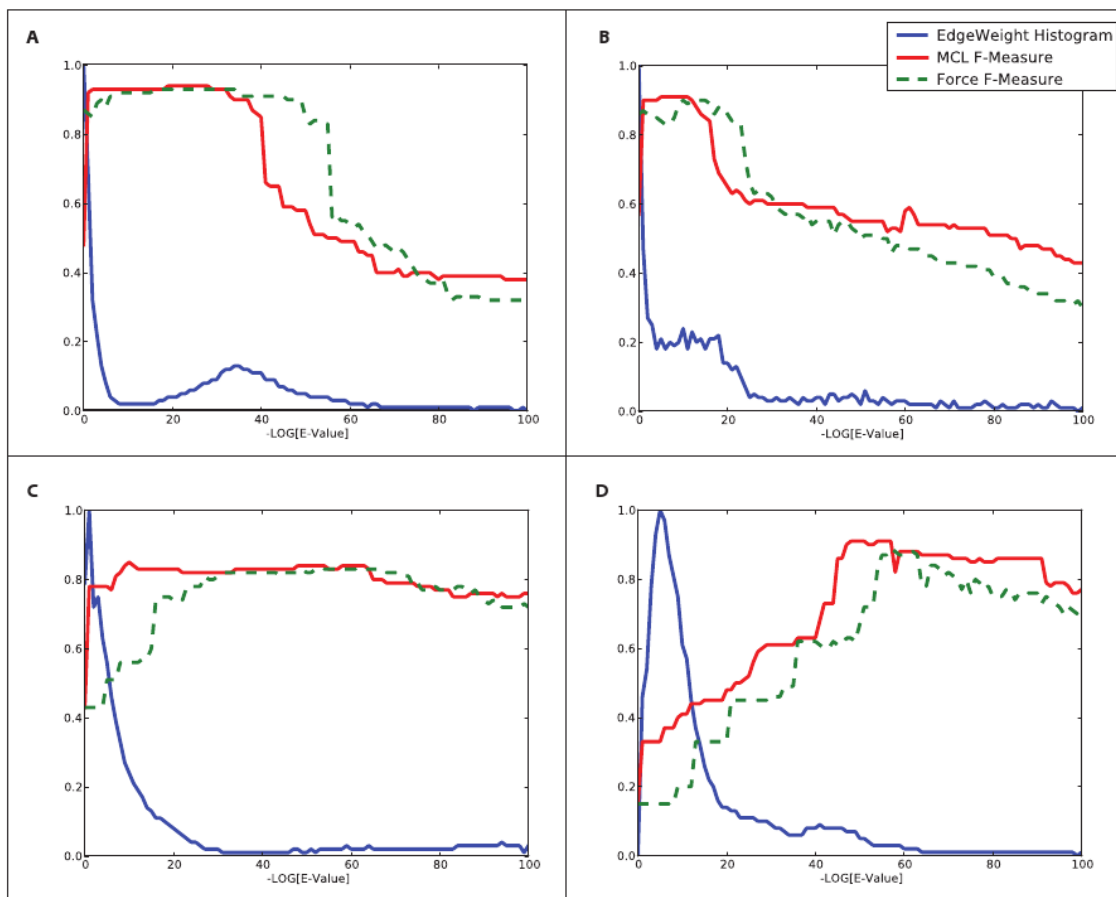


Fig. 2. 1. Clustering performance and edge weight distributions. Each plot shows the F-measure clustering performance metric for both the Force and MCL clustering algorithms over a range of binned $-\log(\text{E-value})$ thresholds, together with a normalized edge weight distribution. (A) Amidohydrolase; edge weight distribution is rapid-descent. (B) SLC; edge weight distribution is rapid-descent. (C) Enolase; edge weight distribution is gradual-descent. (D) Kinase; edge weight distribution is gradual-descent.

2.3.2 MCL Performance over Threshold Range

MCL algorithm performance follows the same general trend for all four datasets. Initially, the unthresholded MCL clustering results produce a low-performance F-measure. For three of the four datasets (Fig. 2.1A, C and D), the initial MCL F-measure is below 0.5. The MCL inflation parameter is known to influence the granularity of the clustering. We therefore explored the impact of alternate inflation parameters on the initial MCL F-measures for these superfamilies (Supplementary Table S2.1). These alternate inflation values did not yield significant improvements.

As the edge weight threshold increases and the edge weight distribution begins to decrease from the maximum, the MCL performance increases in quality. When the edge weight distribution approaches the local minimum, the MCL performance measure finally plateaus at its maximum value. For three of the four datasets (Fig. 2.1A, B and D), the maximum performance plateau is above 0.9. Finally, as the edge weight distribution decays completely to zero, MCL performance also drops significantly. Since MCL performance is greatly dependent on the shape of the edge weight distribution, it is not surprising that performance improves at a greater rate and plateaus at a lower threshold in the rapid-descent histograms than it does in the gradual-descent histograms.

2.3.3 Force Performance over Threshold Range

Force algorithm performance diverges across the two categories of edge weight distributions. In the two rapid-descent histograms, Force performs well even when no initial threshold is present. Force performance for both the rapid-descent histograms lies between 0.8 and 0.9 at edge weight zero. Performance then rises to approximately 0.9 as additional thresholds are considered. Eventually, when the threshold becomes too great, Force performance quickly decays in a manner similar to MCL performance. These results indicate that thresholding

provides the Force clustering algorithm with little additional benefits when a rapid-descent distribution is present. Furthermore, unthresholded Force clustering will outperform unthresholded MCL clustering in a rapid-descent histogram.

For the two gradual-descent histograms, the opposite holds true. Initially, unthresholded Force performance is poor, falling below 0.5. As the threshold rises from zero, performance increases. This increase, however, is more gradual than the increase in MCL performance over the same threshold range. Eventually, Force performance plateaus at a maximum value equal to the best MCL performance. However, because of the gradual performance increase, Force reaches its maximum value at a greater threshold than MCL. Eventually, the threshold becomes too large, and algorithm performance decays.

2.3.4 Developing an Automated Threshold Selection Heuristic from Edge Weight Distributions

As the edge weight distribution drops rapidly, we observe an increase in clustering quality. From these observations, we may assume that at low-level thresholds, most of the edges removed exist between protein families. The presence of these intercluster edges is effectively noise, which impacts the overall clustering results. Eventually, when the threshold is high enough, a boundary is reached at which most intercluster edges have already been removed. The boundary represents the optimal threshold separating intercluster edges from intracluster edges. At this boundary, the protein family components may begin to be affected by the filtration process, and certain loosely connected nodes may break off from the main network. However, the final increase in clustering precision overcomes any decrease in clustering recall, and the overall clustering quality noticeably improves. Thus, we would like to use this as our

threshold in order to maximize the filtration of intercluster edges while minimizing the disruption of the protein family clusters.

Our goal was to estimate this boundary automatically, without knowing in advance the identity of the proteins in the network. We heuristically approximated this boundary b using two available network properties. The first property, $Nn(Th)$, is the number of nodes connected by one or more edges at threshold Th . The second property is $SE(Th)$, the number edges remaining after threshold Th is applied. We combined these two properties into a single network summary value, Nsv , where $Nsv(Th) = SE(Th)/Nn(Th)$. Conceptually, $Nsv(Th)$ is equivalent to the average weighted node degree at threshold Th .

We chose to examine Nsv because its derivative with respect to the threshold, $dNsv(Th)/dTh$, could potentially reveal interesting behaviors pertaining to the network. At low value thresholds, most of the filtered edges are between families, while the individual family components remain strongly connected. At these thresholds, the value of SE decreases while the value of $Nn(Th)$ remains stable. Thus, as we begin to increase the threshold and filter out the noisy intercluster edges, we expect the value of $dNsv(Th)/dTh$ to be negative.

When $Th = b$, we expect most intercluster edges to have been already removed. We also expect a few poorly connected outlier nodes to disconnect completely from the network, leading to a decrease in $Nn(Th)$. $SE(Th)$ will also continue to decrease, but at a lesser rate than at lower thresholds, due to a slowdown in the initial rapid decay observed in the edge weight distribution. If the decrease in $Nn(Th)$ is great enough, and the decrease in $SE(Th)$ is low enough, then the value Nsv may actually increase. In this case, $dNsv(Th)/dTh$ will take on a positive value at a threshold proximate to the boundary, but not at lower thresholds. This leads to the following threshold estimation heuristic: b is approximate to the minimum threshold Th at

which $dNsv(Th)/dTh > 0$. If Nsv does not increase at any point in the distribution, then no threshold is returned.

We applied this heuristic to all four networks, generating thresholds of 1.0 for SLC, 1.0 for Amidohydrolase, 20.0 for Enolase and 69.0 for Kinase. Plots of $dNsv(Th)/dTh$ for all four networks are available in the Supplementary Figure S2.1. The heuristically determined thresholds lead to a performance increase in three of the four datasets (Fig. 2.1A, C and D) for both the Force and MCL algorithms, relative to the unthresholded performance of these same algorithms. For the fourth dataset (Fig. 2.1B), thresholding leads to an increase in MCL performance and a slight decrease of .01 in Force performance. For all four datasets, MCL performance at the heuristically determined threshold is greater than Force performance at that same threshold.

We qualitatively confirmed the improvement in clustering quality by visualizing the generated clusters prior to and after filtering with the heuristically determined thresholds described above (Figs 2.2 and 2.3) using Cytoscape (Shannon et al., 2003), a biological network visualization and analysis tool with both MCL and Force clustering capabilities, and through the use of its ClusterMaker plugin (<http://www.cgl.ucsf.edu/cytoscape/cluster/clusterMaker.html>).

Visualization was carried out by removing all edges from each similarity network that did not correspond to pairs of proteins within the same cluster. Afterwards, the clusters within each network were made visible through Cytoscape's Organic layout, which is a force-directed layout algorithm similar to Fruchterman-Reingold (Fruchterman and Reingold, 1991). Nodes in the visualized network were colored by known protein family assignment to allow for visual assessment of clustering quality.

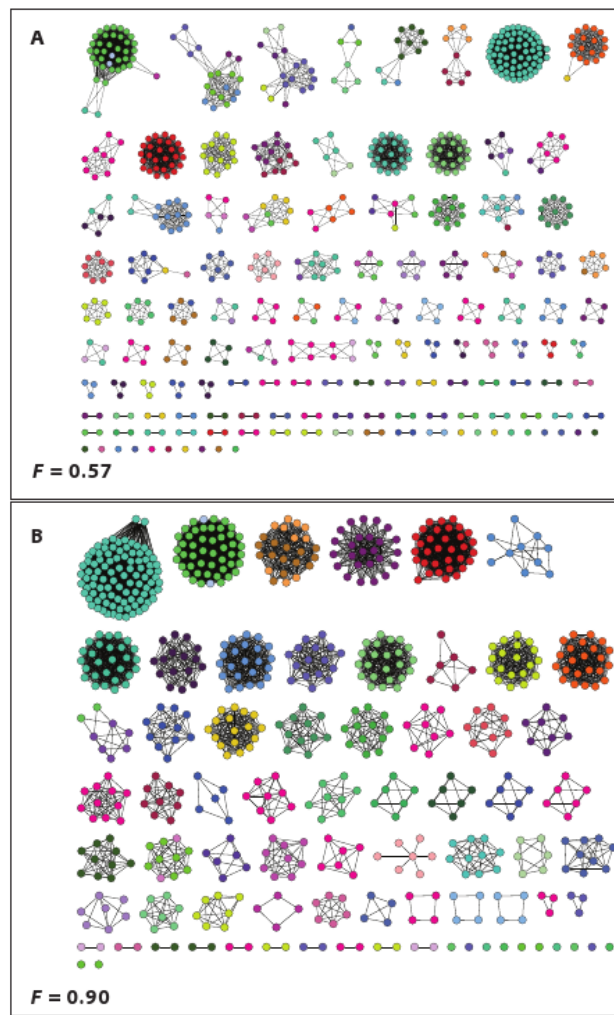


Fig. 2.2. Visualizing MCL Clusters for the SLC Superfamily. Each set of clustering results has been visualized in Cytoscape using the Force-directed layout algorithm. Each node represents a protein, colored by the currently best available family assignments. Edges between nodes that are not in the same cluster have been removed from the similarity network prior to visualization. The unthresholded clustering results are shown in (A) and the thresholded clustering results are shown in (B). The same thresholded network is shown unclustered in

Supplementary Figure S2.4b. The mapping of node colors to family assignments is shown in Supplementary Figure S2.5a.

Figure 2.2A shows the unthresholded MCL clustering output for the SLC superfamily, where many false negatives are present. Multiple proteins belonging to the same family are grouped across distinct clusters, instead of being grouped together. In Figure 2.2B, the heuristically selected threshold has been applied and the number of false negatives has correspondingly decreased. Eliminating certain redundant edges between subsets of proteins within families prevents these subsets from clustering into distinct groups. As a result, more proteins within the same family are clustered together.

Figure 2.3A shows the unthresholded MCL clustering output for the kinase superfamily. The two resulting clusters provide little discrimination among sequences that are known to belong to different functional families. Figure 2.3B shows the corresponding change in clustering output after applying a heuristically selected threshold. Many of the families that have previously clustered together now separate out into their own distinct clusters. This same pattern as seen in the kinase family also holds for Amidohydrolase (Supplementary Fig. S2.2) and Enolase (Supplementary Fig. S2.3) superfamilies.

Visualization of the clustering of well-annotated protein families reaffirms that the increase in F-measure values after automated thresholding corresponds to a genuine increase in clustering quality. Thresholding eliminates false positives in some networks, and eliminates false negatives in others, leading to a relevant improvement in the accuracy of the final clustered results.

Visualizing the thresholded networks prior to clustering (Supplementary Fig. S2.4) emphasizes the role of thresholding in the improvement of clustering quality. When the thresholded SLC,

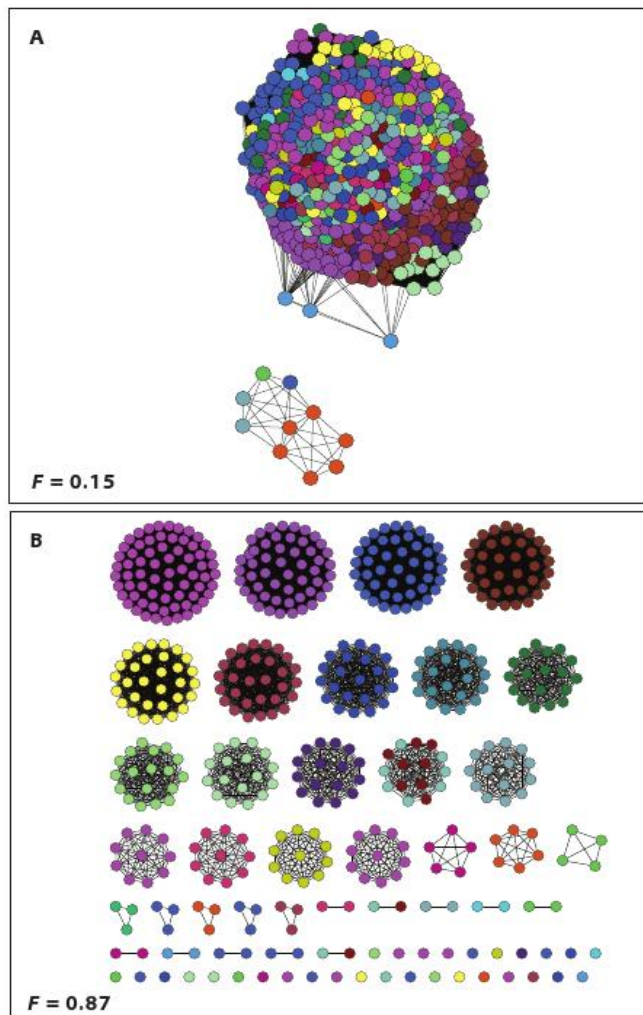


Fig. 2.3. Visualizing MCL Clusters for the Kinase Superfamily. Each set of clustering results has been visualized in Cytoscape using the Force-directed layout algorithm. Each node represents a protein, colored by the currently best available family assignments. Edges between nodes that are not in the same cluster have been removed from the similarity network prior to visualization. The unthresholded clustering results are shown in (A) and the thresholded clustering results are shown in (B). The same thresholded network is shown as unclustered in

Supplementary Figure S2.4d. The mapping of node colors to family assignments is shown in Supplementary Figure S2.5b.

Amidohydrolase and Enolase networks are displayed using a force-directed layout, the boundaries between individual families clearly become visible, even though edges continue to connect the families. In the thresholded Kinase network, clusters of individual families separate out completely, resulting in high-quality input for any protein family-specific clustering algorithm.

To investigate further why families separate out from the thresholded Kinase network, but not from the other three datasets, we hypothesized that this network consisted of tightly connected Kinase families with unusually high edge weights. To confirm this, we calculated the average edge weight within families, and also between families, for each of the four datasets (Supplementary Table S2.2). We found that while the average intrafamily edge weight for Kinase ranked highly at 99.7, the average intrafamily edge weight for Enolase ranked even higher at 114.9. The key to understanding what made Kinase distinct lay in the average edge weight between families. The average interfamily edge family edge weight for Kinase was very low, at 13.8, relative to its high intrafamily edge weight. Furthermore, the Kinase superfamily was the only dataset assigned a threshold greater than its average interfamily edge weight. In the other three datasets, our heuristic assigned a threshold that filters some but not all of the intercluster edges. From these results, we draw the conclusion that our threshold heuristic will filter out some network noise without completely disrupting network connectivity, except in those cases when there is a significant difference between interfamily and intrafamily edge weights.

Additionally, our results showed that the average intrafamily edge weights for SLC and Amidohydrolase were 38.7 and 51.6, respectively. This leads us to hypothesize that the difference in the shape of gradual-descending and rapid-descending distributions is related to the connectivity between families. In the two rapid-descending datasets, connectivity exists at lower edge weights, resulting in a more rapid transition between interfamily edges and intrafamily edges. We believe this more rapid transition results in a more rapid edge weight decay, as observed in our plots.

2.3.5 Automated Threshold Selection Performance on Additional Clustering Algorithms

Table 2.1 lists the performance of the MCL, Force, TransClust, SPCS and Affinity Propagation algorithms for both the unthresholded networks, as well as the networks filtered using our automated threshold selection heuristic. MCL outperforms the other four algorithms, but only when the automated threshold is applied.

TransClust ranks third in clustering performance. Both the thresholded and unthresholded SLC networks score an F-measure of 0.87. The thresholded Kinase network scores an F-measure of 0.82, improving significantly from the unthresholded F-measure of 0.15. Thresholding the Enolase and Amidohydrolase networks also improves the TransClust output, but not to a significant extent. Both these thresholded networks score an F-measure of less than 0.70.

SCPS performs exceedingly poorly when its primary parameter values remain unchanged. Although clustering improvements are observed in all four networks after thresholding is applied, F-measure values score below 0.70 for three of our four datasets. In an effort to improve these, we attempted to adjust the SPCS epsilon parameter, which is responsible for

Table 2.1. F-measure scores of clustering algorithms for thresholded and unthresholded superfamilies.

	Amidohydrolase		SLC		Enolase		Kinase	
	U	T	U	T	U	T	U	T
MCL	0.48	0.92	0.57	0.90	0.43	0.83	0.15	0.87
Force	0.87	0.86	0.86	0.87	0.43	0.74	0.15	0.84
TransClust	0.49	0.59	0.87	0.87	0.46	0.66	0.15	0.82
SCPS	0.30	0.37	0.12	0.65	0.40	0.65	0.15	0.88
SCPS Epsilon=1.1	0.33	0.37	0.70	0.80	0.40	0.72	0.15	0.88
AP	0.16	0.16	0.14	0.15	0.14	0.17	0.16	0.16

Force ranks second in overall performance. It produces results with F-measure greater than 0.80 for two of the unthresholded networks and three of the thresholded networks. The fourth thresholded network, Enolase, scores an F-measure of 0.74 under Force. This is a great improvement over the unthresholded F-measure of 0.43, but is still less than the 0.83 F-measure associated with the MCL clustering of the thresholded Enolase network.

tighter clustering at higher values. By sampling the epsilon parameter along increments of 0.01, we determined that an epsilon value of 1.1 leads to better clustering than the primary epsilon value of 1.0. At epsilon 1.1, SCPS clustering of the thresholded SLC and Kinase networks results in F-measure scores that are equal to or greater than 0.80. The thresholded F-measure for Enolase is 0.72, which is a significant improvement over the unthresholded F-measure of 0.40.

Unfortunately, the Amidohydrolase F-measure remains exceedingly poor, scoring at less than 0.40, even after thresholding.

Affinity Propagation is unable to cluster the four protein networks into functionally meaningful families. All F-measures fall below 0.20. Sampling alternate Affinity Propagation parameters did not yield the improvement.

In order to confidently reconfirm these quantitative observations, we recalculated the clustering table using the Geometric Separation statistic (Brohee and van Helden, 2006) initially developed to compare the quality of protein interaction network clustering approaches. The Geometric Separation results (Supplementary Table S2.3) confirm the conclusions drawn from the F-measure table. The use of an automatically selected threshold improves the Separation statistic, and the thresholded MCL Separation scores rank the highest relative to the other clustering algorithms in the table.

2.4 DISCUSSION

The results indicate that the shape of a protein similarity network edge weight distribution correlates with how well the network clusters over a range of thresholds. It is this relationship between the distribution and clustering potential that allows our simple threshold selection heuristic to improve the quality of clustering results in the variety of networks we studied. This is in contrast to the more complicated approach taken by Harlow et al. in which they performed single linkage hierarchical clustering on MCL results (Harlow et al., 2004). Although these observations are limited to superfamily-based sequence similarity networks of medium size, they nonetheless represent a valuable step in solving the difficult problem of clustering proteins into family groups that may be informative of their different functions. Researchers interested in

clustering larger, more diverse datasets may now efficiently group the data into superfamilies using algorithms like SCPS, and afterwards clustering each superfamily into families with the aid of our threshold selection heuristic.

Our results also show that MCL outperforms other common algorithms in the task of clustering proteins into families, after the appropriate threshold is applied. The Force algorithm ranks second. This is in contrast to previous research (Wittkop et al., 2007), which showed Force outperforming MCL, as indicated by F-measure. Previous performance comparisons have all been carried out on unthresholded networks. The conclusion that Force outperforms MCL is to be expected when network thresholding is not taken into account. Based on our results, when a threshold is not provided, Force outperforms MCL in a network containing a rapid-descent edge weight distribution and performs just as poorly as MCL in a network with a gradual-descent edge weight distribution. However, as we have demonstrated, an appropriate threshold is easy to extract from an input edge weight distribution. Once that threshold is applied, MCL performs as well as or better than Force. By extending both algorithms to include a preliminary automated threshold selection step, the performance difference between the two approaches can be minimized.

Eliminating the performance gap between Force and MCL is an important development because of the large difference in execution times of the two algorithms. As the size of the network increases, the execution time required for running Force goes up significantly (Wittkop et al., 2007). On a modern desktop computer, the Amidohydrolase network takes 5 h to cluster with Force, while MCL clusters the same network in less than 2 min under the same conditions. Given this difference in runtime, and our results that show MCL clustering quality is equal to or better than Force after a heuristically selected threshold is applied, we argue that MCL should be the

algorithm of choice. This choice can be especially important when processing large high-throughput protein similarity datasets. For example, the Amidohydrolase superfamily has more than 20 000 members. Using the current implementation of Force would not be feasible for such a superfamily. Applying heuristically selected thresholding to such a massive dataset allows us to cluster the proteins using the faster-performing MCL algorithm without fear of sacrificing accuracy for the sake of speed.

Finally, our general comparison of biological network clustering approaches illustrates the importance of properly distinguishing between categories of networks prior to selecting an appropriate clustering algorithm. Not all biological networks are equal, and not all network-related problems are equal. Although most of the algorithms we tested showed improvement after thresholding, not all algorithms improved equally. This is because some algorithms are more adept for certain types of problems than for others. SCPS, which was designed to group large sequence sets into superfamilies, clustered reasonably well but did not score as high as the more family-specific MCL and Force algorithms. Affinity Propagation, a general purpose algorithm for clustering nodes in networks, had difficulty in processing protein similarity networks of the scale used here. Thus, it is vital for researchers to proceed with caution before selecting a clustering algorithm appropriate for the problem at hand.

Ultimately, application of sequence similarity networks for functional inference requires clustering results that correspond, to the extent possible, with functionally relevant relationships. A critical step in achieving this goal is automated clustering of sequence similarities without benefit of knowledge about their functional properties. As illustrated here, our approach provides a useful heuristic to improve network clustering in this regard. More

research will be required, however, to better understand the relationship of functional divergence to clustering of sequences using similarity networks.

2.5 CONCLUSIONS

We have examined the role that edge weight distribution plays in network clustering and shown how it may be used to improve the performance of several popular network clustering algorithms. Our automated threshold selection heuristic provides a simple approach for determining an appropriate threshold for network clustering. This threshold may then be employed to eliminate the current gap in clustering quality between MCL and other algorithms, thus alleviating the need to incur the heavily penalty in execution time needed with alternate algorithms such as Force. In addition our results, as shown in Table 1, suggest that thresholding generally improves clustering quality for four out of the five tested clustering algorithms. In the context of using protein similarity networks for functional inference, the significant improvement in clustering quality for these algorithms suggests that any future algorithms designed for this application include a threshold heuristic.

More importantly, our research demonstrates that the predictive potential of the similarity network edge weight distribution is an area of study worth exploring in more detail. Future examination of edge weight distributions may help produce better threshold selection approaches, as well as possibly leading to the development of more accurate network clustering algorithms. Furthermore, additional study of edge weight distribution shape could also provide a deeper understanding of protein similarity networks as a whole.

2.6 SUPPLEMENTARY FIGURES AND TABLES

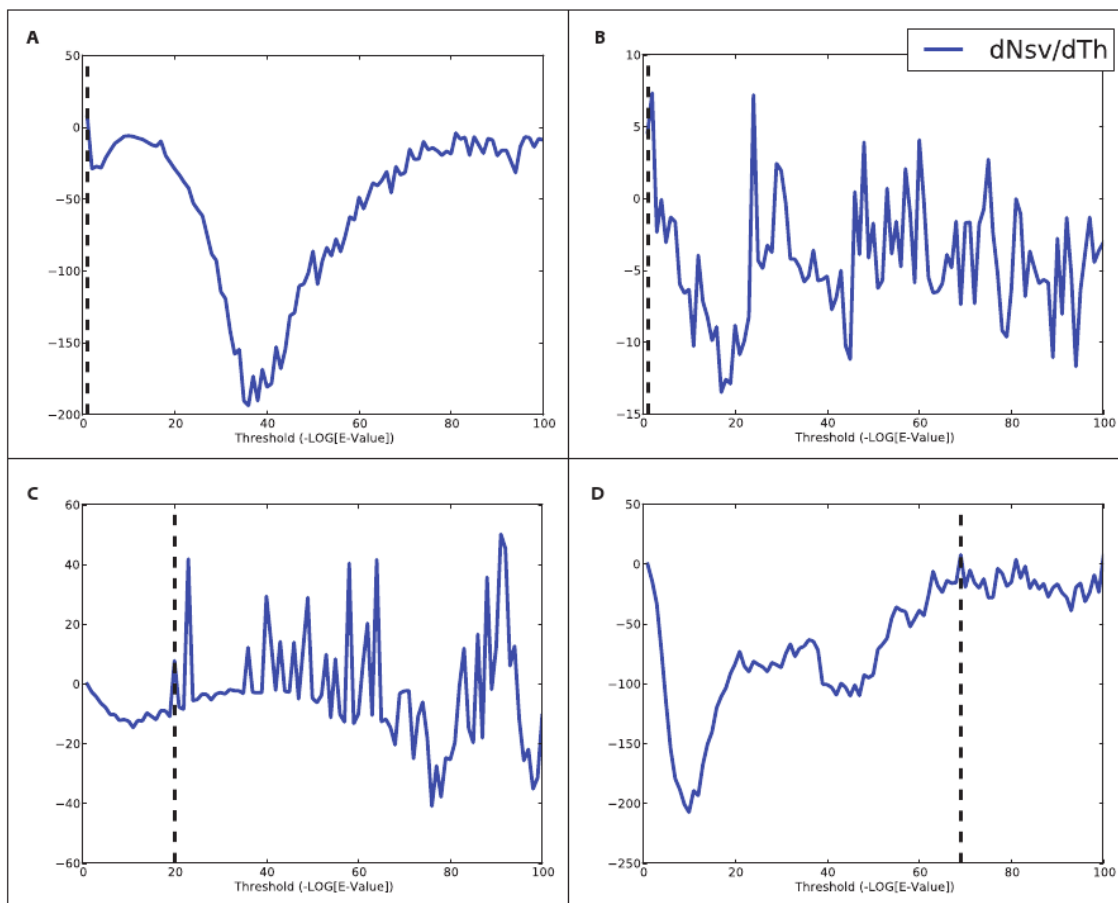


Fig. S2.1. Network summary value (Nsv) from our threshold selection heuristic. Each plot shows the first derivative of Nsv with respect to the threshold. The minimal threshold at which the derivative equals a positive value is marked by a dashed line: (A) Amidohydrolase; threshold of 1.0. (B) SLC; threshold of 1.0. (C) Enolase; threshold of 20.0. (D) Kinase; threshold of 69.0.

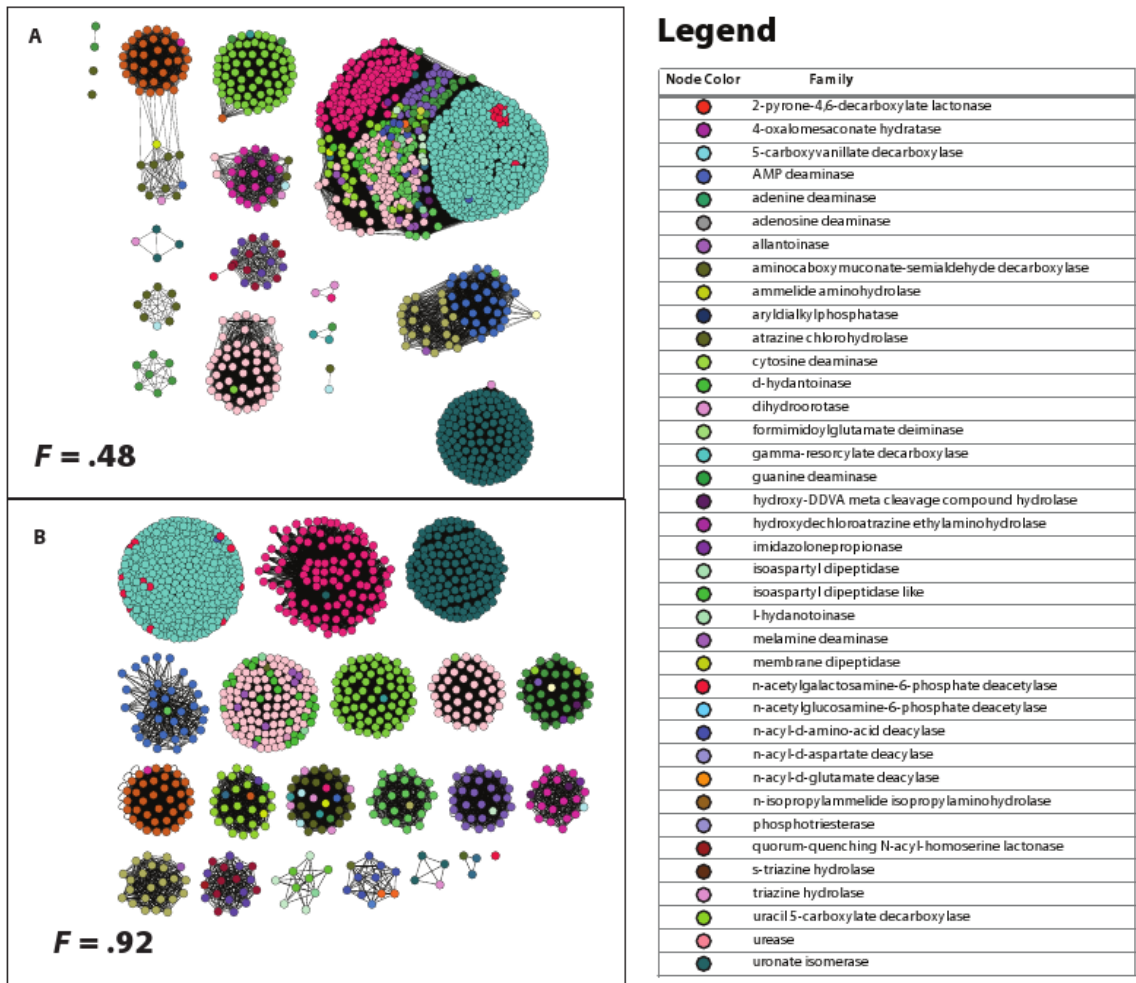


Fig. S2.2. Visualizing MCL clusters for the Amidohydrolase superfamily. Each set of clustering results has been visualized in Cytoscape using the Force-directed layout algorithm. Each node represents a protein, colored by the currently best available family assignments. Edges between nodes that are not in the same cluster have been removed from the similarity network prior to visualization. The unthresholded clustering results are shown in (A) and the thresholded clustering results are shown in (B).

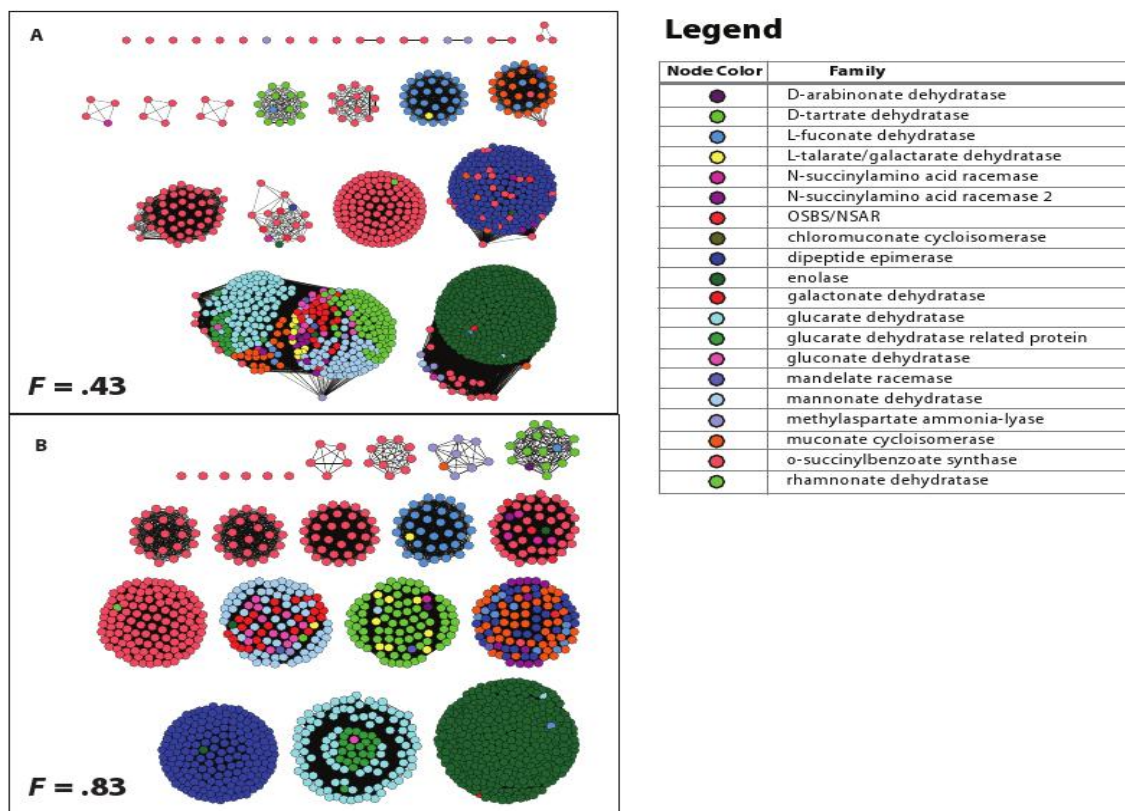


Fig. S2.3. Visualizing MCL clusters for the Enolase superfamily. Each set of clustering results has been visualized in Cytoscape using the Force-directed layout algorithm. Each node represents a protein, colored by the currently best available family assignments. Edges between nodes that are not in the same cluster have been removed from the similarity network prior to visualization. The unthresholded clustering results are shown in (A) and the thresholded clustering results are shown in (B). The same thresholded network is shown unclustered in Supplementary Figure S2.4b. The mapping of node colors to family assignments is shown in the legend to the right of the clusters.

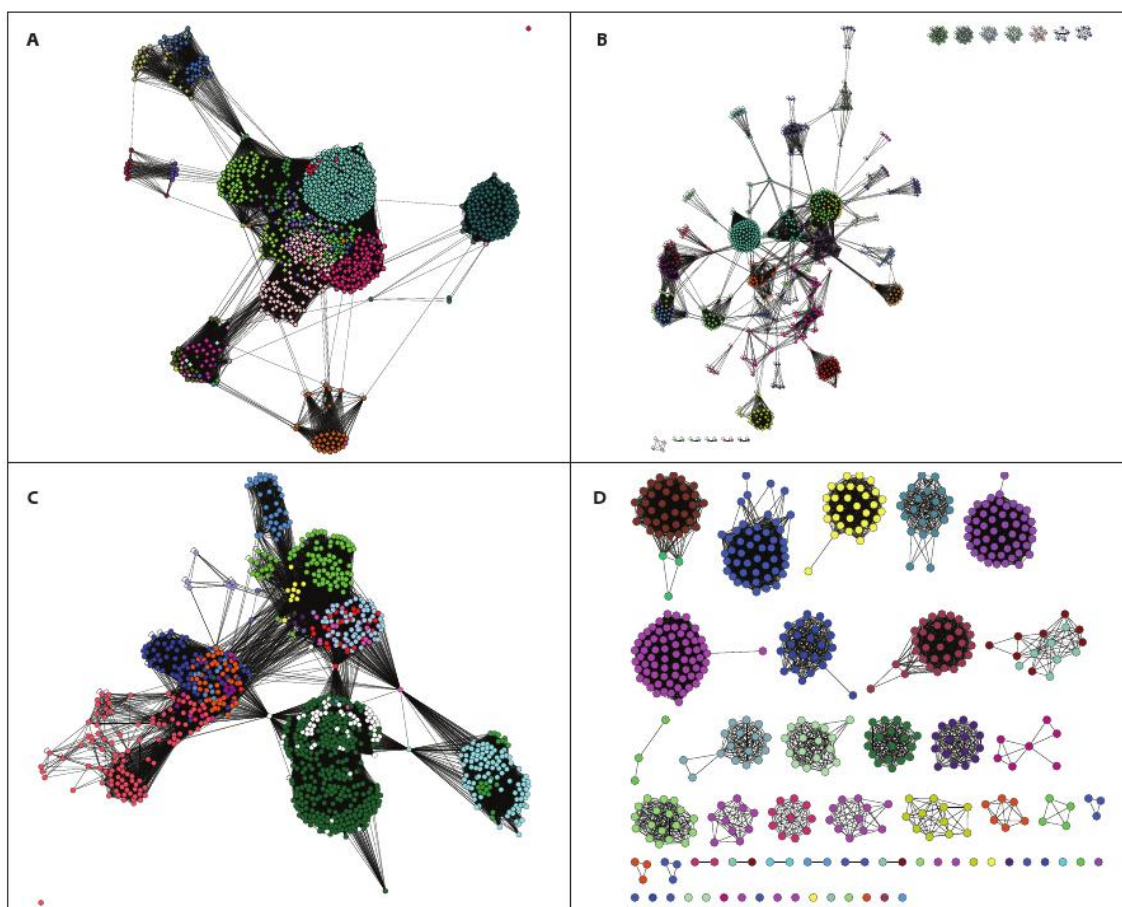


Fig. S2.4. Visualizing the thresholded networks prior to clustering. Applying the automatically selected thresholded filters out noise from each network, making the boundary separations between clusters of families more pronounced after visualization. Nonetheless, three of the four networks remain connected after thresholding, indicating that the post-thresholding process of clustering these networks is not a trivial matter. In the fourth, Kinase network, the clusters separate out completely after thresholding. (A) Amidohyrolase. (B) SLC. (C) Enolase. (D) Kinase.

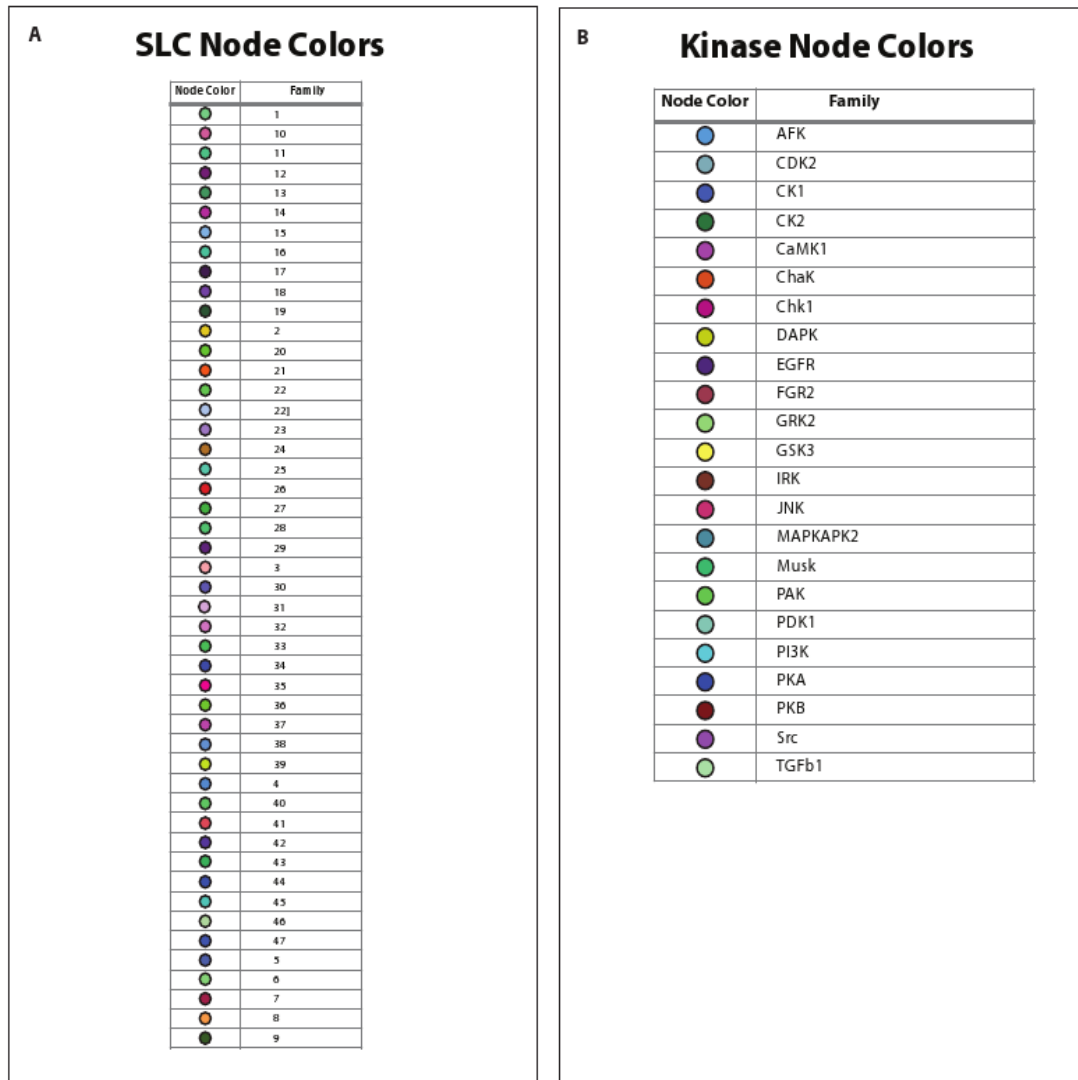


Fig. S2.5. The mapping of node colors to family assignments for the SLC and Kinase superfamilies.

Table S2.1. Exploring inflation parameter for MCL clustering

Inflation_Parameter	Amidohydrolase	SLC	Enolase	Kinase
1	0.43	0.55	0.47	0.14
1.5	0.47	0.55	0.41	0.14
2	0.48	0.57	0.43	0.15
2.5	0.50	0.49	0.46	0.28
3	0.72	0.40	0.48	0.29
3.5	0.62	0.36	0.43	0.37
4	0.47	0.31	0.43	0.37
4.5	0.34	0.29	0.43	0.38
5	0.25	0.26	0.43	0.37

The left-most column lists the MCL inflation parameter, ranging from 1 to 5 (2 is the standard default, shown in bold). The next four columns represent each of the four superfamilies. Each cell in the body of the table contains the F-measure associated with an unthresholded superfamily network that clustered using MCL, with the corresponding inflation parameter. Note that no value of the MCL inflation parameter produces results as high in performance as thresholded networks.

Table S2.2. Average edge-weights, by category, within the superfamily networks

	Average Interfamily Edgeweight	Average Intrafamily Edgeweight
Amidohydrolase	6.74	51.58
SLC	4.22	38.7
Enolase	52	114.93
Kinase	13.83	99.69

The left-most column lists the four superfamilies. The next two columns list the average edge weights between families, and within families for each superfamily network. Note that the average interfamily edge family edge weight for kinase is very small, 13.8, relative to its large intrafamily edge weight. Because edge weights are $-\log(\text{e-value})$ of Blast scores, larger values are more statistically significant.

Table S2.3. Geometric separation scores across clustering algorithms for thresholded and unthresholded superfamilies.

	Amidohydrolase		SLC		Enolase		Kinase	
	U	T	U	T	U	T	U	T
MCL	0.39	0.58	0.47	0.76	0.27	0.44	0.21	0.76
Force	0.44	0.47	0.75	0.75	0.25	0.30	0.31	0.56
TransClust	0.35	0.38	0.78	0.80	0.20	0.35	0.31	0.56
SCPS Epsilon=1.1	0.20	0.32	0.44	0.69	0.25	0.40	0.20	0.80
AP	0.29	0.19	0.25	0.25	0.22	0.24	0.23	0.27

The left-most full column of the table lists the clustering algorithms tested. The next four full columns represent the Geometric Separation scores, as described in Brohee and van Helden, for clustering results across each of the four superfamilies. Each full superfamily column subdivides into two sub-columns; U and T. U represents the Geometric Separation for the clustered, unthresholded superfamily networks. T represents the Geometric Separation for the clustered, thresholded superfamily networks. Geometric separation scores are an alternate means of evaluating relative clustering algorithm performance, but the conclusions drawn from this table are similar to those shown in Table 2.1.

2.7 REFERENCES

- Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
- Apweiler,R. *et al.* (2004) UniProt: the Universal Protein knowledgebase. *Nucleic Acids Res.*, **32**, D115–D119.
- Atkinson,H.J. *et al.* (2009) Using sequence similarity networks for visualization of relationships across diverse protein superfamilies. *PLoS ONE*, **4**, e43-e45.
- Brohee,S and van Helden,J. (2006) Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, **7**, 488.
- Brown,S.D. *et al.* (2006) A gold standard set of mechanistically diverse enzyme superfamilies. *Genome Biol*, **7**, R8.
- Chim,H. and Deng,X. (2007) A new suffix tree similarity measure for document clustering, *In Proceedings of the 16th international conference on World Wide Web*, Association for Computing Machinery, Alberta, Canada, pp. 121-130.
- Enright, A.J. and Ouzounis,C.A. (2001) BioLayout—an automatic graph layout algorithm for similarity visualization. *Bioinformatics*, **17**, 853-854.
- Enright,A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, **30**, 1575–1584.
- Frey,B.J. and Dueck,D. (2007) Clustering by passing messages between data points. *Science*, **315**, 972–976.
- Frivolt,G. and Pok,O. (2006) Comparison of Graph Clustering Approaches. *In Proceedings in IIT.SRC*, Slovak University of Technology, Veliko Turnovo, Bulgaria, pp. 168–175.
- Fruchterman,T.M. and Rheingold,E.M. (1991) Graph drawing by force directed placement. *Softw. Exp. Pract.*, **21**, 1129–1164.

- Gerlt, J.A. *et al.* (2005) Divergent evolution in the enolase superfamily: The interplay of mechanism and specificity. *Arch. Biochem. Biophys.*, **433**, 59–70.
- Glasner, M.E. *et al.* (2006) Evolution of structure and function in the o-succinylbenzoate synthase/N-acylamino acid racemase family of the enolase superfamily. *J. Mol. Biol.*, **360**, 228-250.
- Lu, F. *et al.* (2005) Framework for kernel regularization with application to protein clustering. *Proc. Natl. Acad. Sci.*, **10**, 12332–12337.
- Manning, G. *et al.* (2002) Evolution of protein kinase signaling from yeast to man. *Trends Biochem. Sci.*, **27**, 514–520.
- Noble, W.S. *et al.* (2005). Identifying remote protein homologs by network propagation. *FEBS J.*, **272**, 5119–5128.
- Paccanaro, A. *et al.* (2006) Spectral clustering of protein sequences. *Nucleic Acids Res.*, **34**, 1571-1580.
- Pegg, S.C.H. *et al.* (2006), Leveraging enzyme structure-function relationships for functional inference and experimental design: The Structure-Function Linkage Database. *Biochemistry*, **45**, 2545-2555.
- Ponting, C.P. (2001) Issues in Predicting Protein Function from Sequence. *Brief. Bioinformatics*, **2**, 19-29. Rahmann, S. *et al.* (2007) Exact and heuristic algorithms for weighted cluster editing. *Comput. Syst. Bioinformatics. Conf.*, **6**, 391–401.
- Rodriguez-Esteban, R. (2009) Biomedical text mining and its applications. *PLoS Comput. Biol.*, **5**, e1000597.
- Schaeffer, S.E. (2007) Graph clustering. *Comp. Sci. Review*, **1**, 27-64.
- Schlessinger, A. *et al.* (2010) Comparison of Human Solute Carriers. *Protein Sci.*, **19**, 412-428.

Seffernick, J.L. *et al.* (2001) Melamine deaminase and atrazine chlorohydrolase: 98 percent identical but functionally different. *J. Bacteriol.*, **183**, 2405–2410.

Seibert, C.M. and Raushel, F.M. (2005) Structural and catalytic diversity within the amidohydrolase superfamily. *Biochemistry*, **44**, 6383–6391.

Shannon, P. *et al.* (2003) Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome. Res.* **13**, 2498–2504.

Wittkop, T. *et al.* (2007) Large scale clustering of protein sequences with FORCE -A layout based heuristic for weighted cluster editing. *BMC Bioinformatics*, **8**, 396.

Wittkop, T. *et al.* (2010) Partitioning biological data with transitivity clustering. *Nature Methods*, **7**, 419-420.

Chapter 3

Using Aggregated Network Clustering Techniques to Hypothesize the Functions of Uncharacterized Proteins

A portion of the material in this chapter has been published in
BMC Bioinformatics as:

clusterMaker: A Multi-algorithm Clustering Plugin for Cytoscape

John H. Morris*, Leonard Apeltsin*, Aaron M. Newman*, Jan Baumbach, Tobias
Wittkop, Gang Su, Gary D. Bader, Thomas E. Ferrin

**These authors contributed equally to this work.*

Abstract

Motivation: In the post-genomic era, the rapid increase in high-throughput data calls for computational tools capable of integrating data of diverse types and facilitating recognition of biologically meaningful patterns within them. For example, large protein similarity networks may be clustered in a variety of ways, and proper visualization of the clustering results is a

necessary step to better understanding possible identities of uncharacterized proteins in the networks. Here we present *clusterMaker*, a Cytoscape plugin that implements several clustering algorithms and provides network views of the results.

Results: We analyzed the vicinal oxygen chelate (VOC) superfamily enzyme using the *clusterMaker* plugin. Based on the clustering output, we were able to explore in detail the possible annotation of a protein as a methylmalonyl-CoA epimerase within the VOC superfamily. We also used alignment data to hypothesize the possible identities of other clustered proteins over various grades of likelihood.

3.1 INTRODUCTION

More than 40% of all known proteins lack any annotations within public databases (Jaroszewski et al., 2009) As a result, millions of proteins are completely uncharacterized. Nothing about them is known other than sequence and possibly predicted domain architectures.

Bioinformatics techniques can allow us to filter through this immense collection of unknowns and assign a subset of the proteins some predicted biological characterization. A simple way to carry out large-scale functional prediction is through protein similarity network clustering.

Under this approach, all characterized and uncharacterized proteins are represented as nodes in a network. Edge-weights between nodes reflect the sequence similarity between each pairwise set of proteins. Network clustering algorithms can then organize the network based on predicted functional similarity. Predicted functions may afterwards be assigned to a subset of unknowns that cluster together with functionally characterized proteins.

Of course, clustering all available protein sequence is an exceedingly difficult problem from a computational standpoint. A simpler approach is to provide researchers with the tools to cluster

their individual sequence datasets of interest. This approach is advantageous because large protein databases frequently aggregate an assortment of smaller, more particularized databases from a multitude of research labs. Each individual database might contain no more than a few hundred or a few thousand sequences, tailored to specific laboratory interests. Some of these sequences will likely be unknowns, included because of similarity to other proteins in the set.

There are many algorithms that have been applied to the clustering and categorization of proteins. These include Spectral Clustering of Protein Sequences (SCPS; Paccanaro et al., 2006), TransClust (Wittkop et al., 2010), Markov Clustering Algorithm (MCL; Enright et al., 2002), Affinity Propagation (Frey and Dueck, 2007), and FORCE (Wittkop et al., 2007). Ideally, we would like to cluster input sequence datasets using a subset of these algorithms in order to both quantitatively and visually observe the consistency of the categorization. Unfortunately, there are no tools that provide a convenient platform for linking multiple protein clustering algorithms with a straightforward interface for visualization and analysis. The absence of such tools for hypothesis formulation and protein categorization will grow even more significant as new experimental results and techniques become available.

A promising foundation for development in this area is Cytoscape (Shannon et al., 2003), an open-source, cross-platform software package for visualizing and analyzing biological networks. Cytoscape provides an extensive plugin application programming interface (API) that allows programmers to extend the native capabilities of Cytoscape to provide new functionality. We used this API to implement *clusterMaker*, a plugin that links 10 common clustering algorithms with Cytoscape's built-in data visualization capabilities. While existing clustering approaches have not proven to be sufficient to provide definitive categorization of proteins, these approaches can be extremely useful as initial steps in an overall curation pipeline.

clusterMaker allows researchers and database curators to rapidly cluster their datasets and immediately compare the resulting output through visual analysis. By mapping protein function to visualized node properties, the curator may immediately discern those clusters that include both the unknowns and the functionally characterized proteins. The availability of multiple clustering algorithms allows the curator to assign a greater confidence to those predictions that appear consistently across multiple clustering outputs. Working with *clusterMaker* allows the curator to rapidly generate new functional predictions for immediate public use. This approach can significantly reduce the overall curation timeline, particularly in the early stages of analysis before other approaches such as Hidden Markov Models (HMMs) are applicable.

In this chapter, we examine the use of *clusterMaker* as a curation aid for the Structure-Function-Linkage Database (SFLD – sfld.rbvi.ucsf.edu). The SFLD is a gold-standard resource tool linking sequence information from mechanistically diverse enzyme superfamilies to their characterized structural and functional properties (Pegg et al., 2006). The SFLD provides a three-level classification for proteins: superfamily – proteins that catalyze the same partial reaction, family – proteins that catalyze a unique reaction, and subgroup – a mid-level classification containing multiple families with shared functional residue motifs. All sequences in the SFLD are assigned a superfamily classification, but in numerous cases, family and subgroup assignments remain incomplete. *clusterMaker* allowed us to hypothesize the possible identities of multiple uncharacterized sequences within the SFLD.

3.2 METHODS

3.2.1 Implementation

clusterMaker (<http://www.rbvi.ucsf.edu/cytoscape/cluster/clusterMaker.html>) is implemented as a plugin to the Cytoscape package. *clusterMaker* extends Cytoscape's capabilities by adding implementations of various clustering algorithms and associated visualizations and linking those in an intuitive fashion to the network visualization provided by Cytoscape. *clusterMaker* is entirely written in Java to allow easy portability to any platform supporting the Java virtual machine.

clusterMaker exposes parameters for each clustering algorithm. When a user selects an algorithm, a dialog appears for specifying the node or edge attributes to use for the data source, along with any algorithm-specific parameters, such as the expansion factor for MCL. All of the clustering methods allow selection of a single edge attribute for clustering. For network clustering algorithms this is assumed to be a distance metric. If no attribute is provided, a default distance value of one is assigned to each edge in the network. Each of the ten algorithms provided by *clusterMaker* has been ported into the *clusterMaker* source to provide a consistent user interface and operation.

clusterMaker provides an intuitive visualization for viewing the clustering results of protein similarity networks constructed using any of the following network clustering algorithms: Affinity Propagation, MCL, SCPS, MCODE (Bader and Hogue, 2003), Glay (Su et al., 2010), and TransClust. Each newly constructed network shows only the intra-cluster edges (all inter-cluster edges are dropped). The network is automatically laid out using the Cytoscape force-directed layout. The user may also choose to add the inter-cluster edges back in after the network has been laid out to highlight inter-cluster relationships.

Cytoscape 2.8.1 with *clusterMaker* plugin version 1.8 loaded was used for all of the analyses described here. Cytoscape is available from <http://www.cytoscape.org> and the *clusterMaker*

plugin is available through the Cytoscape plugin manager. *clusterMaker* exports a number of Cytoscape commands to allow other Cytoscape plugins to take advantage of the visualizations and algorithms it provides.

3.2.2 Data Sources

From the 23 available superfamilies present in the SLFD, we have chosen to cluster the vicinal oxygen chelate (VOC) superfamily which comprises a group of metal-dependent enzymes that share a common fold motif and catalyze a variety of reactions (Armstrong et al., 2000). This superfamily is a particularly difficult superfamily to discriminate specific functions due to multiple, perhaps serial permutations and other rearrangements in its evolutionary history (Babbitt, 2011).

The VOC superfamily dataset was composed of 10,437 protein sequences, partially classified among along seven subgroups and 17 families. Less than half of these sequences included both a family and subgroup classification. 224 sequences contained a subgroup classification but not a family classification. The remaining 168 sequences were completely uncharacterized.

3.2.3 Protocol

The SFLD data analysis interface allowed for the immediate importing of the VOC superfamily into Cytoscape using the “download network” button in the “Sequence Similarity” sub-section of the toolbox menu with an e-value cutoff of $1e^{-1}$. Nodes in the network represent individual proteins, with family and subgroup classifications already specified among the properties of the nodes. Edges in the network represent protein similarities based on the BLAST e-values of each pairwise sequence alignment.

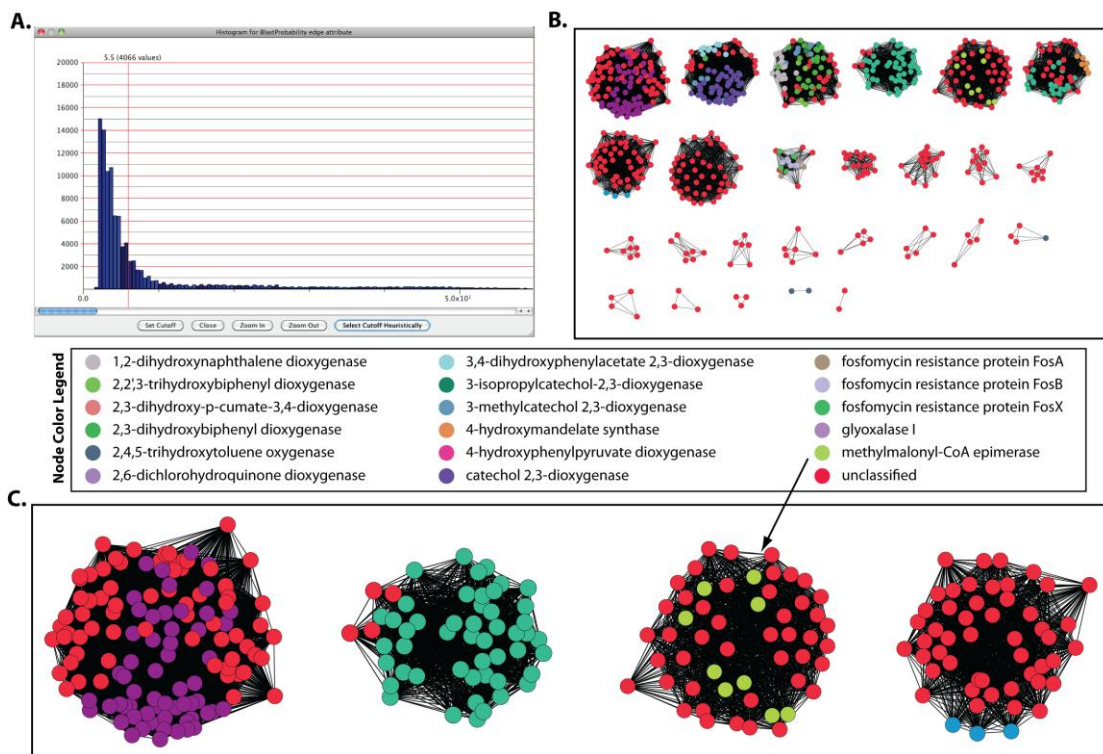


Fig. 3.1. Protein similarity network clustering indicates possible family membership for uncharacterized proteins. (A) A distribution of edge weights (binned $-\log(E\text{-values})$) of the VOC superfamily is shown, with a cutoff value of 5.5 indicated by a red vertical line. The cutoff was determined by a heuristic described in the previous chapter and was used for subsequent clustering. (B) MCL clusters for the VOC superfamily are displayed with nodes colored by family assignment. Red nodes represent proteins with unknown function. (C) Four clusters within the MCL clustering results show only proteins from a single family or proteins of unknown function. These clusters are easily distinguished from all other MCL clustering results in B. Three of these four clusters also appear in the TransClust results.

After we loaded the network, we used *clusterMaker* to heuristically generate a cutoff prior to clustering. *clusterMaker* implements an automated cutoff selection heuristic, which selects a cutoff based on properties of the network edge weight distribution (Figure 3.1A). This heuristic, described in detail in the previous chapter, has been shown to improve the accuracy with which a protein similarity network gets clustered into families (Apeltsin et al., 2011). A heuristically determined cutoff value of 5.5 was used for all our clustering runs here.

We ran three clustering algorithms on the VOC dataset: MCL, TransClust and SCPS. No initial parameters were altered other than the number of MCL iterations, which we raised from eight to 15. Clustering outputs were then visualized by coloring each node based on the known family assignments for each enzyme. This allowed us to immediately pick out those clusters which were composed of a single characterized family and multiple uncharacterized nodes. We then compared the presence of such clusters across all of our clustering results.

3.3 RESULTS

Prior to visualizing the clustering results, we examined the number of clusters returned by each algorithm. MCL generated 26 clusters and TransClust generated 28 clusters. These numbers adequately approximated the presence of 17 distinct families in 50% of the VOC dataset. SCPS on the other hand, generated only four clusters, which indicated an overabundance of false positives in the SCPS clustering data. We therefore disregarded the SCPS clusters and focused our comparison on the MCL and TransClust clustering results.

The TransClust and MCL results (Figure 3.1B) are dominated by uncharacterized proteins (colored red in the figure). Certain clusters are composed entirely of uncharacterized proteins, which makes it impossible to hypothesize their function solely from this data. Other clusters are

composed of uncharacterized proteins as well as two more families, as indicated by two or more additional colors in the nodes. These heterogeneous clusters also give little indication as to which families to assign to uncharacterized proteins and suggest that sequence information alone is not enough of a discriminant to functionally assign these proteins. The most interesting clusters contain just two colors, representing the grouping of uncharacterized proteins with a single VOC family. These clusters allow us to hypothesize the identity of the uncharacterized proteins.

Three such single-family clusters are present in almost equal measures across both the TransClust and MCL results (Figure 3.1C), one of which is the methylmalonyl-CoA epimerase subgroup of 50 proteins (see arrow in Figure 3.1C). This includes the nine characterized members of the methylmalonyl-CoA epimerase family and 41 sequences that lack a family classification in the SFLD, although they are in the same subgroup. The size of the cluster is 52 in the TransClust results and 53 in the MCL results. The additional few nodes represent sequences lacking a subgroup classification and that appear in both the TransClust and MCL results, suggesting that putatively assigning these to the methylmalonyl-CoA epimerase subgroup would be reasonable.

In an effort to seek out additional evidence of family and subgroup membership, we explored in some detail one of the uncharacterized proteins within the methylmalonyl-CoA epimerase cluster. The hypothetical (predicted) protein BH2212 from *Bacillus halodurans* (gi:15614775) lacks both a family and subgroup assignment. We aligned its sequence with that of methylmalonyl-CoA epimerase from *Propionibacterium shermanii* (gi:15826388). Four of the five functionally critical active site residues align perfectly with the uncharacterized sequence. These four residues bind the active-site metal ion needed for catalysis. In the initial alignment,

Table 3.1. HMM alignments of clustered uncharacterized proteins to critical active site residues within the families and subgroups into which they cluster

Gi Number	Family Cluster	Subgroup Residues Align	Family Residues Align
15895460	Methylmalonyl-CoA epimerase	Yes	No
23099316	Methylmalonyl-CoA epimerase	Yes	No
15614775	Methylmalonyl-CoA epimerase	No	No
13473208	2,6-dichlorohydroquinone dioxygenase	No	Yes
27365357	Glyoxalase I	Yes	Yes
21243096	Glyoxalase I	Yes	Yes
23011551	Glyoxalase I	Yes	Yes
15902908	Glyoxalase I	Yes	Yes
30021288	Glyoxalase I	Yes	Yes
48825814	Glyoxalase I	Yes	Yes
29346990	Glyoxalase I	Yes	Yes
21401095	Glyoxalase I	Yes	Yes
7488556	Glyoxalase I	No	No
27886881	Glyoxalase I	Yes	Yes
19703698	Glyoxalase I	Yes	Yes
15806698	Glyoxalase I	No	No
22958029	Glyoxalase I	Yes	Yes
29841036	Glyoxalase I	No	No
15790201	Glyoxalase I	Yes	Yes
27382600	Glyoxalase I	Yes	Yes
29831945	Glyoxalase I	No	No
23121587	Glyoxalase I	Yes	Yes

17552228	Glyoxalase I	No	No
15889727	Glyoxalase I	Yes	Yes
15900839	Glyoxalase I	Yes	Yes
16080888	Glyoxalase I	Yes	Yes
6625562	Glyoxalase I	No	No
27380436	Glyoxalase I	No	No

The first column identifies the uncharacterized protein. The second column indicates the family into which it clusters. The third column indicates whether or not the critical active site residues in the subgroup all align with the unknown. The final column indicates whether or not the critical active site residues in the family align with the unknown.

the fifth residue, a glutamic acid that abstracts a proton from the substrate, is shifted by one position, but minor editing can align it as well without degrading the rest of the alignment. Thus, the unknown protein is most likely capable of binding the active site metal and may also perform the epimerization of (2R)-methylamonyl-CoA.

As part of a wider analysis, we aligned each uncharacterized protein with all sequences from the family and subgroup into which it clustered. These alignments were carried out using the built-in alignment functionality of the SFLD, in which precomputed hidden Markov models (HMMs) help ensure the accuracy of the final alignment. For every uncharacterized sequence, we recorded whether or not the HMM alignment with the family and subgroup resulted in a complete overlap among the functionally critical residues (Table 3.1), as defined in the SFLD. For 19 uncharacterized sequences, the overlap between functionally critical subgroup residues was

perfect. 17 of these sequences also included a perfect overlap with the functional critical residues in the clustered family. All of these 17 sequences clustered with the glyoxalase I family.

3.4 DISCUSSION

Our results indicate that it may well be possible to use clustering algorithms in combination with alignment techniques in order to rank the likelihood with which an unknown protein might perform a particular function relative to other uncharacterized proteins in the dataset. An uncharacterized protein that falls within a family cluster may be more likely perform the family's function than a protein which clusters outside of the family. A uncharacterized protein that falls into a family in both the TransClust and MCL results is also likely a better candidate for characterization than a protein which clusters solely within the TransClust results. If the family's functionally critical residues align well with the clustered unknown, it further increases the likelihood of the protein's characterization into that family. By this last standard, 17 of the unknown proteins that cluster within the glyoxalase I family are the most likely candidates for categorization explored in our study.

However, in order to definitively validate the hypothesized functions of the clustered uncharacterized proteins, experimental testing is necessary. The clustering techniques discussed in this chapter are excellent hypothesis generation tools, but the correlation between various granularities of clustering results and actual functional likelihood has not yet been rigorously studied from a statistical standpoint. Future work will focus on not only confirming the VOC superfamily clustering results, but also exploring the interplay between clustering, alignment and function across other superfamily datasets. In the meantime, researchers may use the *clusterMaker* plugin to guide the selection of appropriate functional testing techniques in their efforts to more efficiently characterize large protein datasets.

3.5 CONCLUSIONS

clusterMaker is an important addition to the suite of Cytoscape plugins. The protein similarity clustering algorithms provided by the plugin allow for easier curation of large protein datasets. Our application of *clusterMaker* to the VOC superfamily demonstrated how the plugin may be used to potentially categorize new proteins whose function is not yet known. While all such categorization efforts must eventually be tested experimentally, *clusterMaker* nonetheless offers a valuable tool for hypothesis generation in the data curation process.

3.6 SUPPLEMENTARY FIGURES

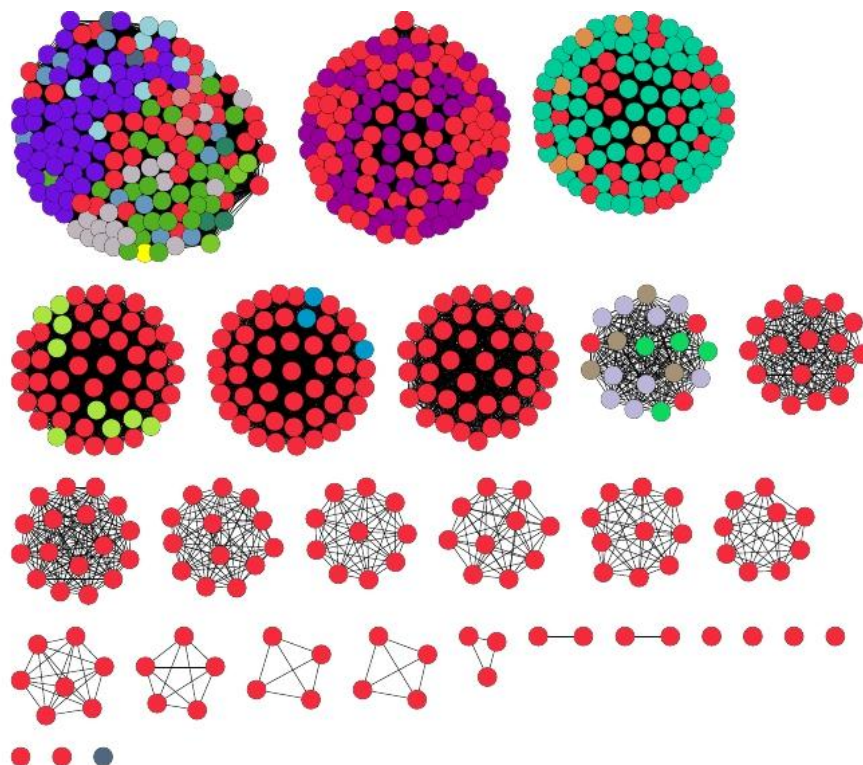


Fig. S3.1. Results of clustering the VOC superfamily using *clusterMaker*'s Transitivity Cluster implementation.

3.7 REFERENCES

- Apeltsin, L. *et al.* (2011) Improving the quality of protein similarity network clustering algorithms using the network edge weight distribution. *Bioinformatics*, **27**, 326–333.
- Armstrong, R.N. *et al.* (2000) Mechanistic diversity in a metalloenzyme superfamily. *Biochemistry*, **39**, 13625–13632.
- Babbitt, P.C. (2011) Exploring the VOC superfamily. *In Submission*.

- Bader,G.D., and Hogue,C.W. (2003). An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, **4**, 2.
- Enright,A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, **30**, 1575–1584.
- Frey,B.J. and Dueck,D. (2007) Clustering by passing messages between data points. *Science*, **315**, 972–976.
- Jaroszewski,L. *et al.* (2009) Exploration of uncharted regions of the protein universe. *PLoS Biol.*, **7**, e1000205.
- Morris,J.H. *et al.* (2011). clusterMaker: A multi-algorithm clustering plugin for Cytoscape. *BMC Bioinformatics*, **12**, 436.
- Paccanaro,A. *et al.* (2006) Spectral clustering of protein sequences. *Nucleic Acids Res.*, **34**, 1571-1580.
- Pegg,S.C.H. *et al.* (2006) Leveraging Enzyme Structure-Function Relationships for Functional Inference and Experimental Design: The Structure-Function Linkage Database. *Biochemistry*, **45**, 2545-2555.
- Shannon,S. *et al.* (2003) Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Research*, **13**, 2498-2504.
- Su,G. *et al.* (2010) GLay: community structure analysis of biological networks. *Bioinformatics*, **26**, 3135-3137.
- Wittkop,T. *et al.* (2007) Large scale clustering of protein sequences with FORCE -A layout based heuristic for weighted cluster editing. *BMC Bioinformatics*, **8**, 396.
- Wittkop,T. *et al.* (2010) Partitioning biological data with transitivity clustering. *Nature Methods*, **7**, 419-420.

Chapter 4

A Network Filtration Protocol for Elucidating

Relationships between Families in a Protein

Similarity Network

Abstract

Motivation: The study of diverse enzyme superfamilies can provide important insight into the relationships between protein sequence, structure and function. It is often challenging, however, to discover these relationships across a large and diverse superfamily. Contemporary similarity network visualization techniques allow researchers to aggregate sequence similarity information into a single global view. Network visualization provides a qualitative estimate of functional diversity within a superfamily, but is unable to quantitate explicit boundaries, when present, between neighboring families in sequence space. This limits the potential of existing sequence-based algorithms to generate functional predictions from superfamily datasets.

Results: By building on current network analysis tools, we have developed a new algorithm for elucidating pairs of homologous families within a sequence dataset. Our algorithm is able to filter through a dense similarity network in order to estimate both the boundaries of individual families and also how the families neighbor one another. Globally, these neighboring families define a topology across the entire superfamily. The topology is simple to interpret by visualizing

the network output generated by our filtration protocol. We have compared the network topology within the kinase superfamily against available phylogenetic data. Our results suggest that neighbors within the filtered kinase network are more likely to share structural and functional properties than more distant network clusters.

4.1 Introduction

Some homologous but highly divergent sets of proteins have evolved to perform substantially different molecular functions. These include a wide range of membrane transporters (George et al., 2004) as well as mechanistically diverse enzyme superfamilies (Pegg et al., 2006).

Mechanistically diverse enzyme superfamilies are sets of evolutionarily related proteins with similar structural and functional properties. All members of such superfamilies share the same structural scaffold and use a conserved subset of active site residues that can be associated with an underlying aspect of catalysis, often a partial reaction (Babbitt and Gerlt, 1997), (Babbitt and Gerlt, 2001). A superfamily can further be subdivided into individual families. Each family catalyzes a unique overall reaction which, together with a distinct set of catalytic residues, differentiates it from all the other families in the set (Pegg et al., 2005). Each individual family within a superfamily can usually be further differentiated by its substrates and products.

Given a superfamily with a few hundred or more protein sequences, it would be valuable to summarize how families within the superfamily relate to one another. More specifically, we would like to extract individual families from the dataset and determine which pairs of families share the strongest degree of functional similarity with each other. We restrict ourselves here only to sequence information because it is widely available and provides us access to large amounts of data. Aggregating these pairs of “neighboring” families allows us to define a

topology that in some cases can be associated with functional transitions within the superfamily and this, in turn, is helpful in predicting the function of previously uncharacterized sequences.

Determining superfamily topology without first knowing the identities of the protein families in the data set is not an easy task. It requires us to calculate boundaries in sequence space based solely on sequence similarity while keeping in mind that the relationships between sequence, structure, and function within a protein superfamily are complex and far from clear. Two closely related superfamily members may share nearly identical sequences, with a few amino acids accounting for the different functions they perform (Seffernick et al., 2001). More divergent families within a superfamily may still share a similar structure in which at least the active site residues associated with the superfamily-common partial reaction are conserved despite sharing a low level of sequence identity (Brenner et al., 1998), (Glasner et al., 2006). Consequently, we are unable to draw reliable conclusions from local sequence-sequence comparisons.

Fortunately, when we aggregate all local sequence comparisons into large-scale protein similarity networks, the results we obtain are much more informative (Enright and Ouzounis, 2000) although they typically lack sufficient resolution to detect topological boundaries between neighboring families. The approach we describe here builds on available similarity network analysis techniques to design a process for identifying topological boundaries in a given superfamily sequence set.

Much of the current research in the field of sequence similarity network analysis has focused on qualitative analysis based on network visualization. Tools such as BioLayout (Enright and Ouzounis, 2001) and CLANS (Fickey and Lupas, 2004) are able to take an all-by-all BLAST-scored (Altschul et al., 1997) network associated with a set of protein sequences and output a visual representation of that network in two-dimensional and three-dimensional space, respectively.

They do this by employing the Fruchterman-Reingold force-directed layout algorithm (Fruchterman and Reingold, 1991), which models the network as a physical network in Euclidian space. The algorithm places the nodes in the network into visually discernible clusters whose distance to one another is a function of their connectivity and BLAST scores. These groups might represent subsets within a monofunctional family, or a collection of strongly related families. Individual groups close to each other in Euclidian space may represent functionally separable families that are nonetheless very similar to one another. The groups that are far apart due to little or no direct connectivity (as can be captured using BLAST as a comparison tool) are a result of sequence divergence. As the distance between groups increases, the degree of functional overlap between proteins represented in the network decreases (Adai et al., 2004). By visualizing these spatial properties of a network, we obtain a reasonable global representation of all sequence data within a superfamily.

While network visualization is a useful tool for hypothesis generation, it does not always accurately define a topology between functional classes of proteins within a superfamily. In order to improve the definition of topology, it is first necessary to delineate boundaries between all distinct pairs of neighboring families in a manner that best approximates functional differences. Visualization constrains us to label these boundaries using cluster distributions in two or three-dimensional space. The network itself, however, is a multidimensional object. If we do not know in advance the distribution of network variance across all possible dimensions we run the risk of inferring the incorrect topology based on statistically insignificant distances (Vlachos et al., 2002).

Even if we ignore the issue of dimensionality, we are still unable to accurately determine topology between functionally distinct protein clusters using just the visual representation of a

network. As the number of edges between neighboring clusters increases in large networks, the visual representation deteriorates. With the Fruchterman-Reingold force-directed layout algorithm noted above, clusters are drawn towards one another by the attractive force proportional to the number of connecting edges. Eventually, the proximity between the clusters blurs the spatial border between them and multiple clusters merge into a single large cluster. The user of the network visualization tool is then left with an incomplete representation of the topological relationships between families in the dataset.

Little previous work has been done to address the issue of visual complexity resulting from excess edges in similarity networks. One current approach is to select a threshold and remove all edges with weights below the threshold (Medini et al., 2006). The threshold is manually adjusted until a value is reached that eliminates many redundant edges while maintaining network connectivity. This approach falls short, however, because not all clusters in the network share equal connectivity. When using a threshold suitable for a majority of edges in the complete network, clusters of outliers connected to the core of the network with very low edge weights may break away, or clusters with multiple poorly weighted connections will disintegrate. It is therefore difficult to maintain network connectivity, which is needed to determine topology, while filtering edges using only a single threshold value.

The goal of the work we describe here is to develop a better filtration approach that maintains network connectivity while highlighting both individual clusters and the topological relationships between them. To do so, we focus on a quantitative analysis of the clusters within similarity networks. The automated clustering of proteins into families based solely on connectivity within protein similarity networks is an expanding area of research. Building on graph theory-based network clustering techniques (Frivolt and Pok, 2006), algorithms such as TribeMCL (Enright et

al., 2002) and RANKPROP (Noble et al., 2005) attempt to isolate tightly integrated sets of nodes using criteria such as edge density and edge weights. These algorithms are parameterized to classify protein sequences into unique families using alignment-based protein similarity networks. The clusters they compute are likely to correspond with spatial clusters of nodes that aggregate together in a force-directed layout, but because the clustering is not based on the spatial proximity of nodes, dimensionality is not an issue.

By clustering the nodes in the network, followed by further analysis, we are able to achieve an effective filtration protocol for reducing the number of edges within a network and elucidating the topology in the associated data set. We accomplish this by first clustering the network into sets of tightly connected components. All edges outside the clusters are then removed from the network, leaving isolated clusters. Next, the clusters are reconnected by reinstating a small set of best-scoring edges between nodes in different clusters. Edges added back into the network represent the boundary between pairs of neighboring clusters and define the topological structure of sequences in the dataset. Finally, we visualize the filtered network using a force-directed layout. The layout of the filtered network qualitatively highlights the topology spanning the clusters, allowing for more intuitive hypothesis generation.

4.2 Methods

4.2.1 Outline of the Network Filtration Protocol

Given an all-by-all BLAST-scored protein similarity network, we want to filter it such that individual families within the network fall into obviously distinguishable clusters and that the sequences most optimally connecting the separate clusters are visible within the network. The protocol for accomplishing this can be summarized as follows:

1. Compute an all-by-all protein similarity network using BLAST;
2. Cluster the nodes in the network and remove all edges that do not connect two nodes in the same cluster;
3. Reconnect the clusters using the minimum number of reasonably weighed edges;
4. Visualize the network using a force directed layout algorithm.

4.2.1.1 Computing the Similarity Network

For any input data set, we carry out an automated BLAST search for every sequence in either the NCBI NR database using default parameters, or a custom database built from selected input sequences. Although skewed expectation values result from running a custom BLAST search compared to running a search against the much larger NCBI NR database, this skew is unimportant relative to the topology of the network itself. The BLAST expectation value (e-value) cutoff for each search is set to one in order not to miss possible connections, although this e-value does not represent a statistically significant match.

Each protein is treated as a node in the similarity network. Whenever a BLAST alignment is returned between two proteins in the data set, we connect these proteins with an edge. Each edge is given a weigh equivalent to the $-\log$ of the BLAST e-value.

4.2.1.2 Clustering the Network

After computing the similarity network, we carry out clustering using techniques discussed in Chapters Two and Three. We prefilter the network and run MCL in order to cluster the nodes into families. MCL is our algorithm of choice because of its speed and its reliability when a threshold is applied.

4.2.1.3 Reconnecting the Clusters

After we have isolated the clusters, our goal is to reconnect these clusters using a minimal subset of edges from the original all-by-all network. We strive to reconnect clusters by maximizing the connectivity between closely related clusters while minimizing the presence of redundant edges. We accomplish this by computing edges from all possible minimum spanning trees (Prim, 1957) connecting all clusters, using a modified version of Kruskal's algorithm (Kruskal, 1956). These edges, defining the topology between clusters, are added back into the network. Edge weights are rounded to integer values when computing all minimum spanning trees to help address the noisy nature of BLAST e-values. The detailed procedure for our cluster reconnection algorithm is as follows:

1. Create an empty graph list gL and an empty edge list eL . Go to step 2.
2. For each cluster X outputted by tribeMCL, create a graph gX such that all edges from the original unfiltered network connecting the nodes in X are present in gX . Add gX to list gL . Go to step 3.
3. Select all intercluster edges from the unfiltered network that are not present in any graph gX in gL . Add these edges to eL . Go to step 4.
4. Sort edges in eL from largest to smallest edge weight. Go to step 5.
5. If the length of eL is zero, return all nodes and edges present in gL . This is the final filtered network. Otherwise go to step 6.

4.2.1.4 Visualizing the Network

We visualize the final filtered network in Cytoscape (Shannon et al., 2003), an open-source Java-based program originally designed to display protein-protein interaction networks. Cytoscape

allows users to assign multiple attributes to the nodes and edges of a given network and then map a set of colors to these attributes. For example, those nodes that represent functionally categorized proteins can be assigned a color based on their family identity. Edges can also be assigned a color based on whether or not they connect nodes from neighboring clusters, as well as on the statistical significance of the corresponding edge weight. The final network is then displayed using Cytoscape's "organic" layout, a force-directed layout algorithm available within the "yfiles" plugin and a standard part of the Cytoscape distribution.

4.2.2 Data Set Selection

4.2.2.1 Designing the Protocol

In order to design our filtration protocol, we used a gold standard collection of manually annotated sequences (Brown et al., 2006) from the enolase superfamily (Babbitt et al., 1996). We downloaded 681 enolase sequences from the Structure Function Linkage Database (SFLD) (Pegg et al., 2006). We used this dataset for the development of our protocol because it represents a highly divergent superfamily in which families evolve at variable rates. All edge-weights were derived using the NCBI NR database.

4.2.2.2 Testing the Significance of the Generated Network Topologies

It has long been established that evolutionary proximity corresponds to structural and functional similarity (Perutz et al., 1965). Protein families rooted directly from the same branch point in a phylogenetic tree share a higher degree of similarity than families that are not. With this axiom in mind, we decided to compare how network topology relates to evolutionary branching in a well-studied phylogenetic tree. Our goal was not to correlate topology with evolution, but

rather to examine the manner in which protein structural and functional similarities could be inferred from a network.

To generate a test dataset, we focused on the kinase superfamily (Manning et al., 2002). In a recent study (Scheeff and Bourne, 2005), the phylogenetic tree for the kinases was generated using rigorous stochastic optimization (Ronquist and Huelsenbeck, 2003) that incorporates both sequence and structural information. The resulting tree encompasses the evolutionary history of 21 kinases, each from a unique family. The families divide into nine different kinase functional classes. We searched for these families in the KinBase [<http://kinase.com/kinbase/>] and KinaseNet [<http://www.kinaset.net>] kinase sequence databases. Thirteen of the families were found in one or both of the databases. These families encompassed all nine classes, and encapsulated a total of 527 sequences. We used the sequences to generate a filtered network representation of the kinase superfamily. All edge-weights were derived using a custom database, rather than the NCBI NR database, for the purpose of quicker computation.

4.3 RESULTS

4.3.1 Visualizing the Topology in the Enolase Superfamily Network

We compared the unfiltered enolase superfamily network from our development dataset to the network output by our filtration protocol. Figure 4.1 shows the unfiltered network. Each node is labeled a distinct color based on the carefully curated family assignment contained in the SFLD.

The color coding of nodes in Figure 4.1 makes clear the strong presence of family based clusters within the similarity network. The boundaries between the clusters, however, are generally not clear. It is also difficult to see, through a purely qualitative analysis, how the families transition from one cluster to another. Furthermore, while we observe the presence of certain separate

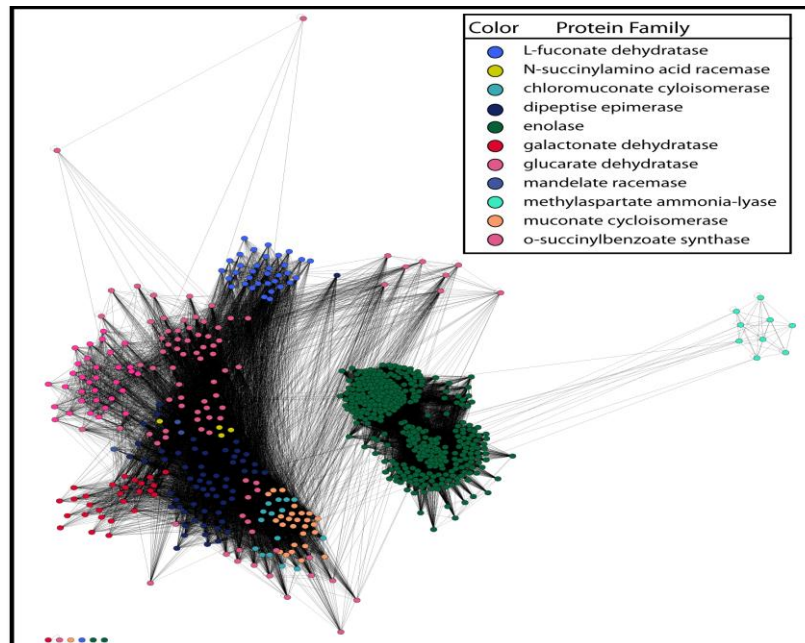


Fig. 4.1 Unfiltered Enolase similarity network. Edge-weighted force-directed representation for the pairwise BLAST similarities in the enolase superfamily. Nodes of the same color group together in two-dimensional space, but it is difficult to distinguish which nodes are responsible for the transition between neighboring families. Certain large spatial clusters are composed of nodes belonging to multiple families. While the nodes in a given family do tend to co-locate, this is only discernible due to their shared color scheme. This would not have been visible if the identities of these families were not known prior to generating the network.

clusters due to the color coding based on characterized family assignment, a researcher visualizing a previously uncharacterized superfamily for which high quality annotation is unavailable would likely be unable to distinguish between adjacent components of the network.

The corresponding filtered network is shown in Figure 4.2A and demonstrates the final output of our network filtration protocol. The enolase superfamily has been separated into clearly distinguishable components by our protocol, corresponding, for the most part, to known protein families. These components are connected by edges that designate pairs of components as neighbors. Edge color defines how closely the components neighbor one another, with the least significant edges shown in blue. The topological relationships between components are easy to detect by direct inspection of the network layout.

The overall connectivity of the family relationships within the enolase superfamily fail in some cases to reflect relationships inferred from highly curated observations derived using experimental methods. For example, the blue edges shown in Figure 4.2A connecting the OSBS/NSAR and the enolase [family] cluster reflect e-values that range from $10^{-0.29}$ - $10^{-0.74}$. Because they are both of low statistical significance and highly complex, the most difficult of the family relationships to capture for this superfamily are those relating the families in the muconate lactonizing enzyme (MLE) subgroup.

4.3.2 Structure, Function, Topology and Evolution in the Muconate Lactonizing Enzyme Subgroup

4.3.2.1 Introduction to the Muconate Lactonizing Enzyme Subgroup

The MLE subgroup is a well-studied subset of the enolase superfamily (Glasner et al., 2006). Our enolase dataset represents six catalytic reactions from the MLE subgroup. These include muconate cycloisomerase (MLE I), chloromuconate cycloisomerase (MLE II), Dipeptide epimerase (DipEp), N-succinylamino acid racemase (NSAR), and o-succinylbenzoate synthase (OSBS). Proteins in the OSBS family are particularly difficult to classify because they are highly

divergent. Some members share less than 15% pairwise sequence identity with other members of the family. Additionally, certain OSBS enzymes are capable of catalyzing both OSBS and NSAR reactions (Palmer et al., 1999; Sakai et al., 2006). Despite this divergence and promiscuity, careful phylogenetic analysis has revealed that members of the OSBS family (including the OSBS/NSAR enzymes) are monophyletic and more closely related to one another than they are to other families in MLE subgroup (Glasner et al., 2006).

We wanted to explore how this messy interplay of sequence, structure, function, and evolution within the MLE subgroup correlates with network topology. We therefore examined in more detail the topology of the subgraph in the filtered enolase network corresponding to the MLE subgroup (Figure 4.2B).

4.3.2.2 Clustering the MLE Subgroup

The MLE I and MLE II families, which catalyze very similar isomerization reactions, group together in a single cluster. The DipEp family is split across four clusters of sizes one, 11, 15, and 32, respectively. As expected, the divergent OSBS family was distributed across multiple clusters of various sizes. Seven clusters were composed of only a single node. Five clusters each contained between three and eight nodes. The remaining three clusters contained between 12 and 27 nodes.

One of the OSBS clusters includes several proteins annotated as NSARs in the SFLD. Three of these have been experimentally characterized and are promiscuous for both OSBS and NSAR activities (Sakai et al., 2006). The functions of the other proteins annotated as NSAR or OSBS in this cluster have not been experimentally determined, but phylogeny and comparative genomics suggest that while some are physiologically required for OSBS activity, others are more likely to function as NSARs in the cell (Glasner et al., 2006).

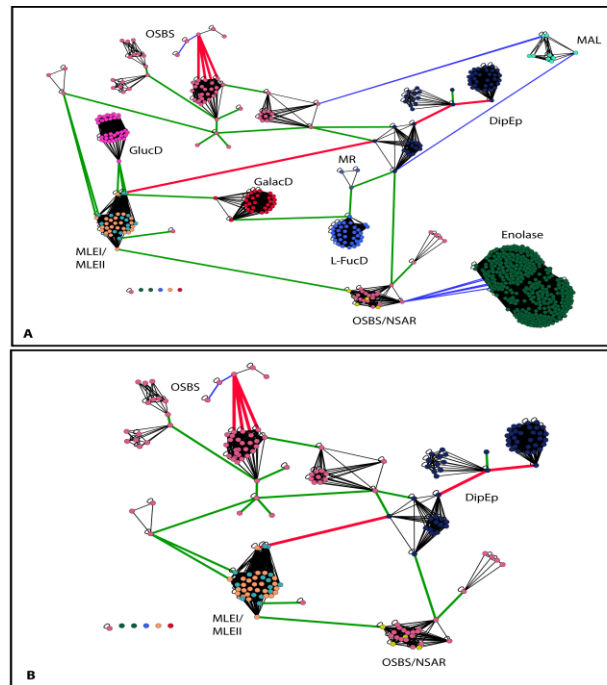


Fig. 4.2 Enolase similarity network. (A) Unweighted force-directed layout representations of the enolase similarity network after processing with our filtration protocol. Edges between nodes in the same cluster are colored black. Edges connecting nodes from neighboring clusters are colored blue, red, and green, based on edge weight. Blue edges have an edge weight of less than 10 (e-value $> 1 \times 10^{-10}$). Green edges have an edge weight between 10 and the prefiltering threshold (33, corresponding to an e-value = 1×10^{-33}). Red edges have an edge weight greater than the threshold (e-value $< 1 \times 10^{-33}$). Parts of the network have been positioned manually to minimize overlap between red edges. Nodes clustered into the same functional class are clearly visible as discrete circular clusters within the network. Many of these clusters are highly homogeneous with respect to the color assignments generated from SFLD annotation (the names of the protein families have been added by hand for this figure). The global topology of the clusters is easy to distinguish. (B) Subgraph of the enolase network in Figure 4.2A containing

just the families from the MLE subgroup. All other families have been deleted from the layout. The OSBS proteins in the OSBS/NSAR cluster do not directly connect to other members of the OSBS family, despite being more closely related to the OSBS family than are other families within the superfamily. Interestingly, structural superposition shows that the structurally characterized OSBS/NSAR from *Amycolatopsis* is more similar to an MLE (lower RMSD) than to other structurally characterized OSBSs (Glasner, 2006).

4.3.2.3 OSBS Connectivity

The OSBS family forms a monophyletic group in the MLE subgroup phylogeny (Glasner et al., 2006). We therefore had expected there to be a direct path connecting all OSBS clusters. For the most part this was the case. Eleven of the clusters were connected by a direct path, uninterrupted by the presence of sequence from other families. Edge weights bridging the gap between these clusters ranged from six to 36. One of the 11 clusters connects to AEE with an edge weight of 18. Another connects to MLE I/II with an edge weight of 13.

Despite the connectivity between most OSBS clusters, the OSBS/NSAR cluster does not directly connect to the other 11 OSBS clusters. Instead, it connects to both AEE and MLE I/II with edge weights of 29. This was quite unexpected. The OSBS proteins in that cluster appear closer in sequence space to members of other families than they do to the members of the family with which they share the same function.

There is little evidence to explain this discrepancy except to note that BLAST e-value is not a good enough metric to resolve this type of complexity. While the topology of a BLAST-based

filtered similarity network is useful as a hypothesis generator, we are unable to use that topology in order to draw definitive conclusions.

4.3.2.4 Interpreting the Significance of Neighboring Clusters in a Filtered Similarity Network

Our investigation of the MLE subgroup revealed that the presence of an edge between two distinct clusters is not necessarily a good indication of evolutionary proximity. Rather, the edge implies that the proteins in the two neighboring clusters share some degree of similarity as it can be identified by the comparison method used, in our case the BLAST algorithm, which in turn implies that the proteins share some degree of functional similarity. For the network shown in Figure 4.2B, all clusters in the group have already been validated as sharing some degree of functional similarity by definition—they are all members of the enolase superfamily, each protein of which performs a common partial reaction mediated by a conserved constellation of active site residues that in most cases are easily identified by BLAST (Babbitt, 1996).

It is important to emphasize that we currently have no way of inferring the degree of similarity between two neighboring clusters. Any conclusions we draw about the similarity between two clusters connected by an edge can only be made relative to all other nodes that these clusters do not neighbor. For example, the 375 member enolase family is a direct neighbor to the cluster of OSBS/NSAR sequences. The edge weight connecting the two clusters is zero, indicating that the BLAST alignment between enolase and OSBS/NSAR is not statistically significant. Based on this data we are unable to interpret how much functional similarity is shared between the enolase and OSBS/NSAR clusters. We can, however, hypothesize that because the enolase family has no other neighbors in the network, the degree of overlap between enolase and all

other proteins in the dataset is no more significant than the degree of overlap between enolase and OSBS/NSAR. As illustrated by this example, when drawing a hypothesis from a given network topology, it is important to consider not only all pair-wise neighbors, but also the set of all pair-wise clusters that do not neighbor one another.

4.3.3 Examining Protein Kinase Network Topology

4.3.3.1 Summary of the Kinase Network Topology

We generated an all-by-all kinase similarity network (Figure 4.3A), which we then filtered using our protocol (Figure 4.3B) to produce 20 individual clusters. Nineteen pairs of neighboring clusters define the topology, indicating that no cycles are present. Three of the clusters are composed of multiple families belonging to the same functional class. Functional classes in the kinase superfamily designate groups of evolutionary related families frequently subject to similar functional regulation within the cell (Hanks and Hunter, 1995). Twelve clusters encompass all sequences from a single family within the dataset, while the five remaining clusters each contain a subset of sequences from a unique family.

Eight of the nine functional classes are well connected (Figure 4.3B). Any non-cyclic path between two members of a single well connected functional class contains only sequences from that particular functional class. This does not apply to the atypical kinases (AKs). No atypical kinase family connects directly to a second atypical kinase family.

Cluster degree, defined as the number of neighbors to a given cluster, is not uniform across the network. One cluster has degree eight, one cluster has degree four, three clusters have degree three, two clusters have degree two, and thirteen clusters have degree one. Cluster hubs, which

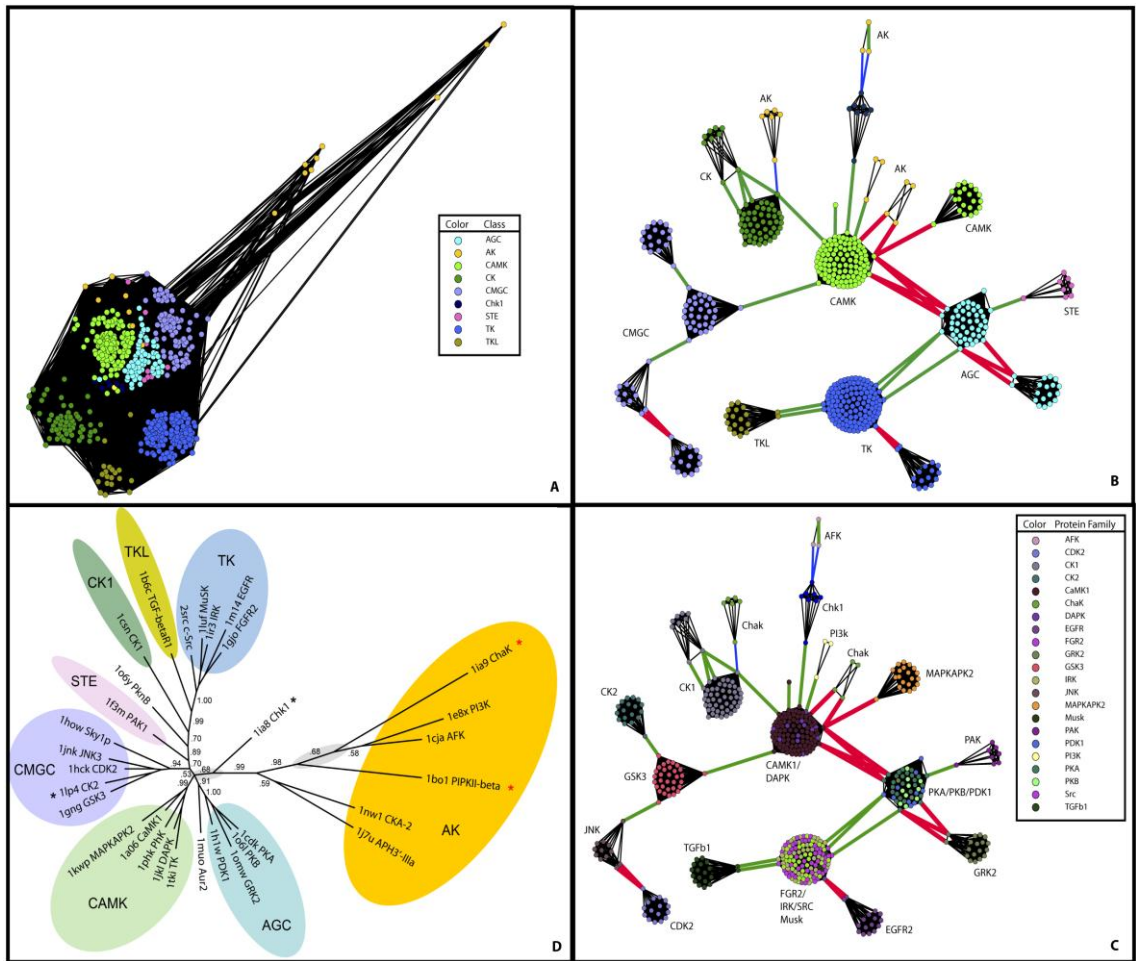


Fig. 4.3 Kinase similarity network. (A) Edge-weighted force-directed representations for the pairwise BLAST similarities in the kinase superfamily. Nodes are colored by functional class and individual functional classes group by color within the network. Node classifications to functional classes and families were obtained from the KinBase and KinaseNet databases. No direct connectivity is discernible from this cluttered network representation. (B) Unweighted force-directed layout representation of the filtered similarity network colored by functional class. Edges connecting nodes from neighboring clusters are colored blue, red, and green, based on edge weight. Blue edges have an edge weight of less than 10. Green edges have an edge

weight between 10 and the prefiltering threshold of 42. Red edges have an edge weight greater than the threshold. Parts of the network have been positioned manually to minimize overlap between intercluster edges. The clusters correspond to either individual families or individual functional classes. Eight of the nine functional classes are well connected. According to the topology, the CAMK functional class is a central hub in the network, connecting five of the functional classes. (C) Unweighted force-directed layout representation of the filtered similarity network colored by family. (D) The kinase superfamily phylogenetic tree, optimized with Mr. Bayes using both sequence and structural data. Both families and functional classes are indicated in the tree. Leaves in the tree correspond to individual families. The labeled ovals encompass multiple families corresponding to functional class, as defined by Scheeff and Bourne (Scheeff and Bourne, 2005). Each oval, signifying a unique functional class, is labeled a unique color. Chk1 is the closest of the typical kinases to the AK functional class. Kinases labeled with a black asterisk are classified differently in the tree compared with the classification produced by Manning (Manning et al., 2002). (Figure 4D from Scheeff and Bourne, 2005).

neighbor multiple clusters at the same time, are clearly distinguishable in the network. This clustering information can be ascertained directly just by looking at the final network layout.

4.3.3.2 Comparing Network Topology to Phylogenetic Branching

We analyzed branching in the evolutionary tree from the 2005 Scheeff and Bourne study (Scheeff and Bourne, 2005) (Figure 4D). Three pairs of functional classes connect directly to a single internal node, while nine pairs of families also descend directly from a single branch point. Seven of these pairs are present in our data set. We are therefore able to compare network

Table 4.1. Comparing phylogenetic divergence to filtered network topology data in kinase superfamily

Kinases With Direct Common Ancestor	Classification	In Same Cluster	Neighbors	Hop Distance
TK – TKL	Class	No	Yes	1
CAMK – AGC	Class	No	Yes	1
Chk1 – AK	Class	No	Yes	1
EGFR – FGFR2	Family	No	Yes	1
Musk – IRK	Family	Yes	No	0
CAMK1 – MAPKAPK2	Family	No	Yes	1
PKB – PKA	Family	Yes	No	0
JKN3 – CDK2	Family	No	Yes	1
CK2 – GSK3	Family	No	Yes	1
AFK – PI3K	Family	No	No	3

Column 1 contains pairs of families and functional classes that are believed to have evolved directly from the same common ancestor. Column 2 specifies whether the kinase pairs are classified as families or functional classes. Column 3 specifies whether or not the kinases appear in the same cluster. Column 4 specifies whether or not the kinases appear in neighboring clusters. Column 5 specifies the hop distance between the kinases, which we define as the

minimum number of clusters that must be traversed across the filtered network to connect a given kinase pair. The average hop distance is 1.0.

topology with phylogenetic branching in ten functional classes and family pairs by measuring the hop distance between each of the pairs across the network. We defined hop distance as the minimum number of cross-cluster traversals that separate two distinct kinase groups. A hop distance of zero indicates that two groups are in the same cluster. A hop distance of one indicates that the two groups are found in adjacent clusters that neighbor one another. The hop distances between all ten pairs of kinase groups are listed in Table 1. The average hop distance and the median hop distance for the ten pairs are both equal to one. In contrast, the mean hop distances between all pairs of functional classes and all pairs of families, are 2.22 and 2.51 respectively. These results informally imply that for this system, protein functional groups evolving directly from a single ancestor have a greater propensity to neighbor each other or cluster together in the filtered network. In other words, if two functional groups are not neighbors then they are less likely to have evolved directly from a single ancestor.

Eight of the 19 neighboring cluster pairs corresponding to seven pairs of functional classes were of indeterminate significance. These indeterminate pairs (TK-AGC, STE-AGC, CAMK-AK, CK-AK, CMGC-CAMK, Chk1-CAMK, CK-CAMK) consisted of neighboring clusters from distinct functional classes that had not diverged directly from a single common ancestor. The significance of these pairs is not known at this time. We are, however, able to state that for our BLAST-based network, over half of the neighboring clusters in the filtered kinase network are consistent with known evolutionary relationships.

4.3.3.3 Determining the Nearest Neighbor to the Atypical Kinases

The proteins in the atypical kinase class differ from other members of the kinase superfamily in that they do not share certain sequence and structural motifs common to all typical kinase proteins. One of the goals of the Scheeff and Bourne study was to determine which kinase class had the greatest evolutionary proximity to the atypical kinases. According to their phylogenetic tree, the AK class and the channel kinase (Chk1) class directly evolved from the same common ancestor. However, the bootstrap value connecting AK and Chk1 to an internal branch point was not reliable enough for the authors to draw a definitive conclusion. Furthermore, a second phylogenetic tree stochastically optimized using just sequence data showed that the choline kinase (CK) class, rather than the Chk1 class, connected to the atypical kinases, albeit again at a very low bootstrap value. The authors presented arguments demonstrating that both CK and Chk1 make good candidates as the closest evolutionary link to the atypical kinases, and that one or the other is the actual link.

In our filtered network representation of the kinase superfamily (Figure 4C), both CK and Chk1 connect to members of the AK class. Chk1 neighbors the actin-fragmin kinase (AFK) family, while CK neighbors a subset of the channel kinase (Chak) family. The remaining atypical kinase sequences, which include the phosphoinositide 3-kinase (PI3K) family and a subset of the Chak family, connect to the calcium/calmodulin-dependent kinase (CAMK) class, which connects directly to both CK and Chk1 in the network. Although the BLAST e-values underlying these results are not statistically significant, the results themselves are consistent with the two candidates for nearest evolutionary neighbor derived using phylogenetic analysis.

4.4 DISCUSSION

4.4.1 Network Topology as a Metric of Functional Similarity

Our results indicate that protein families which are not neighbors in the kinase network are less likely to descend directly from the same common ancestor. Since evolutionary distance reflects structural and functional proximity, these results suggest that a filtered network topology may be useful for developing hypotheses about structural and functional similarity. Individual clusters within a filtered network correspond to whole families or sets of functionally similar families within a superfamily. The topology between these clusters suggests the degree of functional similarity between distinct families and functional classes. Families that do not neighbor one another are less likely to share structural and functional properties than neighbors within the network.

These properties suggest that filtered similarity networks are a useful tool for discriminating sequence clusters in order to provide a starting point for predicting functional relationships and properties in poorly understood protein data sets. A researcher examining a large superfamily with few functionally characterized members will be able to apply our protocol and generate a simple visual representation of all sequences in the data set. Upon visual inspection it should be clear which uncharacterized proteins group together with members of known families. These proteins are likely to be functionally similar to the families with which they cluster, influencing the scope of the experimental assays necessary to characterize function. Additionally, certain clusters will be composed entirely of uncharacterized sequences, indicating the presence of new families. The characterized properties of clusters neighboring unknown families could help constrain the possible functions of these uncharacterized sequences. The network topology will influence hypothesis generation, which in turn allows the researcher to prioritize functional assays in order to efficiently characterize new functions within a superfamily.

Based on the intuitive nature of the filtered network layout, it is possible to investigate functional properties just by visual inspection of the network. However, unlike in the all-by-all network view, the topological boundaries between clusters in the filtered network are clearly defined prior to visualization in two- or three-dimensional space. Our filtration protocol allows researchers to automate the process of network generation relevant to function prediction, without relying on Euclidian distances across dimensionally reduced spatial representations of large multidimensional sequence datasets.

4.4.2 Contrasting Network Analysis with Phylogenetic Analysis

Using our protocol, we are able to suggest relationships between typical and atypical kinases that have previously required combining data from two separate phylogenetic trees. At the same time, we are unable to recapitulate the conclusion that atypical kinase families interconnect to form the AK functional class. Clearly, a similarity network topology does not hold the same statistical significance as a stochastically optimized phylogenetic tree. It is, however, possible to foresee research problems that lend themselves better to network analysis than to phylogenetic analysis.

Filtered homology networks are not as rigorous as phylogenetic trees in representing sequence relationships. The topology of phylogenetic trees is based on detailed mathematical models of protein evolution (Cavalli-Sforza and Edwards, 1967). In contrast, our protocol uses a heuristic approach that elucidates structural and functional similarity from global sequence comparisons without being restricted by any one model. This heuristic approach provides a useful additional tool for researchers seeking to extract potentially important features within a large sequence data set. A few minutes of computation time is all that is required to filter and visualize a similarity network based on several thousand sequences. The process is entirely automated,

requiring no a priori assumptions about the functional identity of the sequences within the network. By contrast, an optimal phylogenetic tree can only be computed using a limited subset of sequences in a multiple sequence alignment due to the computational complexity required to properly align a large and diverse set of sequences. In a large data set, the subset of sequences in the multiple sequence alignment captures only a small fraction of the total available information. Furthermore, selecting the best multiple sequence alignment subsets is a subjective task for the researcher, leading to the risk of bias in the results derived from the data. Even when a well-prepared data set is ready for phylogenetic analysis, evaluating the optimal evolutionary tree usually takes hours of computation time (Laget and Simon, 1999). Therefore, a filtered similarity network serves as a good substitute to a phylogenetic tree in those cases when rapid hypothesis generation across a large, diverse dataset takes priority over rigorous statistical significance.

It is also worth emphasizing that the network topology representation of a sequence data set includes connectivity properties not accessible through a dendrogram or phylogenetic tree. As shown in the kinase network, the degree of connectivity varies from cluster to cluster. This variability allows us to distinguish CAMK as a major hub in the network, which connects five distinct functional classes. This is not at all clear from the phylogenetic representation, where individual proteins connect indirectly through pathways of interior nodes.

The significance of hubs in protein similarity networks is unknown at this time. We hypothesize that such hubs may serve as indicators of proximity to phylogenetic branch points. Future studies will test these and other hypotheses in order to determine if the presence of hubs signifies evolutionary relationships that are not discernible within a phylogenetic tree. Filtered

network topologies provide context and terminology that make it possible to examine the importance of hubs in more detail.

4.4.3 Caveats

In our current implementation we are unable to display inter-family relationships with complete accuracy, partially because our greedy approach to reconnecting the clusters relies solely on a simple scoring metric for edges (e-value) of limited precision. Local alignments made by BLAST across pairs of motifs with significantly different lengths may lead to misleading connections within a similarity network. More sophisticated similarity comparisons, such as profile-profile alignments and hidden Markov models, could lead to more accurate network topologies.

We are also aware that the quality of a network topology depends on how well the functional groups in the dataset separate out in the first place. Kinase functional groups are more discrete with respect to one another than families in the MLE subgroup, for example. This in turn leads to better clustering, and a more meaningful topology. Knowing in advance the separability of functionally similar groups in a network would give us some measure of topological reliability. Currently, we are unable to infer the discreteness of network components from only sequence data.

4.5 CONCLUSIONS

We have developed a protocol for filtering protein superfamily similarity networks. The protocol divides an input network into discrete components while at the same time emphasizing the topology that best connects the components together. We have shown that individual clusters in the filtered networks correspond to families and classes of functionally similar proteins. Additionally, we provide evidence that neighboring clusters represent more similar sets of

proteins than clusters that are distant. Our results suggest that network topologies in a protein similarity graph, as defined by our filtration protocol, embody a meaningful representation of structural and functional similarities between individual functional groups within a protein superfamily.

In addition to defining topology, our filtration protocol also leads to a more meaningful visualization of the data within the network. An unfiltered network resembles a “hairball,” where clusters are often difficult to distinguish from one another and overlapping edges make it difficult to see significant connections. By filtering the network prior to visualization using a force-directed layout algorithm, we are able to directly count the number of clusters and see precisely how these clusters connect to one another. This direct global view provides a useful alternative for summarizing a large data set in a single easy-to-comprehend image. Our protocol can be used to output a simple representation of otherwise complex information, thereby facilitating the generation of useful hypotheses relevant to the data set in question.

The use of global protein similarity networks in the bioinformatics research community continues to rise. Our filtration protocol builds on existing network techniques to yield a comprehensive understanding of protein superfamily data. We believe the protocol serves as a foundation for developing new techniques capable of making meaningful structural and functional predictions based only on sequence information.

4.6 REFERENCES

Adai, A.T. *et al.* (2004) LGL: Creating a Map of Protein Function with an Algorithm for Visualizing Very Large Biological Networks. *J. Mol. Biol.*, **340**, 179-190.

- Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**,3389-3402.
- Babbitt,P.C. and Gerlt,J.A. (1997) Understanding enzyme superfamilies. Chemistry as the fundamental determinant in the evolution of new catalytic activities. *J. Biol. Chem.*, **272**, 30591-30594.
- Babbitt,P.C. and Gerlt,J.A. (2001) Divergent evolution of enzyme function: Mechanistically diverse superfamilies and functionally distinct suprafamilies. *Annu. Rev. Biochem.* **70**, 209-246.
- Babbitt,P.C. *et al.* (1996) The Enolase Superfamily: A General Strategy for Enzyme-Catalyzed Abstraction of the alpha-Protons of Carboxylic Acids. *Biochemistry*, **35**, 16489-16501.
- Brown,S.D. *et al.* (2006) A gold standard set of mechanistically diverse enzyme superfamilies. *Genome Biol.* **7**, R8.
- Brenner,S.E. *et al.* (1998) Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Natl. Acad. Sci. USA*, **95**, 6073-6078.
- Cavalli-Sforza,L.L. and Edwards,A.W. (1967) Phylogenetic Analysis. Models and Estimation Procedures. *American Journal of Human Genetics*, **19**, 233-257.
- Enright,A.J. and Ouzounis,C.A. (2000) GeneRAGE: a robust algorithm for sequence clustering and domain detection. *Bioinformatics*, **16**, 451-457.
- Enright,A.J. and Ouzounis,C.A. (2001) BioLayout—an automatic graph layout algorithm for similarity visualization. *Bioinformatics*, **17**, 853-854.
- Enright,A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, **30**, 1575-1584.
- Fickey,T. and Lupas,A. (2004) CLANS: a Java application for visualizing protein families based on pairwise similarity. *Bioinformatics*, **20**, 3702-3704.

- Frivolt,G. and Pok,O. (2006) Comparison of Graph Clustering Approaches. In *IIT.SRC 2006: Student Research Conference:168-175 April 2006; Bratislava, Slovakia*, 168-175.
- Fruchterman,T.J. and Reingold,M.R. (1991) Graph Drawing by Force-Directed Placement. *Software – Practice And Experience*, **21**, 1129-1164.
- George,R.A. *et al.* (2004) SCOPEC: A database of protein catalytic domains. *Bioinformatics*, **20**, 130-136.
- Glasner,M.E. *et al.* (2006) Evolution of structure and function in the o-succinylbenzoate sythase/N-acylamino acid racemase family of the enolase superfamily. *J. Mol. Biol.*, **360**, 228-250.
- Hanks,S.K. and Hunter,T. Protein kinases 6. (1995) The eukaryotic protein kinase superfamily: kinase (catalytic) domain structure and classification. *FASEB.*, **9**, 576-596.
- Kruskal,J.B. On the shortest spanning subtree and the traveling salesman problem. *Proc. Amer. Math. Soc.*, (1956) **7**, 48–50.
- Laget,B and Simon,D.L. (1999) Markov Chain Monte Carlo Algorithms for the Bayesian Analysis of Phylogenetic Trees. *Mol. Biol. Evol.*, **16**, 750-750.
- Medini,D. *et al.* (2006) Protein Homology Network Families Reveal Step-Wise Diversification of Type III and Type IV Secretion Systems. *PLoS Comput. Biol.*,**2**, e173.
- Noble,W.S. *et al.* (2005) Identifying remote protein homologs by network propagation. *FEBS Journal*, **20**, 5119–5128.
- Palmer,D.R.J. *et al.* (1999) Unexpected divergence of enzyme function and sequence: “N-acylamino acid racemase” is “o-Succinylbenzoate Synthase”. *Biochemistry*, **38**, 4252-4258.
- Pegg,S.C.H. *et al.* (2005) Representing Structure-Function Relationships in Mechanistically Diverse Enzyme Superfamilies. *Pac. Symp. Biocomput.*, **10**, 358-369.

- Pegg,S.C.H. *et al.* (2006) Leveraging Enzyme Structure-Function Relationships for Functional Inference and Experimental Design: The Structure-Function Linkage Database. *Biochemistry*, **45**, 2545-2555.
- Perutz,M.F. *et al.* (1965) Structure and function of haemoglobin II. Some relations between polypeptide chain configuration and amino acid sequence. *J. Mol Biol.*, **13**, 669-678.
- Prim,R.C. (1957) Shortest connection networks and some generalisations. *Bell System Technical Journal*, **36**, 1389–1401.
- Sakai,A. *et al.* (2006) Evolution of enzymatic activities in the enolase superfamily: N-succinylamino acid racemase and a new pathway for the irreversible conversion of D- to L-amino acids. *Biochemistry.*, **45**, 4455-4462.
- Scheeff,D.E. and Bourne,P.E. (2005) Structural Evolution of the Protein Kinase-Like Superfamily. *PLoS Comput Biol.*, **1**, e49.
- Seffernick,J.L. *et al.* (2001) Melamine Deaminase and Atrazine Chlorohydrolase: 98 Percent Identical but Functionally Different. *J. Bacteriol.*, **183**, 2405–2410.
- Shannon,S. *et al.* (2003) Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Research*, **13**, 2498-2504.
- Vlachos,M. *et al.* (2002) Non-Linear Dimensionality Reduction Techniques for Classification and Visualization. In *Proceedings of the Eight ACM SIGKDD International Conference on Knowledge Discovery and Data Mining: 645-651 July 2002; Alberta.*, 645–651.

Chapter 5

Validating Filtered Network Topologies Using a Functional Residue Prediction Algorithm

Abstract

Motivation: Indirect validation of filtered similarity networks from phylogenetic data is insufficient to study such networks on a larger scale. A more quantitative approach is necessary.

Results: We have developed an algorithm capable of predicting the location of functionally significant sequence residues given a protein similarity network. Since the quality of the final predictions depends directly on the network topology, these predictions may be used as an assessment of a given network's biological significance. We tested the algorithm on the network representation of a transmembrane protein superfamily. The algorithmic predictions overlapped with experimentally determined functional data for the superfamily, thereby validating the presence of certain hubs within the superfamily network.

5.1 Introduction

In the previous chapter we explored relationships between clustered protein families using filtered network topologies. We examined the overlap between network topology and evolutionary branching in the Kinase superfamily in order to draw certain broad conclusions about the biological validity of connections in the network. These conclusions were based on preliminary results and have yet to be rigorously proven, so we refer to them as the

“conjectured general principles” supporting the filtered network data. Because evolutionary distance reflects structural and functional proximity, we were able to state as a conjectured general principle that neighboring clusters in the filtered networks share a greater degree of functional similarity than non-neighbors. This functional similarity is related directly to structure, and is explicated in more detail later in this chapter. Our conclusions concerning functional similarity and network topologies were drawn from purely qualitative observations of relationships between topology and phylogeny. Nonetheless, this simple comparison of observable similarities yielded some initial validation of our filtered network technique.

There were, however, certain critical limitations to our comparison-based analysis. As discussed in section 4.4.2 of the previous chapter, network hubs present in the Kinase topology could not be related to the branching in the phylogenetic tree. The hubs connected to an above-average number of families, and such inter-connectivity may not be observed in phylogenetic data. We found no way to infer whether such hubs were an anomaly, or whether they were integral to the structure of the network. Thus we were unable to evaluate the biological significance of the network topology in its entirety. Qualitative comparisons allowed us to validate certain distinct relationships within the network, but not the network as a whole.

The limitation of a qualitative approach led us to ask a much broader question; how does one validate an entire protein similarity network in a quantitative way? To determine an answer it is first necessary to define what we mean by “validity.” In a valid network topology, the conjectured general principles pertaining to the similarities and differences between neighboring and non-neighboring families must hold true in some quantifiable way. The more accurate the conjectures, the more valid the network will become. This correlation between conjecture and validity in an organizational framework may be better understood by following

the history of the periodic table. In his initial set of conjectures, Mendeleev stated that the elements are ordered by atomic weights. In 1914, Henry Moseley altered this conjecture, ordering the elements by nuclear charge instead. The change in conjecture led to a more valid table. Certain elements were placed in new positions more compatible with their chemical properties. Argon, for example, was finally put in a column with the rest of the noble gasses. More importantly, the appearance of gaps in the reordered table resulted in the discovery of two new radioactive elements; Technetium and Promethium. Thus, although Mendeleev's original set of guiding principles held true to a certain extent, Moseley's table proved more valid because of its advanced predictive capacity.

We treat the general principles associated with network topology in an analogous way. If the general principles hold true, then all topological relationships, including hubs within the network, will be biologically significant. Therefore, the network validation process requires that we test and confirm our conjectured general principles about the relationships represented in the network. To do so, we must first use these principles as a basis for an algorithm capable of predicting certain testable protein properties. The accuracy of the algorithm will depend on the degree to which the general principles hold true. The more accurate the predictions, the more valid the clusters and the network topology will be relative to our conjectured principles. Likewise a lack of predictive potential will cast doubt on the biological significance of the filtered network. Thusly, the predictive algorithm can serve as a scoring function for validating the significance of any input network.

In this chapter, we develop one such algorithm and put it to the test. We show how it is possible to infer the location of functionally critical residues from the topological relationships with a

filtered sequence similarity network. Functional residue prediction is then used to validate a potentially critical hub within the network of a transmembrane protein superfamily.

5.2 Predicting Protein Properties from a Filtered Network

Topology

A set of broadly conjectured general principles underlie our interpretation of filtered network topologies. First and foremost, we conjecture that a connection between two proteins in the similarity network is indicative of a functional similarity rooted directly in sequence and structure. This similarity depends on a function being defined as the total set of possible entropically unlikely interactions between a protein and all other molecules normally found in biological systems. The definition includes all protein-protein, protein-DNA, and protein-ligand interactions. It does not include interactions with other domains in a quaternary structure; these are treated as equivalent to structural self-interactions in a single protein domain¹. Functional biochemical interactions physical depend on a protein's structural fold, which is encoded in its one-dimensional amino acid sequence. Two proteins sharing a similar sequence homology are more likely to share a similar structure, thus resulting in a similar set of biochemical interactions (Babbitt and Gerlt, 2001). If we were to represent these interactions as vectors of experimentally recorded measurements, we theoretically could calculate this similarity directly. However, such broad experimental coverage is not currently available. We must therefore rely

¹ Functional sites by necessity differ across similar families, because they perform different functions. Meanwhile, structural scaffolds are conserved between these families. Similar families sharing the same structural scaffold will display the same quaternary structure with the same interaction points across domains. These interaction points are likely to be conserved across families. As a result, we should not treat them as functional sites. We therefore do not include multimeric interactions in our definition of function. Instead, they are treated as structural interactions necessary to create a scaffold that will allow the family to carry out the unique set of functions by which it is defined. However, multimeric interactions and functional sites are not necessarily mutually exclusive, since in certain proteins the functional site location overlaps with points of multimer assembly (Kim et al., 2005).

on conjecturing that the binary binning of similar and dissimilar proteins is represented adequately by the presence and absence of edges in the filtered network topologies. As a corollary, the adjacent clusters in each the network ought to share greater sequence, structural, and functional similarity than the non-neighboring clusters.

Expanding on the previous conjecture, we can further state that the non-adjacent clusters also share a limited transitive similarity, by virtue of their connectivity within the network topology. We may examine transitive similarity in an example network containing just three clusters; Cluster A, Cluster B and Cluster C. In this sample network topology, Cluster B is connected to both Cluster A and Cluster C, but clusters A and C are not connected to each other. If additional data is available for a protein in Cluster A, that data may be used to make inferences relating to proteins in Cluster B. For example, the structure of a protein in Cluster A may be used to model the structure of a protein in Cluster B. Afterwards, the same attributes may be propagated and applied to proteins in Cluster C. Thus, a modeled structure in Cluster B may be used as a template to model an additional structure in Cluster C. As a result, there is a transitive link between Cluster A and Cluster C; the properties of one may be inferred by propagating through the network the properties of the other. Of course, in larger networks such propagation inevitably alters the preset properties of proteins in a cluster receiving the attribute information, requiring additional propagation to neighbors of that cluster. The propagation may invariably continue until an equilibrium point is reached in which no new attribute information is being passed from one cluster to another.

An additional conjecture equates each cluster with a single protein family. As defined in previous chapters, the members of a family share the same structural scaffold and perform a unique set of functions. These functions are rooted in physical interactions between

biomolecules and certain critical residues within the structural scaffold of each protein (Pegg et al., 2005). All family members perform the same functions, and specific combinations of biochemical interactions are unique to each individual family. Under such conditions, a subset of the critical functional residues must be “class-specific,” meaning they are conserved uniquely within a given family, but not across its neighbors. Meanwhile, neighboring families connected in the network must also share certain structural and functional properties, based on the discussion in the previous two paragraphs. Overlapping functional characteristics between neighboring families can only result if particular residues in common across the neighbors lead to shared biochemical patterns of interaction (Babbitt and Gerlt, 1997), (Babbitt and Gerlt, 2001). We therefore conjecture the existence of “invariant” functional residues, which are conserved across the functional sites of neighboring families in the networks.

These conjectured general principles allow us to infer that the functional site motifs responsible for the molecular interactions associated with a protein’s activity are composed of both invariant and class-specific residues. As a result, local residue conservation between neighboring families may be used to elucidate particular functional residue motif segments, which are part of a greater motif. Transitivity allows for the global propagation of locally derived motif segments across the entire network. The segments may afterwards be recombined into complete functional residue motifs. Thus, searching for patterns of residue conservation across a filtered network topology should lead to the algorithmic identification of functional sites, for all protein sequences in that network.

Functional residue prediction from conservation patterns in sequence data is not a novel idea. Evolutionary Trace (ET), a commonly used algorithm for functional residue prediction, has been around for fifteen years (Lichtarge et al. 1996). This algorithm takes as input a phylogenetic tree

computed from all sequences in a dataset. Initializing at the root of the tree, the algorithm iteratively descends breadth first down the branches, subdividing sequences into subgroups based on the number of branch points. At each iteration, a multiple sequence alignment is carried out for all subgroups, and a consensus sequence is calculated for every subgroup alignment. The consensus sequences are then compared with one another. If, at a particular position, a residue is identical in all consensus sequences, then the residues at that position are labeled as invariant. If on the other hand, the residues at that position are conserved within each subgroup, but vary between subgroups, then these residues are labeled as class-specific. Otherwise, all residues at that position are labeled as neutral. All specific residues are assigned a rank, which represents the minimum number of branches that the tree must be divided for that residue to receive a specific label. The algorithm iterates until all residues have been assigned an evolutionary rank. The lowest ranking residues are assumed to represent the highest evolutionary functional importance. These residues are most likely to appear within the functional site of the protein structure.

The simple sequence-based approach behind ET has inspired the development of new algorithms, which combine multiple sequence alignments with available structural data to elucidate functionally significant residues (Landgraf, et al. 2001), (Aloy, et al. 2001). Also, the evaluation of residue conservation in the multiple alignment step of the ET algorithm has grown increasingly more sophisticated. Techniques have been developed to score and penalize gaps in the multiple sequence alignments (Madabushi, et al., 2002). Furthermore, the use of informational entropy to keep track of residue diversity within the alignments has also been explored (Mihalek, et al., 2004).

Building on ET, we have developed “Protein Space Trace” (PST), an algorithm that relies on a filtered input network to properly rank residues by functional significance. We treated PST as a simple baseline technique for topology-based predictions. PST relies on nothing more than simple multiple alignments and network connectivity. It does not take into account structural data, and does not integrate the more sophisticated alignment analysis discussed in the previous paragraph. Future iterations of Protein Space Trace are possible and could build on the baseline approach to improve the prediction quality, for example. For now, this simplified implementation demonstrates that similarity network-based predictions are possible.

The PST algorithm is dependent on a conservation threshold which iteratively decreases from one to zero. At each iteration, the conservation threshold is used to identify a set of conserved residues within each family. These conserved residues are then compared locally across all pairwise neighbors. Patterns of specific and invariant residues are treated as functional motifs. Afterwards, all functional site information is propagated globally across all families in order to generate a more complete set of motifs. When the final iteration reaches completion, each residue is assigned a rank equivalent to the minimum number of iterations required to place that residue within a functional motif. Since the conservation threshold decreases with every iteration, the number of required iterations is inversely proportional to residue conservation within a motif. Thus, the lowest ranking residues are assumed to hold the greatest degree of functional significance.

It is important to note the predictions made by PST are open to direct experimental validation through mutagenesis studies. Such validation would corroborate not only the algorithm, but also our conjectures pertaining to the biological significance of filtered protein similarity networks.

5.3 Developing and Implementing Protein Space Trace

The following sections discuss and justify each step of the PST algorithm.

5.3.1 Defining Invariance and Class-Specificity in PST

Assume we are given Cluster A and Cluster B, two neighboring clusters in a filtered protein similarity network. Alignment A and Alignment B are the multiple alignments of sequences in A and B, respectively. We take it for granted that certain columns (positions) in each multiple alignment are conserved, and that we can distinguish conserved from non-conserved positions using a set of criteria to be defined later.

The boundary between Cluster A and Cluster B is bridged by a pair of “bridge sequences,” Bridge A and Bridge B, that best connect Cluster A and Cluster B together. Since Bridge A and Bridge B both appear in Alignment A and Alignment B, each residue in a bridge is associated with a particular position in a multiple alignment. We can carry out a pair-wise alignment of Bridge A and Bridge B, and examine all pairs of residues in the pair-wise alignment that corresponds to conserved positions in both Alignment A and Alignment B. For all such pairs of residues, we check to see if the residues are equal. If they are equal, then the associated positions in Alignment A and Alignment B are labeled as invariant. If the residues are not equal, then the associated positions are labeled as class-specific.

5.3.2 Propagation of Class-Specificity

Clusters in the network may share multiple neighbors. It is therefore possible that a conserved position within a cluster alignment may be labeled as invariant in relation to one neighbor, and class-specific in relation to another. In such cases, class-specificity takes precedence. While

closely related families may share the same residue at a particular position, that residue varies across other related families in the network, and is therefore not invariant.

The priority assigned to class-specific positions allows for the propagation of class-specificity across the network. If a position is designated as invariant for all connected clusters except for one, then the outlier cluster will propagate its position assignment of class-specificity to its neighbors, whom will in turn propagate the assignment to their neighbors, and so on, eventually converging to a state where no further propagation is possible. This does not necessarily mean that the assignment will propagate to all clusters in the network. For propagation to occur, a bridge sequence residue associated with the class-specific position must align with a conserved residue in the neighboring bridge sequence. This will likely hold true for clusters close to the origin of propagation, but not for more distance clusters. Therefore, an alternate set of conserved residues might align across the bridge sequences, and the class-specific residue from one bridge sequence might align with a residue that is not conserved at all. At this point, class-specificity will not propagate to that neighboring cluster.

5.3.3 Defining Functional Significance in PST

To determine which positions in the alignments are associated with residues that are functionally significant, we look at relationships between neighboring positions. Neighboring positions are defined by an “adjacency threshold” that is set at a low value, say one or two. If the distance gap between two columns in a multiple sequence alignment is less than or equal to the adjacency threshold, the columns are neighbors to one another.

We use the following two assumptions to define functional significance along the columns of a multiple sequence alignment:

1. If an invariant column neighbors a class-specific column, then both columns are functionally significant.
2. If a conserved column neighbors a previously determined functionally significant column, then the conserved column is also assigned a functionally significant label.

The first assumption is based on our interpretation of how sequence similarity relates to structural and functional similarity. Neighboring clusters share similar but distinct functions. This is a result of both overlapping and diverging residues being present in the functional sites of the protein structures associated with the clusters. Conserved residues within the functional sites appear as a series of motifs distributed across protein sequences. Certain motifs are likely to reflect the combination of overlapping and diverging residues in the functional site by containing invariant and class-specific residues that neighbor one another. When these motifs appear within an alignment, we can assume they are likely to be associated with the functional site.

Furthermore, if two conserved residues are neighbors, then they belong to the same motif. If one of those residues is functionally significant, then the entire motif is functionally significant. Therefore, all residues that neighbor a functionally significant residue are also functionally significant, thus justifying the second assumption. Together, the two assumptions allow us to isolate individual motifs segments, which are then expanded into complete motifs through the propagation step of the algorithm.

5.3.4 Defining Column Conservation

Functional characterization is dependent on being able to distinguish conserved from non-conserved residue positions. However, we have not yet defined what it means for a position to

be conserved. In the simplest, most stringent definition, a column in a multisequence alignment is only conserved when no residue varies within that column. We believe this definition is too restrictive. If all residues except for one within a column are the same, then, while not fully conserved, the column is highly conserved nonetheless. Thus, there are degrees of conservation, relative to one another, that should be taken into account. We explore these degrees of conservation using a "Column Conservation Threshold" (CCT). The CCT is a value, ranging from zero to one. We compare the CCT to the fraction of the column occupied by the most frequently occurring residue. If that fraction is greater than or equal to the CCT, then we classify the column as conserved. Otherwise, it is not. The maximum CCT under which a column is conserved underlies the degree of conservation associated with that column. A CCT of one implies the column is fully conserved. A CCT of .9 implies it is highly conserved. A CCT of less than .5 implies that conservation within that column is, for the most part, not significant.

While the CCT is a good metric for relating column conservation within the columns of a single cluster alignment, it is not the best metric for comparing conservation between columns of distinct clusters. This is because the clusters themselves are meant to represent distinct protein families, and some protein families are more divergent than others. A large, diverse family may neighbor a smaller, more homogenous family in the filtered similarity network. Both families might share similar functional site residues, but the larger family will show a greater tendency towards variety within that functional site. In this example, a lower CCT holds a greater significance for the large family than it does for the small family. In order to properly relate conservation between one family and the other, a second, more global parameter is needed. This parameter, the "Global Conservation Threshold," (GCT), must assign each family a CCT based on the size and divergence of that family, in a manner that gives equal significance to all conserved columns within the entire dataset.

It is possible to define the GCT by assuming that at each resolution of conservation, all clusters across the network share approximately an equal number of conserved residues. That is, at a high level of conservation, one percent of residues within each cluster are conserved (for example), while at a medium level of conservation, 10 percent of all residues within each cluster are conserved, and so on. The GCT, a parameter between 0 and 1, is able to define the appropriate level of conservation for all clusters. For any given value of the GCT, a custom CCT is set for each of the clusters. The CCT associated with each cluster equals the minimum value in which at least $100 \times \text{GCT}$ percent of the residues in the cluster alignment are conserved.

When the GCT is low, all residues specified as conserved are highly conserved, even if some of those residues come from more divergent clusters. As the GCT increases, more residues are included. Those residues that are conserved at a higher GCT but not a lower one are of less importance than residues that are conserved when the GCT is low. The GCT can therefore be used to globally rank all residues within all alignments in the network.

5.3.5 Ranking Functional Significance with PST

The PST algorithm builds on these definitions, assumptions, and parameters. When initialized, the GCT is set to a low value. At each iteration, the GCT is incremented and the conserved residues are recomputed. Afterwards, invariant and class-specific conserved residues are located and propagated across the network. Conservation is then used to recalculate the position of all functionally significant residues. If a particular column in an alignment is categorized as functionally significant at that iteration, but was not so categorized for all previous iterations, then the residues in that column are assigned a score equal to the GCT. The score serves as a ranking of functional significance for those residues. Residues with a lower score are assumed to be more functionally significant than residues with a higher score.

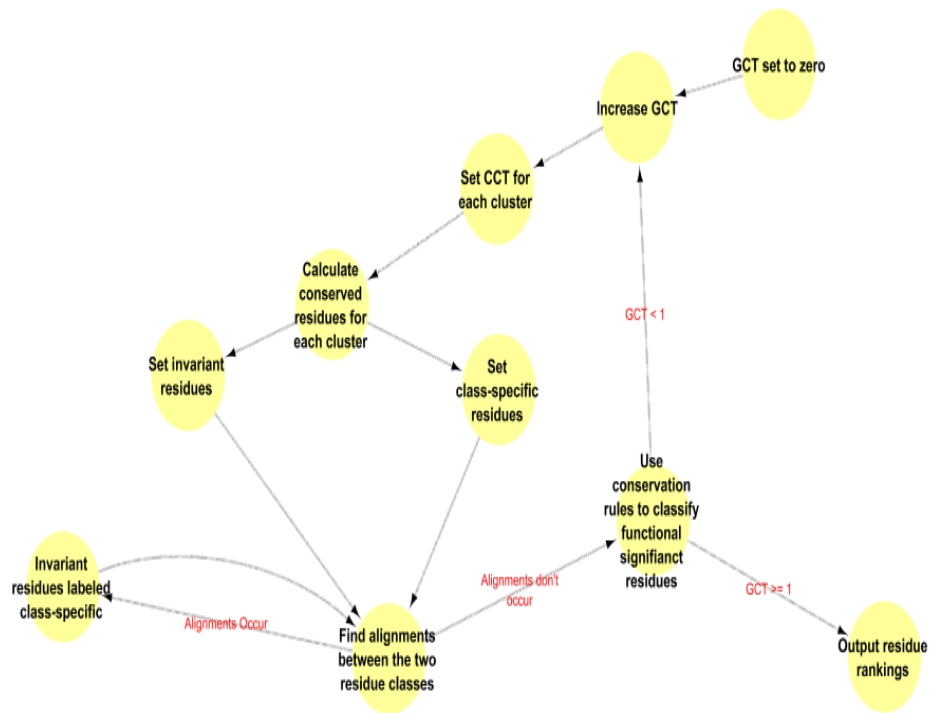


Fig. 5.1 Flow diagram of the PST algorithm

5.4 Testing Protein Space Trace on a Transmembrane

Superfamily

Some protein superfamilies have very few structures available for available for analysis. For these superfamilies, sequence-based functional residue prediction is of particular importance because determining functional residues from the limited structural data can be challenging. The Solute Carrier Transporter (SLC) superfamily represents one such group of proteins. Solute carrier transporters are transmembrane proteins that control the uptake and efflux of crucial components such as sugars, amino acids, nucleotides, inorganic ions, and drugs into the cell

(Schlessinger et al. 2010). These transporters are theorized to function based on an alternative access mechanism, in the solute binds to the transporter at specific binding site outside the intercellular membrane (Schlessinger et al. 2010), (Jardetzky, 1966). After the solute interacts with the binding cavity, the transporter undergoes a change in confirmation and shifts the solute to the other side of the membrane. Thus, the most critical functional residues are located within the solute binding cavity of the transporter.

Transporters can serve as either drug targets or drug delivery systems, making them crucial to new pharmaceutical developments. Unfortunately, like all membrane proteins, they are notoriously difficult to crystallize and few SLC structures are available. We were therefore interested in testing whether functionally critical SLC residues could be extracted from just sequence data using the PST algorithm. We generated a network from the 683 SLC sequences discussed in Chapter 2, and processed it using our Network Filtration Protocol. The resulting protein similarity network was then used as input into Protein Space Trace.

To test the quality of the residue rankings generated by our algorithm we relied on a recent mutagenesis study of the OAT3 gene (Erdman et al. 2006). OAT3 is a human gene belonging to the SLC22 family. In the study, 10 distinct OAT3 coding regions were identified in DNA samples from 270 individuals. Clones of each variant were created by site-directed mutagenesis, expressed in cells, and tested for function. Three of the 10 variants led to a complete loss of function. One of these variants eliminated function through the formation of a premature stop codon. Erdman *et. al* hypothesized that the loss of transport resulting from the remaining variants was due to the substitution of a chemically different amino acid at a functionally crucial location, although they were quick to point out that additional studies must be performed to rule out differences in mRNA or protein expression as a cause of the differences in function.

We wanted to test how our PST residue rankings for OAT3 would overlap with Erdman's loss-of-functions variants. Our reasons for exploring this system were four-fold. First, agreement between our predictions and Erdman's experimental work would contribute to validating our hypothesis that each variant is located at a functionally critical site. Second, our predictions could lead to the discovery of new functionally critical residues within the SLC superfamily. Third, a successful test would help validate PST as a useful functional residue prediction algorithm. Finally, our results would help validate the biological significance of the SLC network, leading us to potentially treat the hubs in the network as priority targets for crystallization.

5.5 RESULTS

5.5.1 The SLC Network Topology

Running our network prediction protocol on the SLC dataset produced a network that was not completely connected. A cutoff of 1.0 was applied to the network, prior to clustering it with MCL. 407 proteins fell into clusters of families that did not connect to each other. The other 289 sequences formed a connected subgraph with 16 distinct families (Figure 5.2), and SLC22 formed a hub in this subgraph by neighboring five other families. We used the subgraph as our input into PST.

5.5.2 Top Ranking Functional Residues in OAT3

After running PST, we examined the top ranking residues in OAT3. 12 residues occupied the top three ranks. The residues clustered into three distinct regions in the gene's amino acid amino acid sequence (Table 5.1). Two of these regions overlapped completely with two of Erdman's experimentally characterized loss-of-function sites. The third region did not match any of the 10

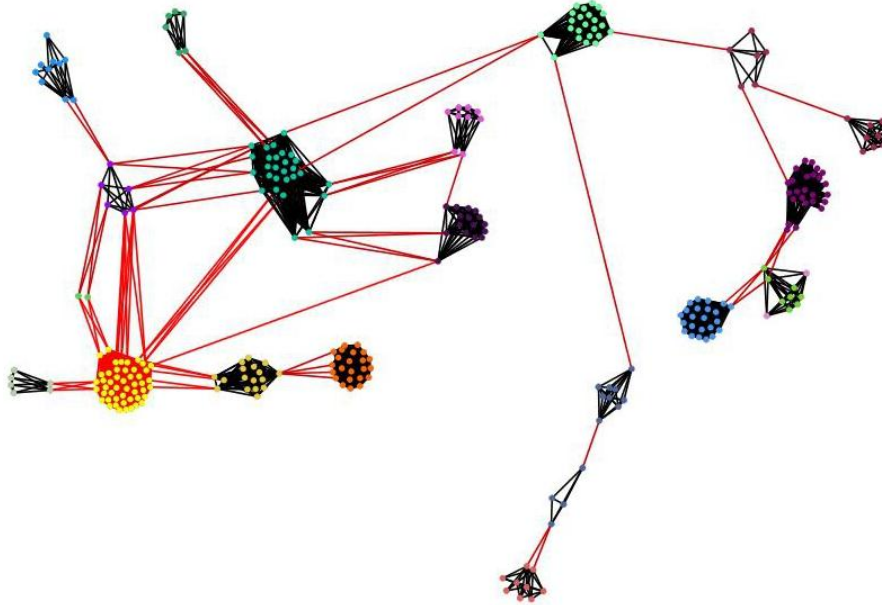


Fig. 5.2 289 sequences forming a connected subgraph with 16 distinct families in the filtered SLC protein similarity network created with Cytoscape using the clusterMaker plugin (Morris et al. 2011). Nodes have been colored by family. Edges between nodes in the same cluster are colored black. Edges connecting nodes from neighboring clusters are colored red. The cluster containing the SLC22 family has been highlighted in yellow. The SLC22 cluster is a hub that serves as a neighbor to five other families within the network. The mapping of node colors to family assignments is shown in Supplementary Figure S2.5a.

known variants described by Erdman *et. al.*

In order to evaluate the statistical significance of the found three motifs, we processed the sequences in the SLC22 cluster using the MEME motif discovery tool (Bailey and Elkan, 1994). We set the MEME input parameters requesting the 20 motifs ranging from three to six residues. The motifs were returned and ranked by e-value. The e-value for each motif estimated the

Table 5.1. The top ranking residues as determined by PST overlaid with other available data.

OAT3 Motifs	AB056422 Motifs	Transmembrane Topology
DRFG[R]	DRLG[R]	Cytoplasm
ES[I]RWL	ES[A]RWL	Cytoplasm
LPE	LPE	Cytoplasm

Column one contains residues associated with the OAT3 gene. Column two contains residues associated with the AB056422 gene that align to the residues in column one. Column three contains the predicted transmembrane topology for the residues in column two. The twelve top ranking residues are colored blue. Red brackets surround the two loss-of-function variants discussed in Erdman *et. al.* The top ranking PST residues cluster into three regions, two of which overlap with the loss-of-function variants. All three regions are computationally predicted to localize in the cytoplasm.

expected number of motifs with similar properties that one would find in an equal-sized dataset of random sequences. The e-value motifs in the final MEME output ranged from 1.0e-142 to 1.0e-22. All three PST motifs appeared in the output. The two loss-of-function site motifs occupied the second and third ranked position in the MEME results, with e-values of 1.6e-139 and 6.0e-122. The third motif occupied the seventh position, with an e-value of 8.9e-80. Thus, we confirmed that three PST motifs were unlikely to be a result of random output. Also, the comparison indicated that the PST algorithm selected its motifs in a more discriminate manner

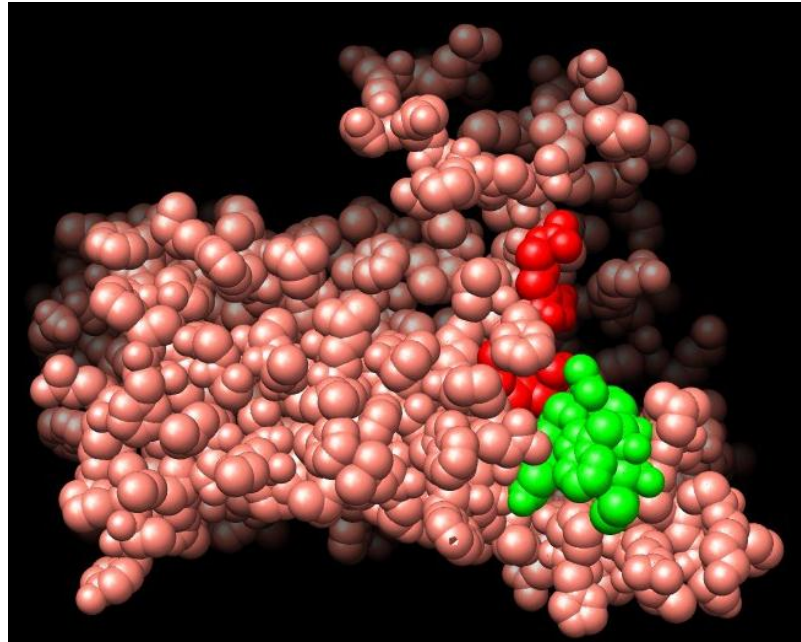


Fig. 5.3 The MODBASE model structure for OAT3, model id number 05e68042930a21b75ddc866cd5fe321c. The model target includes the first two high ranking PST regions, which overlap with the loss-of-function sites. It does not include the third region motif. The two included regions are colored red and green. They appear adjacent to each other near the edge of the model's barrel shaped structure, indicating a possible solute binding site.

the MEME, an algorithm not designed to differentiate between structural and functional site motifs.

We compared our findings to the results of a recent bioinformatics analysis of the AB056422 gene, a member of the SCL22 family found in mice (Wu et al. 2009). The analysis computationally predicted the transmembrane topology of AB056422. An alignment of OAT3 present in and AB056422 showed that the three regions discussed in the previous paragraph are

also AB056422 with a noticeable degree of conservation (Table 5.1). All three regions are predicted to fall within the cytoplasm in the transmembrane topology.

We were also able to locate a protein structure model for OAT3 in the MODBASE model database (Sanchez and Sali, 1999). Unfortunately the sequence identity between the input sequence and the template was only 12%, possibly limiting the accuracy of the model.

Furthermore, the model target region did not include the third, previously unstudied motif.

Nonetheless, we were able to map the first two motifs onto the model structure (Figure 5.3).

These motifs appeared adjacent to each other at the edge of the barrel shaped structure, where a solute could potentially bind.

These results led us to hypothesize that all three regions determined by our PST algorithm neighbor one another in the protein's barrel shaped tertiary structure, thus potentially forming a single functionally critical site where solutes may bind prior to being expelled from the cytoplasm. It would be most interesting to see what effect mutating the residues in the third region would have on OAT3 function; our prediction is that mutating these residues would also lead to a loss-of-function similar to the other two regions.

5.6 DISCUSSION

5.6.1 Validating SLC Network Topology and the OAT3 Functionally Critical Sites

The PST algorithm accurately confirmed the location of two experimentally determined loss-of-function sites within the OAT3 protein in the SLC22 family. More importantly, because PST was designed to elucidate functionally significant residues, we are able to conclude that these two

regions represent functionally critical locations within the protein. This diminishes the possibility, raised in the original OAT3 paper, that changes in mRNA or protein expression subsequent to mutation are potentially responsible for the loss-of-function. Thus, our PST algorithm, together with our network filtration protocol, helped corroborate the final hypothesis drawn by Erdman *et. al.*

These results help demonstrate the predictive capacity of the PST algorithm. They also help validate certain key conjectures on which the algorithm itself was founded. The conjectures in question pertain to the biological significance of the input filtered similarity network.

Meaningful predictions of functionally significant sites indicate that the SLC network used as input to PST is actually biologically relevant. This leads us to hypothesize that the SLC22 family serving as a hub in the network is not some random anomaly of the network filtration process.

We believe that SLC22 is a hub precisely because it shares certain structural and functional similarities with other families. Based on the previously mentioned conjectures, the neighbors

of this particular family share a more homologous structural scaffold with SLC22 than they do with each other. Thus, knowing the structure of SLC22 could lead to valuable homology models

of five additional families. This has important implications for experimentalists because

crystallization is a difficult and costly process, particularly for transmembrane proteins such as the solute carrier transferases. Any such X-ray crystallography efforts represent a significant

investment of time and resources, so the determined crystal structure should ultimately yield information that allows for further hypothesis generation. Based on the predictive capacity of

our filtered SLC network, we conclude that SLC22 might represent a greater priority target for crystallization than most other members of the SLC superfamily. Additional experimental

validation of predicted functionally significant residues would also help bolster our confidence in this conclusion.

5.6.2 Caveats and Future Directions

Although our preliminary results look promising, we have to date limited the testing of the PST algorithm to only the SLC superfamily. Additional testing on multiple superfamilies is required to statistically confirm the algorithm's capacity to validate filtered protein similarity networks and our ability to predict functionally important residues. Since mutagenesis data is typically not available for any given superfamily, future testing will necessitate the integration of other data sources into our evaluation of the algorithm. Structural data is a particularly useful supplement to mutagenesis, especially in enzymes whose crystallized structures feature a ligand bound to the active site. Using a properly calibrated scoring method that classifies residue predictions as true positives, false positives, true negatives, and false negatives, it should be possible to evaluate the PST residue rankings based on the distance between highly ranked residues and bound elements within the protein structure. Ultimately, the integration of structural and mutagenesis data will allow us to carry out a large-scale study of PST performance over multiple available superfamilies.

The eventual scaling of the PST validation process may lead to other interesting possibilities. In the future, we shall be able to process a multitude of superfamily datasets through an automated pipeline, in which similarity networks are generated, filtered, and scored based on how the PST predictions align and misalign with available superfamily structures and experimentally determined functional data. Network topologies scoring above an experimentally determined significance threshold will be categorized as being biologically significant, and will undergo additional analysis, which will include the extraction of information rich hubs from within the network.

5.7 CONCLUSIONS

We have developed Protein Space Trace, an algorithm for predicting the position of functionally significant residues from a filtered protein similarity network. The design of the algorithm is based on certain general-principle conjectures about the biological significance of network connectivity. Thus, the validation of the algorithm's predictive capacity indirectly validates the biological significance of an input protein similarity network. We tested Protein Space Trace using the SLC superfamily. Our predictions of three critically conserved residues matched two previously experimentally characterized functionally critical regions within the OAT3 gene, while predicting the location of a third, previously unstudied region in the sequence. These predictions also help validate the network of the SLC superfamily, leading us to hypothesize that the SLC22 network hub is a high priority target for crystallization. Thus, we have showed how the PST algorithm simultaneously serves as a predictive technique and a validation technique, when applied to the SLC superfamily. Although further analysis is needed to fully explore the algorithm's potential, our initial results represent a fruitful first step to eventually developing a quantitative metric for scoring similarity networks based on biological significance.

5.8 REFERENCES

- Aloy,P. *et al.* Automated structure-based prediction of functional sites in proteins: applications to assessing the validity of inheriting protein function from homology in genome annotation and to protein docking. *J. Mol. Biol*, **311**, 395-408.
- Babbitt,P.C. and Gerlt,J.A. (1997) Understanding enzyme superfamilies. Chemistry as the fundamental determinant in the evolution of new catalytic activities. *J. Biol. Chem.*, **272**, 30591-30594.
- Babbitt,P.C. and Gerlt,J.A. (2001) Can sequence determine function? *Genome Bio.*, **5**, 1-10.

- Babbitt,P.C. and Gerlt,J.A. (2001) Divergent evolution of enzyme function: Mechanistically diverse superfamilies and functionally distinct suprafamilies. *Annu. Rev. Biochem.* **70**, 209-246.
- Erdman,A.R. *et al.* (2006). The human organic anion transporter 3 (OAT3; SLC22A8): genetic variation and functional genomics. *Am J Physiol Renal Physiol*, **290**, F905-F912.
- Jardetzky,O. (1966). Simple allosteric model for membrane pumps. *Nature*, **211**, 969-970.
- Kim,S.Y. *et al.* (2005). Novel type of enzyme multimerization enhances substrate affinity of oat beta-glucosidase. *J Struct Biol*, **150**, 1-10.
- Landgraf,R, *et al.* (2001) Three-dimensional cluster analysis identifies interfaces and functional residue clusters in proteins. *J. Mol. Biol*, **307**, 1487-1502.
- Litcharge, O. *et al.* (1996) The evolutionary trace method defines the binding surfaces common to a protein family. *J. Mol. Biol*, **257**, 342-358.
- Madabushi,S. *et al.* (2002). Structural clusters of evolutionary trace residues are statistically significant and common in proteins. *J. Mol. Biol*, **44**, 4595-4614.
- Morris,J.H. *et al.* (2011). clusterMaker: A multi-algorithm clustering plugin for Cytoscape. *BMC Bioinformatics*, **12**, 436.
- Mihalek,I. *et al.* (2004). A family of evolution-entropy hybrid methods for ranking protein residues by importance. *J. Mol. Biol*, **5**, 1265-1282.
- Pegg,S.C.H. *et al.* (2005) Representing Structure-Function Relationships in Mechanistically Diverse Enzyme Superfamilies. *Pac. Symp. Biocomput.*, **10**, 358-369.
- Sanchez,R. and Sali,A. (1999) MODBASE: A database of comparative protein structure models. *Bioinformatics*. **15**, 1060-1061.
- Schlessinger,A. *et al.* (2010) Comparison of Human Solute Carriers. *Protein Science*. **19**, 412-428.

Bailey, T.L. and Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in polymers. *ISMB*. **2**, 22-36.

Wu, W. *et al.* (2009). Analysis of a large cluster of SCL22 transporter genes, including novel USTs, reveals species-specific amplification of subsets of family members. *Phys. Genomics*. **2**, 116-124.

Chapter 6

Conclusion

The goal of this thesis has been to develop a simple organizational framework for arranging protein sequence datasets in a way that increases their hypothesis generation potential. Prior to building this framework, we set down a few prerequisite requirements. The framework needed to reasonably capture how the input protein sequences clustered into functional families. Furthermore, it was important that the framework represent topological relationships between similar and dissimilar family clusters. Finally, we needed a way to evaluate the biological significance of a processed input dataset based on the predictive capacity of the organized protein sequences.

We began by searching for a better way to cluster protein sequences into families. In our first full chapter, we examined a variety of commonly used protein sequence clustering algorithms. All these algorithms took as input a sequence similarity network generated from the starting dataset. By studying the properties of edge weight distributions in four different similarity networks, we were able to develop a simple thresholding heuristic for filtering out unnecessary edges prior to clustering. Applying the threshold to our input networks improved the quality of the overall clustering results across multiple algorithms. One of the algorithms that showed noticeable improvement was MCL, a clustering technique with a relatively rapid runtime. Thus, we combined the thresholding heuristic with the MCL clustering algorithm to form the first step in our organizational framework protocol.

The next chapter probed in more detail the capacity of thresholded MCL clustering to assist in the generation of new hypotheses pertaining to the identity of uncharacterized protein sequences. We ran our technique on 683 sequences in the vicinal oxygen chelate superfamily. Less than half of these sequences featured a family categorization. Certain uncharacterized proteins fell into clusters with previously classified proteins that all belonged to a single family. Examining how the uncharacterized sequences align with the families into which they cluster revealed the existence of overlaps between functionally significant family residues and a subset of the uncharacterized sequences. This allowed us to predict the function of uncharacterized proteins while simultaneously ranking the priority of each prediction by the degree of residue overlap. The generation of ranked hypotheses makes it easier for researchers to focus on specific proteins when analyzing complex datasets. Thus, the individual protein clusters in our organizational framework formed a meaningful hypothesis generation tool.

Next, we extended our research beyond individual protein clusters and examined how these clusters may relate topologically to one another. We developed a baseline reconnection protocol in which each cluster is treated as a single network metanode. The extracted intercluster edges from the unfiltered starting network are then used to calculate the union of all minimum spanning trees that connect the metanode clusters. We carried out the reconnection algorithm on clusters from the kinase superfamily. A qualitative comparison between the resulting topology and a carefully calculated kinase phylogenetic tree revealed certain similar sets of relationships between proteins in the superfamily. The results implied that the connected clusters within the topology shared a greater degree of structural and functional similarity than non-connected clusters. Thus, a visualized sequence topology can potentially give researchers an immediate birds-eye view of intricate relationships within a complex protein

dataset. We believe our protocol for generating such topologies is a useful data organizational tool for researchers to employ.

Finally, we presented a technique for quantifying the biological significance of a protein similarity network topology. The technique relied on the predictive capacity of the topologies in question. Using several general hypotheses on significance of clusters and cluster connections within a topology, we developed the Protein Space Trace algorithm. This algorithm was designed to find the location of functionally significant residues within the proteins composing an input topology. The quality of these predictions reflects the biological significance of the topology used as input. We tested Protein Space Trace on the topology for the solute carrier transferase superfamily. The algorithm produced three functional site motifs. Two of these motifs have previously been confirmed experimentally as functionally significant and a third is yet to be confirmed. These results indicate that the topological relationships within a superfamily are useful for making biologically significant predictions. This helped validate our use of the solute carrier transferase network topology as a hypothesis generation tool and, thus, we were able to demonstrate how algorithmic prediction may be used to probe the validity of protein organizational schemas.

In this thesis, we developed and tested a framework for organizing protein sequence datasets. Future work will focus on extending these techniques beyond mere sequence data in order to take into account additional protein properties such as structure and function. Eventually we hope to be able to unify all existing protein data in a single organizational framework. The unified data structure will be tested and validated for its predictive capacity and will then be employed to predict the identities of all uncharacterized proteins and to prioritize experiments needed to confirm these predictions.

Appendix

The python code for implementing the automated threshold selection heuristic discussed in Chapter 2 is available at http://www.rbvi.ucsf.edu/Research/cytoscape/threshold_scripts.zip.

Instructions for installing the clusterMaker Cytoscape plugin discussed in Chapter 3 are available at <http://www.cgl.ucsf.edu/cytoscape/cluster/clusterMaker.html>. The python code for


implementing the network filtration protocol discussed in Chapter 4 is available at http://www.rbvi.ucsf.edu/Research/cytoscape/protocol_scripts.zip.

Publishing Agreement

It is the policy of the University to encourage the distribution of all theses, dissertations, and manuscripts. Copies of all UCSF theses, dissertations, and manuscripts will be routed to the library via the Graduate Division. The library will make all theses, dissertations, and manuscripts accessible to the public and will preserve these to the best of their abilities, in perpetuity.

Please sign the following statement:

I hereby grant permission to the Graduate Division of the University of California, San Francisco to release copies of my thesis, dissertation, or manuscript to the Campus Library to provide access and preservation, in whole or in part, in perpetuity.



Author Signature

12/01/11
Date