2011

Modular Algorithms for Biomolecular Network Alignment

Fadi George Towfic
Iowa State University

Modular algorithms for biomolecular network alignment

by

Fadi George Towfic

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Bioinformatics and Computational Biology

Program of Study Committee:

Vasant Honavar, Co-major professor

M. Heather West Greenlee, Co-major professor

Drena Dobbs

Robert Jernigan

Christopher Tuggle

Iowa State University

Ames, Iowa

2011

DEDICATION

This dissertation is dedicated to my parents, for their endless support, constant encouragement and sage advice.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

As the saying goes: *Nanos Gigantium Humeris Insidentes*[1]. This work would not have been possible if I had not stood upon the shoulders of gentle giants. Without my advisor, Dr. Vasant Honavar, this dissertation would not have taken shape. His insight, patience and focus have been unparalleled and have inspired me to contribute all that I have to the pursuit of knowledge. My co-major advisor, Dr. Heather Greenlee, has been instrumental in the development and discussion of all the algorithms and techniques in this dissertation. Her contributions have been instrumental to chapters 2 and 4 of this dissertation. I am also thankful to Dr. Drena Dobbs, on whose shoulders I stood upon to further my understanding of protein structures and their interactions, documented in three publications early in my research career. I am in debt to Dr. Robert Jernigan, who inspired chapter 3 of this dissertation and guided me in exploring the relationship between mathematics and data in my research. I am also thankful to continuous discussions and insights with Dr. Christopher Tuggle. Such collaborations gave fruition to ideas described in chapter 4 of this dissertations.

I would also like to take this opportunity to thank the two giants in my life, my dad Dr. George Towfic and my mom Dr. Samira Kettoola, who raised me and my siblings to always pursue knowledge in all its forms. The support of my brother, Zaid, and my sister, Farah, have been instrumental during turbulent times.

Of course, I cannot begin to measure the value of the numerous discussions I've had with friends and colleagues regarding research, life and the interactions of those concepts: Cornelia Caragea, Oliver Couture, Yasser El-Manzalawy, Bob Farnham, Ataur Katebi, Neeraj Kaul, Sumudu Leelananda, Harris Lin, Sender Lkhagvadorj, Olga Nikolova, Dr. Don Sakaguchi, Dr. Jeanne Serb, Adrian Silvescu, Jivko Sinapov, Jia Tao, Dr. Jeff Trimarchi, John Van Hemert, Rasna Walia, Li Xue, Oksana Yakhnenko, Xia Zhang and Michael Zimmermann.

[1] *Quidquid latine dictum sit, altum sonatur.*

ABSTRACT

Comparative analyses of biomolecular networks constructed using measurements from different conditions, tissues, and organisms offer a powerful approach to understanding the structure, function, dynamics, and evolution of complex biological systems. The rapidly advancing field of systems biology aims to understand the structure, function, dynamics, and evolution of complex biological systems in terms of the underlying networks of interactions among the large number of molecular participants involved including genes, proteins, and metabolites. In particular, the comparative analysis of network models representing biomolecular interactions in different species or tissues offers a powerful means of identifying conserved modules, predicting functions of specific genes or proteins and studying the evolution of biological processes, among other applications.

The primary focus of this dissertation is on the biomolecular network alignment problem: Given two or more networks, the problem is to optimally match the nodes and links in one network with the nodes and links of the other. We describe a suite of modular, extensible, and efficient algorithms for aligning biomolecular network models including: (1) undirected graphs in their weighted and unweighted variations (2) undirected graphs in their labeled and unlabeled variants. The resulting algorithms have been implemented as part of the Biomolecular Network Alignment (BiNA) Toolkit, an open source, user-friendly suite of software for comparative analysis of networks.

Our experiments show that BiNA is (i) competitive with the state-of-the-art network alignment tools with respect to the quality of alignments (based on a variety of performance measures) and (ii) able to align large networks ranging in size from a few hundreds of nodes and a few thousand edges to tens of thousands of nodes with millions of edges. We describe several applications of BiNA including (1) construction of phylogenetic trees based on protein-protein interaction networks, and (2) identification of biochemical pathways involved in ligand recognition in B cells by aligning gene co-expression networks constructed from mRNA profiles of B cells exposed to different ligands.

CHAPTER 1. INTRODUCTION

1.1 Overview of network models and systems biology

Biological processes are orchestrated by networks of interactions among nucleic acids, proteins, metabolites and other ligands, both within and between cells, in response to internal or external stimuli. Recently, several high-throughput techniques have emerged for measuring gene expression under different conditions or perturbations, interactions among proteins, and among genes, proteins, regulatory RNAs, small ligands and other signaling agents. Thus, it has become possible to make system-wide measurements of biological variables (45; 84; 126). Against this background, network models of protein-protein interactions (144; 78; 156), regulatory relationships between genes (38), metabolic pathways (80), and their combinations (5; 12) have been successfully applied in the rapidly expanding field of systems biology (29; 165). Numerous studies have successfully utilized network models to: comprehend how temporal and spatial clusters of genes, proteins, and signaling agents correspond to genetic, developmental and regulatory networks (160; 85; 137); uncover the biophysical basis and essential macromolecular sequence and structural features of such interactions (147; 109); infer interactions between proteins in a target species based on experimentally characterized interactions in a source species (169); discover conserved pathways among different species (88; 145); find protein groups that are relevant to disease (77; 108); predict protein function (172; 92); discover the chemical mechanism of metabolic reactions (134; 91); discover topological and other characteristics of biomolecular networks (94; 86; 87); and explain the emergence of systems-level properties of networks from the interactions among their parts (1; 16; 17). Furthermore, driven by the need for computational tools for exploiting network models in biological sciences, several groups have developed databases for storing networks (12; 109; 8) and query languages and tools for retrieving networks that match specified criteria (140; 109); identifying optimal matches for a source pattern e.g., a set of proteins linked by an undirected path (89; 140) or those that form a specific pattern or motif (14; 15) in one or more target networks; and for aligning protein-protein interaction networks (89; 97; 81; 56; 149), regulatory networks (169; 139) and metabolic networks (127; 6).

Because the available data is often of variable quantity, quality and granularity, there is a need for several classes of network models at varying levels of abstraction, to explore different questions in diverse applications. Of particular interest are:

- **Undirected graphs** in which nodes represent genes or proteins and links between nodes denote interactions (e.g., protein-protein interaction networks (144; 78; 156)). Such networks provide a global picture of gene-gene or protein-protein interactions that can further be analyzed to identify putative functional modules (144; 78; 156), nodes that play important roles (e.g., hubs) (79); or to determine topological features (degree distribution, hierarchical structure, modularity, etc. (51; 132; 168; 90)). Comparative analysis of two or more networks of the same type from different species can help identify conserved functional modules (139; 145; 114; 170; 121), transferring functional annotations across species, etc. Although most of the work has focused on undirected graphs with a single type of links, many applications call for network models that can accommodate *multiple types of links* (e.g., interaction, co-localization, etc. in the case of protein-protein interaction networks), or *multiple types of nodes* (e.g., in the case of networks that simultaneously model the interactions among proteins, RNA, DNA, etc.), or *both.*

- **Undirected weighted graphs** e.g., gene coexpression networks in which the nodes represent genes and weights on the links model the similarity of expression patterns between genes (e.g., gene expression correlation networks (145)). Such networks can be analyzed to identify clusters of genes that display similar expression patterns, e.g., using spectral clustering techniques (158; 98; 119); Comparative analysis of two or more networks from different tissues from the same species can be used to identify key differences in gene coexpression patterns; Comparative analysis of gene coexpression networks obtained under comparable conditions from different species can be used as a basis of inferring functional similarities between the corresponding genes, etc.

- **Directed graphs** that model influences between genes where nodes represent genes and directed, unlabeled or labeled links denote regulatory interactions. Pathway databases such as TRANSPATH (99), PathCase (48), and KEGG (84) present examples of richly annotated directed graphs. Tracing of directed paths in such graphs can uncover sequences of regulatory events, redundant regulatory mechanisms, etc; directed cycles indicate feedback regulation. Comparison of pathways can reveal common subgraphs, putative evolutionary relations, etc. Topological analysis can reveal the distributions and average numbers of regulators per gene.

- **Undirected or directed multi-graphs** where the each node and each edge has associated with it a *set of labels* (e.g., nodes labeled with their *Gene ontology* functional annotation, subcellular localization, etc.) as well as their **weighted** counterparts.

The primary focus of this dissertation is on modular algorithms that are equally applicable to aligning undirected graphs and undirected, weighted graphs. The algorithms may also be extended to deal with directed graphs or multigraphs (see chapter 7 for more details).

1.2 The network alignment problem

Network alignment methods present a powerful approach for detecting conserved modules across several networks constructed from different species, conditions or timepoints. The detection of conserved network modules may allow the discovery of disease pathways, proteins/genes critical to basic biological functions, and the prediction of protein functions. The problem of aligning two networks, in the absence of the knowledge of how each node in one network maps to one or more nodes in the other network, requires solving the subgraph isomorphism problem, which is known to be computationally intractable (NP-complete) (61). Consequently, several heuristics have been explored for striking a balance between the speed, accuracy and robustness of the alignment of large biological networks. For instance, The PathBLAST algorithm searches for nodes/proteins that share sequence homology and the same order in the two pathways being aligned. The runtime complexity of this algorithm, which is factorial in the length of the pathways being aligned, prevents it from being viable for aligning large networks with thousands of nodes (140). MaWISh (97) is a pairwise network alignment algorithm with a runtime complexity of $O(mn)$ (where m and n are the number of vertices in the two networks being compared) that relies on a scoring function that takes into account protein duplication events as well as interaction loss/gain events between pairs of proteins to detect conserved protein clusters.

Bruckner et al.'s algorithm (Torque) attempts to address the problem whereby the topology among the nodes for the query network is not known (28). The running time complexity of Torque is $O(3^k m)$ where k is the number of vertices in the query and m is the number of edges. Hopemap is an iterative clustering-based alignment algorithm for protein-protein interaction networks. HopeMap starts by clustering homologs based on their sequence similarity and already known KEGG Orthology status. The algorithm then proceeds to search for strongly connected components and outputs the conserved components that satisfy a predefined user threshold (149). Graemlin 2.0 is a linear time algorithm that relies on a feature-based scoring function to perform an approximate global alignment of multiple networks. The scoring function for Graemlin 2.0 takes into account protein deletion, duplication, mutation, presence and count as well as edge/paralog deletion across the different networks being aligned (56). NetworkBLAST-M (81) is a progressive multiple network alignment algorithm that constructs a layered alignment graph, where each layer corresponds to a network and edges between layers connect homologs across different networks. Highly conserved subnetworks from networks from different species are first aligned based on highly conserved orthologous clusters, then the clusters are expanded using an iterative greedy local search algorithm (81).

In the following sections, we provide a detailed sketch of the network alignment algorithms that have been proposed in the literature. We also provide an analysis of the running time complexity for each of the algorithms. Finally, we provide a statement for the significance of efficient network alignment algorithms and how such algorithms may be used to address important biological questions.

1.3 Formal mathematical definition of network alignment

The graphs dealt with in this section are node-labeled, undirected and unweighted. A graph $G(V, E, \rho)$ consists of a sets of vertices V and edges E and vertex label function ρ. V denotes $\{v_1, v_2, v_3, ...v_n\}$ and E denotes $\{e_1, e_2, e_3, ...e_k\}$, where $k \leq \frac{n(n-1)}{2}$. ρ is a function that assigns labels to the vertices of G. We match labels of nodes/vertices across protein-protein interaction networks from different species using BLAST (3). $H(V_2, E_2, \rho_2)$ is said to be a subgraph of $G(V_1, E_1, \rho_1)$ if $V_2 \subset V_1$, $\rho_2(i) = \rho_1(i)\ \forall i \in V_2$, and $E_2 \subset E_1$ where E_2 consists only of edges whose end points are in V_2. We associate with the graphs $G_1(V_1, E_1, \rho_1)$ and $G_2(V_2, E_2, \rho_2)$ sets subgraphs $S_1 = \{C_1, C_2, C_3, ...C_n\}$ and $S_2 = \{Z_1, Z_2, Z_3, .., Z_m\}$(respectively), where $C_i(K_i, O_i, \mu_i)$ $1 \leq i \leq n$ is a subgraph of G_1 and $Z_j(W_j, Q_j, \kappa_j)$ $1 \leq j \leq m$ is a subgraph of G_2. Our basic strategy is to find a best match for each subgraph in S_1 from S_2 by optimizing a scoring function, $K(C_i, Z_i)$, such that we obtain: (i) a set of vertices that satisfy $\mu_i(u) = \kappa_j(v)$, where $v \in W_j$ and $u \in K_i$ (ii) a set of edges whereby: if $(\mu_i(u_1), \mu_i(u_2))$ is an edge in O_i, then $(\kappa_j(v_1), \kappa_j(v_2))$ is an edge in Q_j where $\mu_i(u_1) = \kappa_j(v_1)$ and $\mu_i(u_2) = \kappa_j(v_2)$. In this section, we present two different choices of graph kernels for $K(C_i, Z_j)$: the shortest path kernel (22) and random walk kernel (157). The resulting solution to the network alignment problem satisfies the condition that each subgraph in S_1 has at most one matching subgraph in S_2. Thus, a pairwise alignment of the networks $G_1(V_1, E_1, \rho_1)$ and $G_2(V_2, E_2, \rho_2)$ is expressed in terms of an optimal alignment among the sets of the corresponding sets of subgraphs in S_1 and S_2.

1.4 Brief overview of state-of-the-art methods

1.4.1 MaWISh

MaWISh (Maximum-Weight Induced Subgraph) is a local pairwise alignment algorithm for protein-protein interaction networks that focuses on discovering highly conserved subgraphs in the interactome of a pair of species. The problem is modeled as a graph optimization problem, while taking into account duplication/divergence models (see Figure 1.1). It is a greedy algorithm that finds a set of nodes in each graph such that the alignment score is

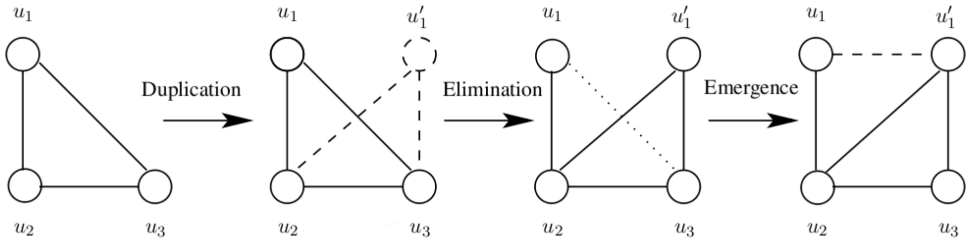


Figure 1.1 Original figure from Koyuturk et al. (97). The Duplication/Elimination/Emergence model considered in MaWISh. Starting with three interactions between three proteins, protein u_1 is duplicated to add u_1 into the network together with its interactions (dashed circle and lines). Then, u_1 loses its interaction with u_3 (dotted line). Finally, an interaction between u_1 and u_1 is added to the network (dashed line)

highest. Specifically, MaWISh searches for hubs in a graph, then adds neighbors to each hub based on a heuristic that measures the conservation of the module across several graphs. The runtime complexity of MaWISh is $O(mn)$ (where m and n are the number of vertices in the two networks being compared).

1.4.2 NetworkBLAST-M

Kalaev et al.'s extension to the NetworkBLAST algorithm to align multiple protein-protein interaction networks consists of stacking the protein-protein interaction networks in to multiple layers, then connecting the nodes across the layers (using inter-layer edges) using sequence homology based on a pre-computed phylogenetic tree. The algorithm searches for high scoring subnets (multiple k-spines, see Figure 1.2) and outputs the high scoring subnets as possible alignments across the various input networks from different species. The algorithm starts by computing a seed subnetwork that consists of 2 spines, then expands the alignment iteratively around the seed spines. The initial seed spines are found by imposing a strict topology on the connected nodes from each species. For example, in Figure 1.2, although there is no edge connecting the nodes in species U_1 and U_2, the nodes are still reachable from each other due to the fact that there is a path from U_1 to U_3 and from U_3 to U_2. The seed searching algorithm assumes that such a topology is equivalent to the case where there are edges connecting the nodes from U_1 to U_2 and from U_2 to U_3. Furthermore, the algorithm assumes that a homologous protein must exist in every species/network in the alignment. Thus the k-spines contain k proteins, one from every species in the alignment. The seed spines are expanded by searching for spines that contain only nodes that are at most two hops away from the seed spines.

1.4.3 Graemlin

Graemlin is a linear time Multiple Alignment algorithm for protein-protein interaction networks that relies on a parameter-learning algorithm to decompose networks into specific feature vectors and compute the similarity based on such features. Graemlin also provides a parameter-learning algorithm that can automatically weight the contribution of each feature based on a precomputed alignment. The features for nodes considered in Graemlin 2.0 are:

- Protein presence (the maintenance of proteins in both species)
- Protein count (the maintenance of more than one protein in both species)
- Protein deletion (the loss of a protein in one of the two species)
- Protein duplication (the duplication of a protein in one of the two species)
- Protein mutation (the divergence in sequence of two proteins in different species)
- Paralog mutation (the divergence in sequence of two proteins in the same species)

The features considered for edges in Graemlin 2.0 are:

- Edge deletion (the loss of an interaction between two pairs of proteins in different species)
- Paralog edge deletion (the loss of an interaction between two pairs of proteins in the same species)

Graemlin 2.0 relies on a phylogenetic tree to sum the pairwise features over pairs of species adjacent in the tree, including ancestral species. The feature functions also take into account the evolutionary distance between the species being compared (see Figure 1.3).

1.4.4 GRAAL

GRAph ALigner (GRAAL) is a strictly topological alignment algorithm that relies on graphlet distributions to compare networks (101). GRAAL has been successfully utilized for reconstructing phylogenetic relationships between bacterial species based on the topologies of the species' protein-protein interaction networks. Briefly, GRAAL relies on the computation of graphlets up to four nodes in size around each node between the graphs being compared. Each node in a network is given a score denoting how many graphlets they participate in. The scores for nodes across two networks are then compared using Milenkoviæ et al.'s (115) formula for averaging node-based scores in a graph:

$$S(u_x^1, v_y^2) = \frac{|\log(S(u_x^1) + 1) - \log(S(v_y^2) + 1)|}{\log(\max(S(u_x^1), S(v_y^2)) + 2)}$$

Where $S(u_x^1)$ and $S(v_y^1)$ are the scores for the nodes from $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, where $u_x^1 \in V_1$ and $v_y^2 \in V_2$. The above formula produces a normalized score for each node-based graphlet score.

1.5 Limitations of current methods

Current approaches to biomolecular network alignment summarized above suffer from several limitations: Most of the biomolecular network alignment algorithms described above deal with a specific type of network (e.g., protein-protein interaction networks). Because the scoring functions for matching nodes across networks, or for aligning the networks based on matches between nodes across networks, and the heuristics used to speed up the alignment are typically hard-coded into the implementation of the respective algorithms, it is not straightforward to extend the existing implementations (e.g., for aligning protein-protein interaction networks) to handle more general classes of biomolecular networks (e.g., networks that model multiple types of interactions between multiple types of molecular entities). Nor is it easy to replace or modify specific components of the alignment algorithms (e.g., the scoring function used for matching nodes across networks) to meet the needs of specific biological applications, or to easily specify at runtime the specific characteristics of the biomolecular networks that can be exploited by the alignment algorithm (73; 142). Some of the algorithms, because of computational considerations, make some simplifying assumptions that are at odds with the known characteristics of biomolecular networks (142).

1.6 Significant contributions of dissertation

This dissertation provides a class of flexible (in terms of ease of modification), scalable (in terms of computational running time), and accurate (in terms of biological significance) algorithms for comparing and aligning biomolecular networks while making minimum assumptions about the source of the networks. The networks can be labeled (e.g., sequence labeled, or nodes can be matched based on orthology) or unlabeled (networks can be aligned strictly based on topology). The following sections describe the main contributions of this dissertation against the background of the current literature in the field.

1.6.1 First highly modular algorithm in the field

Chapter 2 describes the Biomolecular Network Alignment (BiNA) toolkit in detail. This algorithm is the first algorithm in the field whose scoring (comparison) functions and partition (clustering) functions are independent. Furthermore, this algorithm uses the proven divide and conquer strategy to enable the future addition of new techniques for partitioning and scoring without changing the overall method.

1.6.2 Highly scalable algorithm

BiNA can run on desktop machine to clusters, aligning networks from 100's of edges to several millions. Chapter 6 describes the implementation details and scalability of the algorithm in detail. The running time of the various methods that comprise this algorithm are described in detail in chapter 2.

1.6.3 First highly flexible algorithm

BiNA can align undirected, unweighted protein interaction networks and undirected, weighted gene-coexpression networks. BiNA can align within the same organism or across species, can align based on topology alone or using node labels or BLAST correspondence. Experiments on aligning networks from different species are provided in chapters 2, 3 and 4. Experiments outlining the alignment of networks within the same organism are provided in chapter 5. The alignment techniques based on strict topology and discussion of applications of topology to the alignment problem are provided in chapter 3.

1.6.4 Highly portable

BiNA has been implemented purely in Java to achieve maximum portability on Windows, Mac and Linux/Unix systems). The BiNA webserver is user-friendly and accessible. The architecture and implementation of the algorithm are discussed in chapter 6.

1.6.5 High accuracy in terms of biological performance

BiNA has been evaluated in several respects to assess the biological relevance of the algorithm's output. Several assessments currently available in the literature are:

- Detection of enriched GO Terms (chapters 2 and 3)
- Construction of phylogenies based on labeled and unlabeled protein-protein interaction networks (chapter 3)
- Detection of orthologs (chapter 4)

1.6.6 Applied to important biological problems

BiNA has been applied to several important biological questions. Two of the applications currently available in the literature are:

- Detection of orthologs based on protein-protein and gene coexpression networks (chapter 4)
- Detection of expression patterns in B-Cells (chapter 5)

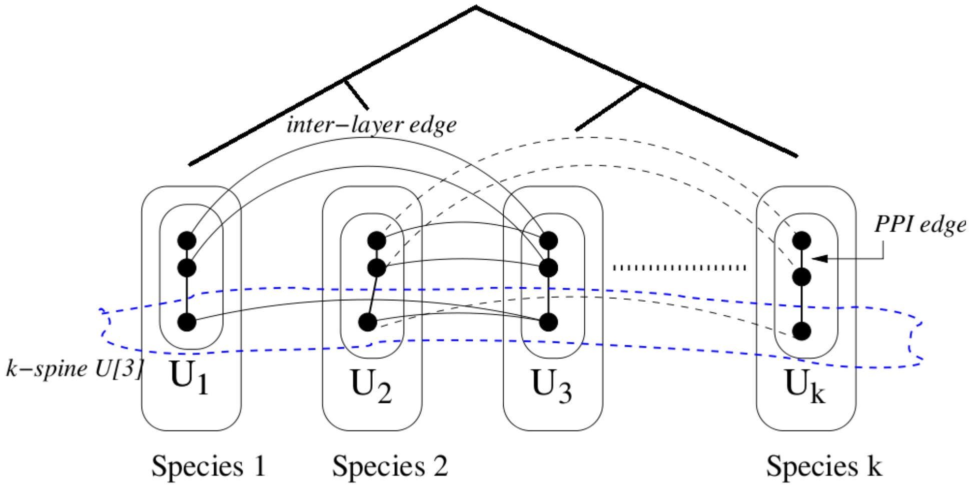
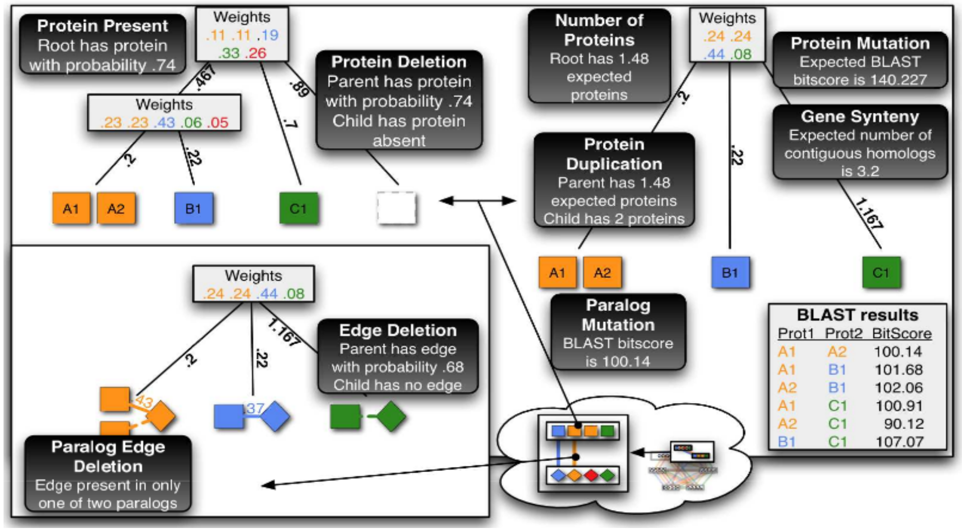


Figure 1.2 Slightly modified figure from Kalaev et al. (81). A seed defined by a d-identical-spine subnet ($d = 3$ in the above example since there are 3 k-spines), where the k-spines are restricted to be paths with identical topology. The dashed blue line encloses one of the three k-spines. The phylogenetic tree used to order the connection operation of the inter-layer edges of the k-spines is shown at the top of the figure



BLAST results

Prot1	Prot2	BitScore
A1	A2	100.14
A1	B1	101.68
A2	B1	102.06
A1	C1	100.91
A2	C1	90.12
B1	C1	107.07

Figure 1.3 Original figure from Flannick et al. (56). This figure shows the set of evolutionary events that are computed by Graemlin's node and edge feature functions. Graemlin 2.0 uses a phylogenetic tree with branch lengths to determine the events. First, the species weight vectors (shown as gray boxes) at each internal node of the tree are constructed; the weight vector represents the similarity of each extant species to the internal node. Graemlin 2.0 uses these weight vectors to compute the likely evolutionary events (shown as black boxes) that occur

CHAPTER 2. BIOMOLECULAR NETWORK ALIGNMENT (BiNA) TOOLKIT

Based on a paper titled "Aligning Biomolecular Networks using Modular Graph Kernels", accepted for publication in WABI 2009[1]

Fadi Towfic, M. Heather West Greenlee and Vasant Honavar

Abstract

Comparative analysis of biomolecular networks constructed using measurements from different conditions, tissues, and organisms offer a powerful approach to understanding the structure, function, dynamics, and evolution of complex biological systems. We explore a class of algorithms for aligning large biomolecular networks by breaking down such networks into subgraphs and computing the alignment of the networks based on the alignment of their subgraphs. The resulting subnetworks are compared using graph kernels as scoring functions. We provide implementations of the resulting algorithms as part of BiNA, an open source biomolecular network alignment toolkit. Our experiments using *Drosophila melanogaster*, *Saccharomyces cerevisiae*, *Mus musculus* and *Homo sapiens* protein-protein interaction networks extracted from the DIP repository of protein-protein interaction data demonstrate that the performance of the proposed algorithms (as measured by % GO term enrichment of subnetworks identified by the alignment) is competitive with some of the state-of-the-art algorithms for pair-wise alignment of large protein-protein interaction networks. Our results also show that the inter-species similarity scores computed based on graph kernels can be used to cluster the species into a species tree that is consistent with the known phylogenetic relationships among the species.

[1] *Reproduced with permission from Springer*

2.1 Background and Motivation

The rapidly advancing field of systems biology aims to understand the structure, function, dynamics, and evolution of complex biological systems (29). Such an understanding may be gained in terms of the underlying networks of interactions among the large number of molecular participants involved including genes, proteins, and metabolites (165; 62). Of particular interest in this context is the problem of comparing and aligning multiple networks e.g., those generated from measurements taken under different conditions, different tissues, or different organisms (139). Network alignment methods present a powerful approach for detecting conserved modules across several networks constructed from different species, conditions or timepoints. The detection of conserved network modules may allow the discovery of disease pathways, proteins/genes critical to basic biological functions, and the prediction of protein functions.

The problem of aligning two networks, in the absence of the knowledge of how each node in one network maps to one or more nodes in the other network, requires solving the subgraph isomorphism problem, which is known to be computationally intractable (NP-Hard) (61). However, in practice, it is possible to establish correspondence between nodes in the two networks to be aligned and to design heuristics that strike a balance between the speed, accuracy and robustness of the alignment of large biological networks. For instance, MaWISh (97) is a pairwise network alignment algorithm with a runtime complexity of $O(mn)$ (where m and n are the number of vertices in the two networks being compared) that relies on a scoring function that takes into account protein duplication events as well as interaction loss/gain events between pairs of proteins to detect conserved protein clusters. Hopemap (149) is an iterative clustering-based alignment algorithm for Protein-Protein Interaction networks. HopeMap starts by clustering homologs based on their sequence similarity and already known KEGG/InParanoid Orthology status. The algorithm then proceeds to search for strongly connected components and outputs the conserved components that satisfy a predefined user threshold (149). Graemlin 2.0 is a linear time algorithm that relies on a feature-based scoring function to perform an approximate global alignment of multiple networks. The scoring function for Graemlin 2.0 takes into account protein deletion, duplication, mutation, presence and count as well as edge/paralog deletion across the different networks being aligned (56). NetworkBLAST-M (81) is a progressive multiple network alignment algorithm that constructs a layered alignment graph, where each layer corresponds to a network and edges between layers connect homologs across different networks. Highly conserved subnetworks from networks from different species are first aligned based on highly conserved orthologous clusters, then the clusters are expanded using an iterative greedy local search algorithm (81).

Against this background, we explore a class of algorithms for aligning large biomolecular networks using a *divide and conquer* strategy that takes advantage of the *modular* substructure of biological networks (67; 132; 70). The basic idea behind our approach is to align a pair of networks based on the optimal alignments of the subnetworks of one network with the subnetworks of the other. Different ways of decomposing a network into subnetworks in combination with different choices of measures of *similarity* between a pair of subnetworks yield different algorithms for aligning biomolecular networks.

We utilize variants of state-of-the-art *graph kernels* (22; 23), first developed for use in training support vector machines for classification of graph-structured patterns, to compute the *similarity* between two subgraphs. The use of graph kernels to align networks offers several advantages: It is easy to substitute one graph kernel for another (to incorporate different application-specific criteria) without changing the overall approach to aligning networks; it is possible to combine multiple graph kernels to create more complex kernels (23) as needed. Our experiments with the fly, yeast, mouse and human protein-protein interaction networks extracted from DIP (Database of Interacting Proteins) (136) demonstrate the feasibility of the proposed approach for aligning large biomolecular networks.

The rest of the paper is organized as follows: Section 2 precisely formulates the problem of aligning two biomolecular networks and describes the key elements of our proposed solution. Section 3 describes the experimental setup and experimental results. Section 4 concludes with a summary of the main contributions of the paper in the broader context of related literature and a brief outline of some directions for further research.

2.2 Problem Formulation

We consider the problem of pair-wise alignment of protein-protein interaction networks. We model protein-protein interaction networks as undirected and unweighted graphs. In a protein-protein interaction network, the vertices in the graph correspond to proteins and the edges denote interactions between the two proteins. Let the graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ denote two protein-protein interaction networks where $V_1 = \{v_1^1, v_2^1, v_3^1, ...v_n^1\}$ and $V_2 = \{v_1^2, v_2^2, v_3^2, ...v_m^2\}$, respectively, denote the vertices of G_1 and G_2; and E_1 and E_2 denote the edges of G_1 and G_2 respectively. Let a matrix $\mathbf{P}$ with $|V_1|$ rows and $|V_2|$ columns (i.e, $n \times m$ matrix) denote a set of matches between the vertices of G_1 and G_2. The mapping matrix $\mathbf{P}$ is defined such that for any two vertices v_x^1 and v_y^2 (where $1 \leq x \leq n$ and $1 \leq y \leq m$) from graphs G_1 and G_2, respectively, $P_{v_x^1 v_y^2} = 1$ if v_x^1 from G_1 is matched to v_y^2 from G_2 and $P_{v_x^1 v_y^2} = 0$ if v_x^1 in G_1 is not a match to v_y^2 in G_2. For example, the matches between nodes may be based on homology between the sequences of the corresponding proteins. Thus, each node in G_1 is matched to 0 or more nodes of G_2 and vice versa. Note that the number of such matches for any node in G_1 is much smaller than the total number of nodes in G_2 and vice versa.

$C_1(L_1, O_1)$ is said to be a subgraph of $G_1(V_1, E_1)$ if $L_1 \subset V_1$ and $O_1 \subset E_1$ where O_1 consists only of edges whose end points are in L_1. We associate with the graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ sets of subgraphs $S_1 = \{C_1, C_2, C_3, ...C_l\}$ and $S_2 = \{Z_1, Z_2, Z_3, .., Z_w\}$ (respectively), where $C_i(L_i, O_i)$ $1 \leq i \leq l$ is a subgraph of G_1 and $Z_j(W_j, Q_j)$ $1 \leq j \leq w$ is a subgraph of G_2. Our basic strategy is to find a best match for each subgraph in S_1 from S_2 by optimizing a scoring function, $K(C_i, Z_j)$, such that we obtain: (i) a set of vertices that satisfy $P_{v_x^1 v_y^2} = 1$, where $v_x^1 \in L_i$ and $v_y^2 \in W_j$ and (ii) a set of edges where: if (v_x^1, v_d^1) is an edge in O_i, then (v_y^2, v_g^2) is an edge in Q_j where $P_{v_x^1 v_y^2} = 1$ and $P_{v_d^1 v_g^2} = 1$. The resulting solution to the network alignment problem satisfies the condition that each subgraph in S_1 has at most one matching subgraph in S_2. Thus, a pairwise alignment of the networks $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ is expressed in terms of an optimal alignment among the sets of the corresponding sets of subgraphs in S_1 and S_2.

2.3 Algorithm

2.3.1 Divide: Partitioning methods

As noted earlier, our basic approach to aligning a pair of protein-protein interaction networks involves (a) decomposing each network into a collection of smaller subnetworks; (b) compute the alignment of the two networks in terms of the optimal alignments of the subnetworks of one network with the subnetworks of the other. Different choices of methods for decomposing a network into subnetworks in combination with different choices of measures of *similarity* between a pair of subnetworks yield different algorithms for aligning protein-protein interaction networks. In our current implementation, we establish the matches between nodes in the two protein-protein interaction networks to be aligned based on reciprocal BLASTp (3) hits between the corresponding protein sequences. Thus, $P_{v_x^1 v_y^2} = 1$ if and only if the corresponding protein sequences of v_x^1 and v_y^2 are reciprocal BLASTp hits (74) for each other (at some chosen user-specified threshold). Alternatively, the mapping can be established based on known homologies (e.g between the human WNT1 and mouse Wnt1 proteins) (96; 30).

2.3.1.1 K-Hop

A k-hop neighborhood-based approach to alignment uses the notion of k-hop neighborhood. The k-hop neighborhood of a vertex $v_x^1 \in V_1$ of the graph $G_1(V_1, E_1)$ is simply a subgraph of G_1 that connects v_x^1 with the vertices in V_1 that are reachable in k hops from v_x^1 using the edges in E_1. Given two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, a mapping matrix $\mathbf{P}$ that associates each vertex in V_1 with zero or more vertices in V_2 and a user-specified parameter k, we construct for each vertex $v_x^1 \in V_1$ its corresponding k-hop neighborhood C_x in G_1. We then use the mapping matrix $\mathbf{P}$ to obtain the set of matches for vertex v_x^1 among the vertices in V_2; and construct the k-hop neighborhood Z_y for each matching vertex v_y^2 in G_2 and $P_{v_x^1 v_y^2} = 1$. Let $S(v_x^1, G_2)$ be the resulting collection of k-hop neighborhoods in G_2 associated with the vertex v_x^1 in G_1. We compare each k-hop subgraph C_x in G_1 with each member of the corresponding collection $S(v_x^1, G_2)$ to identify the k-hop subgraph of G_2 that is the best match for C_x (based on a chosen similarity measure). This process is illustrated in figure 2.1. The runtime complexity of the

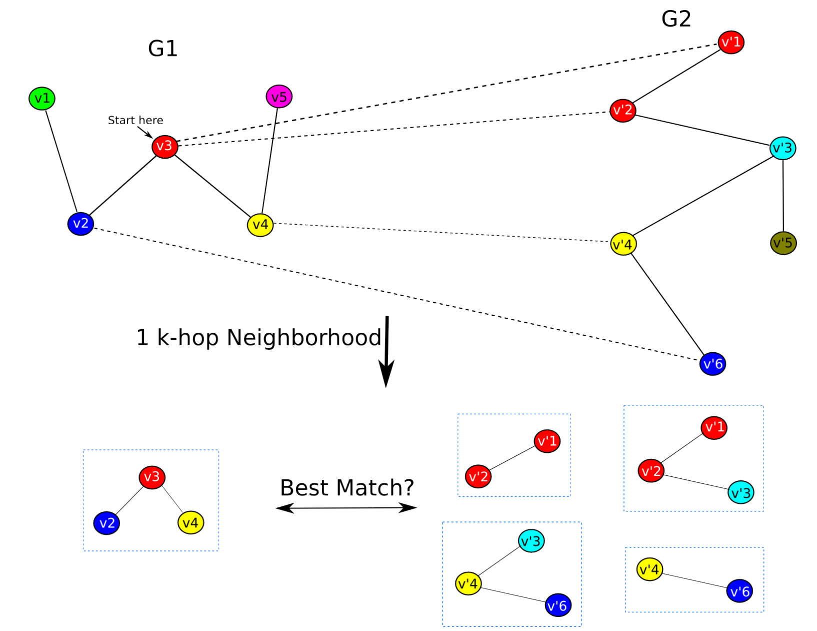


Figure 2.1 General schematic of the k-hop neighborhood alignment algorithm. The input to the algorithm are two graphs (G_1 and G_2) with corresponding relationships among their nodes using mapping matrix $\mathbf{P}$ (similarly colored nodes are sequence homologous according to a BLAST search, for example $P_{v_2 v'_6} = 1$). The algorithm starts at an arbitrary vertex in G_1 (red vertex in the figure) and constructs a k-hop neighborhood around the starting vertex (1-hop neighborhood in the figure). The algorithm then matches each of the nodes in the 1-hop neighborhood subgraph from G_1 to nodes in G_2 using mapping matrix $\mathbf{P}$. 1-hop subgraphs are then constructed around each of the matching vertices. The 1-hop subgraphs from G_2 are then compared using a scoring function (e.g. a graph kernel) to the 1-hop subgraph from G_1 and the maximum scoring match is returned.

k-hop neighborhood based network alignment algorithm is $O(bmg)$ where m is the number of nodes in the query network G_1, b is the maximum number of matches in the target network G_2 for any node in the query network, and g is the running time of the similarity measure or scoring function used to compare a pair of k-hop subnetworks.

2.3.1.2 Decomposing Networks Into Clusters

A graph clustering based alignment algorithm works as follows: Given two node-labeled graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, and a mapping matrix $\mathbf{P}$ that associates each vertex in V_1 with zero or more vertices in V_2, we first extract collections of subgraphs $H_1 = \{C_1, C_2, C_3, ...C_l\}$ and $H_2 = \{Z_1, Z_2, Z_3, ...Z_w\}$ from G_1 and G_2 respectively. In principle, any graph clustering

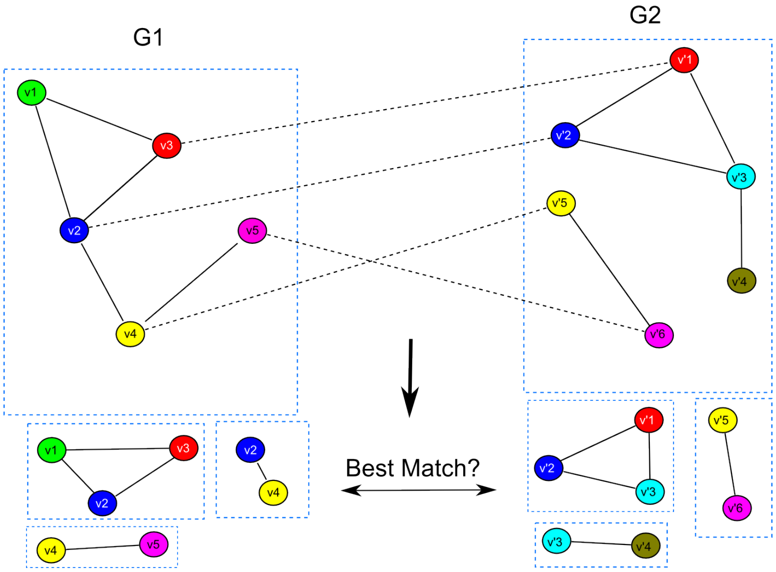


Figure 2.2 Schematic for the cluster-based alignment algorithm. The input to the algorithm are two graphs (G_1 and G_2) with corresponding relationships among their nodes using mapping matrix $\mathbf{P}$ (similarly colored nodes are sequence homologous according to a BLAST search, for example $P_{v_2 v_2'} = 1$). Subgraphs are generated from G_1 and G_2 using a graph clustering algorithm (e.g. bicomponent clusterer that finds biconnected subgraphs) and the subgraphs from G_1 are compared against the subgraphs from G_2 to find the best matching subgraphs using an appropriate scoring function.

algorithm may be used to construct the subgraph sets H_1 and H_2. In our experiments, we used the bicomponent clusterer as implemented in the JUNG (Java Universal Network/Graph) framework (123; 163) to extract H_1 and H_2. Briefly, the bicomponent clusterer searches for all biconnected components (graphs that cannot be disconnected by removing a single node/vertex (68)) by traversing a graph in a depth-first manner (please see (111) for more details). Once the subgraph sets H_1 and H_2 of the biconnected subgraphs of G_1 and G_2 (respectively) are extracted, an all vs. all comparison is conducted to identify for each subgraph in H_1, the best matching subgraph in H_2 using a scoring function (e.g. a graph kernel, see figure 2.2). The running time complexity of this algorithm is $O(lwg)$ where l is the number of clusters extracted from the query network G_1, w is the number of clusters extracted from the target network G_2 , and g is the running time of the scoring function used to compare a pair of clusters (subgraphs).

2.3.2 Conquer: Scoring Functions

We now proceed to describe the similarity measures or scoring functions used to compare a pair of subgraphs (e.g., a pair of k-hop subgraphs or a pair of bi-component clusters described above).

2.3.2.1 Shortest Path Graph Kernel

The shortest path graph kernel was first described by Borgwardt and Kriegel (22). As the name implies, the kernel compares the length of the shortest paths between any two nodes in a graph based on a pre-computed shortest-path distance. The shortest path distances for each graph may be computed using the Floyd-Warshall algorithm as implemented in the CDK (Chemistry Development Kit) package (143). We modified the Shortest-Path Graph Kernel to take into account the sequence homology of nodes being compared as computed by BLAST (3). The shortest path graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs, bicomponent clusters extracted from G_1 and G_2 respectively) is given by:

$$K(Z_{G_1}, Z_{G_2}) = \log \left[\sum_{v_i^1, v_j^1 \in Z_{G_1}} \sum_{v_k^2, v_p^2 \in Z_{G_2}} \delta(v_i^1, v_k^2) \times \delta(v_j^1, v_p^2) \times d(v_i^1, v_j^1) \times d(v_k^2, v_p^2) \right] \tag{2.1}$$

where $\delta(v_x^1, v_y^2) = \frac{BlastScore(v_x^1, v_y^2) + BlastScore(v_y^2, v_x^1)}{2}$. $d(v_i^1, v_j^1)$ and $d(v_k^2, v_p^2)$ are the lengths of the shortest paths between v_i^1,v_j^1 and v_k^2,v_p^2 computed by the Floyd-Warshall algorithm. The runtime of the Floyd-Warshall Algorithm is $O(n^3)$. The shortest path graph kernel has a runtime of $O(n^4)$ (where n is the maximum number of nodes in larger of the two graphs being compared). Please see figure 2.3 for a general outline of the comparison technique used by the shortest-path graph kernel.

2.3.2.2 Random Walk Graph Kernel

The random walk graph kernel (157) has been previously utilized by Borgwardt et al. (23) to compare protein-protein interaction networks. The random walk graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs, bicomponent clusters extracted from G_1 and G_2 respec-

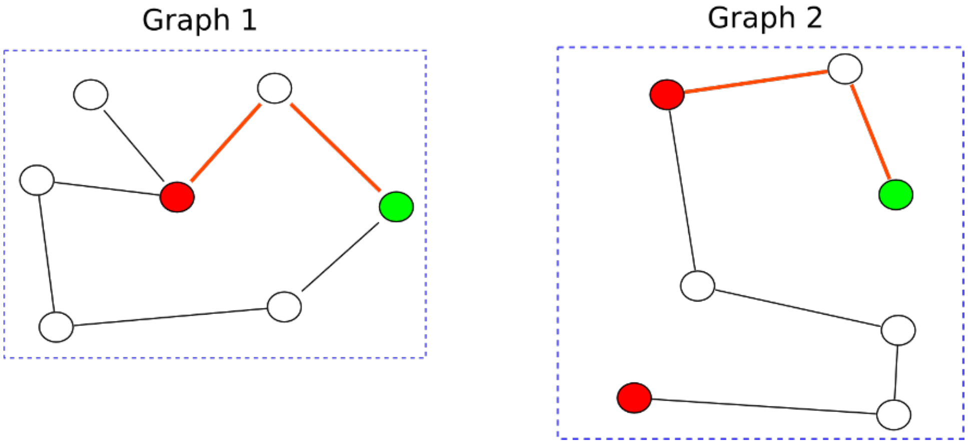


Figure 2.3 An example of the graph matching conducted by the shortest path graph kernel. Similarly colored nodes are sequence homologous according to a BLAST search. As can be seen from the figure, the graph kernel compares the lengths of the shortest paths around homologous vertices across the two graphs. The red edges show the matching shortest path in both graphs as computed by the graph kernel. The shortest path distance graph kernel takes into account the sequence homology score for the matching vertices across the two graphs as well as the distances between the two matched vertices within the graphs.

tively) is given by:

$$K(Z_{G_1}, Z_{G_2}) = p \times (\mathbf{I} - \lambda K_x)^{-1} \times q \tag{2.2}$$

where $\mathbf{I}$ is the identity matrix, λ is a user-specified variable controlling the length of the random walks (a value of 0.01 was used for the experiments in this paper), K_x is an $nm \times nm$ matrix (where n is the number of vertices in Z_{G_1} and m is the number of vertices in Z_{G_2} resulting from the Kronecker product $K_x = Z_{G_1} \otimes Z_{G_2}$, specifically,

$$K_{\alpha\beta} = \delta(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}), \alpha \equiv m(i-1) + k, \beta \equiv m(j-1) + l \tag{2.3}$$

Where $\delta(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}) = \frac{BlastScore(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}) + BlastScore(Z_{G_{2_{kl}}}, Z_{G_{1_{ij}}})}{2}$; p and q are $1 \times nm$ and $nm \times 1$ vectors used to obtain the sum of all the entries of the inverse expression $((\mathbf{I} - \lambda K_x)^{-1})$.

We adapted the random walk graph kernel to align protein-protein interaction networks by taking advantage of the reciprocal BLAST hits (RBH) among the proteins in the networks from different species (74). Naive implementation of our modified random-walk graph kernel, like the original random-walk graph kernel (157), has a runtime complexity of $O(r^6)$ (where $r = max(n, m)$). This is due to the fact that the product graph's adjacency matrix is $nm \times nm$,

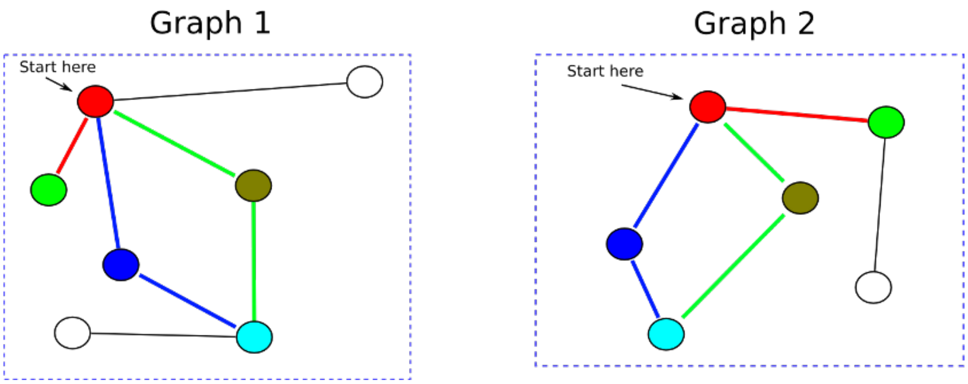


Figure 2.4 An example of the graph matching conducted by the random walk graph kernel. Similarly colored vertices are sequence homologous according to a BLAST search. As can be seen from the figure, the graph kernel compares the neighborhood around the starting vertices in each graph using random walks. Colored edges indicate matching random walks across the two graphs of up to length 2. The random walk graph kernel takes into account the sequence homology of the vertices visited in the random walks across the two graphs as well as the general topology of the neighborhood around the starting vertex.

and the matrix inverse operation takes $O(h^3)$ time, where h is the number of rows in the matrix being inverted (thus, the total runtime is $O((rm)^3)$ or $O(r^6)$ where $r = max(n,m)$). However, runtime complexity of the random walk graph kernel (and hence our modified random walk graph kernel) can be improved to $O(r^3)$ by making use of the Sylvester equations as proposed by Borgwardt et al. (23). Figure 2.4 illustrates the computation of the random walk graph kernel.

2.3.2.3 Page Rank (topology based)

Based on the work of Brin and Page (27) and implemented in the Java Universal Network/Graph Framework (123), the Page Rank score is calculated by first constructing a function measuring the transition probability around each node u in the undirected graph $G(V,E)$ as

$$(1-\alpha) * \left(\frac{1}{degree(u)}\right) + \alpha * \left(\frac{1}{|V|}\right)$$

where $|V|$ is the number of nodes/vertices in G, $degree(u)$ is the number of neighbors of node u and α is a constant parameter describing the influence from each node u. In our experiments,

α is set to 0.15. For nodes with no neighrbors, $\frac{1}{degree(u)}$ is set to 0. The transition probability of the Markov chain is then used to calculate the stationary probability of transitioning to each node in the graph. Thus, this scoring function compares the transition probabilities around each node in the graphs being compared and outputs a high score for graphs that have similar topologies as measured by their transition probabilities and a low score otherwise.

2.3.2.4 Kullback–Leibler divergence of degree distributions (topology based)

In lieu of the Euclidean distance function used above, the Kullback–Leibler (KL) divergence (103) can be used to calculate the difference between the two degree distributions from the k-hop subgraphs $H(Q,W)$ and $K(R,T)$. First, the degree distributions from the graphs are converted to n-dimentional vectors $\mathbf{h}$ and $\mathbf{k}$, respectively and the KL divergence between the distributions can then be calculated as follows:

$$\sum_{i=1}^{n} \frac{\mathbf{h}_i}{sum(\mathbf{h})} \log_2 \left(\frac{\frac{\mathbf{h}_i}{sum(\mathbf{h})}}{\frac{\mathbf{k}_i}{sum(\mathbf{k})}} \right)$$

The advantage of this approach is that it is not as sensitive as Euclidean distance to bin size or size of the graph due to the normalization procedure required to convert the degree frequencies to probabilities. It is also relatively quick to calculate compared to the Random Walk and Shortest Path Distance graph kernels.

2.3.2.5 Chi-square test between degree distributions (topology based)

The chi-square test (141) can also function as a similarity measure between degree distributions. The degree distributions from the k-hop subgraphs $H(Q,W)$ and $K(R,T)$ are converted to n-dimentional vectors $\mathbf{h}$ and $\mathbf{k}$, respectively and Pearson's cumulative test statistic is calculated between the distributions as follows:

$$\sum_{i=1}^{n} \frac{\mathbf{k}_i - \mathbf{h}_i}{\mathbf{h}_i}$$

Although this approach is slightly sensitive to large differences in the sizes of the graphs being compared, it provides a rigid statistical comparison between the distributions and is less likely to be skewed by slight fluctuations between the degree distributions.

2.3.2.6 Pearson correlation between degree distributions (topology based)

Pearson's product moment correlation coefficient measures the linear dependence between the two degree distributions represented as n-dimentional vectors $\mathbf{h}$ and $\mathbf{k}$ from the k-hop subgraphs $H(Q,W)$ and $K(R,T)$.

$$\frac{\sum_{i=1}^{n}(\mathbf{h}_i - \bar{\mathbf{h}})(\mathbf{k}_i - \bar{\mathbf{k}})}{\sqrt{\sum_{i=1}^{n}(\mathbf{h}_i - \bar{\mathbf{h}})^2}\sqrt{\sum_{i=1}^{n}(\mathbf{k}_i - \bar{\mathbf{k}})^2}}$$

2.3.2.7 Spearman Rank correlation between degree distributions (topology based)

Spearman's rank correlation measures the linear dependence between the ranks of the n-dimentional vectors $\mathbf{h}$ and $\mathbf{k}$ from the k-hop subgraphs $H(Q,W)$ and $K(R,T)$. This non-parametric correlation measure is more robust in dealing with frequency distributions that may have a large discrepancy in their frequencies compared to their ranks. Spearman's rank correlation is defined as

$$1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}$$

Where d_i is defined as the difference between the ranks of the raw frequency counts $\mathbf{h}_i$ and $\mathbf{k}_i$

2.4 Summary and Discussion

Aligning biomolecular networks from different species, tissues and conditions allows offers a powerful approach to discover shared components that can help explain the observed phenotypes. Specifically, applications of network alignment allow the discovery of conserved pathways among different species (88; 145), finding protein groups that are relevant to disease (77; 108), discovery of the chemical mechanism of metabolic reactions (134; 91) and more

(172; 92; 137; 17; 1). We have explored a novel class of graph kernel based polynomial time algorithms for aligning biomolecular networks. The proposed algorithms align large biomolecular networks by decomposing them into easy to compare substructures. The resulting subnetworks are compared using graph kernels as scoring functions. The modularity of kernels (35) offers the possibility of constructing composite kernel functions using existing kernel functions that capture different but complementary notions of similarity between graphs (23).

The runtime complexity of the k-hop neighborhood based alignment algorithm is $O(bmg)$ where m is the number of nodes in the query network G_1, b is the maximum number of matches in the target network G_2 for any node in the query network, and g is the running time of the similarity measure or scoring function used to compare a pair of k-hop subnetworks. The running time complexity of this algorithm is $O(lwg)$ where l is the number of clusters extracted from the query network G_1 , w is the number of clusters extracted from the target network G_2 , and g is the running time of the scoring function used to compare a pair of clusters (subgraphs). In comparison, the run-time complexity of NetworkBLAST-M ($O((np)^d s 3^s)$), where n is the number of nodes in each of the networks, s the number of networks, p an upper bound on the node degree and d the number of *seed spines* used to generate the alignment. In the special case of pairwise network alignment (s=2), the run-time complexity of NetworkBLAST reduces to $O((np)^d)$. The runtime complexity of HopeMap is linear in terms of the total number of nodes and edges in the alignment graph (149), which is $O(2n + 2n^2)$ in terms of the input graphs (where each input graph has at most n nodes).

The k-hop network neighborhood based and bicomponent clustering based protein-protein interaction network alignment algorithms are implemented in BiNA (`http://www.cs.iastate.edu/~ftowfic`), an open source Biomolecular Network Alignment toolkit. The current implementation includes variants of the shortest path and random walk graph kernels for computing similarity between pairs of subnetworks. The modular design of BiNA allows the incorporation of alternative strategies for decomposing networks into subnetworks and alternative similarity measures (e.g., kernel functions) for computing the similarity between subnetworks. Some interesting directions for further work on the biomolecular network alignment algorithms include:

- Design of alternative measures of performance for assessing the quality of the generated network alignments.

- Algorithms for aligning networks that contain directed links, such as transcriptional regulatory networks, multiple types of nodes (proteins, DNA, RNA) and multiple types of links.
- Extensions that allow the alignment of multiple networks.
- The use of more sophisticated graph-clustering algorithms (such as MCL (49)).
- Automated tuning of parameters (e.g λ for the random walk kernel) using parameter learning techniques (56).
- Optimizations that reduce the runtime memory requirements of the algorithm.

Acknowledgments

This research was supported in part by an Integrative Graduate Education and Research Training (IGERT) fellowship to Fadi Towfic, funded by the National Science Foundation grant (DGE 0504304) to Iowa State University and a National Science Foundation Research Grant (IIS 0711356) to Vasant Honavar. The authors are grateful to the WABI-09 anonymous referees for their helpful comments on the manuscript.

CHAPTER 3. COMPARATIVE ANALYSIS OF TOPOLOGICAL VS. NODE LABEL-BASED NETWORK ALIGNMENT

Paper submitted to ACM Bioinformatics and Computational Biology Conference

Fadi Towfic and Vasant Honavar

Abstract

With the advent of high-throughput methods for the generation of protein interaction and gene-expression networks, an increasing number of systematic studies comparing protein and gene interactions across tissues, organisms and systems are becoming available. As more expression and interaction data become available, algorithms for analyzing networks constructed from such large datasets must be able to deal with tens to hundreds of networks that have thousands to tens of thousands of genes and millions of edges. We have explored a set of scoring functions that measure similarity between networks based on node-annotation as well as local topology. Our results suggest a two-step framework for speeding up alignments of large networks by (1) optimally exploiting topological information to quickly compare global properties of networks based on their structure, and (2) refining the comparison by conducting a thorough alignment that exploits node labels and other external information for finding matching nodes. We provide implementations of our algorithms as part of the Biomolecular Network Alignment (BiNA) Toolkit.

3.1 Introduction

The advent of high-throughput methods for the generation of protein interaction and gene-expression networks has enabled systematic studies comparing protein and gene interactions across tissues, organisms and systems. As more expression and interaction data become available, algorithms for analyzing such networks must be able to deal with large datasets of tens to hundreds of networks that have thousands to tens of thousands of genes and millions of edges (142; 73). Of particular interest is the problem of comparing and aligning multiple networks (e.g., those generated from measurements taken under different conditions, different tissues, or different organisms) (139).

Finding conserved subnetworks among a set of input networks may be utilized for the discovery of conserved pathways among different species (88; 145), finding protein groups that are relevant to disease (77; 108), discovery of the chemical mechanism of metabolic reactions (134; 91) and more (92; 137; 17; 1). Currently, several algorithms are available for comparing and aligning protein interaction networks. One class of network alignment algorithms utilizes node-labels based on sequence, phylogenetic, or orthology annotation information (81; 82; 57; 56; 149; 107; 152). Some examples of algorithms in this class are: MaW-ISh (97) that takes into account protein duplication events as well as interaction loss/gain events between pairs of proteins to align networks; IsoRankN (107) that maximizes the overall match across a set of input networks by relying on similarity of neighborhoods between nodes; Hopemap (149) which uses sequence homology together with InParanoid orthology groups (120) and GO annotations to establish correspondences between proteins across two networks being aligned. Hopemap-ko, a variant of Hopemap (149), uses KEGG orthologs (84) to align pairs of proteins across the two networks. NetworkBLAST-M (81) uses phylogeny to drive the alignment whereas NetworkBLAST-ko exploits KEGG orthologs (84) to establish correspondences between proteins across networks. Graemlin 2.0 (56) that utilizes a feature-based scoring function (incorporating penalties for protein deletion, duplication, mutation) and a phylogenetic (species) tree to guide an approximate global alignment of multiple networks. Yu et al. (169) have proposed an algorithm for aligning gene regulatory networks by by identifying DNA binding sites that are conserved across proteins in the two networks. Pinter et al. (127) and Ay et al. (6) have recently introduced heuristic algorithms for aligning metabolic pathways.

Recently, another class of network comparison algorithms has been proposed. This class relies strictly on local network topology to draw correspondence between nodes across two or more networks (102; 101). GRAAL (101) (GRAph ALigner), which utilizes graphlet topological signatures around nodes to measure the similarity of neighborhoods around nodes is currently the only algorithm that can align networks strictly based on topology. The information encoded in the topology of networks alone has been shown to contain enough signal to ascertain phenotype (129), essential proteins (79), and cellular states (31; 94). Furthermore, network alignments in general have been successfully utilized to reconstruct phylogenetic relationships among sets of species (101; 152). While topological information has been very useful for comparing networks and extracting important biological information from network models, the relationship between node labels and topology have not been fully explored in the context of network alignment. Specifically, for a fixed alignment strategy, the relative contributions of topological and node-label information have not been systematically explored in the literature.

Against this background, we adapted our network alignment algorithm, BiNA (Biomolecular Network Alignment) (152; 151; 154), to include several scoring functions that calculate similarity between networks strictly based on the topologies of the networks. Our topology-based scoring functions include measures based on Page Rank, Kullback-Leibler divergence, Chi-square test, pearson correlation, and spearman rank correlation between degree distributions. We sought to explore how node-labels in networks, specifically sequence-based labels, contribute to network alignment performance in the context of finding subgraphs with significantly enriched GO (Gene Ontology) terms and reconstructing phylogenetic relationships between species. Our results suggest a two-step framework for speeding up alignments of large networks by (1) optimally exploiting topological information to quickly compare global properties of networks based on their structure, and (2) refining the comparison by conducting a thorough alignment that exploits node labels and other external information for finding matching nodes.

The rest of the paper is organized as follows: Section 2 introduces the dataset and methods for aligning two biomolecular networks and describes our approach for exploiting the neighborhood similarity measures for aligning networks and our experimental setup. Section 3 describes our experimental results. Section 4 concludes with a summary of the main contributions of the paper in the broader context of related literature and a brief outline of some directions for further research.

3.2 Materials and methods

The following section will introduce the datasets used in the analysis and define the network alignment approach as well as each of the scoring functions used for matching nodes across all the protein-protein interaction networks used in the experiments. Furthermore, we introduce our evaluation approaches to determine how the performance of the alignment algorithm changes with respect to the different information utilized in the alignment (node-label based alignments vs pure topological alignments) and different scoring functions used in the experimental setup.

3.2.1 Network Alignment Algorithm

The proteins in the DIP protein-protein interaction networks for mouse, human, yeast, and fly were matched using BLAST as shown in figure 3.1. As can be seen from the figure, protein-protein interaction networks are represented as two labeled graphs (graphs 1 and 2) with weighted edges connecting sequence-homologous nodes across the two graphs. The BLAST similarity scores are taken into account when comparing the neighborhoods around each of the vertices in the graphs to reconstruct the KEGG orthologs. This graph representation is similar to the representations used by NetworkBLAST (81), HopeMap (149), and Graemlin 2.0 (56). A k-hop neighborhood-based approach to alignment uses the notion of k-hop neighborhood. The k-hop neighborhood of a vertex $v_x^1 \in V_1$ of the graph $G_1(V_1, E_1)$ is simply a subgraph of G_1 that connects v_x^1 with the vertices in V_1 that are reachable in k hops from v_x^1 using the edges in E_1. Given two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, a mapping matrix $\mathbf{P}$ that associates each vertex in V_1 with zero or more vertices in V_2 (the matrix $\mathbf{P}$ can be constructed based on BLAST matches) and a user-specified parameter k, we construct for each vertex $v_x^1 \in V_1$ its corresponding k-hop neighborhood C_x in G_1. We then use the mapping matrix $\mathbf{P}$ to obtain the set of matches for vertex v_x^1 among the vertices in V_2; and construct the k-hop neighborhood Z_y for each matching vertex v_y^2 in G_2 and $P_{v_x^1 v_y^2} = 1$. Let $S(v_x^1, G_2)$ be the resulting collection of k-hop neighborhoods in G_2 associated with the vertex v_x^1 in G_1. We compare each k-hop subgraph C_x in G_1 with each member of the corresponding collection $S(v_x^1, G_2)$ to identify the k-hop subgraph of G_2 that is the best match for C_x (based on a chosen similarity measure).

3.2.1.1 Matching Based on Node Labels

In the k-Hop alignment algorithm, potential matches between nodes may be drawn based on node labels (e.g., sequence homology between nodes based on BLASTp scores). The schematic of the k-hop matching algorithm for $k = 1$ is shown in figure 3.1. The algorithm starts by constructing a 1-hop vertex-induced subgraph around each node in Network 1 (list of nodes is shown in the "Node from Network 1 column). After each of the 1-hop subgraphs is constructed for the nodes in network 1 (see "1-hop subgraph from Network 1" column), a 1-hop vertex-induced subgraph is also constructed around each homologous node in network 2 (see "Possible match from Network 2" for possible matching subgraphs from network 2, and "Corresponding node from Network 2" for each respective matching node from Network 2). A scoring function (outlined in the "Scoring Functions" section below) is then used to estimate the best matching subgraph from network 1 for each 1-hop neighborhood graph around nodes from network 2.

3.2.1.2 Matching based on topology

The k-Hop alignment algorithm, potential matches between nodes may be calculated strictly based on the topology of the neighborhoods around each possible matching nodes. Using this method, sequence homology (node colors) are completely ignored and, instead, a 1-hop vertex induced subgraph is constructed for each node in network 2. The scoring function used must then be able to differentiate good matches around the nodes strictly based on graph topology with no sequence information to restrict the possible matches. The algorithm starts by constructing a 1-hop vertex-induced subgraph around each node in Network 1. After each of the 1-hop subgraphs is constructed for the nodes in network 1, a 1-hop vertex-induced subgraph is also constructed around each node in network 2. A scoring function (outlined in the "Scoring Functions" section below) is then used to estimate the best matching subgraph from network 1 for each 1-hop neighborhood graph around nodes from network 2.

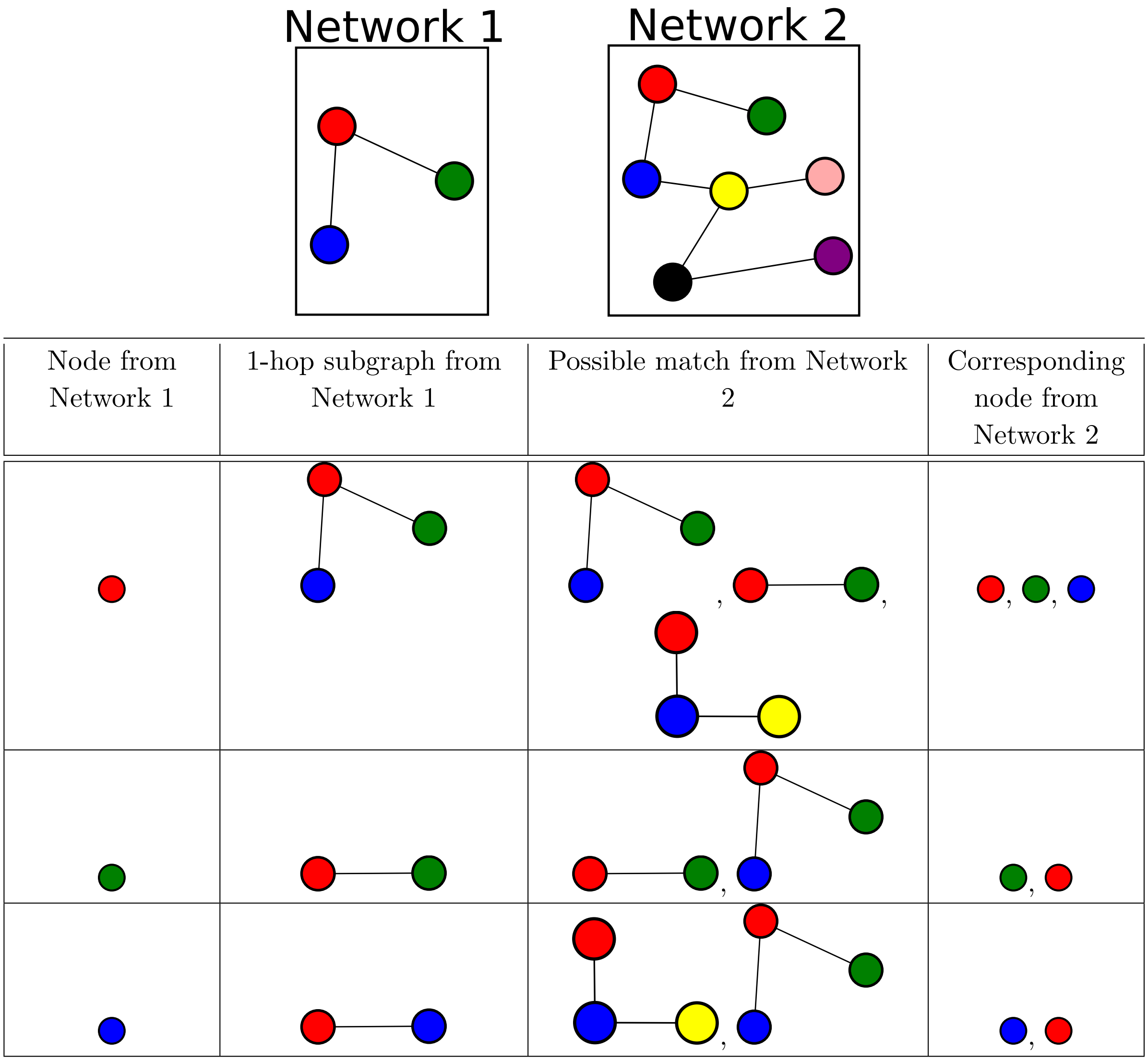

Network 1
Network 2
Node from Network 1
1-hop subgraph from Network 1
Possible match from Network 2
Corresponding node from Network 2

Figure 3.1 Schematic of k-hop network alignment algorithm (with $k = 1$ in this example) using sequence-homology to label (color) nodes. In this figure, sequence-homologous nodes as detected by BLASTp are given the same color. Please refer to the text for a full description of the algorithm. Briefly, the algorithm constructs a 1-hop neighborhood for each node in network 1 and uses a scoring function to calculate the best matching neighborhood in network 2 based on homologous nodes (similarly colored nodes) in network 2.

3.2.2 Scoring Functions

In this section, we will introduce our functions used to calculate scores for possible matches between k-hop subgraphs based on node-labels and topology (dubbed "Node-label based" below) or strictly topology based scoring functions based on comparing the degree distributions or pagerank of k-hop subgraphs (dubbed "topology based" below).

3.2.2.1 Shortest path distance graph kernel (node-label based)

As originally described by Borgwardt and Kriegel (22), the shortest path graph kernel measures the similarity of two given graphs based on the number of matching shortest path distances between them. We have previously adapted this kernel to take into account the node-labels as measured by BLAST homology scores between nodes (152). The kernel compares the length of the shortest paths between any two nodes in a graph based on a pre-computed shortest-path distance. The shortest path distances for each graph may be computed using the Floyd-Warshall algorithm, for example. We modified the Shortest-Path Graph Kernel to take into account the sequence homology of nodes being compared as computed by BLAST (3). Thus, the shortest path graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs from G_1 and G_2 respectively) is given by:

$$K(Z_{G_1}, Z_{G_2}) = \log\left[\sum_{v_i^1, v_j^1 \in Z_{G_1}} \sum_{v_k^2, v_p^2 \in Z_{G_2}} prod(v_i^1, v_j^1, v_k^2, v_p^2)\right]$$

where $prod(v_i^1, v_j^1, v_k^2, v_p^2) = \delta(v_i^1, v_k^2) \times \delta(v_j^1, v_p^2) \times d(v_i^1, v_j^1) \times d(v_k^2, v_p^2)$. The BLAST homology score is defined as

$$\delta(v_x^1, v_y^2) = \frac{BlastScore(v_x^1, v_y^2) + BlastScore(v_y^2, v_x^1)}{2}$$

$d(v_i^1, v_j^1)$ and $d(v_k^2, v_p^2)$ are the lengths of the shortest paths between v_i^1,v_j^1 and v_k^2,v_p^2 computed by the Floyd-Warshall algorithm. The shortest path graph kernel has a runtime of $O(n^4)$ (where n is the maximum number of nodes in larger of the two graphs being compared).

3.2.2.2 Random walk graph kernel (node-label based)

As originally described by Vishwanathan et al. (157) and utilized by Borgwardt et al. (23), the random walk graph kernel compares the transition probabilities from one node to another across two graphs. In our previous work, we modified this kernel to take into account the node-labels as measured by BLAST homology scores between nodes (152). Briefly, the random walk graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs extracted from G_1 and G_2 respectively) is given by:

$$K(Z_{G_1}, Z_{G_2}) = p \times (\mathbf{I} - \lambda K_x)^{-1} \times q \tag{3.1}$$

where $\mathbf{I}$ is the identity matrix, λ is a user-specified variable controlling the length of the random walks (a value of 0.01 was used for the experiments in this paper), K_x is an $nm \times nm$ matrix (where n is the number of vertices in Z_{G_1} and m is the number of vertices in Z_{G_2} resulting from the Kronecker product $K_x = Z_{G_1} \otimes Z_{G_2}$, specifically,

$$K_{\alpha\beta} = \delta(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}), \alpha \equiv m(i-1)+k, \beta \equiv m(j-1)+l \tag{3.2}$$

Where $\delta(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}) = \frac{BlastScore(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}) + BlastScore(Z_{G_{2_{kl}}}, Z_{G_{1_{ij}}})}{2}$; p and q are $1 \times nm$ and $nm \times 1$ vectors used to obtain the sum of all the entries of the inverse expression $((\mathbf{I} - \lambda K_x)^{-1})$.

Our modified random walk graph kernel can align protein-protein interaction networks and gene-coexpression networks by taking advantage of the reciprocal BLAST hits (RBH) among the proteins in the networks from different species (74). Naive implementation of our modified random-walk graph kernel, like the original random-walk graph kernel (157), has a runtime complexity of $O(r^6)$ (where $r = max(n,m)$). This is due to the fact that the product graph's adjacency matrix is $nm \times nm$, and the matrix inverse operation takes $O(h^3)$ time, where h is the number of rows in the matrix being inverted (thus, the total runtime is $O((rm)^3)$ or $O(r^6)$ where $r = max(n,m)$). However, runtime complexity of the random walk graph kernel (and hence our modified random walk graph kernel) can be improved to $O(r^3)$ by making use of the Sylvester equations as proposed by Borgwardt et al. (23).

3.2.2.3 Page rank (topology based)

Based on the work of Brin and Page (27) and implemented in the Java Universal Network/Graph Framework (123), the Page Rank score is calculated by first constructing a function measuring the transition probability around each node u in the undirected graph $G(V,E)$

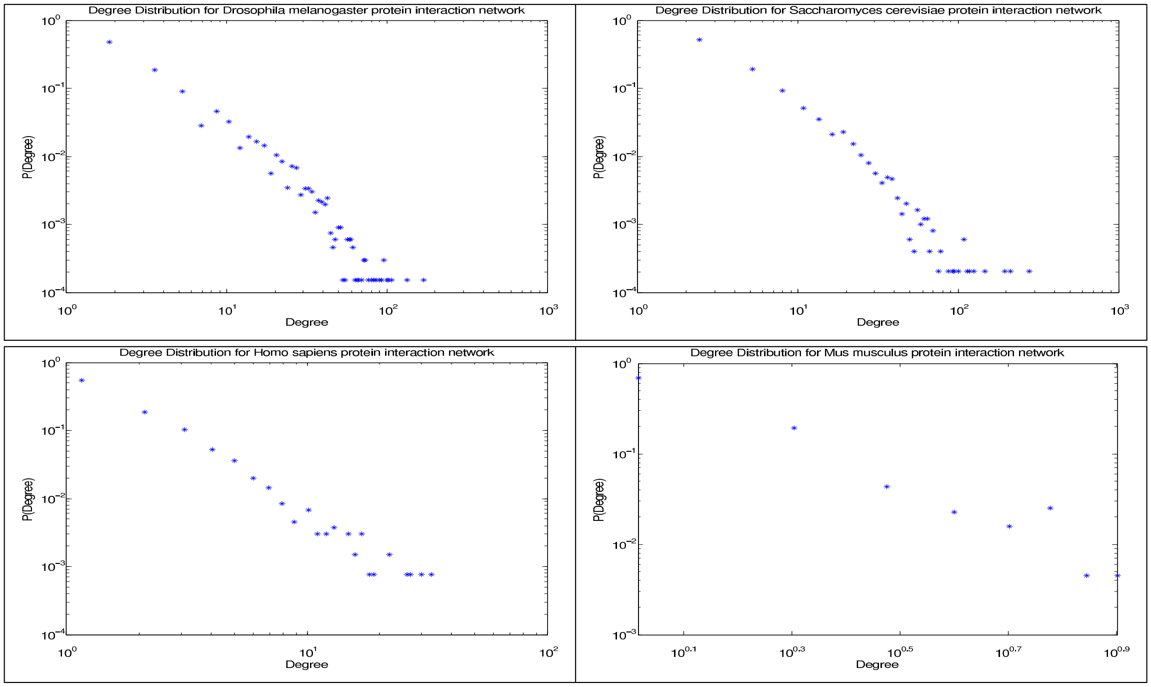


Figure 3.2 Example of degree distributions used for Euclidean distance, Kullback–Leibler divergence, Chi-square test, Pearson correlation, and Spearman Rank correlation for topology-based scoring between pairs of k-hop subgraphs. The x-axis is the degree of a node and the y-axis is the number of nodes with that degree (P(Degree)). As can be seen from the figure, the protein interaction networks exhibit scale-free like behavior as described by Barabasi and Oltvai (17)

as

$$(1-\alpha) * \left(\frac{1}{degree(u)}\right) + \alpha * \left(\frac{1}{|V|}\right)$$

where $|V|$ is the number of nodes/vertices in G, $degree(u)$ is the number of neighbors of node u and α is a constant parameter describing the influence from each node u. In our experiments, α is set to 0.15. For nodes with no neighrbors, $\frac{1}{degree(u)}$ is set to 0. The transition probability of the Markov chain is then used to calculate the stationary probability of transitioning to each node in the graph. Thus, this scoring function compares the transition probabilities around each node in the graphs being compared and outputs a high score for graphs that have similar topologies as measured by their transition probabilities and a low score otherwise.

3.2.2.4 Kullback–Leibler divergence of degree distributions (topology based)

In lieu of the Euclidean distance function used above, the Kullback–Leibler (KL) divergence (103) can be used to calculate the difference between the two degree distributions from the k-hop subgraphs $H(Q, W)$ and $K(R, T)$. First, the degree distributions from the graphs are converted to n-dimensional vectors $\mathbf{h}$ and $\mathbf{k}$, respectively and the KL divergence between the distributions can then be calculated as follows:

$$\sum_{i=1}^{n} \frac{\mathbf{h}_i}{sum(\mathbf{h})} \log_2 \left(\frac{\frac{\mathbf{h}_i}{sum(\mathbf{h})}}{\frac{\mathbf{k}_i}{sum(\mathbf{k})}} \right)$$

The advantage of this approach is that it is not as sensitive as Euclidean distance to bin size or size of the graph due to the normalization procedure required to convert the degree frequencies to probabilities. It is also relatively quick to calculate compared to the Random Walk and Shortest Path Distance graph kernels.

3.2.2.5 Chi-square test statistic between degree distributions (topology based)

The chi-square test (141) statistic can also function as a similarity measure between degree distributions. The degree distributions from the k-hop subgraphs $H(Q, W)$ and $K(R, T)$ are converted to n-dimensional vectors $\mathbf{h}$ and $\mathbf{k}$, respectively and Pearson's cumulative test statistic is calculated between the distributions as follows:

$$\sum_{i=1}^{n} \frac{\mathbf{k}_i - \mathbf{h}_i}{\mathbf{h}_i}$$

Although this approach is slightly sensitive to large differences in the sizes of the graphs being compared, it provides a rigid statistical comparison between the distributions and is less likely to be skewed by slight fluctuations between the degree distributions.

3.2.2.6 Pearson correlation between degree distributions (topology based)

Pearson's product moment correlation coefficient measures the linear dependence between the two degree distributions represented as n-dimensional vectors $\mathbf{h}$ and $\mathbf{k}$ from the k-hop subgraphs $H(Q, W)$ and $K(R, T)$.

$$\frac{\sum_{i=1}^{n}(\mathbf{h}_i - \bar{\mathbf{h}})(\mathbf{k}_i - \bar{\mathbf{k}})}{\sqrt{\sum_{i=1}^{n}(\mathbf{h}_i - \bar{\mathbf{h}})^2}\sqrt{\sum_{i=1}^{n}(\mathbf{k}_i - \bar{\mathbf{k}})^2}}$$

3.2.2.7 Spearman rank correlation between degree distributions (topology based)

Spearman's rank correlation measures the linear dependence between the ranks of the n-dimensional vectors $\mathbf{h}$ and $\mathbf{k}$ from the k-hop subgraphs $H(Q, W)$ and $K(R, T)$. This non-parametric correlation measure is more robust in dealing with frequency distributions that may have a large discrepancy in their frequencies compared to their ranks. Spearman's rank correlation is defined as

$$1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)}$$

Where d_i is defined as the difference between the ranks of the raw frequency counts $\mathbf{h}_i$ and $\mathbf{k}_i$

3.2.3 Datasets

The yeast, fly, mouse and human protein-protein interaction networks were obtained from the Database of Interacting Proteins (DIP) release 1/26/2009 (136). The sequences for each dataset were obtained from uniprot release 14 (11). The DIP sequence ids were matched against their uniprot counterparts using a mapping table provided on the DIP website. All proteins from DIP that had obsolete uniprot IDs or were otherwise not available in release 14 of the uniprot database were removed from the dataset. The fly, yeast, mouse and human protein-protein interaction networks consisted of $6,645$, $4,953$, 424 and $1,321$ nodes and $20,010$, $17,590$, 384 and $1,716$ edges, respectively. The protein sequences for each dataset were downloaded from uniprot (11). BLASTp (3) with a cutoff of 1×10^{-10} was used to match protein sequences across species. The KEGG (Kyoto Encyclopedia of Genes and Genomes) (83) orthology and uniprot annotations for all species were downloaded from the KEGG website and matched against the uniprot id's for the proteins in the DIP datasets.

3.2.4 Evaluation of Alignment

3.2.4.1 Gene-ontology enrichment of matching subgraphs

We utilize Kalaev et al.'s approach to evaluate network alignments as described in the NetworkBLAST (82) and the HopeMap (149) papers. Recall from "Network Alignment Algorithm" section that the output of the alignment algorithm is a set of subgraphs S_1 and S_2 (corresponding to the query and target networks, respectively). The set of subgraphs $S_2 = \{Z_1, Z_2, Z_3, ..., Z_w\}$ in the target network are queried for overrepresented Gene Ontology (GO) categories from the biological process GO hierarchy (4). An implementation of the GO enchrichment algorithm (GOTermFinder (24) tool) was used to calculate the enrichment p-values (with p-value significance cutoff = 0.05) and corrected for multiple testing using the false discovery rate. GOTermFinder computes p-values given a set of GO annotations for a set of proteins in subgraphs $Z_{1..w}$ based on the number of proteins in the subgraph Z_x (where $1 \leq x \leq w$, and the number of vertices in Z_x is r) and the number of proteins in the genome of the target network (n) and their respective GO annotation. The hypergeometric distribution is utilized to calculate the p-value is computed based on the probability of k or more out of r proteins being assigned a given annotation (where k is the number of proteins in the subgraph Z_x possessing the GO category of interest), given that y of n proteins possess such an annotation in the genome in general. The number of subgraphs, f, that had one or more GO categories overrepresented is calculated (where $f \leq w$) and the fraction of subgraphs from the target network that had a significant number of GO categories overrepresented is then computed ($\frac{f}{w} \times 100$, % coherent subnetworks). Specificity of the alignment method is measured by the percent of coherent subnetworks discovered for each species while the sensitivity is indicated by the number of distinct GO categories covered by the functionally coherent subnetworks. The purpose of this evaluation approach is to determine whether or not the matching subgraphs found in the target network represent a functional module/pathway (functionally coherent subgraphs) based on the GO annotation of the proteins in the subgraph. We compare the results from running the network alignments using the various comparison strategies described in the "Scoring Functions" sections to our previous results (152) compared against NetworkBLAST-M and

HopeMap.

3.2.4.2 Construction of phylogenetic trees based on network alignment and bootstrapping

A set of symmetric 4×4 distance matrices using the alignment scores across the 4 networks was constructed. Each matrix was constructed using one of the seven scoring functions discussed in section 2. The distance matrix was normalized such that the diagonals contained 0 and the off diagonals contained the distance comparing the network from row i with network in column j where $1 \leq i, j \leq 4$ (where a distance of 0 implied a perfect match and distances greater than 0 denoted increasingly worse matches). A phylogeny based on each distance matrix was constructed using Phylip's (53) neighbor-joining program. The tree produced by phylip was bootstrapped (46; 52) by sampling randomly (with replacement) from all the nodes in the 4 networks 100 times and reconstructing the distance matrices 100 times, once for each bootstrap iteration. This random resampling results in 100 distance matrices that are then fed into the same neighbor-joining algorithm to construct 100 phylogenetic trees. Phylip's "consense" program was used to merge the 100 trees and to compute majority-rule consensus trees. The majority rule consensus approach has been shown to minimize the number of false groupings and provides a good summary of the posterior distribution over the trees that were used to construct the consensus tree (75). TreeView (124) was used to visualize the trees.

3.3 Results

3.3.1 Performance as Measured by GO Enrichment

Detection of conserved subnetworks with a a significant number of enriched GO terms provides a general idea of the alignment algorithm's capability of detecting generally similar regions across two networks. Previously, we showed that BiNA is capable of detecting significantly GO-term enriched regions in networks compared to algorithms that exploit orthology relationships between nodes, as opposed to just sequence-level information that was utilized by BiNA (152). Furthermore, we showed that BiNA is also capable of detecting orthologs based on protein-protein interaction networks and gene-coexpression networks (154), making BiNA a viable basis for exploring how topology-based measures can best be utilized for aligning networks. Our first experiment utilized BiNA's label-based k-hop matching algorithm described in the "Matching Based on Node Labels" section. Briefly, this approach relies on sequence information to narrow down possible candidate matches for nodes from network 1 to nodes from network 2. Once the candidate nodes are obtained, their neighborhoods are compared based on one of the seven scoring functions described in section 2. The results from this experiment are shown in table 1.

As can be seen from table 1, scoring functions that utilized both sequence-level as well as topological signals (i.e., the Shortest Path and Random Walk functions) generally performed better compared to scoring functions that relied on topological information alone (i.e., Page Rank, Kullback-Leibler divergence, Chi-squared test, pearson and spearman correlation) if the observation is limited to strictly the same number of hops. However, increasing the size of the neighborhood around potential match candidates to 2 and 3 hops generally improves the performance of all scoring functions, especially the topological scoring functions.

This pattern is seen again in table 2, which compares the performance of scoring functions using an alignment that does not use any node-label information at all (see "Matching Based on Topology" section for details). This is expected since a larger neighborhood helps improve the topological signal around each node resulting in a more defined degree distribution. Although none of the topological scoring functions completely match the performance of the scoring functions that also exploit node labels (in the form of alignment scores between sequences), it should be noted that the running times for the topological functions are generally much quicker compared to the Shortest Path kernel ($O(n^4)$) and Random Walk Kernel ($O(n^6)$). Specifically, the Page Rank scoring function has $O(n^4)$ and the degree-distribution based functions have $O(n)$ running time, where n is the number of nodes in the neighborhoods being compared. Together, those results suggest that topology carries significant information that can help in detecting matching regions between any two networks. However, comparing the results of the same scoring functions between tables 1 and 2, it is clear that node labels are helpful in improving the performance (as measured by GO Term enrichment in matched subgraphs).

This suggests that algorithms that are able to exploit topology very well to align networks can further improve their performance by considering node labels.

3.3.2 Reconstruction of Phylogenetic Relationships

Although GO enrichment can provide a general measure of performance indicating the cohesiveness of detected matches (82; 81), the assumption of independence between GO Terms and gaps in the Gene Ontology annotation of some genes makes this measure's use for extrapolating the performance results to biological annotations problematic. As network alignments in general have been successfully utilized to reconstruct phylogenetic relationships among sets of species (101; 152). We sought to quantify the performance of the topological and node-label based scoring functions based on their ability to reconstruct known biological relationships between the species being aligned. Our full procedure is described in section 2. Briefly, we construct bootstrapped phylogenetic trees based on each of our seven scoring functions and alignment approach (label-based or pure topology based). The bootstrap values on the branches provide a confidence measure of the alignment based on the scoring function described. Figure 3 shows an example of the bootstrapped trees constructed based on the Random Walk Graph Kernel. The left panel of figure 3 shows the bootstrapped tree based on the labeled alignment and the right panel shows the bootstrapped tree based on the purely topological alignment. As can be seen from the figure, although both trees show similar topologies, the bootstrap values are higher for the alignment utilizing the node labels alongside the local network topology. This result is more clearly shown in table 3, which compares the bootstrap performance in reconstructing phylogenetic relationships between mouse-human and fly-yeast branches based on the label and strictly topological alignments. Taken together with the GO enrichment results, the experiments shown here indicate that node labels, if available, can be very useful for improving the performance of network alignment. Furthermore, topology-based alignments, if used as a general comparison to detect initial networks for alignment, can be used to significantly speed

Method	% GO in Sc	# GO in Sc	% GO in Dm	# GO in Dm	% GO in Mm	# GO in Mm	% GO in Hs	# GO in Hs
Network-BLAST-ko	**100**	**9**	**100**	**8**	**-**	**-**	**-**	**-**
Hope-Map-ko	**100**	**24**	**92**	**24**	**-**	**-**	**-**	**-**
SP 1Hop	**100**	**51**	**78**	**22**	**53**	**19**	**85**	**70**
RW 1Hop	**100**	**71**	**85**	**19**	**100**	**1**	**100**	**8**
SP 2Hop	**100**	**46**	**76**	**9**	**94**	**4**	**100**	**13**
RW 2Hop	**100**	**107**	**100**	**1**	**94**	**4**	**100**	**17**
PR 1Hop	91	62	54	36	50	30	66	47
KL 1Hop	79	292	32	135	40	51	44	59
Pearson 1Hop	79	293	32	135	48	46	59	41
Spearman 1Hop	79	293	32	135	48	46	59	41
Chi 1Hop	79	292	32	135	40	51	45	59
PR 2Hop	99	63	68	37	72	23	85	32
KL 2Hop	97	187	62	108	64	41	73	42
Pearson 2Hop	97	185	62	108	69	32	90	22
Spearman 2Hop	97	185	62	108	69	32	90	22
Chi 2Hop	97	187	68	37	64	41	74	41
PR 3Hop	100	8	76	7	68	13	86	18
KL 3Hop	98	45	62	24	68	31	67	25
Pearson 3Hop	98	45	61	24	69	27	66	13
Spearman 3Hop	98	45	61	24	69	26	66	13
Chi 3Hop	98	45	63	24	69	31	68	25

Table 3.1 Comparison of Graph Kernel Performance using BLAST to match initial node centers in K-Hop alignment between human (Hs), mouse (Mm), yeast (Sc) and fly (Dm). Bold entries are adapted from our previous results on K-hop alignments (152). The methods are denoted as SP (Shortest Path), RW (Random Walk), PR (Page Rank), KL (Kullback–Leibler divergence), Pearson (Pearson correlation), Spearman (Spearman rank correlation) and Chi (Chi-squared test statistic).

Method	% GO in Sc	# GO in Sc	% GO in Dm	# GO in Dm	% GO in Mm	# GO in Mm	% GO in Hs	# GO in Hs
PR 1Hop	99	9	63	11	98	4	31	5
KL 1Hop	70	9	80	4	2	1	3	1
Pearson 1Hop	70	9	80	4	0	0	3	2
Spearman 1Hop	89	13	80	4	0	0	3	2
Chi 1Hop	70	9	16	3	0	0	2	1
PR 2Hop	98	16	93	12	92	9	74	7
KL 2Hop	89	12	80	4	57	4	40	3
Pearson 2Hop	89	13	80	4	58	5	41	3
Spearman 2Hop	70	9	80	4	58	5	41	3
Chi 2Hop	89	12	80	4	59	4	41	3

Table 3.2 Comparison of Graph Kernel Performance using pure topological alignment

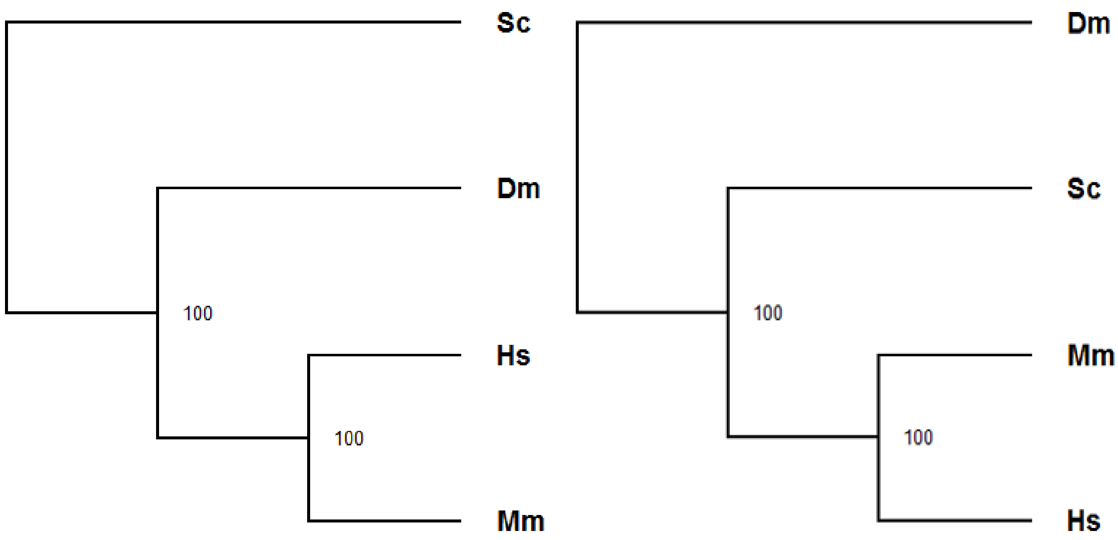


Figure 3.3 Comparison of bootstrapped trees constructed based on the labeled alignment using the KL scoring function **(left)** and the purely topological global comparison using the same scoring function **(right)** between human (Hs), mouse (Mm), yeast (Sc) and fly (Dm)

up network alignments. This is especially important in the case of gene-coexpression networks that can grow to tens of thousands of nodes and millions of edges.

3.4 Discussion and conclusions

With the availability of high-throughput methods for the generation of protein interaction and gene-expression networks, large systematic studies comparing protein and gene interactions across tissues, organisms and systems have become more common. Such studies regularly produce large gene expression and protein interaction data in the form of gene-coexpression and protein-protein interaction networks. Thus, algorithms for analyzing such networks must be able to deal with large datasets of tens to hundreds of networks that have thousands to tens of thousands of genes and millions of edges (142; 73). Of particular interest is the problem of comparing and aligning multiple networks (e.g., those generated from measurements taken under different conditions, different tissues, or different organisms) (139). Specifically, as more and more large networks become available for comparison, strategies for speeding up alignment algorithms become very important. While topological information has been very useful for comparing networks and extracting important biological information from network models, the

Method	Labeled Mouse-Human Bootstrap	Labeled Fly-Yeast Bootstrap	Topol. Mouse-Human Bootstrap	Topol. Fly-Yeast Bootstrap
SP 2Hop	100	100	100	0
RW 2Hop	100	100	100	0
PR 2Hop	97	99	0	100
KL 2Hop	100	100	100	0
Pearson 2Hop	100	100	100	0
Spearman 2Hop	100	100	100	0
Chi 2Hop	100	100	100	0
Average	**99.57**	**99.85**	85.71	14.28

Table 3.3 Comparison of the bootstrap performance in reconstructing phylogenetic relationships between mouse-human and fly-yeast branches

relationship between node labels and topology have not been fully explored in the context of network alignment. Specifically, the question of how information from network topology and node-labels can interplay and affect alignment performance given a fixed alignment strategy has not been fully addressed in the literature.

We have explored a set of scoring functions that measure similarity between networks based on node-annotation as well as local topology (Random Walk and Shortest Path scoring functions), as well as scoring functions that are strictly topology based (Page Rank, Kullback-Leibler divergence, Chi-squared test, pearson and spearman correlation). While the latter group of functions is significantly faster to compute (having computational complexity of $O(n)$ for chi-squared, pearson, and KL, $O(n \log(n))$ for spearman rank correlation, where n is number of nodes in the largest subgraph being compared) and generally perform well with respect to reconstructing biological/phylogenetic relationships, we have shown that node annotations can improve the performance even further at the cost of computational time (the shortest path graph kernel has a computational complexity of $O(n^4)$ and the random walk graph kernel has a complexity of $O(n^6)$).

In general, our label-based k-hop approach has a running time complexity of $O(bmg)$ (152)

where m is the number of nodes in the query network, b is the maximum number of matches (e.g., BLAST-based matches) in the target network for any node in the query network, and g is the running time of the similarity measure or scoring function used to compare a pair of k-hop subnetworks ($O(n)$ for Kullback-Leibler divergence, Chi-squared test and pearson, $O(n \log(n))$ for spearman correlation and $O(n^4)$ for Shortest path kernel and $O(n^6)$ for random walk kernel, where n is number of nodes in the largest subgraph being compared). In the naive case where no node labels/sequence similarity information is considered, b is equal to l, the total number of nodes in the target network. On the other hand, when node labels/sequence similarity information is used in determining the matches, $b << l$.

Thus, our results suggest a two-step framework for speeding up alignments of large networks by (1) optimally exploiting topological information to quickly compare global properties of networks based on their structure, and (2) refining the comparison by conducting a thorough alignment that exploits node labels and other external information for finding matching nodes[1].

The network alignment algorithms, both node-label and strictly topological, are implemented in BiNA (http://www.cs.iastate.edu/~ftowfic), an open source Biomolecular Network Alignment toolkit. The modular design of BiNA allows the incorporation of alternative strategies for decomposing networks into subnetworks and alternative similarity measures (e.g., scoring functions) for computing the similarity between nodes. Some interesting directions for further work on the biomolecular network alignment algorithms include the exploration of the use of topology in different types of networks (such as gene co-expression networks and transcriptional regulatory networks) for detecting topological matches and exploring integrated methods for exploiting new combinations of node labels generate speedy alignments without losing matching accuracy.

Acknowledgement This research was supported in part by a Cornette Fellowship award and an Integrative Graduate Education and Research Training (IGERT) fellowship to Fadi Towfic, funded by the National Science Foundation (NSF) to Iowa State University and through

[1] We have also conducted experiments to compare the global topology of the networks, which we unfortunately cannot show due to space constraints. The results from those experiments closely match our pure topological alignment results shown in tables 2 and 3

TeraGrid resources provided by San Diego Super Computer Center. This work has benefited from useful discussions with Rasna Walia, Li Xue and Yasser El-Manzalawy. The work of Vasant Honavar was supported by the National Science Foundation, while working at the Foundation. Any opinion, finding, and conclusions contained in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

CHAPTER 4. DETECTION OF GENE ORTHOLOGY FROM GENE CO-EXPRESSION AND PROTEIN INTERACTION NETWORKS

Paper originally appeared in BMC Bioinformatics, vol 7, 2010[1]

Fadi Towfic, Susan VanderPlas, Casey A. Oliver, Oliver Couture, Christopher K. Tuggle, M. Heather West Greenlee and Vasant Honavar

Abstract

Background: Ortholog detection methods present a powerful approach for finding genes that participate in similar biological processes across different organisms, extending our understanding of interactions between genes across different pathways, and understanding the evolution of gene families.

Results: We exploit features derived from the alignment of protein-protein interaction networks and gene-coexpression networks to reconstruct KEGG orthologs for *Drosophila melanogaster*, *Saccharomyces cerevisiae*, *Mus musculus* and *Homo sapiens* protein-protein interaction networks extracted from the DIP repository and *Mus musculus* and *Homo sapiens* and *Sus scrofa* gene coexpression networks extracted from NCBI's Gene Expression Omnibus using the decision tree, Naive-Bayes and Support Vector Machine classification algorithms.

Conclusions: The performance of our classifiers in reconstructing KEGG orthologs is compared against a basic reciprocal BLAST hit approach. We provide implementations of the resulting algorithms as part of BiNA, an open source biomolecular network alignment toolkit.

[1] *Copyright retained by authors*

4.1 Introduction

With the advent of fast and relatively inexpensive sequencing technology, it has become possible to access and compare genomes from a wide range of organisms including many eukaryotes as well as bacteria and archea through databases such as GenBank (18), Ensembl (58), PlantGDB (43) and others (33; 25; 21). The availability of genomes from such a wide range of organisms has enabled the comparison and analysis of evolutionary relationships among genes across organisms through the reconstruction of phylogenies (161), common pathways (83; 112), and comparing gene functions (133; 47). Of particular interest in this context is the problem of finding genes originating from a single gene from a common ancestor of the compared genomes (orthologs) (96). Ortholog detection methods present a powerful approach for finding genes that participate in similar biological processes across different organisms, extending our understanding of interactions between genes across different pathways, and understanding the evolution of gene families.

Several sequence-based approaches currently exist for finding orthologous genes among a set of genomes. For instance, one of the simplest methods is to utilize reciprocal best BLAST hits (3) across a set of species to identify orthologs (74). The COGs (Clusters of Orthologous Groups) approach (148), for example, defines orthologs as sets of proteins that are reciprocal best BLAST hits across a minimum of three species. Another possible approach utilized by databases such as InParanoid (120) and OrthoMCL (106) consists of an iterative BLAST search to construct the reciprocal BLAST hits, and a second step that clusters the reciprocal hits to achieve greater sensitivity. InParanoid uses a pre-defined set of rules to construct its clusters, while OrthoMCL utilizes a sequence-based Markov clustering algorithm for clustering its proteins/genes into ortholog groups. Other approaches, such as PhyOP (65), RAP (44) and others (133; 83; 161; 47) identify orthologous genes/proteins by utilizing phylogenetic analysis to explicitly exploit the evolutionary rates across the species being compared. Such approaches account for the different mutation rates accumulated by the various species being compared, thus allowing greater sensitivity in detecting the pairs of genes/proteins to be classified as orthologous. Methods such as those utilized by Fu et al. consider gene order and rearrangements in detecting orthologs (60). Recently, with the availability of large-scale analysis of protein-protein interactions, protein-protein interaction networks have also been considered in detecting orthologous genes. Ogata et al. utilized a graph comparison algorithm to compare protein-protein interaction networks and determined orthologs by matching the nodes in the protein-protein interaction graphs (122). Bandyopadhyay et al. utilized the PathBLAST pathway alignment algorithm to detect orthologs (13). Another method utilized by databases such as KEGG is to manually construct orthology groups based on a combination of features such as sequence similarity, pathway interactions, and phylogenetic analysis (112; 83).

Against this background, we explore a set of graph features that may be utilized in detecting orthologs based on sequence similarity as well as the similarity of their neighborhoods in protein-protein interaction and gene coexpression networks. Furthermore, we construct a set of classifiers that utilize the above features and compare the classifiers to the reciprocal BLAST hits approached for the reconstruction of KEGG orthologs (83). The basic idea behind our approach is to align a pair of protein-protein interaction/gene coexpression networks and scan the alignment for all possible matches that a node (protein) from one network can pair with in the other network. We then train decision tree (164), Naive-Bayes (117), Support Vector Machine (36), and an ensemble classifier (41) that utilize features from the alignment algorithm to identify KEGG orthologs and we compare the performance of the classifiers to the reciprocal BLAST hit method.

We utilize the alignment algorithms available as part of the BiNA (Biomolecular Network Alignment) toolkit (153) as well as graph features extracted from the aligned networks such as degree distribution, BaryCenter (163), betweenness (162) and HITS (Hubs and Authorities) (93) centrality measures. Our experiments with the fly, yeast, mouse and human protein-protein interaction networks extracted from DIP (Database of Interacting Proteins) (136) as well as the mouse and human gene expression data extracted from NCBI's Gene Expression Omnibus (GEO) (45) demonstrate the feasibility of the proposed approach for detecting KEGG orthologs.

4.2 Materials and methods

4.2.1 Dataset

The yeast, fly, mouse and human protein-protein interaction networks were obtained from the Database of Interacting Proteins (DIP) release 1/26/2009 (136). The sequences for each dataset were obtained from uniprot release 14 (11). The DIP sequence ids were matched against their uniprot counterparts using a mapping table provided on the DIP website. All proteins from DIP that had obsolete uniprot IDs or were otherwise not available in release 14 of the uniprot database were removed from the dataset. The fly, yeast, mouse and human protein-protein interaction networks consisted of $6,645$, $4,953$, 424 and $1,321$ nodes and $20,010$, $17,590$, 384 and $1,716$ edges, respectively. The protein sequences for each dataset were downloaded from uniprot (11). BLASTp (3) with a cutoff of 1×10^{-10} was used to match protein sequences across species. The KEGG (Kyoto Encyclopedia of Genes and Genomes) (83) orthology and uniprot annotations for all species were downloaded from the KEGG website and matched against the uniprot id's for the proteins in the datasets.

For detecting orthologs based on gene-coexpression networks, Affymetrix gene expression data was collected from the GEO database for experiments in selected tissues in pigs (*Sus scrofa*) (54), humans (*Homo sapiens*) (166), and mice (*Mus musculus*) (146). The collected tissues were: adrenal gland, hypothalamus, spleen, thyroid, liver, small intestine, stomach, fat, lymph node, skeletal muscle, olfactory bulb, ovary, and testes. All expression data were taken from healthy animals. Data from each tissue for a given species were obtained from the same Affy platform. Probe IDs contained in the data were matched with gene IDs, and all available probe expression values for each gene were averaged to obtain one expression value per gene per tissue. Gene sequences were collected from NCBI Entrez (110) and compared across species bidirectionally to identify gene homology. BLASTn (3) with a cutoff of 1×10^{-10} was used to match gene sequences across species. The KEGG (Kyoto Encyclopedia of Genes and Genomes) (83) orthology and entrez gene id annotations for all species were downloaded from the KEGG website and matched against the gene id's for the genes in the datasets. The microarray expression measures were utilized to compute the pairwise Spearman rank correlations between all pairs of genes were calculated, with links with with an absolute value correlation cutoff of 0.8 or higher being retained in the resulting weighted graph.

4.2.2 Graph representation of BLAST orthologs

The proteins in the DIP protein-protein interaction networks for mouse, human, yeast, and fly as well as the gene coexpression networks for mouse, human and pig from GEO were matched using BLAST as shown in Figure 4.1. As can be seen from the figure, each protein-protein interaction network or gene coexpression network is represented as a labeled graph (graphs 1 and 2). In the case of protein interaction networks, the graphs (graphs 1 and 2) are unweighted, whereas in the case of gene coexpression networks, the graphs are weighted (where the weights on the edges denote the pairwise correlation in the expression of the corresponding genes). The BLAST similarity scores are taken into account when comparing the neighborhoods around each of the vertices in the graphs to reconstruct the KEGG orthologs. Please note that the sequence homologous nodes across the two graphs in Figure 4.1 have the same color. A k-hop neighborhood-based approach to alignment uses the notion of k-hop neighborhood. The k-hop neighborhood of a vertex $v_x^1 \in V_1$ of the graph $G_1(V_1, E_1)$ is simply a subgraph of G_1 that connects v_x^1 with the vertices in V_1 that are reachable in k hops from v_x^1 using the edges in E_1. Given two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, a mapping matrix $\mathbf{P}$ that associates each vertex in V_1 with zero or more vertices in V_2 (the matrix $\mathbf{P}$ can be constructed based on BLAST matches) and a user-specified parameter k, we construct for each vertex $v_x^1 \in V_1$ its corresponding k-hop neighborhood C_x in G_1. We then use the mapping matrix $\mathbf{P}$ to obtain the set of matches for vertex v_x^1 among the vertices in V_2; and construct the k-hop neighborhood Z_y for each matching vertex v_y^2 in G_2 and $P_{v_x^1 v_y^2} = 1$. Let $S(v_x^1, G_2)$ be the resulting collection of k-hop neighborhoods in G_2 associated with the vertex v_x^1 in G_1. We compare each k-hop subgraph C_x in G_1 with each member of the corresponding collection $S(v_x^1, G_2)$ to identify the k-hop subgraph of G_2 that is the best match for C_x (based on a chosen similarity measure). Figure 4.1 illustrates this process.

Shortest path graph kernel score

The shortest path graph kernel was first described by Borgwardt and Kriegel (22). As the name implies, the kernel compares the length of the shortest paths between any two nodes in a graph based on a pre-computed shortest-path distance. The shortest path distances for each graph may be computed using the Floyd-Warshall algorithm as implemented in the CDK (Chemistry Development Kit) package (143). We modified the Shortest-Path Graph Kernel to take into account the sequence homology of nodes being compared as computed by BLAST (3). The shortest path graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs, bicomponent clusters extracted from G_1 and G_2 respectively) is given by:

$$S = \sum_{v_i^1, v_j^1 \in Z_{G_1}} \sum_{v_k^2, v_p^2 \in Z_{G_2}} \delta(v_i^1, v_k^2) \times \delta(v_j^1, v_p^2) \times d(v_i^1, v_j^1) \times d(v_k^2, v_p^2)$$

$$K(Z_{G_1}, Z_{G_2}) = \log\left[S\right]$$

where $\delta(v_x^1, v_y^2) = \frac{BlastScore(v_x^1, v_y^2) + BlastScore(v_y^2, v_x^1)}{2}$. $d(v_i^1, v_j^1)$ and $d(v_k^2, v_p^2)$ are the lengths of the shortest paths between v_i^1,v_j^1 and v_k^2,v_p^2 computed by the Floyd-Warshall algorithm. For gene-coexpression network, the Floyd-Warshall algorithm takes into account the weight of the edges (correlations) in the graphs. The runtime of the Floyd-Warshall Algorithm is $O(n^3)$. The shortest path graph kernel has a runtime of $O(n^4)$ (where n is the maximum number of nodes in larger of the two graphs being compared). Please see Figure 4.2 for a general outline of the comparison technique used by the shortest-path graph kernel.

4.2.3 Random walk graph kernel score

The random walk graph kernel (23) has been previously utilized by Borgwardt et al. (23) to compare protein-protein interaction networks. The random walk graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs, bicomponent clusters extracted from G_1 and G_2 respectively) is given by:

$$K(Z_{G_1}, Z_{G_2}) = p \times (\mathbf{I} - \lambda K_x)^{-1} \times q \tag{4.1}$$

where $\mathbf{I}$ is the identity matrix, λ is a user-specified variable controlling the length of the random walks (a value of 0.01 was used for the experiments in this paper), K_x is an $nm \times nm$ matrix (where n is the number of vertices in Z_{G_1} and m is the number of vertices in Z_{G_2} resulting from the Kronecker product $K_x = Z_{G_1} \otimes Z_{G_2}$, specifically,

$$K_{\alpha\beta} = \delta(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}), \alpha \equiv m(i-1)+k, \beta \equiv m(j-1)+l \tag{4.2}$$

Where $\delta(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}) = \frac{BlastScore(Z_{G_{1_{ij}}}, Z_{G_{2_{kl}}}) + BlastScore(Z_{G_{2_{kl}}}, Z_{G_{1_{ij}}})}{2}$; p and q are $1 \times nm$ and $nm \times 1$ vectors used to obtain the sum of all the entries of the inverse expression $((\mathbf{I} - \lambda K_x)^{-1})$. We adapted the random walk graph kernel to align protein-protein interaction networks by taking advantage of the reciprocal BLAST hits (RBH) among the proteins in the networks from different species (74). Naive implementation of our modified random-walk graph kernel, like the original random-walk graph kernel (23), has a runtime complexity of $O(r^6)$ (where $r = max(n, m)$). This is due to the fact that the product graph's adjacency matrix is $nm \times nm$, and the matrix inverse operation takes $O(h^3)$ time, where h is the number of rows in the matrix being inverted (thus, the total runtime is $O((rm)^3)$ or $O(r^6)$ where $r = max(n, m)$). However, runtime complexity of the random walk graph kernel (and hence our modified random walk graph kernel) can be improved to $O(r^3)$ by making use of the Sylvester equations as proposed by Borgwardt et al. (23). Figure 4.3 illustrates the computation of the random walk graph kernel. The random walk graph kernel can take into account the weight of the edges of the graphs in the case of gene-coexpression networks. The weights for the edges across the two networks must be similar for the two networks to be considered matches.

4.2.4 BaryCenter score

The BaryCenter score is calculated based on the total shortest path of the node. The shortest path distances for each node in a graph is calculated and the score is assigned to the node based the sum of the lengths of all the shortest paths that pass through the node (163). More central nodes in a connected component will have smaller overall shortest paths, and 'peripheral' nodes on the network will have larger overall shortest paths.

4.2.5 Betweenness score

Betweenness is a centrality measure of a vertex within a graph. Vertices that occur on many shortest paths between other vertices have a higher betweenness score than nodes that do not occur on many paths (162). For a graph $G_1(V_1, E_1)$, the betweenness score for vertex $v_x^1 \in V_1$is defined as:

$$B(v_x^1) = \sum_{v_i^1 \neq v_x^1, v_j^1 \neq v_x^1, v_i^1 \neq v_j^1, v_{x,i,j}^1 \in V_1} \frac{\delta_{v_i^1 v_j^1}(v_x^1)}{\delta_{v_i^1 v_j^1}}$$

Where $\delta_{v_i^1 v_j^1}$ is the number of the shortest paths from v_i^1 to v_j^1 and $\delta_{v_i^1 v_j^1}(v_x^1)$ is the number of shortest paths from v_i^1 to v_j^1 that pass through vertex v_x^1.

4.2.6 Degree distribution score

The degree distribution score is a simple node importance ranker based on the degree of the node. Nodes with a high number of connections will get a high score while nodes with a smaller number of connections will receive a lower score.

4.2.7 HITS score

The HITS score represents the "hubs-and-authorities" importance measures for each node in a graph (93). The score is computed iteratively based on the degree connectivity of the nodes in the graph and the "authoritativeness" of the neighbors around each node. For a graph $G_1(V_1, E_1)$, each node v_x^1 is assigned two scores: $\alpha(v_x^1)$ and $\gamma(v_x^1)$. Vertices that are connected to many vertices are marked as hubs, and thus their $\alpha(v_x^1)$ scores are large. On the other hand, a vertex that points to highly connected vertices is referred to as an authority and is assigned a high $\gamma(v_x^1)$ score. Some nodes can be highly connected (have high $\alpha(v_x^1)$ score) and have neighbors that are highly connected (thus, have a high $\gamma(v_x^1)$); such nodes would have a high HITS score.

4.2.8 Scoring candidate orthologs based on sequence and network similarity

In order to establish orthologs between fly, yeast, human, pig and mouse, the 1 hop and 2 hop shortest path and random walk scores, BLAST score, BaryCenter score, betweenness score, degree distribution score and HITS score were computed for each pair of homologs detected by BLAST (total of 9 features). The BaryCenter, betweenness, degree distribution and HITS scores were combined using Milenkoviæ et al.'s (115) formula for averaging node-based scores in a graph:

$$S(u_x^1, v_y^2) = \frac{|\log(S(u_x^1)+1) - \log(S(v_y^2)+1)|}{\log(\max(S(u_x^1), S(v_y^2)) + 2)}$$

Where $S(u_x^1)$ and $S(v_y^1)$ are the scores for the nodes from $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, where $u_x^1 \in V_1$ and $v_y^2 \in V_2$. The above formula produces a normalized score for each node-based feature (BaryCenter, betweenness, degree distribution, and HITS scores) for each pair of homologs while adjusting for any bias in magnitude differences in the scores for the graphs (e.g, G_1 may have much more nodes than G_2, thus the node-based scores for G_1 may be more likely to be greater than the node-based scores for G_2).

4.2.9 Ortholog detection

We utilized three broad classes of methods for detecting orthologs:

- Reciprocal BLAST hits method (120; 148). The gene/protein sequences for each of the two species (A and B) being compared are BLASTed against each other. This yields for each gene/protein (from species A, the target) a list of candidate orthologs in species B (and vice versa). Suppose the averaged BLAST scores of gene/protein a_i in species A and the genes/proteins $b_1, \cdots, b_m$ in species B are $s_{i1}, \cdots, s_{im}$. The method predicts the gene/protein in species B that has the highest averaged BLAST score as the ortholog to gene/protein a_i in species A.

- The reciprocal BLAST score-based classifier takes as input the averaged BLAST scores for each possible pair of genes/proteins and outputs a prediction as to whether the pair are orthologous to each other. This method can predict multiple orthologs from species B for each gene/protein from species A (and vice versa).

- The network-based classifier takes as input a vector of pairwise scores (see "Scoring candidate orthologs based on sequence and network similarity" section) computed using the gene-coexpression or protein-protein interaction networks (1 hop and 2 hop Random Walk graph kernel and Shortest Path graph kernel scores as well as the degree distribution, BaryCenter (163), betweenness (162) and HITS (Hubs and Authorities) (93) centrality measures). The classifier outputs a prediction for each pair of genes/proteins as to whether the pair are orthologous to each other. This method can predict multiple orthologs from species B for each gene/protein from species A (and vice versa).

The KEGG (83) ortholog database is used to label the instances in the dataset for training and testing the classifiers.

4.2.10 Performance evaluation

We compare the performance of the simple methods for detecting orthologs based on reciprocal BLAST hits with the decision tree (164), Naive-Bayes (117), Support Vector Machine (36), and ensemble classifier (41) trained using the BLAST scores as well as the graph-based scores (see "Ortholog detection" section) with 10-fold cross-validation. We used the average ranks of the methods based on their performance estimated using the area under the receiver operating characteristic curve (AUC) to compare their overall performance. Although Demsar's (40) non-parametric test can be used to compare machine learning algorithms, the use of this test requires the number of data sets to be greater than 10 and the number of methods to be greater than 5 (40). Thus, it cannot be applied directly to our analysis (since we have only 7 datasets and 5 methods). In such a setting, the average ranks of the classifiers provide a reasonable basis for comparing their overall performance (40). We also report the area under the receiver operating characteristic curve AUC as an additional measure of performance for each of the methods.

4.3 Analysis and results

4.3.1 Reconstructing KEGG orthologs using BLAST

We compare predictions based only on the BLAST score as well as predictions based on the network features discussed in materials and methods section. The results in Table 4.4 show the performance of the reciprocal BLAST hits method in reconstructing the orthologs between the fly, yeast, human and mouse datasets from DIP (136). The last column of of Table 4.4 shows the performance of the reciprocal BLAST hits method in reconstructing the orthologs between the mouse and human gene-coexpression networks. As can be seen from the table, the reciprocal BLAST method performs fairly well in reconstructing the KEGG orthologs for each dataset. As noted by Bandyopadhyay et al. (13), this may be due to the fact that most ortholog detection schemes, at least in part, depend on sequence homology analysis. For example, although KEGG orthologs use information other than sequence homology (such as metabolic pathway comparison and manual curation) (83), sequence homology plays an important role in the definition of KEGG orthologs.

Table 4.4 shows the performance of classifiers using only the BLASTp scores to detect KEGG orthologs between fly, yeast, mouse and human. The logistic regression classifier in WEKA (164) has the best performance overall (according to the average rank shown in Table 4.4), however, it does not outperform the reciprocal BLAST hit method shown in Table 4.4. The results from the gene-coexpression network from mouse and human are comparable overall to the results from the protein-protein interaction networks for the same species.

4.3.2 Reconstructing KEGG orthologs using sequence, protein-protein interaction network, and gene-coexpression data

Table 4.4 shows a comparison of the classifiers trained on the 1 hop and 2 hop Random Walk graph kernel and Shortest Path graph kernel scores as well as the degree distribution, BaryCenter (163), betweenness (162) and HITS (Hubs and Authorities) (93) centrality measures described in materials and methods section. We utilized the approach of Hall et al. (66) as implemented in WEKA (164) to rank the features based on their contribution to the classification performance. We found that the random-walk and shortest-path graph kernel scores were the top two ranked features in terms of their predictive ability. As seen from Table 4.4, most of the classification methods show some improvement over the classifiers trained only on the BLASTp scores shown in Table 4.4. Notably, the ensemble classifier on the mouse-human datasets substantially outperforms its BLASTp counterpart on both the protein-protein interaction networks and the gene-coexpression data. Table 4.4 shows a few representative orthologous pairs that are missed by a regression-based classifier trained on BLASTp scores but are detected by the ensemble classifier trained on the network features and Figure 4.4 shows the network neighborhood for one of such pairs (the TNF receptor-associated factor 2). This suggests that the combination of sequence homology with network-derived features may present a more reliable approach than simply relying on reciprocal BLASTp hits in identifying orthologs.

4.4 Discussion and future work

The availability of genomes from a wide range of organisms has enabled the comparison and analysis of evolutionary relationships among genes across organisms through the reconstruction of phylogenies (161), common pathways (83; 112), comparing gene functions (133; 47), and network alignment (81; 149; 56; 153; 171; 97; 89; 101; 127; 6). Ortholog detection methods present a powerful approach for finding genes that participate in similar biological processes across different organisms, extending our understanding of interactions between genes across different pathways, and understanding the evolution of gene families. We have explored a set of graph-based features that may be utilized for the detection of orthologs among different genomes by combining sequence-based evidence (such as BLAST-based sequence homology) with the network alignment algorithms available as part of the BiNA (Biomolecular Network Alignment) toolkit (153) as well as graph features extracted from the aligned protein-protein interaction networks such as degree distribution, BaryCenter (163), betweenness (162) and HITS (Hubs and Authorities) (93) centrality measures. To the best of our knowledge, this is the first time such an analysis has been carried out based on the comparison of weighted gene-coexpression networks. The features may be used to score orthologous nodes in large biomolecular networks by comparing the neighborhoods around each node and scoring the nodes based on the similarity of their neighborhoods in the corresponding protein-protein interaction and gene-coexpression networks. Classifiers can then be trained using the scores to generate predictions as to whether or not a given pair of nodes are orthologous. Our results suggest that the algorithms that rely on orthology detection methods (e.g., for genome comparison) can potentially benefit from this approach to detecting orthologs (e.g., in the case of the comparison between mouse and human). The proposed method can also help identify proteins that have strong sequence homology but differ with respect to their interacting partners in different species (i.e., proteins whose functions may have diverged after gene-duplication).

Our experiments with the fly, yeast, mouse and human protein-protein interaction datasets as well as the gene-coexpression data suggest that the accuracy of identification of orthologs using the proposed method is quite competitive with that of reciprocal BLAST method for detecting orthologs. The improvements obtained using information about interacting partners in the case of the mouse-human data (96.18% for the protein-protein interaction network-based method and 96.10 for the gene-coexpression methods as opposed to 90.31% AUC for the reciprocal BLASTp method) suggest that the proposed technique could be useful in settings that benefit from accurate identification of orthologs (e.g., genome comparison). Using the methods described in this paper, we have predicted the mouse and human orthologs for the pig genes, for which currently there is no KEGG ortholog data (please see Additional file 1 and Additional file 2 for our predictions).

The network neighborhood-based homology detection algorithm is implemented in BiNA (http://www.cs.iastate.edu/~ftowfic), an open source Biomolecular Network Alignment toolkit. The current implementation includes variants of the shortest path and random walk graph kernels for computing orthologs between pairs of subnetworks and the computation of various graph-based features available in the Java Universal Graph Framework library (123) such as the degree distribution, BaryCenter (163), betweenness (162) and HITS (Hubs and Authorities) (93) centrality measures. The modular design of BiNA allows the incorporation of alternative strategies for decomposing networks into subnetworks and alternative similarity measures (e.g., kernel functions) for computing the similarity between nodes. It would be interesting to explore variants of methods similar to those proposed in this paper for improving the accuracy of detection of orthologous genes or proteins using other sources of data (e.g., gene regulatory networks or metabolic networks).

Competing interests

The authors declare that they have no competing interests.

Authors contributions

SVP, CO and OC assembled and verified the datasets for the analysis. FT wrote the algorithms, ran the experiments and wrote the initial draft of the manuscript. CT, MHWG and VH supervised the analysis, design of the algorithms and revisions to the manuscript.

Acknowledgements

This research was supported in part by an Integrative Graduate Education and Research Training (IGERT) fellowship to Fadi Towfic, funded by the National Science Foundation (NSF) grant (DGE 0504304) to Iowa State University and a NSF Research Grant (IIS 0711356) to Vasant Honavar.

Datasets	AUC
Mouse-Human (PPI)	90.39
Mouse-Fly (PPI)	92.62
Mouse-Yeast (PPI)	96.14
Human-Fly (PPI)	88.89
Human-Yeast (PPI)	85.63
Yeast-Fly (PPI)	75.03
Mouse-Human (gene-coexpression)	90.40

Table 4.1 Performance of the Reciprocal BLAST hit method on the fly, yeast, human and mouse protein-protein interaction datasets from DIP as well as the gene coexpression networks for mouse and human from GEO

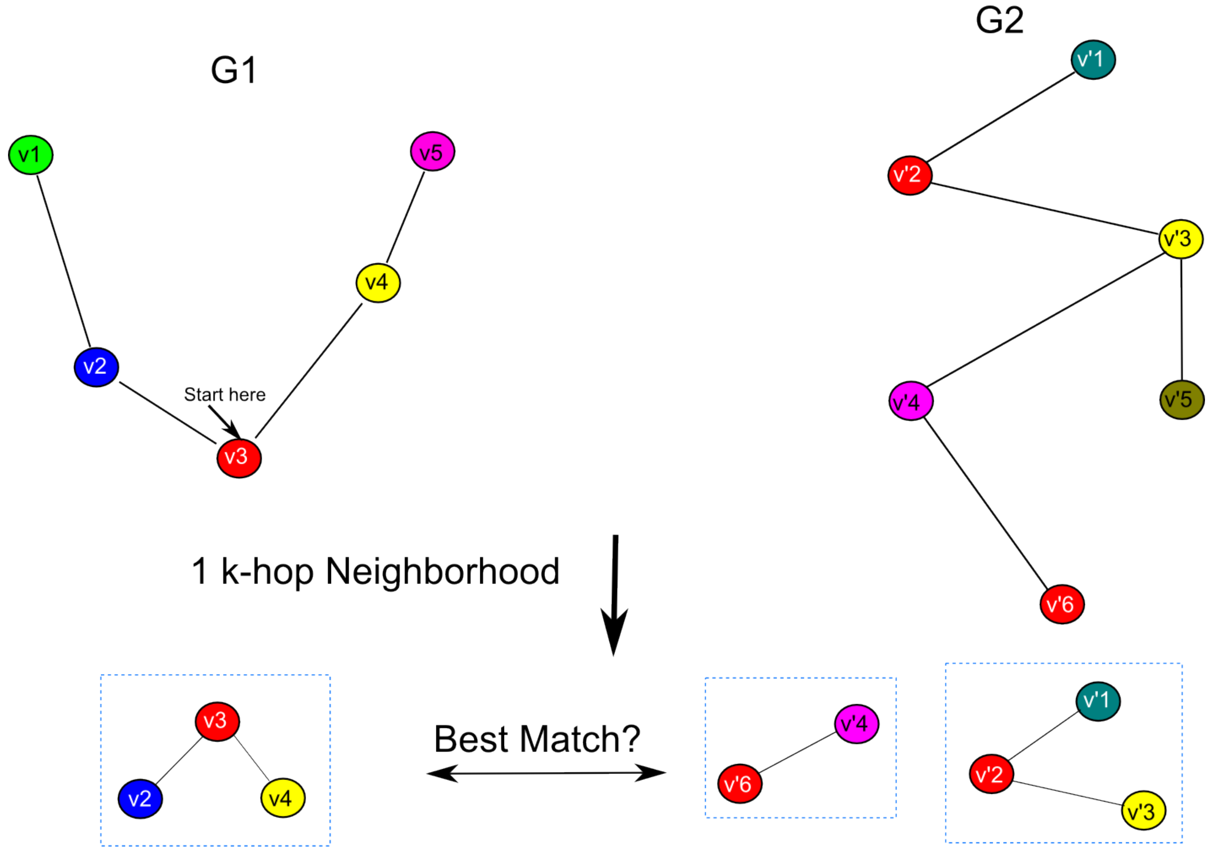


Figure 4.1 A schematic of the graph representation of the BLAST orthologs based on protein-protein interaction networks and gene coexpression networks. The networks are represented as two labeled graphs (G1 and G2) with corresponding relationships among their nodes (similarly colored nodes are sequence homologous according to a BLAST search). Nodes from G1 (e.g., v3) are compared to their sequence-homologous counterparts in G2 (e.g., v'2 and v'6) based on the topology of their neighborhood and sequence homology of the neighbors. In the figure, v'2 has the same number of neighbors of v3 and one of the neighbors of v'2 (i.e., v'3) is sequence-homologous to v4. Thus, v'2 is scored higher (more likely to be an ortholog to v3) compared to v'6. Protein-protein interaction networks are represented as unweighted graphs, while gene coexpression networks incorporate weights (as calculated by correlations) into their edges

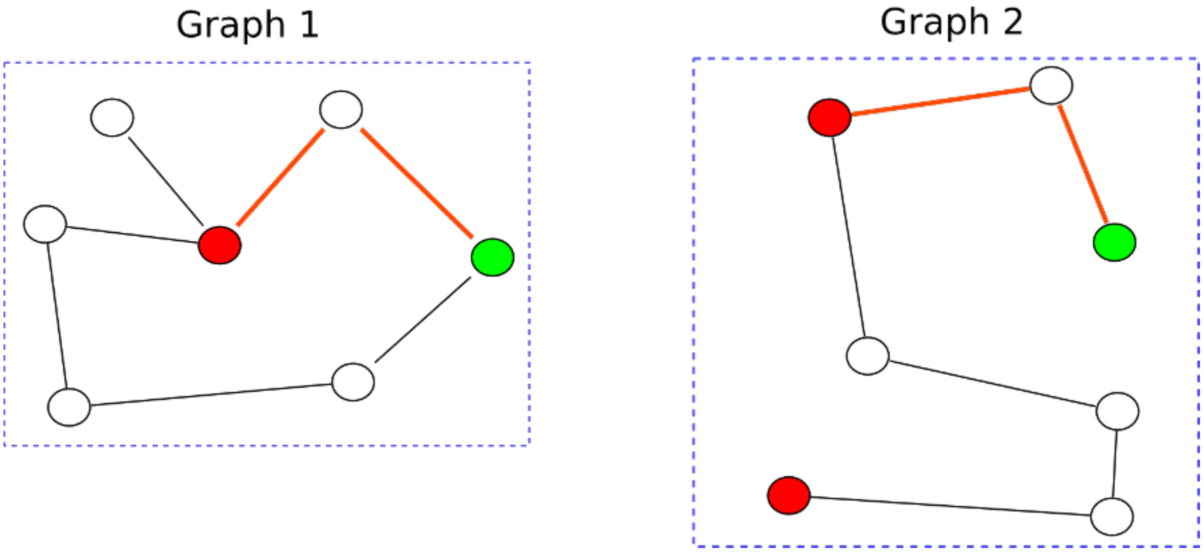


Figure 4.2 An example of the graph matching conducted by the shortest path graph kernel. Similarly colored nodes are sequence homologous according to a BLAST search. As can be seen from the figure, the graph kernel compares the lengths of the shortest paths around homologous vertices across the two graphs (taking into account the weights of the edges, if available). The red edges show the matching shortest path in both graphs as computed by the graph kernel. The shortest path distance graph kernel takes into account the sequence homology score for the matching vertices across the two graphs as well as the distances between the two matched vertices within the graphs

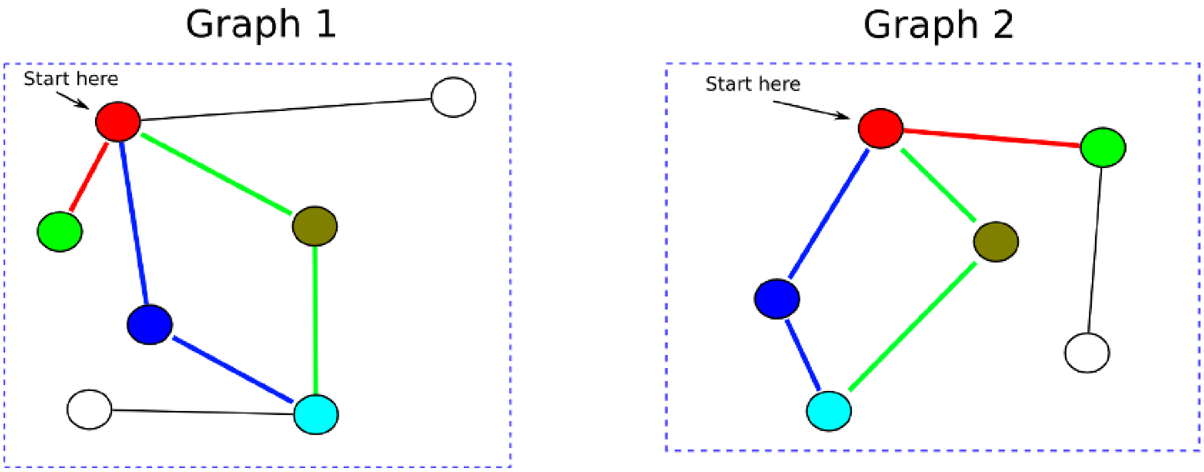


Figure 4.3 An example of the graph matching conducted by the random walk graph kernel. Similarly colored vertices are sequence homologous according to a BLAST search. As can be seen from the figure, the graph kernel compares the neighborhood around the starting vertices in each graph using random walks (taking into account the weights of the edges, if available). Colored edges indicate matching random walks across the two graphs of up to length 2. The random walk graph kernel takes into account the sequence homology of the vertices visited in the random walks across the two graphs as well as the general topology of the neighborhood around the starting vertex

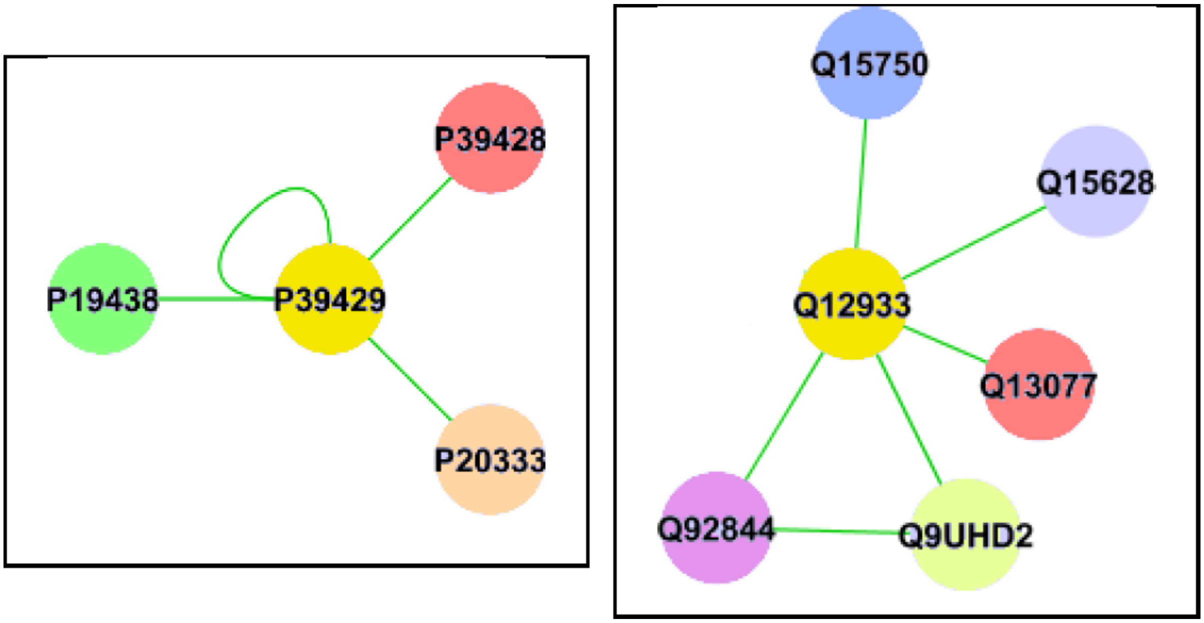


Figure 4.4 A sample 1 hop neighborhood around one of the matched orthologs (TNF receptor-associated factor 2 “P39429” in mouse and “Q12933” in human) according to the graph features (**left:** 1 hop network around the “P39429” protein for mouse, **right:** 1 hop neighborhood around the “Q12933” protein for human). Similarly colored nodes are sequence homologous. The graph properties search for similar topology and sequence homology around the neighborhood of the nodes being compared

Datasets	Adaboost j48 AUC	NB AUC	SVM AUC	Log. Reg. AUC	Ensemble AUC
Mouse-Human (PPI)	87.79 (4)	90.15 (3)	77.31 (5)	90.29 (2)	90.30 (1)
Mouse-Human (gene-coexpression)	89.80 (4)	70.4 (5)	90.40 (1)	90.40 (1)	90.40 (1)
Mouse-Fly (PPI)	87.58 (4)	88.47 (3)	70.17 (5)	92.01 (1)	88.89 (2)
Mouse-Yeast (PPI)	89.85 (5)	91.89 (2)	90.78 (3)	95.46 (1)	91.45 (4)
Human-Fly (PPI)	81.35 (4)	87.70 (2)	65.90 (5)	88.90 (1)	84.42 (3)
Human-Yeast (PPI)	82.97 (3)	81.26 (4)	63.68 (5)	85.50 (1)	84.19 (2)
Yeast-Fly (PPI)	73.02 (3)	72.49 (4)	56.80 (5)	74.86 (1)	74.48 (2)
Average Rank (PPI Only)	3.83	3	4.67	1.17	2.33
Average Rank (PPI + GeneCoexpression)	3.86	3.28	4.28	1.28	2.28

Table 4.2 Performance of the Reciprocal BLAST hit score as a feature to the decision tree (j48), Naive Bayes (NB), Support Vector Machine (SVM) and Ensemble classifiers on the fly, yeast, human and mouse protein-protein interaction datasets from DIP as well as the gene coexpression networks for mouse and human from GEO. Values in parenthesis are the ranks for the classifiers on the specified dataset

Datasets	Adaboost j48 AUC	NB AUC	SVM AUC	Log. Reg. AUC	Ensemble AUC
Mouse-Human (PPI)	95.19 (2)	88.72 (5)	90.78 (3)	89.57 (4)	96.18 (1)
Mouse-Human (gene-coexpression)	89.80 (5)	94.1 (4)	97.50 (1)	97.30 (2)	96.10 (3)
Mouse-Fly (PPI)	90.31 (1)	85.81 (3)	81.28 (4)	80.67 (5)	88.94 (2)
Mouse-Yeast (PPI)	92.04 (3)	85.50 (4)	79.63 (5)	95.60 (1)	95.50 (2)
Human-Fly (PPI)	88.18 (1)	83.10 (4)	75.03 (5)	87.04 (3)	87.20 (2)
Human-Yeast (PPI)	82.83 (2)	81.26 (4)	78.22 (5)	81.57 (3)	84.84 (1)
Yeast-Fly (PPI)	74.52 (1)	69.36 (4)	64.57 (5)	74.33 (2)	72.78 (3)
Average Rank (PPI Only)	1.67	4	4.5	3	1.83
Average Rank (PPI + GeneCoexpression)	2.14	4	4	2.86	2

Table 4.3 Performance of all the combined features (Reciprocal BLAST hit score, 1 and 2 hop shortest path graph kernel score, 1 and 2 hop random walk graph kernel score, BaryCenter, betweenness, degree distribution and HITS) as input to the decision tree (j48), Naive Bayes (NB), Support Vector Machine (SVM) and Ensemble classifiers on the fly, yeast, human and mouse protein-protein interaction datasets from DIP as well as the gene coexpression networks for mouse and human from GEO. Values in parenthesis are the ranks for the classifiers on the specified dataset

Mouse Protein	Human Protein	BLASTp score	RW 1HOP	SP 1HOP	RW 2HOP	SP 2HOP	Bary Center	Betweenness	Degree	HITS
P05627	P05412	481	104	197.35	612	290.27	0.71	0.69	0.01	0.26
P36898	P36894	725	28.13	222.85	90.66	576.51	0.35	0.77	0.01	3.06E-10
P39429	Q12933	870	48	126.18	150.47	187.45	0.79	0.11	0.01	1.20E-4

Table 4.4 KEGG orthologs detected using the Ensemble classifier utilizing all network features. The orthologs shown in the above table were missed by the BLAST logistic regression classifier

CHAPTER 5. B CELL LIGAND GENE COEXPRESSION NETWORKS REVEAL REGULATORY PATHWAYS FOR LIGAND PROCESSING

Paper submitted to BMC Systems Biology

Fadi Towfic, Shakti Gupta, Vasant Honavar and Shankar Subramaniam

Abstract

Background

The initiation of B cell ligand recognition is a critical step for the generation of an immune response against foreign bodies. A wide variety of responses may be induced in B cells through the activation of different receptors. Unfortunately, the regulatory mechanisms that are involved in B cell response to antigenic stimulants are not very well understood. We sought to identify the biochemical pathways involved in the B cell ligand recognition cascade and sets of ligands that trigger similar immunological responses.

Results

We utilized several comparative approaches to analyze the gene coexpression networks generated from a set of microarray experiments spanning 33 different ligands. First, we compared the degree distributions of the generated networks. Second, we utilized a pairwise network alignment algorithm (BiNA) to align the networks based on the hubs in the networks. Third, we aligned the networks based on a set of KEGG pathways. We summarized our results by constructing a consensus hierarchy of pathways that are involved in B cell ligand recognition. The resulting pathways that are shared across B cell responses to different ligands were further validated through literature for their common physiological responses (e.g., both PGE and

NPY trigger pathways that contribute to inflammation).

Conclusions

Collectively, the results based on our comparative analyses of degree distributions, alignment of hubs, and alignment based on KEGG pathways showed a high degree of concordance and (i) provide a basis for molecular characterization of the immune response states of B cells and (ii) demonstrate the power of comparative approaches (e.g., gene coexpression network alignment algorithms) in elucidating biochemical pathways involved in complex signaling events in cells.

Background

B cell ligand recognition plays a large role in various immune responses: from the recognition of foreign invaders such as viruses and bacteria to the recognition of cancerous cells. B cells act as the body's most effective line of defense to invaders (32). Several types of responses may be induced in naïve mature B cells through the activation of different receptors (e.g., cytokine and chemokine receptors) (39; 76). Recognition of ligands by the B cell Ag-receptor (BCR) begins with the activation of an array of intracellular effector molecules and end with phenotypic and genotypic modifications that define the cell's response to the stimulus. As more and more players in this process are uncovered, the current schematic of BCR signal transduction has become a "labyrinth" of interconnecting pathways (37). Despite the complicated events that occur during this event, the resultant reaction is very ordered and precise. The activation of various signal-transduction pathways in mature B cells is influenced by the combination of ligands presented to the B cells. The presence of different ligands may trigger cell-proliferation, activation, differentiation, migration, isotype switching and apoptosis (32; 135; 71). Of particular interest in this area is the elucidation of the regulatory mechanisms that are involved in B cell recognition of various ligands. These data provide a detailed look at the finite states B cells can enter upon exposure to ligands. Understanding the genetic interaction that are required for this process allows the design of drugs that are capable of triggering a specific immune response at a given time-point, understanding the mechanisms that underly different auto-immune diseases, and understanding the regulation mechanism for B cells.

Against this background, several studies (104; 173; 118) have examined the changes in expression patterns of B cells in response to exposure to different ligands. These studies used differential gene expression analysis of microarray data (e.g., using Significance Analysis of Microarrays (SAM) (155)) and Gene Ontology (GO) (4) terms to detect genes that were significantly differentially expressed and whose pathway annotations shared significant GO terms. This approach, although well developed and widely used, suffers from an important limitation: it focuses on differences in expression patterns of individual genes across the different treatments or time-points.

In contrast, recently developed techniques for network alignment such as those developed by Koyutürk et al. (97) and Kalaev et al. (81), among others (149; 57; 82; 89; 137; 139; 107) attempt to detect interactions between genes, proteins, or metabolites that are conserved across gene expression, protein-protein interaction and/or metabolic networks. However, most existing network alignment or conserved module finding algorithms work with networks with unweighted links (e.g., protein-protein interaction networks in which the nodes represent proteins and the links between pairs of nodes represent binary interactions between the corresponding proteins). Hence, such methods are not directly applicable for comparing the gene expression pattern in a cell when it is treated with different ligands. Other approaches, for example those utilized by Glaab et al. (64) among others (167; 6; 10) attempt to integrate mRNA expression patterns with protein-protein interaction networks or metabolic networks to construct a weighted network in which the weights on the links represent a measure of confidence in the observed interactions between nodes. However, such methods do not offer a means of directly comparing two or more networks to identify pathways that are similarly regulated or differentially expressed.

Gene-coexpression networks in which the nodes represent genes and the weighted links between pairs of nodes encode the correlations in expression patterns of the corresponding genes offer a useful way to represent cellular responses to each of the different treatments (e.g., exposure to different ligands). Alignment of such networks provides a direct means of comparing cellular responses to different treatments. Hence, we utilized a pairwise network alignment algorithm (BiNA (153)) to align 33 gene coexpression networks generated from a set of microarray experiments spanning 33 different ligands (please see Table 1 for a complete list of the ligands) (173). A network alignment (analogous to a sequence alignment) compares two input networks and returns a set of common pathways across the networks with a score denoting the similarity between the networks being compared. By constructing a symmetric 33×33 distance matrix using the alignment scores across the 33 networks, a hierarchical cluster was constructed based on the distance matrix to visualize relationship across the networks representing the gene expression changes due to exposure to different ligands. The common pathways detected across the most similar networks were examined and the pathways were annotated according to KEGG (84). Using this approach, we examined the regulation mechanisms specific to certain groups of ligands. Based on our network alignment method, we identified a set of specific genes and pathways that appear to be involved in BCR-mediated ligand capture, vesicle function and vesicle trafficking during B cell antigen processing and presentation for the set of 33 ligands we examined. Furthermore, we present a new analysis pipeline based on network alignment that may be utilized on newer datasets in the future to study similar processes.

Results and Discussion

Cells respond to stimuli through myriad pathways. However, they deploy similar modules in their response to distinct ligands. The major objective of this study was to explore the space of signaling responses of B-cells to naturally occurring stimuli and identify the commonality and differences in the ligand response. Such analysis will provide an insight into the space of responses of B-cells in native physiology and provide pathway motifs that can be explored through further experimentation.

We utilized several different approaches for comparing gene co-expression networks constructed from microarray data obtained from B cells treated with different ligands: Comparison of degree distributions of networks using Kolmogorov-Smirnoff statistic (see “Clustering based on degree distribution” section), alignment of the networks based on the top 2000 highly connected nodes (see “Clustering based on alignment of high degree nodes in ligand networks” section), and alignment of the networks based on KEGG pathways that were enriched with high intensity probes (see “Clustering based on ligand similarity across signaling pathways” section). The results of our analyses show a high degree of concordance in terms of the pathways and reactions involved in B cell ligand recognition that are identified by several comparative methods (see “Discussion and Conclusions” section). This enabled us to (i) construct a consensus hierarchy of the pathways that are highly regulated (activated or inhibited) in B cells after their exposure to ligands; and (ii) and group the ligands on the basis of similarity between gene expression patterns across specific biochemical pathways of interest (see Tables 2 and 3, as well as Figure 5 and supplementary material). The resulting pathways that show similar responses to different ligands in B cells were further validated through literature for their common physiological responses (e.g., both PGE and NPY trigger pathways that contribute to inflammation). We now proceed to describe our methods and results in greater detail.

Clustering based on degree distribution

In order to determine the relationships of the ligand networks based on the network topology, we computed the degree distribution (shown in Figure 1 in the supplementary material, the *degree* of a node is the number of edges/links for that node) for each ligand network (a total of 33 networks, see Table 1 for a complete list of the ligands used in this study). The degree distribution plots show the relationship between the degree of a node and the frequency of nodes with that degree ($P(Degree)$).

We compared the resulting 33 distributions using the two-sample Kolmogorov-Smirnov statistic (113). Specifically, we used the Kolmogorov-Smirnov statistic to compute the 33 × 33 pairwise distances from the 33 degree distributions. Thus, we constructed a 33 × 33 matrix $\mathbf{D}^{toplogical}$ where the entry in the ith row and jth column in the matrix corresponds to the distance between the degree distributions of the ith and jth networks as determined by the Kolmogorov-Smirnov statistic. The $\mathbf{D}^{toplogical}$ matrix was then fed into a hierarchical neighbor-joining algorithm to construct the hierarchical cluster shown in Figure 1. Figure 1 shows the relationships between the ligand networks obtained by the topological comparison of the networks based on their degree distributions. Ligand networks with a large number of differentially expressed genes relative to untreated samples (as indicated in (104)) have been highlighted in the figure.

As can be seen from Figure 1, ligand networks with a high number of differentially expressed genes relative to untreated samples share the same subtree/clade in the hierarchical network. This result indicates that the network structure as measured by the degree distribution and compared by the Kolmogorov-Smirnov statistic can be used to detect ligands that elicit similar responses upon exposure to B cells.

Although topological comparison of gene co-expression networks based on their degree distributions is simple, intuitive, and computationally inexpensive, it fails to take into account the node labels or the biological annotation for the nodes in the networks. In order to compare the networks based on both the network topology and the node labels/biological annotation (e.g, signaling pathways, metabolic pathways...etc) for the nodes, we utilized a network alignment algorithm implemented in the Biomolecular Network Alignment (BiNA) toolkit (153; 154).

Clustering based on alignment of high degree nodes in ligand networks

The network alignment algorithm implemented in BiNA allows the comparison of gene co-expression networks based not only on the extent to which they share similar topologies, but also the weights on the links (e.g., similarities in gene coexpression patterns) and the similarities of node and/or edge labels (biological annotations). We used the BiNA toolkit to run all-vs-all comparisons between all 33 ligand networks and construct a 33×33 distance matrix $\mathbf{D}^{hubs}$ whose entries signify the similarity score between ligands. Initially, we reduced the comparison to an alignment of the neighborhood around the top 2000 highly connected nodes (hubs) between all 33 ligand networks. We initially started aligning all nodes in the network, but quickly noticed that the total alignment score between two networks saturated after 2000 hubs. Specifically, to construct $\mathbf{D}^{hubs}$, consider the output of a pairwise alignment between two networks (e.g., between ligand network 1, $L^1(V^1, E^1)$ and ligand network 2, $L^2(V^2, E^2)$) is a set of matched nodes S^1 (for ligand network 1, where $S^1 \subset V^1$) and S^2 (for ligand network 2, where $S^2 \subset V^2$) with a corresponding score set M. The corresponding entries S_i^1 and S_i^2 and M_i signify matching K-hop neighborhoods around the nodes S_i^1 and S_i^2 with a similarity score M_i (where $1 \leq i \leq 2000$ since we are considering 2000 hubs). The overall pairwise similarity score between the two ligand networks is calculated by summing the scores across all matched neighborhoods $\sum_{m \in M} m$ (see alignment subsection in Methods for more information on how neighborhood scores are calculated). The overall similarity scores between all 33 ligand networks were assembled into a similarity matrix $\mathbf{D}^{hubs}$ with each entry in the matrix signifying the similarity score between the ligand networks (e.g., entry $d_{1,2}^{hubs}$ in $\mathbf{D}^{hubs}$ contains the similarity score between ligand network 1 and ligand network 2 as determined by BiNA). The $\mathbf{D}^{hubs}$ matrix was then fed into a hierarchical neighbor-joining algorithm to construct the hierarchical cluster representing the similarity between the ligand networks.

Finally, in order to calculate confidence measures on the branches of the hierarchical cluster produced by the alignment, the tree produced by hierarchical clustering was bootstrapped (46; 52) by sampling randomly (with replacement) from the top 2000 hubs 100 times. This random resampling on the M set, followed by summing the scores of the resampled set for each cell in $\mathbf{D}^{hubs}$ results 100 distance matrices $\mathbf{D}_{1...100}^{bootstrappedhubs}$ which are fed into the same hierarchical neighbor-joining algorithm to construct 100 hierarchical similarity trees. The consensus tree of the hierarchical clusters based on the bootstrapped trees is produced using the Phylip (53) "consense" tool. Figure 2 shows the bootstrapped tree resulting from this method.

Figure 2 shows that ligands with similar induced reaction (e.g., LPS and SDF, both affect pathways involved in cell migration) cluster together. Such an analysis yields not only general similarity relationships between the ligand networks, but also provides specific gene and pathway information as can be seen from clustering based on signaling pathways (see below).

The cluster shown in Figure 2 describes the similarity of expression based on node labels as well as correlation between the genes in the ligand networks. However, the hierarchical cluster from Figure 2 does not provide specific information as to which sets of pathways are shared/similarly regulated across ligand networks that fall under the same clade/subtree in the hierarchical cluster. KEGG (84) annotation of pathways was used to link the node labels in the networks to biological pathways (such as metabolism or signal processing). The additional pathway annotation can be used to determine the specific biological pathways that are involved in B cell ligand recognition, and how those pathways are regulated based on exposure to each ligand. This procedure is described in detail in the next section.

Clustering based on ligand similarity across signaling pathways

We wanted to choose pathways based on the highly regulated genes in the microarray dataset rather than relying on *a priori* knowledge from the literature. The reasons for this choice are two-fold: (i) a choice of pathways that is unbiased by what is currently known in the literature can help identify novel pathways involved in B cell ligand recognition (ii) if the list of pathways determined to be highly regulated based on the microarray data happens to share a high degree of overlap with the list generated based on literature surveys, it helps establish the utility of the approach in settings where the prior knowledge available in the literature is quite sparse.

We choose pathways according to the following procedure:

1. In the fully normalized dataset (all 422 microarray samples), search for genes that meet the following criteria (referred to as "high intensity" genes in what follows). Briefly, we wanted to maximize the sensitivity of detection of genes that are differentially regulated upon exposure of B cells to ligands compared to untreated B cells. This procedure maximizes sensitivity at the cost of specificity. The list of genes generated by this approach will be further reduced by comparing the neighborhoods in the ligand networks using network alignments.

    (a) Calculate the fold difference between the average probe expression level and the expression level for all probes in each sample (see Methods section)

    (b) Select probes whose fold-difference is higher than 1 in at least one of the 422 samples.

    (c) Of the probes selected in step (b), find probes that are expressed at least 1 fold higher compared to the same probes from the untreated samples

2. Once the high intensity probes are selected from step 1-c, map back the probe id's to their respective gene id's

3. Among all the pathways in KEGG, and count the number of genes from step (2) that show up in each KEGG pathway

The results of the preceding steps are summarized in Table 2. As can be seen from Table 2, many of the pathways enriched in high-intensity genes are known to be implicated in the development of the immune system and processing of ligands. It should be noted that although KEGG considers the immune system pathways (KEGG category 5.1) to be a part of organismal system (KEGG category 5), we considered the immune systems pathways separately (see Table 2) since we wanted to specifically examine the immune system pathways.

Figures 3 and 4 present examples of the alignment based on the KEGG metabolism and Genetic Information Processing pathways. The numbers on the branches signify the number of similarly regulated subpathways between any two ligands. As can be seen from the figure, some ligand networks (e.g., TER/BAF and FML/GRH) fall under the same clade/subtree in the two pathways, signifying general similarity in the regulation/signaling of pathways by such ligands. Differences between the trees show that the ligands may have different effects depending on the pathway being observed.

Figure 5 shows a consensus tree based on all 7 general pathway categories highlighted in Table 2. As can be seen from the figure, GRH and FML, for example, fall under the same clade/subtree in the consensus tree in Figure 2 and the consensus tree constructed based on differentially expressed pathways (see Table 2) shown in Figure 5. Overall, this shows that the results of the alignment is consistent across the different pathways chosen to ascertain the similarity hierarchy between the overall networks. The numbers on the branches can also serve as confidence measures for grouping certain leaves/networks with each other. We also utilized specific signaling pathway highlighted in the literature (173; 104) (see table 1 in supplementary material) to align the networks and constructed a cladogram describing the relationship between the ligands. The result is shown in figure 2 in the supplementary material.

Conclusions

Recognition of ligands by the B cell Ag-receptor (BCR) begins with the activation of an array of intracellular effector molecules and end with phenotypic and genotypic modifications that define the cell's response to the stimulus (37). The pathways involved in this process are highly interrelated and, thus, methods for identifying the processes involved must take into account the underlying relationships between the genes that are involved. The activation of various signal-transduction pathways in mature B cells is influenced by the combination of ligands presented to the B cells (32; 135; 71). The goal of this study was to identify the putative biochemical pathways involved in the B cell ligand recognition cascade and to identify sets of ligands that trigger similar B cell (immunological) responses.

Identifying sets of ligands that trigger similar B cell responses provides a basis for elucidating the specific genetic interactions that play a role in the recognition of ligands by B cells. Which, in turn, provides valuable information for designing drugs that are capable of triggering a specific immune response. Furthermore, the knowledge of biochemical pathways that are involved in immune response could lead to better understanding of mechanisms behind different auto-immune diseases, and recognition of the regulation mechanism for B cells. To achieve this goal, we constructed 33 gene coexpression networks that represented the genetic interactions in B cells after exposure to each of the 33 ligands. Each network represents the response of normal splenic B cells to a specific ligand across 4 different time points with 3 replicates per time point. We then utilized several comparative approaches to identify shared subnetworks/pathways among the 33 networks. Based on those pathways (see Table 2), we were able to identify ligands that trigger similar expression changes in each of the pathways (see Table 3, Figures 5 and 6, and supplementary material).

The results from the alignments showed that some ligands tend to have similar expression patterns based on the KEGG pathways used to anchor the pairwise all-vs-all alignments for the 33 ligand networks. Table 3 presents a detailed list of ligands that induce similar expression cascades in the KEGG pathways highlighted in Table 2. Several of the matched ligands (see Figure 5) are actually known to induce similar reactions in B cells based on a literature search we conducted. For example, LPS (Lipopolysaccharide) and SDF (Stromal cell derived factor-1) are known to affect cellular migration, IFG (Interferon-gamma) and LPA (Lysophosphatidic acid) are known to trigger changes in isotype switching (173; 104). PGE (Prostaglandin E2) and NPY (Neuropeptide Y) trigger pathways that contribute to inflammation, M3A (Macrophage inflammatory protein-3)/DIM (Dimaprit)/TGF (Transforming growth factor-beta 1) have several effects: M3A is strongly chemotactic for lymphocytes, TGF provides a chemotactic gradient for leukocytes and down-regulates the activity of immune cells (105). DIM, analog to histamine, activates immune response. Additionally, GRH (Growth hormone-releasing hormone) and FML (formyl-Met-Leu-Phe) are known to affect growth and chemotaxis of cells, respectively. CPG (CpG-Containing Oligonucleotide) and PAF (Platelet activating factor) are known to affect cellular proliferation and stimulation of antibody production (173). NEB (Neurokinin B) and NGF (Nerve Growth Factor) have both been observed to have been shown to be involved in the growth and development of neurons (55; 150). Furthermore, TNF (Tumor necrosis factor-alpha) has been shown to be highly involved in mediating inflammatory and immune responses (128), similar to what has been recently observed using CGS (CGS-21680 hydrochloride) (159). Table 3 shows an abbreviated list of all the pathways that we have identified based on the network alignment between the 33 ligand networks that contribute to each of the above matches (the full list is provided in table 3 of supplementary material).

From the results shown in Table 3 (and expanded table 3 in supplementary material), it can be seen that several major pathways are regulated in B cells in response to the exposure to the 33 ligands shown in Table 1. First, human disease pathways (e.g., cancer, asthma, see Tables 3 and 4 for specific list of KEGG pathways classified as “Human Disease pathways”) are the most prevalent pathways triggered by over half the ligands: 70L, AIG, SLC, LPA, IFG, GRH, FML, IFB, S1P, BOM, LB4, NEB, NGF, TNF, CGS, DIM, TGF. Those ligands constitute a set of molecules that trigger a wide variety of responses in B cells and can be used to further ascertain the conditions under which B cells activate under certain situations in human diseases. Second, cellular process pathways (e.g., endocytosis, apoptosis, see Table 2 for specific list of pathways classified as “Cellular Processes”) seem to be also over-represented among the pathways that significantly change in expression across upon exposure to ligands. Some of the ligands (70L, AIG, SLC, LPA, IFG, GRH, FML, IFB, S1P, TNF, CGS) seem to trigger both human disease and cellular process pathways, while other ligands (PGE, NPY, TER, BAF) only trigger cellular pathways. Such ligands constitute a set of molecules that trigger changes in B cells that may affect their growth and proliferation. The relationship between each of the above ligands as to exactly which ligands trigger similar expression patterns in the selected KEGG subpathways is described in Table 3 and Figure 5 based on our approach. The third major pathway commonly regulated in B cells upon ligand exposure is metabolism with a sizable number of ligands (GRH, FML, PGE, NPY, TNF, CGS, PAF, CPG, TER, BAF, DIM, TGF) triggering pathways in that category. Ligands that only triggered pathways in B cells related to metabolism but not "human diseases" or "cellular processes" are PGE, NPY, PAF, CPG. Since those ligands are known to affect inflammation and antibody production, the metabolic pathways expressed as a result of B cell exposure to those ligands may be important indicators of B cell immune response.

Aligning the 33 ligand networks allowed the detection of the specific relationships between the ligands in terms of the pathways that they regulate in B cells. Additionally, the alignment pointed out specific pathways that share expression patterns across ligands and are involved in BCR activation. We have been able to validate some of the relationships we uncovered based on the immune responses described in the literature in the case of some of the ligands in our dataset. The computation tools and methods we utilized for constructing the alignments and analyzing the results are available online as part of the BiNA (Biomolecular Network Alignment) toolkit http://www.cs.iastate.edu/~ftowfic. An analysis pipeline based on network alignment such as the one used in this study may also serve as a general template for identifying pathways with conserved expression patterns across different conditions in other types of experiments. We have made our data and results available through the supplementary material to this paper. Some promising directions for further work include integration of additional types of information (e.g., protein-protein interaction networks) in our analyses and overlaying our pathways with already known protein-protein interactions to detect specific proteins that are responsible for triggering the signaling cascades for each ligand. Such information can aid in narrowing down the list of pathways to their core protein interactions.

Methods

Microarray Data

The microarray data (104; 173) were collected from the Alliance for Cell Signaling (AfCS) site (2). Briefly, the experiments were designed to examine gene expression changes induced by the 33 single ligands, mouse splenic B cells were cultured with ligands in serum-free medium for 0.5, 1, 2, and 4 h. cDNA synthesized from the RNA of B cells was labeled with Cy5 and hybridized onto custom-made two-color Agilent cDNA arrays (Containing 16273 probes) with a Cy3-labeled cDNA prepared from the RNA of total splenocytes. There were a total of 424 Agilent chips hybridized in this study (104; 173).

The data was processed using MatLab® Bioinformatics toolbox. The background corrected intensity values were used for each chip. Some of the background corrected intensities were negative and created a problem to take the logarithm of the data. To circumvent this problem, a very low positive value (10, a value that was 500 times below the mean intensity of all chips) was assigned to these probes. Each chip was also normalized to its mean intensity. Chip-to-chip normalization was performed via LOWESS normalization method to allow for adequate analysis between chips (130). After the normalization, the replicate chips were averaged. To remove the outliers each replicated probe was subjected to an outlier test. The outlier test was as follows:

1. Calculate the mean and standard deviation (SD) for all replicates of each probe.
2. Select the probes in the range of mean ± 1.2 SD for the calculation of a new mean and SD
3. Discard the probes out of the range of the new mean ± 2 new SD.
4. Calculate the fold change as ligand treated divided by control (untreated) samples for each probe on the chip. The log Fold-change was calculated using R's (131) BioConductor (63) package.

Construction of Gene Coexpression Networks

After obtaining the expression matrices for each of the 33 ligands (33 expression matrices total), we merged expression levels from probesets that mapped onto the same gene. This was done by averaging the log(FC) values were across the probesets that mapped to the same gene as indicated by the microarray chip annotation information provided by Agilent. After obtaining a single expression matrix per ligand (where rows in the matrix are genes and columns are the replicates/timepoints for that particular ligand), pearson correlation was used to obtain the gene-coexpression matrices. We obtained 33 gene co-expression matrices ($\mathbf{E}^{1\ldots 33}$), one for each ligand, then applied a correlation cutoff of ≥ 0.8 to sparsify the matrices. Entries $e^k_{i,j}$ in the matrix $\mathbf{E}^k$ were set to 0 whenever $|e^k_{i,j}| < 0.8$ for $1 \leq k \leq 33$ and $1 \leq i, j \leq n$ where n is the number of genes/rows in the matrix $\mathbf{E}^k$. Remaining entries $|e^k_{i,j}| > 0$ signified edges in the networks that connected genes whose expression patterns were correlated above our chosen cutoff. The resulting networks were treated as undirected, weighted graphs with average of 10 thousand nodes (genes) and 3 million edges ($\binom{10{,}000}{2} \approx 50$ million possible edges in a fully connected graph). We varied the threshold cutoff around our chosen value (0.8) from $[0.78, 0.82]$ in 0.01 increments and the distances between the degree distributions (see for example Figure 1 in supplementary material) of the ligand networks did not significantly ($p < 0.01$) differ as measured by the Friedman test.

Gene Coexpression Network Alignment

Given two gene coexpression networks (graphs 1 and 2), the graphs are treated as weighted (where the weights on the edges denote the pairwise correlation in the expression of the corresponding genes). A k-hop neighborhood-based approach to alignment uses the notion of k-hop neighborhood (see (153; 154) for background on k-hop network alignment algorithm). The k-hop neighborhood of a vertex $v^1_x \in V_1$ of the graph $G_1(V_1, E_1)$ is simply a subgraph of G_1 that connects v^1_x with the vertices in V_1 that are reachable in k-hops from v^1_x using the edges in E_1. Given two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$, a mapping matrix $\mathbf{P}$ that associates each vertex in V_1 with zero or more vertices in V_2 (the matrix $\mathbf{P}$ can be constructed based on BLAST matches or gene id's. In our analysis, we used a 1-to-1 mapping between expression networks based on gene id's) and a user-specified parameter k, we construct for each vertex $v_x^1 \in V_1$ its corresponding k-hop neighborhood C_x in G_1. We then use the mapping matrix $\mathbf{P}$ to obtain the set of matches for vertex v_x^1 among the vertices in V_2; and construct the k-hop neighborhood Z_y for each matching vertex v_y^2 in G_2 and $\mathbf{P}_{v_x^1 v_y^2} = 1$. Let $S(v_x^1, G_2)$ be the resulting collection of k-hop neighborhoods in G_2 associated with the vertex v_x^1 in G_1. We compare each k-hop subgraph C_x in G_1 with each member of the corresponding collection $S(v_x^1, G_2)$ to identify the k-hop subgraph of G_2 that is the best match for C_x (based on a chosen similarity measure). We utilized a k-hop value of 1 for the analysis we discussed in this paper. The analysis was conducted on 8 nodes from the San Diego Supercomputer Center's Triton cluster with 8 cores and 24GB of memory per node.

Shortest path graph kernel score

The shortest path graph kernel was first described by Borgwardt and Kriegel (22). The kernel acts as a scoring function that compares the length of the shortest paths between any two nodes in a graph based on a pre-computed shortest-path distance. The shortest path distances for each graph may be computed using the Floyd-Warshall algorithm. We modified the Shortest-Path Graph Kernel to take into account the labels of the nodes being compared as computed by BLAST (3) or as a mapping in the mapping matrix $\mathbf{P}$. The shortest path graph kernel for subgraphs Z_{G_1} and Z_{G_2} (e.g., k-hop subgraphs) is given by:

$$S = \sum_{v_i^1, v_j^1 \in Z_{G_1}} \sum_{v_k^2, v_p^2 \in Z_{G_2}} \mathbf{P}_{v_i^1 v_k^2} \times \mathbf{P}_{v_j^1 v_p^2} \times d(v_i^1, v_j^1) \times d(v_k^2, v_p^2)$$

$$K(Z_{G_1}, Z_{G_2}) = \begin{cases} 0 & S = 0 \\ log[S] & otherwise \end{cases}$$

where $d(v_i^1, v_j^1)$ and $d(v_k^2, v_p^2)$ are the lengths of the shortest paths between v_i^1,v_j^1 and v_k^2,v_p^2 computed by the Floyd-Warshall algorithm. For gene-coexpression network, the Floyd-Warshall algorithm takes into account the weight of the edges (correlations) in the graphs. The runtime of the Floyd-Warshall Algorithm is $O(n^3)$. The shortest path graph kernel has a runtime of $O(n^4)$ (where n is the maximum number of nodes in larger of the two graphs being compared).

Hierarchical Clustering

A set of symmetric 33×33 distance matrix using the alignment scores across the 33 networks was constructed. Each matrix was constructed based on a specific subset of genes on the microarray chip (e.g., all genes involved in Calcium Signaling Pathway, all genes involved in Notch Signaling Pathway...etc. Please see Table 2 in paper and Tables 1 and 2 in supplementary material for a full list of pathways utilized for comparing the networks). For each matrix, the diagonals contained the sum of the rows in the matrix and the off diagonals contained the alignment score comparing the network from row i with network in column j where $1 \leq i, j \leq 33$. The hierarchical cluster was constructed using a neighbor-joining method based on the distance matrix in Matlab. The hierarchical cluster can be used to visualize relationship across the networks representing the gene expression changes due to exposure to different ligands. TreeView (124) was used to visualize the hierarchical clusters and Phylip's (53) "consense" program was used to merge hierarchical clusters and to compute majority-rule consensus trees. The majority rule consensus approach has been shown to minimize the number of false groupings and provides a good summary of the posterior distribution over the trees that were used to construct the consensus tree (75).

Authors' contributions

FT and SG assembled and verified the datasets for the analysis. FT wrote the algorithms, FT and SG ran the experiments and wrote the initial draft of the manuscript. SB and VH supervised the analysis, design of the algorithms and revisions to the manuscript. The authors declare no conflict of interest related to this work.

Acknowledgments

This research was supported in part by a Cornette Fellowship award and an Integrative Graduate Education and Research Training (IGERT) fellowship to Fadi Towfic, funded by the National Science Foundation (NSF) grant (DGE 0504304) to Iowa State University and NSF grant awards 0939370, 0835541 and 0641037 to Shankar Subramaniam. We also thank Raj Srikrishnan for his help in data processing. The work of Vasant Honavar was supported by the National Science Foundation, while working at the Foundation. Any opinion, finding, and conclusions contained in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Figures

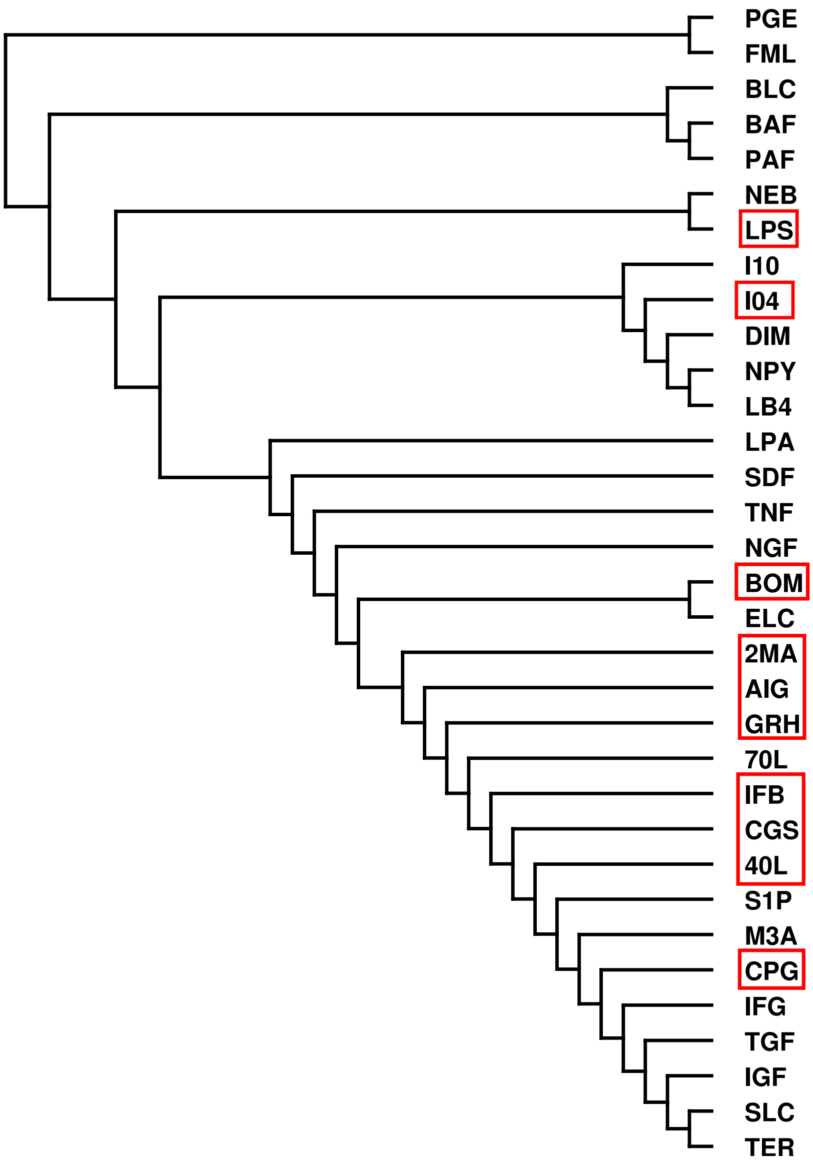


Figure 5.1 Clustering Based on Toplogical Features (Degree Distribution). Ligand networks with a high number of differentially expressed genes relative to untreated samples (as indicated in (104)) have been highlighted in the figure.

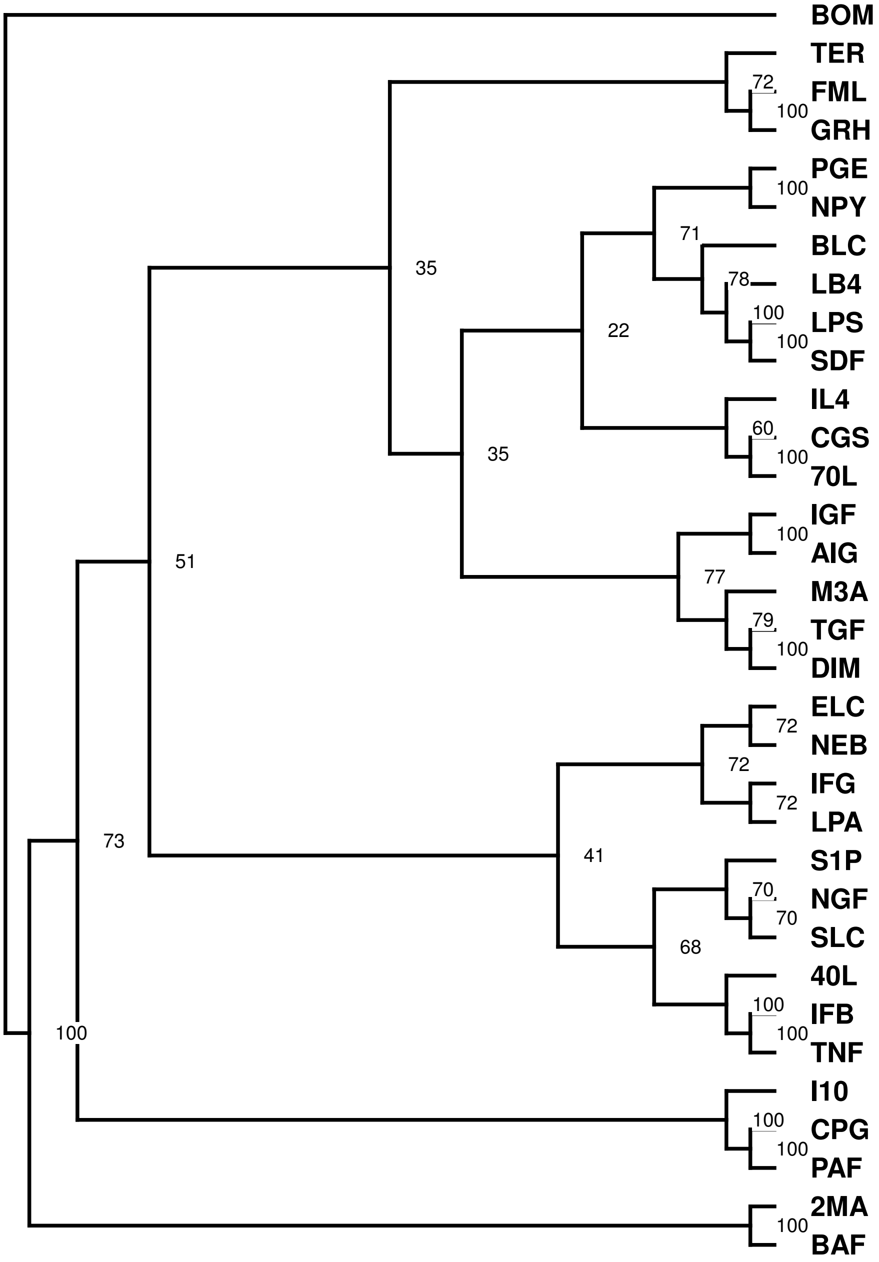


Figure 5.2 Bootstrapped tree showing the relationship between all 33 ligand networks. This tree shows that ligands with similar induced reaction (e.g., LPS and SDF, both affect pathways involved in cell migration) cluster together.

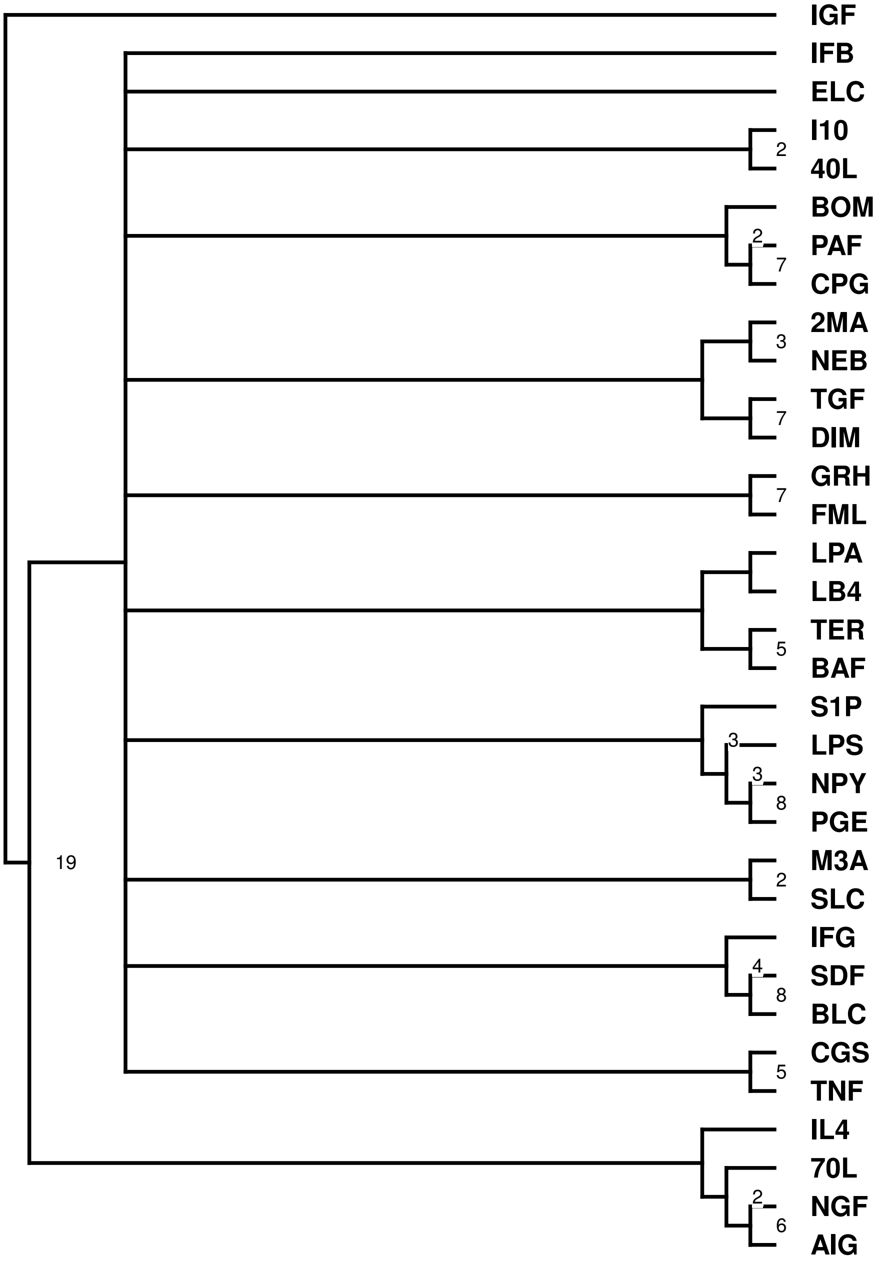


Figure 5.3 Consensus tree constructed based on all metabolism pathways in Table 2. The values on the branches indicate the total number of times the branch appeared across all networks (total of 19). If no value is indicated, the branch appeared only once.

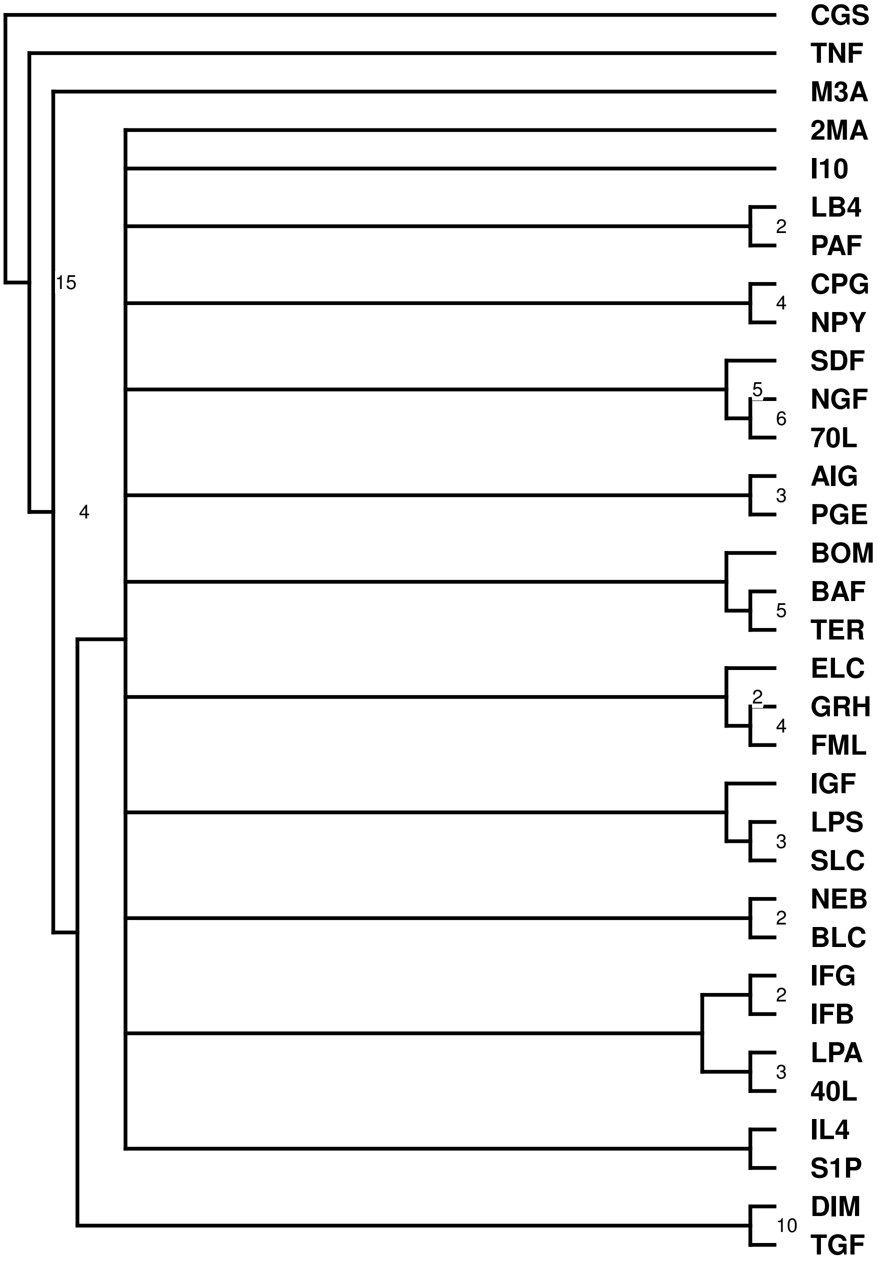


Figure 5.4 Consensus tree constructed based on all Genetic Information Processing pathways in Table 2. The values on the branches indicate the total number of times the branch appeared across all networks (total of 15). If no value is indicated, the branch appeared only once.

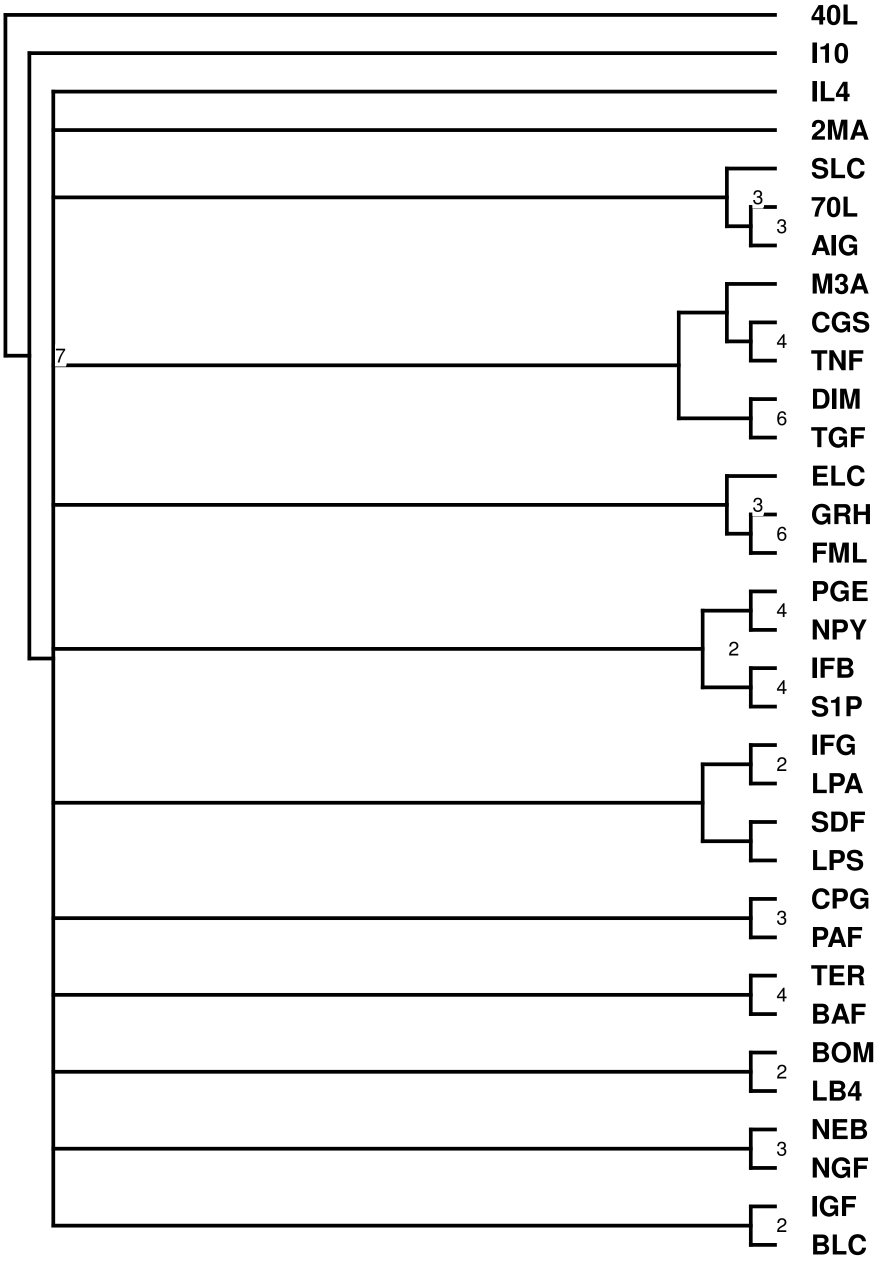


Figure 5.5 Consensus of all pathway categories in Table 2. The values on the branches indicate the total number of times the branch appeared across all networks (total of 7). If no value is indicated, the branch appeared only once.

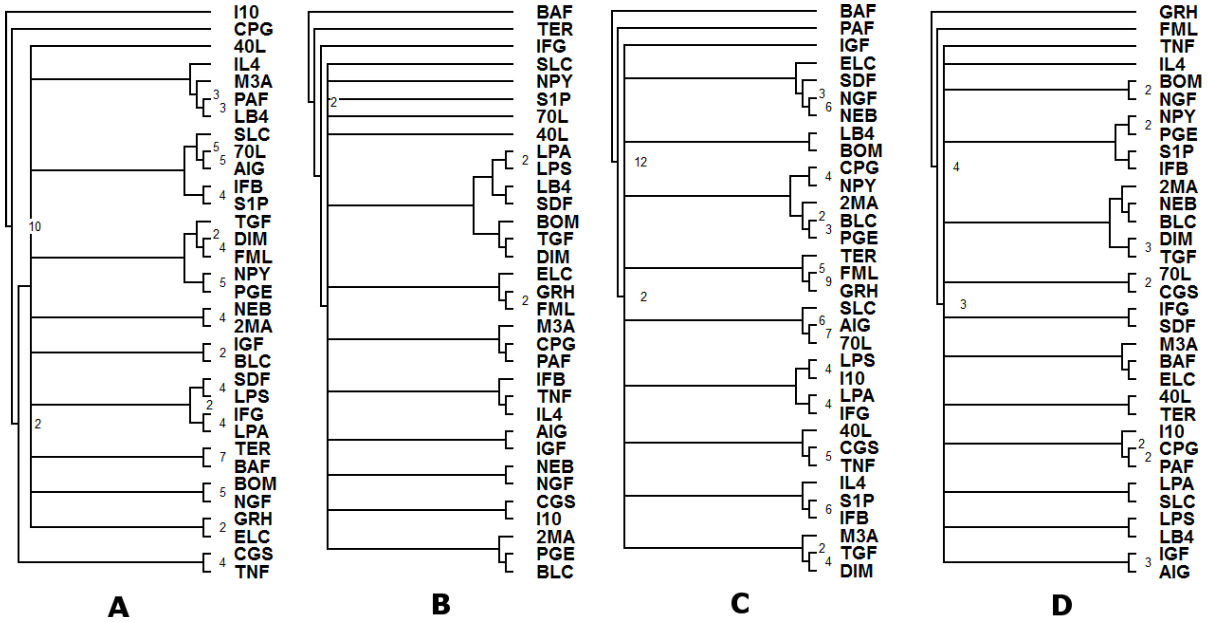


Figure 5.6 **A**: Consensus tree constructed based on all Cellular Processes pathways in Table 2. **B**: Consensus tree constructed based on all Environmental Information Processing pathways in Table 2. **C**: Consensus tree constructed based on all Human Diseases pathways in Table 2. **D**: Consensus tree constructed based on all Immune System pathways in Table 2. The values on the branches indicate the total number of times the branch appeared across all networks (totals of 10, 2, 12, and 4 for **A**, **B**, **C**, and **D** respectively). If no value is indicated, the branch appeared only once.

Tables

Ligand Abbreviation	Ligand Name
2MA	2-Methyl-thio-ATP
AIG	Antigen (Anti-Ig)
BAF	BAFF (B-cell activating factor)
BLC	BLC (B-lymphocyte chemoattractant)
BOM	Bombesin
40L	CD40 ligand
70L	CD70/CD27 ligand
CGS	CGS-21680 hydrochloride (2-p-[2-Carboxyethyl]phenethylamino-5'-N-ethylcarboxamidoadenosine)
CPG	CpG-Containing Oligonucleotide
DIM	Dimaprit
ELC	ELC (Epstein Barr Virus-induced molecule-1 Ligand Chemokine)
FML	fMLP (formyl-Met-Leu-Phe)
GRH	Growth hormone-releasing hormone
IGF	Insulin-like growth factor 1
IFB	Interferon-beta
IFG	Interferon-gamma
I10	Interleukin 10
IL4	Interleukin 4
LPS	Lipopolysaccharide
LB4	Leukotriene B4 (LTB4)
LPA	Lysophosphatidic acid
M3A	MIP3-alpha (Macrophage inflammatory protein-3)

Ligand Abbreviation	Ligand Name
NEB	Neurokinin B
NPY	Neuropeptide Y
NGF	NGF (Nerve Growth Factor)
PAF	Platelet activating factor
PGE	Prostaglandin E2
SDF	SDF1 alpha (Stromal cell derived factor-1)
SLC	SLC (Secondary lymphoid-organ chemokine)
S1P	Sphingosine-1-phosphate
TER	Terbutaline
TNF	Tumor necrosis factor-alpha
TGF	Transforming growth factor-beta 1

Table 5.1 Full list of the ligands and their abbreviations used in the experiments analyzed in this paper. This list was adapted from Lee et al. (104)

KEGG pathway category	Number of subpathways	Subpathway KEGG ID's
Cellular Processes	10	mmu04142, mmu04144, mmu04145, mmu04520, mmu04540, mmu04810, mmu04110, mmu04114, mmu04115, mmu04140
Environmental Information Processing	2	mmu04150, mmu04310
Organismal System	6	mmu04962, mmu04964, mmu04966, mmu04260, mmu04722, mmu04910

KEGG pathway category	Number of sub-pathways	Subpathway KEGG ID's
Genetic Information Processing	15	mmu03020, mmu03022, mmu03030, mmu03040, mmu03050, mmu03060, mmu03410, mmu03420, mmu03430, mmu03440, mmu04120, mmu04130, mmu00970, mmu03010, mmu03018
Human Diseases	12	mmu05100, mmu05210, mmu05212, mmu05214, mmu05215, mmu05216, mmu05219, mmu05222, mmu05010, mmu05012, mmu05014, mmu05016
Immune System	4	mmu04623, mmu04662, mmu04666, mmu04622
Metabolism	19	mmu00020, mmu00030, mmu00051, mmu00072, mmu00100, mmu00130, mmu00190, mmu00230, mmu00240, mmu00260, mmu00290, mmu00460, mmu00510, mmu00511, mmu00563, mmu00630, mmu00670, mmu00740, mmu00900

Table 5.2 List of pathways detected based on high-intensity probes from the microarray data. Please see Table 1 in supplementary material for a more detailed version of this table with pathway names and relative number of genes enriched in the pathway based on the data

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
70L/AIG/SLC	Cellular Processes, Human Diseases, Organismal System	Cell cycle, p53 signaling pathway, Phagosome, Parkinson's disease, Huntington's disease
LPA/IFG	Cellular Processes, Human Diseases	p53 signaling pathway, Bacterial invasion of epithelial cells
GRH/FML	Cellular Processes, Environmental Information Processing, Genetic Information Processing, Human Diseases, Metabolism, Organismal System	Cell cycle, Regulation of autophagy, Aminoacyl-tRNA biosynthesis, Ribosome, RNA degradation, RNA polymerase, DNA replication, Ubiquitin mediated proteolysis, Parkinson's disease, Huntington's disease, Thyroid cancer, TCA cycle, Oxidative phosphorylation, Pyrimidine metabolism, Glyoxylate and dicarboxylate metabolism
PGE/NPY	Cellular Processes, Immune System, Metabolism, Organismal System	Oocyte meiosis, Cytosolic DNA-sensing pathway, Fc gamma R-mediated phagocytosis, TCA cycle, Ubiquinone and other terpenoid-quinone biosynthesis, Oxidative phosphorylation, Pyrimidine metabolism, Riboflavin metabolism, Terpenoid backbone biosynthesis
IFB/S1P	Cellular Processes, Human Diseases, Immune System, Organismal System	Cell cycle, Oocyte meiosis, p53 signaling pathway, Parkinson's disease, Huntington's disease, Bacterial invasion of epithelial cells, Fc gamma R-mediated phagocytosis
BOM/LB4	Human Diseases, Organismal System	Colorectal cancer

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
NEB/NGF	Environmental Information Processing, Human Diseases, Organismal System	Parkinson's disease, Colorectal cancer
TNF/CGS	Cellular Processes, Genetic Information Processing, Human Diseases, Metabolism	Cell cycle, p53 signaling pathway, Ribosome, DNA replication, Mismatch repair, SNARE interactions in vesicular transport, Parkinson's disease, Bacterial invasion of epithelial cells, Steroid biosynthesis, Oxidative phosphorylation, Glyoxylate and dicarboxylate metabolism
PAF/CPG	Environmental Information Processing, Immune System, Metabolism	RIG-I-like receptor signaling pathway, Cytosolic DNA-sensing pathway, Pyrimidine metabolism, Cyanoamino acid metabolism, One carbon pool by folate, Riboflavin metabolism
TER/BAF	Cellular Processes, Environmental Information Processing, Genetic Information Processing, Metabolism	Cell cycle, Oocyte meiosis, p53 signaling pathway, Endocytosis, Aminoacyl-tRNA biosynthesis, RNA degradation, Spliceosome, Ubiquitin mediated proteolysis, TCA cycle, Pentose phosphate pathway, Cyanoamino acid metabolism

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
DIM/TGF	Environmental Information Processing, Genetic Information Processing, Human Diseases, Immune System, Metabolism, Organismal System	Aminoacyl-tRNA biosynthesis, Ribosome, RNA polymerase, Basal transcription factors, Spliceosome, Protein export, Mismatch repair, Bacterial invasion of epithelial cells, Colorectal cancer, RIG-I-like receptor signaling pathway, Cytosolic DNA-sensing pathway, B cell receptor signaling pathway, TCA cycle, Pentose phosphate pathway, Steroid biosynthesis, Oxidative phosphorylation

Table 5.3 Top matched ligands based on expression patterns in the consensus tree shown in Figure 5. The KEGG pathway categories correspond to the pathway categories highlighted in Table 2. Please see Table 3 in the supplementary material for an expanded version of this table

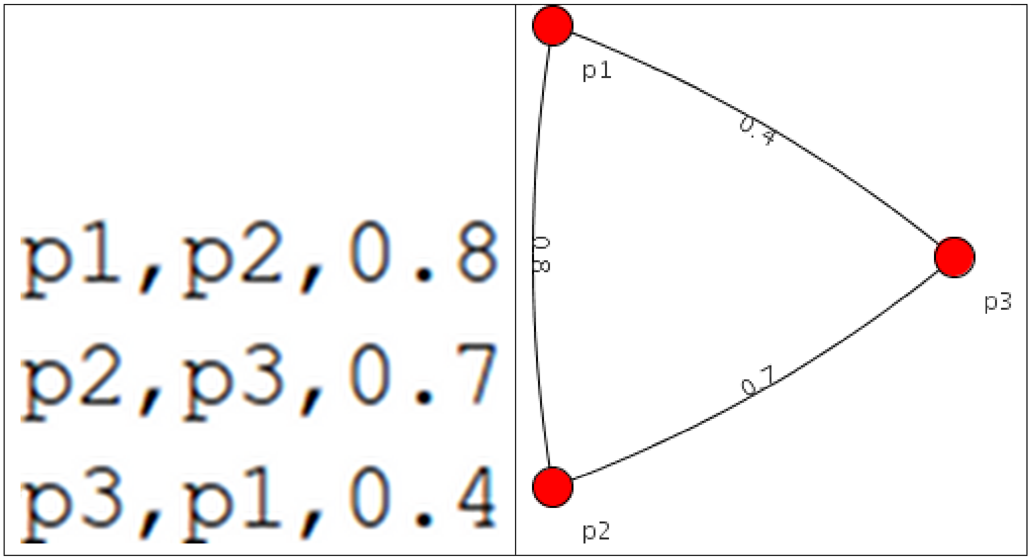
CHAPTER 6. TOOLS

The Biomolecular Network Alignment Algorithm (BiNA) has been implemented as a platform-independent software library written in Java. The library has a command-line interface suitable for deployment on servers and for scripting purposes. A user-friendly webserver that offers many of the same features as the commandline version, plus visualization of the results, has also been implemented. The following sections discuss the features and implementation of the BiNA web server and the BiNA software library.

6.1 BiNA webserver

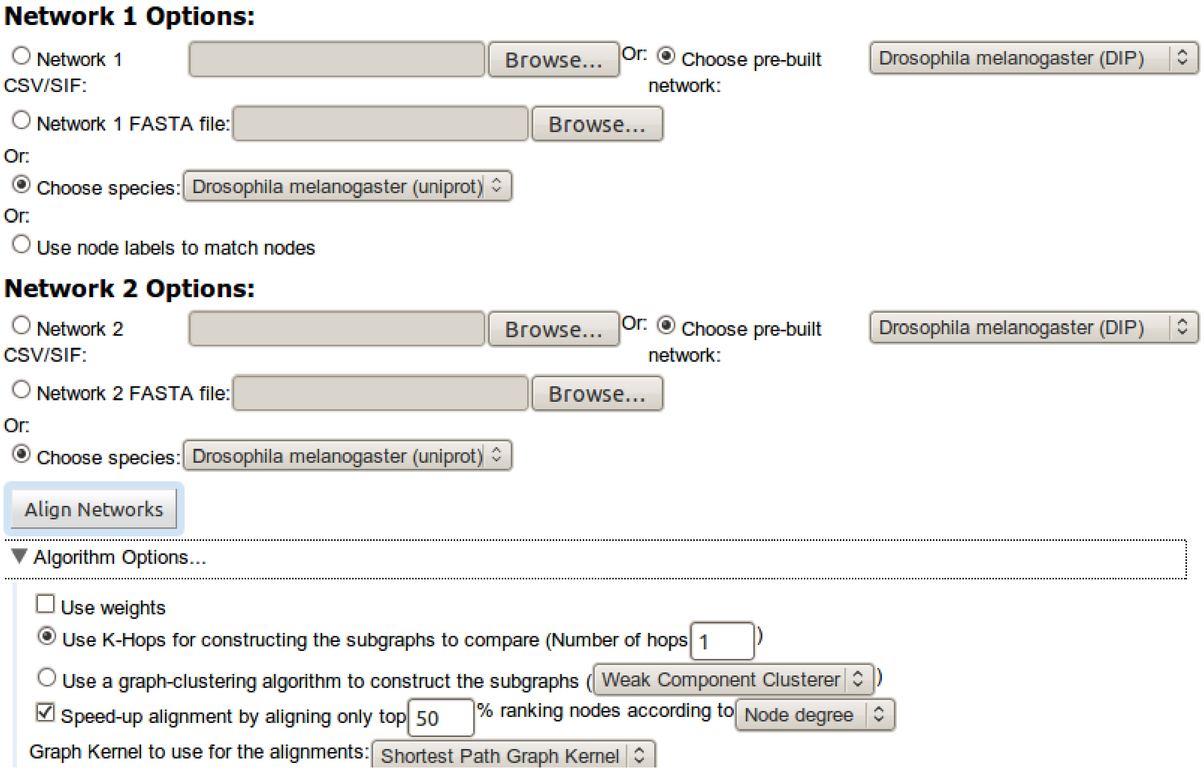
BiNA can align protein-protein interation networks and gene-coexpression networks saved in files as described in figure 6.1. The tool can also make use of sequence-level information to match nodes automatically, taking into account the sequence-conservation score based on BLASTp/n. Alternatively, the user can forgo supplying sequences for nodes in the networks being aligned if both networks to be aligned have the same node ids. BiNA supports weighted and unweighted protein-protein interaction network representations, as well as gene-coexpression networks through the same interface.

The alignment algorithm relies on two basic procedures (1) dividing the networks into smaller subnetworks (2) matching the smaller subnetworks to reconstruct the alignment. The options for the two steps of the alignment are highlighted under “alignment options” of the main page (see figure 6.2). In the divide step, the user may choose various graph partitioning and clustering algorithms to break-down the networks into smaller substructures. The default algorithm for breaking down the networks is the K-Hop algorithm discussed in our earlier publications (152; 154; 151). Briefly, the K-Hop approach constructs a vertex-induced subgraph



```
p1,p2,0.8
p2,p3,0.7
p3,p1,0.4
```

Figure 6.1 **Left:** Sample format for specifying the topology of a weighted, undirected network in CSV format. This file format can be generated by Cytoscape (similar to Simple Interaction File (SIF) format). The separators can be commas or any whitespace (e.g., space or tab) character. **Right:** Visualization of the network described by the file on the left



Network 1 Options:

○ Network 1 CSV/SIF: [ ] Browse... Or: ● Choose pre-built network: Drosophila melanogaster (DIP)

○ Network 1 FASTA file: [ ] Browse...

Or:

● Choose species: Drosophila melanogaster (uniprot)

Or:

○ Use node labels to match nodes

Network 2 Options:

○ Network 2 CSV/SIF: [ ] Browse... Or: ● Choose pre-built network: Drosophila melanogaster (DIP)

○ Network 2 FASTA file: [ ] Browse...

Or:

● Choose species: Drosophila melanogaster (uniprot)

Align Networks

▼ Algorithm Options...

□ Use weights

● Use K-Hops for constructing the subgraphs to compare (Number of hops 1)

○ Use a graph-clustering algorithm to construct the subgraphs (Weak Component Clusterer)

☑ Speed-up alignment by aligning only top 50 % ranking nodes according to Node degree

Graph Kernel to use for the alignments: Shortest Path Graph Kernel

Figure 6.2 Main alignment parameters on the input screen of the BiNA webserver

for each node in the graph by including the node and its neighbors. The neighborhood may be expanded by including the neighbors-of-neighbors (i.e., setting the number of hops to 2) and so on. Increasing the number of hops may improve the alignment at the cost of computational time. In our experiments, setting the number of hops to 1 or 2 produced accurate alignments without adding much computational stress (152; 154; 151).

In the matching step, the user may select different scoring functions (Graph Kernels) to compute the similarity between the clusters resulting from step (1). Currently, the webserver only supports the Shortest Path and Random Walk graph kernels. Briefly, the shortest path kernel matches graphs based on the length of the shortest paths between similarly-labeled nodes (recall that nodes are matched based on BLAST score or node ids) while the random walk kernel matches graphs based on the transition probability between similarly-labeled neighbors. The resulting score depends on the size of the graphs being matched with 0 being a poor score (no substructures matched across the node neighborhoods being compared) while a high score implies a good match. To speed up the alignment, users may also restrict the alignment to the top $X\%$ hubs defined by node-degree, betweeness, Hubs-and-authorities, and a random ranking. The default number of hubs to align is the top 50% according to the node degree.

6.2 BiNA program

BiNA is implemented as a multi-threaded java-based hardware-independent software library. The key elements of the library are shown in Figure 6.3. The library has been designed so as to provide the maximum exibility and accessibility to users through the provided Java API (Application Programming Interface) as well as the implemented command-line and HTML interfaces. The core of the toolkit is a set of APIs for comparing, scoring, and partitioning Undirected and weighted/unweighted graphs. The implementation of the software utilizes the already-established JUNG (Java Universal Graph Framework) and COLT (CERN's highperformance computing library) for manipulating graphs and performing matrix computations. BiNA provides an input interface for submitting datasets from files, databases or URLs. The APIs also allow users to select the number of threads to utilize to speedup computations. Furthermore, the API allows the graph decomposition (clustering), node-matching and graph

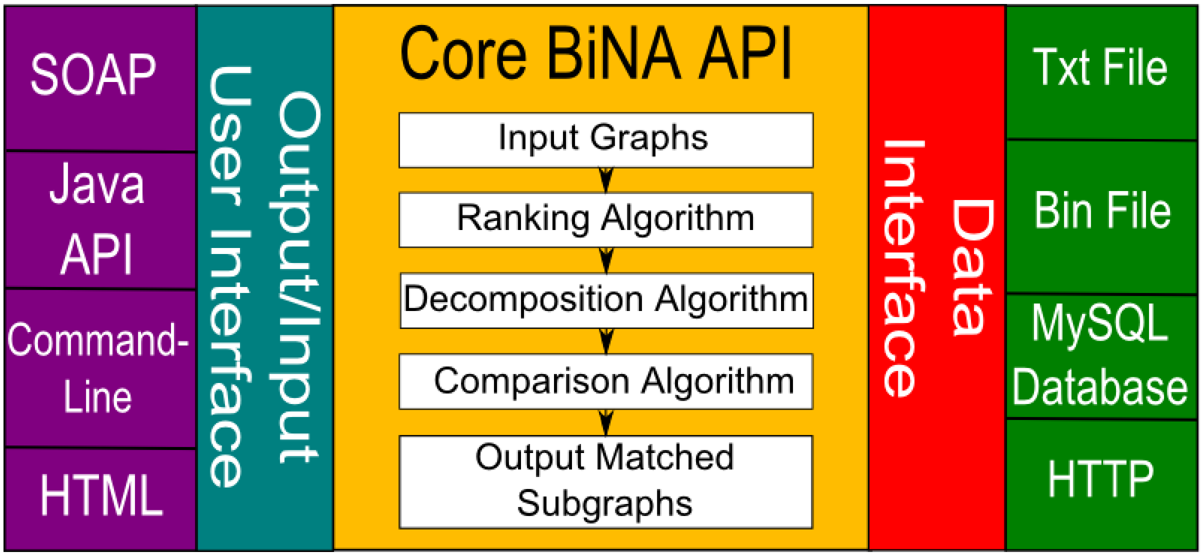


Figure 6.3 Overview diagram of BiNA's service-oriented architecture model

comparison algorithms to be used individually as well as in combination with each other. Additionally, many of the components of BiNA (e.g., the network decomposition algorithms) can be easily modified or extended (in Java or otherwise) so long as the user provides the resulting graph as a supported type in the Data Interface library. BiNA toolkit is extensible by implementing one of the already defined Java interfaces allowing the addition of graph comparison algorithms or data interfaces to the core BiNA toolkit. The network alignment algorithms and API are implemented in Java due to the language's flexibility, hardware-independence, and the wide-availability of libraries for scientific computing to the platform. As such, the program can be run on any Java 1.5-certified JVM on Linux/Unix, Windows, or Mac OS X.

As BiNA can run across multiple processors, the speedup of the algorithm (a measure of how fast one can expect the algorithm to perform by adding more processors) was calculated. Speedup on p processors (S_p) is defined as $S_p = T_1/T_p$ where T_1 is the time it takes the algorithm to run on a single processor (sequential) and T_p is the time it takes the algorithm to run on p processors. As can be seen from figure 6.4, BiNA's implementation achieves linear speedup in most situations. In other words, one can expect that if two processors are allocated to run the alignment, the algorithm will run nearly twice as fast as indicated in figure 6.4. In some situations, superlinear speedup is achieved (i.e., if 6 processors are allocated to run the

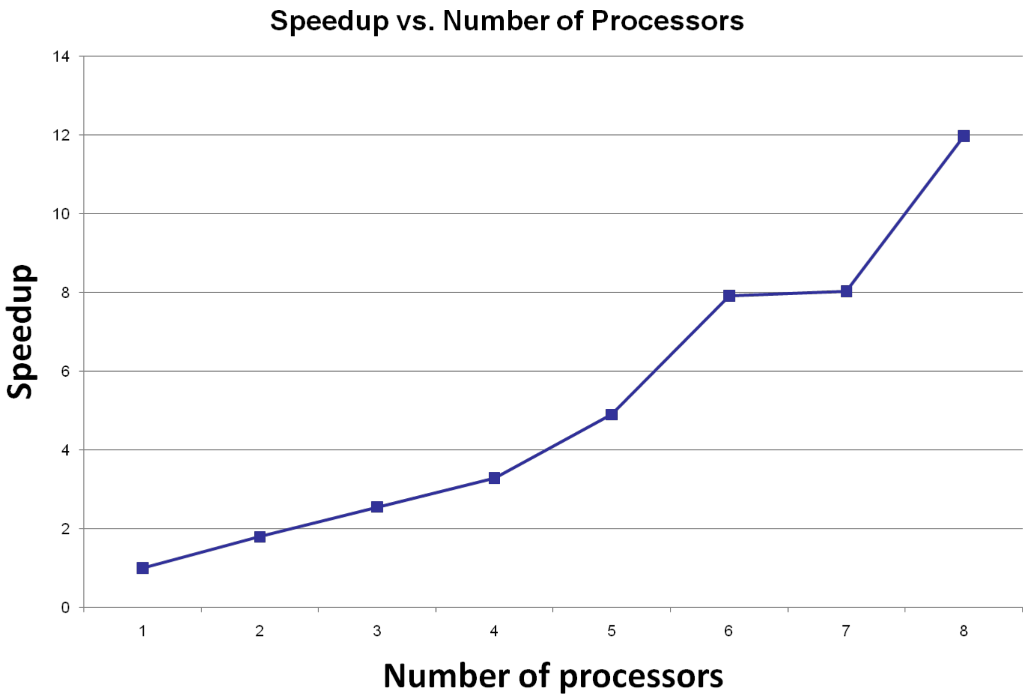


Figure 6.4 BiNA's scalability on multiple processors as measured by speedup. As can be seen from the figure, the implementation of the algorithm is highly scalable, allowing full utilization of additional processors with little performance penalty

algorithm, one can expect the algoritm to run nearly 8 times as fast) due to the fact that the results from some computations are cached in memory, saving some laborious similarity scores from being recomputed.

CHAPTER 7. CONCLUSIONS

With the availability of a wealth of high-throughput data from biological systems (45; 83; 126), the representation of the relationships between the entities (genes, proteins, metabolites) in such datasets as interaction networks offers a powerful approach to analyzing the interdependencies among each of the biomolecular entities in living cells. Specifically, such representations allow for the discovery of conserved pathways among different species (88; 145), finding protein groups that are relevant to disease (77; 108), discovery of the chemical mechanism of metabolic reactions (134; 91) and more (172; 92; 137; 17; 1). The rapidly advancing field of systems biology aims to understand the structure, function, dynamics, and evolution of complex biological systems in terms of the underlying networks of interactions among the large number of molecular participants involved including genes, proteins, and metabolites (29; 165). Of particular interest in this context is the problem of comparing and aligning multiple networks e.g., those generated from measurements taken under different conditions, different tissues, or different organisms (139). Despite the recent appearance of several algorithms for alignment of protein-protein interaction networks (89; 97; 81; 56; 149), regulatory networks (169; 139) and metabolic networks (127; 6), most of the network alignment algorithms exhibit long running times (140), do not leverage biological properties of the networks being aligned (142), or make some unrealistic simplifying assumptions (142). Furthermore, verification of the alignment results of biomolecular networks currently rely on GO keyword enrichment among the alignment modules, which might provide overoptimistic results due to over-generalization of keywords.

This dissertation provides a set of efficient (in terms of the running time complexity) and accurate (in terms of the evaluation criteria) network alignment algorithms for biomolecular networks. Specifically, the algorithms provided as part of this research exploit the node-labels, the various edge types and modularity of biomolecular networks. All the alignment algorithms have been evaluated based on their ability to reproduce biologically relevant alignments and output in terms of the aligned modules.

7.1 Significant contributions of dissertation

This dissertation provides a class of flexible (in terms of ease of modification), scalable (in terms of computational running time), and accurate (in terms of biological significance) algorithms for comparing and aligning biomolecular networks while making minimum assumptions about the source of the networks. The networks can be labeled (e.g., sequence labeled, or nodes can be matched based on orthology) or unlabeled (networks can be aligned strictly based on topology). The following sections describe the main contributions of this dissertation against the background of the current literature in the field.

7.1.1 First highly modular algorithm in the field

Chapter 2 describes the Biomolecular Network Alignment (BiNA) toolkit in detail. This algorithm is the first algorithm in the field whose scoring (comparison) functions and partition (clustering) functions are independent. Furthermore, this algorithm uses the proven divide and conquer strategy to enable the future addition of new techniques for partitioning and scoring without changing the overall method.

7.1.2 Highly scalable algorithm

BiNA can run on desktop machine to clusters, aligning networks from 100's of edges to several millions. Chapter 6 describes the implementation details and scalability of the algorithm in detail. The running time of the various methods that comprise this algorithm are described in detail in chapter 2.

7.1.3 First highly flexible algorithm

BiNA can align undirected, unweighted protein interaction networks and undirected, weighted gene-coexpression networks. BiNA can align within the same organism or across species, can align based on topology alone or using node labels or BLAST correspondence. Experiments on aligning networks from different species are provided in chapters 2, 3 and 4. Experiments outlining the alignment of networks within the same organism are provided in chapter 5. The alignment techniques based on strict topology and discussion of applications of topology to the alignment problem are provided in chapter 3.

7.1.4 Highly portable

BiNA has been implemented purely in Java to achieve maximum portability on Windows, Mac and Linux/Unix systems). The BiNA webserver is user-friendly and accessible. The architecture and implementation of the algorithm are discussed in chapter 6.

7.1.5 High accuracy in terms of biological performance

BiNA has been evaluated in several respects to assess the biological relevance of the algorithm's output. Several assessments currently available in the literature are:

- Detection of enriched GO Terms (chapters 2 and 3)
- Construction of phylogenies based on labeled and unlabeled protein-protein interaction networks (chapter 3)
- Detection of orthologs (chapter 4)

7.1.6 Applied to important biological problems

BiNA has been applied to several important biological questions. Two of the applications currently available in the literature are:

- Detection of orthologs based on protein-protein and gene coexpression networks (chapter 4)
- Detection of expression patterns in B-Cells (chapter 5)

7.2 Open problems

Several open problems in the field of systems biology are related to the network alignment problem (140; 142). The construction and refinement of reference networks, for example, may be enhanced by alignment (142). Other applications include the construction of phylogenetic trees based on network models (101), detection of conserved biological modules (139; 145; 114; 170; 121), and identification of orthologs (13; 154), among others. As noted by Sharan and Ideker, tools for network alignment have the potential to revolutionize network comparisons similar to how tools like BLAST revolutionalized sequence comparisons (139). As such, this field can significantly benefit from developments in the key areas discussed in the following sections.

7.2.1 Evaluation methods

Several evaluation metrics currently exist based on Gene Ontology (4), orthology detection (13; 154; 56), phylogenetic tree construction (101), and significance of alignment based on evolutionary network models (97). However, many of the methods are based on metrics that may not directly measure alignment performance (e.g., GO enrichment, phylogenetic relationship reconstruction), or depend on annotations in databases that may prove problematic for assessing performance on unannotated species (e.g., measures that depend on orthology and GO such as orthology detection, significance based on evolutionary models...etc). As such, more robust evaluation metrics based strictly the networks to be aligned must be developed.

7.2.2 Applicability to more network models

As mentioned in Chapter 1, various network models have been successfully utilized in the literature to study biomolecular interactions. Among the simplest and more straightforward models are the undirected graphs that are addressed in this dissertation. Future improvements to the algorithms discussed in this dissertation should include applications to comparisons between additional network models, such as directed graphs that are used to represent gene regulatory networks (160; 85; 137) and weighted directed graphs that are used to represent

Bayesian networks (19; 42; 59). Furthermore, comparison techniques that take into account heterogeneous models (e.g., compare protein-protein interaction networks with gene regulatory networks) are also important for developing and refining biomolecular network models of biological processes.

7.2.3 More detailed network models

Recently, representations of networks as multi-graphs (graphs that contain multiple edges or multiple labels associated with graph components) has gained attention due to studies that sought to integrate various biological data (e.g., expression, protein interaction, protein modification) into a single network model (20; 34). As more sophistacted mathematical models, such as tensor representations of multigraphs (7; 50; 95), become accessible for dealing with large datasets, network comparison algorithms need to take into account multiple labels associated with edges and nodes in biological networks. Our kernel-based approach can be extended into tensor space by taking advantage of recent developments in the machine-learning literature on tensor kernels (69; 72). Furthermore, due to the inherit uncertainty in some experimental setups for deducing network models (e.g., inaccurate yeast-two-hybrid protein-protein interaction data), network models that associate confidence levels with edges in such networks are gaining popularity (144; 10; 9). Thus, comparison algorithms will need to explicitly take into account experimental confidence values associated with measures used to construct such network models.

7.2.4 Rapid comparisons

Rapid comparison of network models and modules can be very useful for detecting already existing patterns in data. Databases such as GEO (45) and Array Express (125) have long housed expression data (e.g., over $500,000$ expression samples are available in GEO as of March 10, 2011) and those databases will dramatically increase in size as next-gen expression datasets are added. As such, scientists currently do not have a meaningful way of querying this data based on any parameters other than name of the dataset, name of depositing individual/institution, date of deposit, or basic keywords in the dataset description. Recently, a

BLAST search option has been added to allow searching for datasets where a certain gene is known to be available on the microarray platform for that dataset. Network alignment methods provide a natural means of querying the data and annotating datasets based on the expression patterns recorded in the samples. For example, users may be able to search for all datasets where a specific pathway is up or down regulated, find datasets where sets of genes have a specific pattern of regulation/interaction relative to each other and so on. Thus, fast network alignment approaches (e.g., based strictly on topology) may be used to detect gene expression or protein-protein interaction datasets where specific patterns may exist, then the results may be refined by a more detailed alignment approach (based on node labels) to provide a set of datasets that are strongly likely to exhibit the query network/module.

7.2.5 Integrated pipeline for analysis and visualization

The networks dealt with as part of this dissertation typically span hundreds of nodes and thousands of edges (the Drosophila melanogaster protein-protein interaction network, for example, is over 6000 nodes and $20,000$ edges). Thus, due to the size of the networks, visualization has been limited to the comparison of one network relative to others (e.g., human protein-protein interaction network vs. mouse protein-protein interaction network), rather than relationships between individual subgraphs (or pathways) within each network (e.g., comparison of specific interactions lost or gained within each pathway within each organism). This is due to the fact that the visualization of the alignment results, or any large graph structure that contains thousands of nodes and edges, has not been adequately addressed in the literature. Typically, tools such as Cytoscape (138) or GraphCrunch (102; 116) heavily rely on 2D graph layout algorithms, making the display of graphs with numerous nodes highly problematic on typical display or print resolutions. As such, newer visualization methods may need to be developed for specifically displaying aligned graphs (or the alignment graph itself) based on VANLO (26), or CIRCOS (100). Such visualization methods can be integrated as part of a full analysis pipeline based on network alignment for aligning networks, visualizing the result, refining the alignment if necessary, and generating testable hypothesis based on the comparison results.

APPENDIX. ADDITIONAL MATERIAL FOR CHAPTER 6

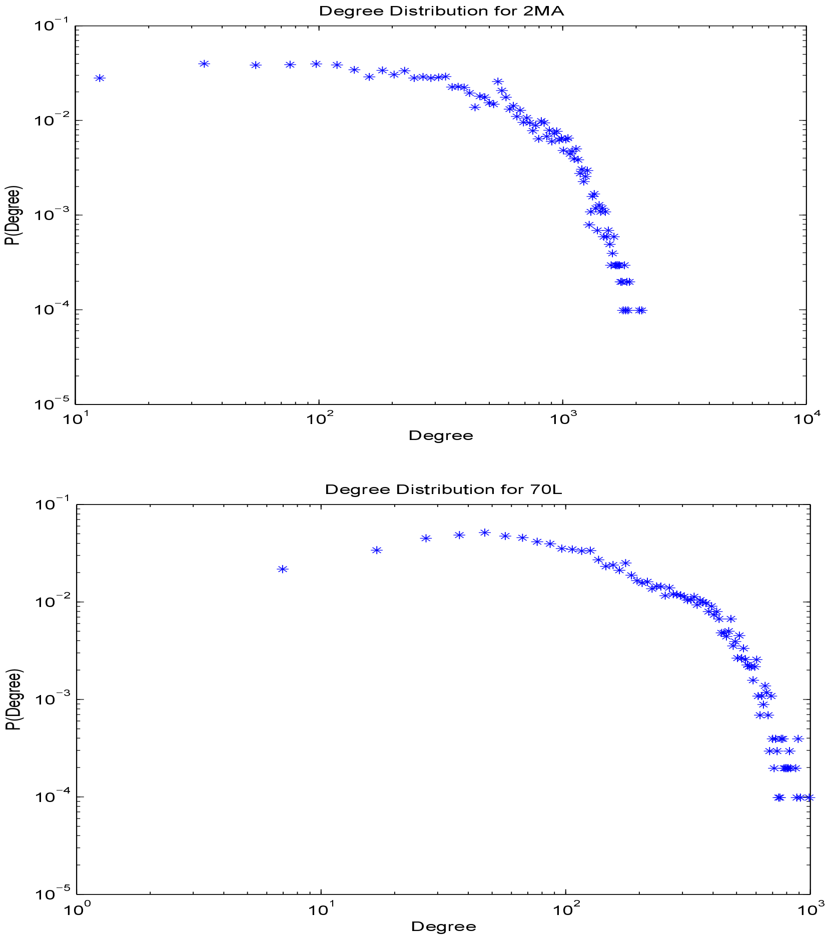


Figure A.1 Example of Degree distributions used for Kolmogorov-Smirnov test for initial clustering of the ligands based on network topology. As can be seen from the figure, the expression networks exhibit scale-free like behavior as described by Barabsi and Oltavai (17).

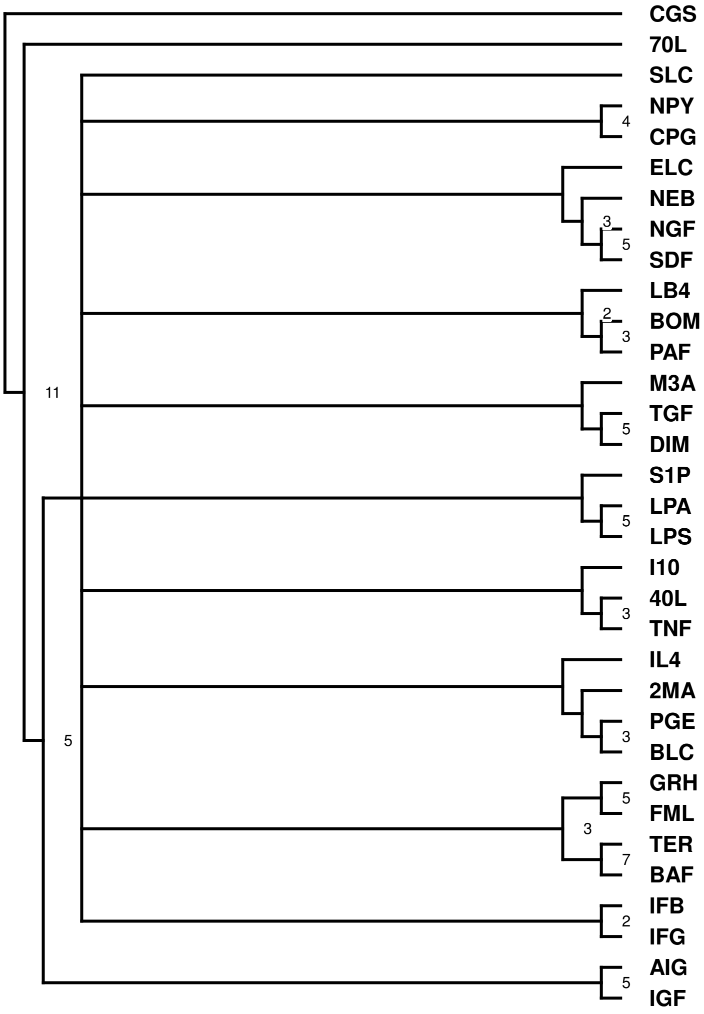


Figure A.2 Consensus tree constructed based on all KEGG pathways in Table A.1 in supplementary material. The values on the branches indicate the total number of times the branch appeared across all networks (total of 11). If no value is indicated, the branch appeared only once.

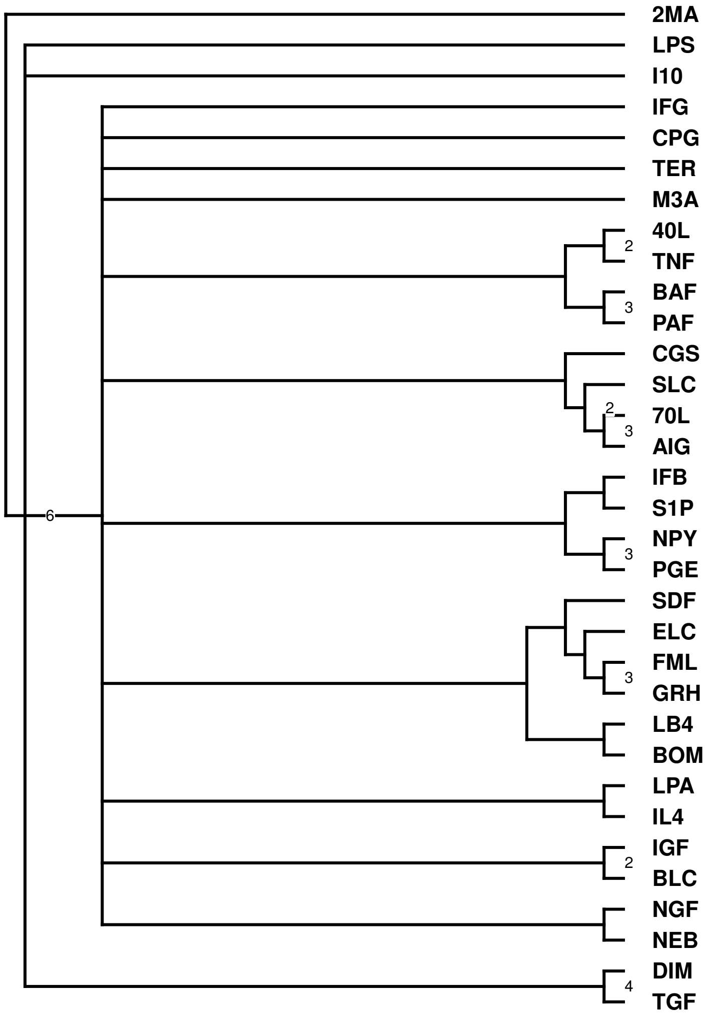


Figure A.3 Consensus tree constructed based on all Organismal System pathways in table 2 in the supplementary material (table 2 in the paper). The values on the branches indicate the total number of times the branch appeared across all networks (total of 6). If no value is indicated, the branch appeared only once.

Signaling Pathway	Number of Neighborhoods Utilized in Alignment	KEGG Pathway Reference (if Applicable)
Calcium	204	mmu04020
ErbB	93	mmu04012
Wnt	162	mmu04310
VEGF	80	mmu04370
TGF-beta	92	mmu04350
Phosphatidylinositol	81	mmu04070
Notch	58	mmu04330
mTOR	56	mmu04150
Jak-STAT	160	mmu04630
Hedgehog	55	mmu04340
MAPK	279	mmu04010
MAPK (ERK-MAPK)	10	N/A
MAPK (p38)	6	N/A
MAPK (JNK-SAPK)	6	N/A

Table A.1 Full list of networks and the number of neighborhoods utilized for comparing the networks for figure 2 in supplementary material.

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Protein export	mmu03060	16	36	0.444444444	2. Genetic Information Processing
Ribosome	mmu03010	47	133	0.353383459	2. Genetic Information Processing

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Citrate cycle (TCA cycle)	mmu00020	11	32	0.34375	1. Metabolism
Proteasome	mmu03050	17	50	0.34	2. Genetic Information Processing
Cyanoamino acid metabolism	mmu00460	2	6	0.333333333	1. Metabolism
One carbon pool by folate	mmu00670	6	19	0.315789474	1. Metabolism
Mismatch repair	mmu03430	7	23	0.304347826	2. Genetic Information Processing
Ubiquitin mediated proteolysis	mmu04120	46	156	0.294871795	2. Genetic Information Processing
Glyoxylate and dicarboxylate metabolism	mmu00630	5	17	0.294117647	1. Metabolism
Nucleotide excision repair	mmu03420	13	45	0.288888889	2. Genetic Information Processing

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Oxidative phosphorylation	mmu00190	49	170	0.288235294	1. Metabolism
Ubiquinone and other terpenoid-quinone biosynthesis	mmu00130	2	7	0.285714286	1. Metabolism
DNA replication	mmu03030	10	36	0.277777778	2. Genetic Information Processing
SNARE interactions in vesicular transport	mmu04130	10	37	0.27027027	2. Genetic Information Processing
Parkinson's disease	mmu05012	49	182	0.269230769	6. Human Diseases
Riboflavin metabolism	mmu00740	4	15	0.266666667	1. Metabolism
RNA polymerase	mmu03020	9	34	0.264705882	2. Genetic Information Processing

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Spliceosome	mmu03040	39	151	0.258278146	2. Genetic Information Processing
p53 signaling pathway	mmu04115	19	76	0.25	4. Cellular Processes
Aminoacyl-tRNA biosynthesis	mmu00970	11	44	0.25	2. Genetic Information Processing
Basal transcription factors	mmu03022	9	36	0.25	2. Genetic Information Processing
Regulation of autophagy	mmu04140	9	36	0.25	4. Cellular Processes
Pentose phosphate pathway	mmu00030	7	28	0.25	1. Metabolism
Huntington's disease	mmu05016	59	239	0.246861925	6. Human Diseases
Cell cycle	mmu04110	34	140	0.242857143	4. Cellular Processes
Pyrimidine metabolism	mmu00240	25	105	0.238095238	1. Metabolism

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Terpenoid backbone biosynthesis	mmu00900	4	17	0.235294118	1. Metabolism
Steroid biosynthesis	mmu00100	4	17	0.235294118	1. Metabolism
Oocyte meiosis	mmu04114	30	128	0.234375	4. Cellular Processes
Thyroid cancer	mmu05216	7	31	0.225806452	6. Human Diseases
B cell receptor signaling pathway	mmu04662	19	85	0.223529412	5.1 Immune System
Fc gamma R-mediated phagocytosis	mmu04666	23	103	0.223300971	5.1 Immune System
RNA degradation	mmu03018	16	73	0.219178082	2. Genetic Information Processing
RIG-I-like receptor signaling pathway	mmu04622	15	70	0.214285714	5.1 Immune System
Cytosolic DNA-sensing pathway	mmu04623	12	58	0.206896552	5.1 Immune System

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Bacterial invasion of epithelial cells	mmu05100	15	76	0.197368421	6. Human Diseases
Alzheimer's disease	mmu05010	54	283	0.190812721	6. Human Diseases
Cardiac muscle contraction	mmu04260	18	95	0.189473684	5. Organismal System
Purine metabolism	mmu00230	33	176	0.1875	1. Metabolism
Other glycan degradation	mmu00511	3	16	0.1875	1. Metabolism
Amyotrophic lateral sclerosis (ALS)	mmu05014	13	70	0.185714286	6. Human Diseases
Homologous recombination	mmu03440	5	27	0.185185185	2. Genetic Information Processing
Vasopressin-regulated water reabsorption	mmu04962	8	44	0.181818182	5. Organismal System
Valine	mmu00290	2	11	0.181818182	1. Metabolism

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Small cell lung cancer	mmu05222	17	94	0.180851064	6. Human Diseases
Neurotrophin signaling pathway	mmu04722	26	144	0.180555556	5. Organismal System
mTOR signaling pathway	mmu04150	10	56	0.178571429	3. Environmental Information Processing
Colorectal cancer	mmu05210	13	74	0.175675676	6. Human Diseases
Phagosome	mmu04145	33	191	0.172774869	4. Cellular Processes
Collecting duct acid secretion	mmu04966	5	29	0.172413793	5. Organismal System
Fructose and mannose metabolism	mmu00051	6	35	0.171428571	1. Metabolism
Base excision repair	mmu03410	9	54	0.166666667	2. Genetic Information Processing
Synthesis and degradation of ketone bodies	mmu00072	2	12	0.166666667	1. Metabolism

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
N-Glycan biosynthesis	mmu00510	8	49	0.163265306	1. Metabolism
Glioma	mmu05214	12	74	0.162162162	6. Human Diseases
Pancreatic cancer	mmu05212	12	75	0.16	6. Human Diseases
Bladder cancer	mmu05219	7	44	0.159090909	6. Human Diseases
Gap junction	mmu04540	15	95	0.157894737	4. Cellular Processes
Regulation of actin cytoskeleton	mmu04810	36	229	0.15720524	4. Cellular Processes
Glycine	mmu00260	5	32	0.15625	1. Metabolism
Wnt signaling pathway	mmu04310	25	162	0.154320988	3. Environmental Information Processing
Glycosylphosphatidylinositol (GPI) -anchor biosynthesis	mmu00563	4	26	0.153846154	1. Metabolism
Adherens junction	mmu04520	12	79	0.151898734	4. Cellular Processes
Insulin signaling pathway	mmu04910	22	146	0.150684932	5. Organismal System

Pathway Name	KEGG Pathway ID	Number of Genes With High Intensity on The Chip	Total Number of Genes in Pathway (According to KEGG)	Gene On Chip / Total Genes In Pathway	Category
Proximal tubule bicarbonate reclamation	mmu04964	3	20	0.15	5. Organismal System
Prostate cancer	mmu05215	14	94	0.14893617	6. Human Diseases
Lysosome	mmu04142	19	129	0.147286822	4. Cellular Processes
Endocytosis	mmu04144	35	239	0.146443515	4. Cellular Processes

Table A.2 List of pathways detected based on high-intensity probes from the microarray data. As can be seen from the table, many of the pathways enriched in high-intensity genes are known to be implicated in the development of the immune system and processing of antigens

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
70L/AIG/SLC	Cellular Processes, Human Diseases, Organismal System	mmu04110 (Cell cycle), mmu04115 (p53 signaling pathway), mmu04142 (Lysosome), mmu04145 (Phagosome), mmu04540 (Gap junction), mmu05012 (Parkinson's disease), mmu05016 (Huntington's disease), mmu05212 (Pancreatic cancer), mmu05214 (Glioma), mmu05219 (Bladder cancer), mmu05222 (Small cell lung cancer), mmu04722 (Neurotrophin signaling pathway), mmu04910 (Insulin signaling pathway), mmu04962 (Vasopressin-regulated water reabsorption)
LPA/IFG	Cellular Processes, Human Diseases	mmu04115 (p53 signaling pathway), mmu04144 (Endocytosis), mmu04145 (Phagosome), mmu04810 (Regulation of actin cytoskeleton), mmu05100 (Bacterial invasion of epithelial cells), mmu05210 (Colorectal cancer), mmu05212 (Pancreatic cancer), mmu05222 (Small cell lung cancer)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
GRH/FML	Cellular Processes, Environmental Information Processing, Genetic Information Processing, Human Diseases, Metabolism, Organismal System	mmu04110 (Cell cycle), mmu04140 (Regulation of autophagy), mmu04810 (Regulation of actin cytoskeleton), mmu04150 (mTOR signaling pathway), mmu04310 (Wnt signaling pathway), mmu00970 (Aminoacyl-tRNA biosynthesis), mmu03010 (Ribosome), mmu03018 (RNA degradation), mmu03020 (RNA polymerase), mmu03030 (DNA replication), mmu04120 (Ubiquitin mediated proteolysis), mmu05010 (Alzheimer's disease), mmu05012 (Parkinson's disease), mmu05014 (Amyotrophic lateral sclerosis), mmu05016 (Huntington's disease), mmu05210 (Colorectal cancer), mmu05212 (Pancreatic cancer), mmu05214 (Glioma), mmu05216 (Thyroid cancer), mmu05219 (Bladder cancer), mmu00020 (TCA cycle), mmu00190 (Oxidative phosphorylation), mmu00230 (Purine metabolism), mmu00240 (Pyrimidine metabolism), mmu00510 (N-Glycan biosynthesis), mmu00563 (Glycosylphosphatidylinositol(GPI)-anchor biosynthesis), mmu00630 (Glyoxylate and dicarboxylate metabolism), mmu04910 (Insulin signaling pathway), mmu04964 (Proximal tubule bicarbonate reclamation), mmu04966 (Collecting duct acid secretion)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
PGE/NPY	Cellular Processes, Immune System, Metabolism, Organismal System	mmu04114 (Oocyte meiosis), mmu04142 (Lysosome), mmu04144 (Endocytosis), mmu04145 (Phagosome), mmu04810 (Regulation of actin cytoskeleton), mmu04623 (Cytosolic DNA-sensing pathway), mmu04666 (Fc gamma R-mediated phagocytosis), mmu00020 (TCA cycle), mmu00072 (Synthesis and degradation of ketone bodies), mmu00130 (Ubiquinone and other terpenoid-quinone biosynthesis), mmu00190 (Oxidative phosphorylation), mmu00230 (Purine metabolism), mmu00240 (Pyrimidine metabolism), mmu00740 (Riboflavin metabolism), mmu00900 (Terpenoid backbone biosynthesis), mmu04260 (Cardiac muscle contraction), mmu04722 (Neurotrophin signaling pathway), mmu04966 (Collecting duct acid secretion)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
IFB/S1P	Cellular Processes, Human Diseases, Immune System, Organismal System	mmu04110 (Cell cycle), mmu04114 (Oocyte meiosis), mmu04115 (p53 signaling pathway), mmu04810 (Regulation of actin cytoskeleton), mmu05010 (Alzheimer's disease), mmu05012 (Parkinson's disease), mmu05016 (Huntington's disease), mmu05100 (Bacterial invasion of epithelial cells), mmu05212 (Pancreatic cancer), mmu05214 (Glioma), mmu04666 (Fc gamma R-mediated phagocytosis), mmu04260 (Cardiac muscle contraction)
BOM/LB4	Human Diseases, Organismal System	mmu05210 (Colorectal cancer), mmu05214 (Glioma), mmu04260 (Cardiac muscle contraction)
NEB/NGF	Environmental Information Processing, Human Diseases, Organismal System	mmu04150 (mTOR signaling pathway), mmu05012 (Parkinson's disease), mmu05014 (Amyotrophic lateral sclerosis), mmu05210 (Colorectal cancer), mmu05214 (Glioma), mmu04722 (Neurotrophin signaling pathway)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
TNF/CGS	Cellular Processes, Genetic Information Processing, Human Diseases, Metabolism	mmu04110 (Cell cycle), mmu04115 (p53 signaling pathway), mmu04142 (Lysosome), mmu04540 (Gap junction), mmu03010 (Ribosome), mmu03030 (DNA replication), mmu03430 (Mismatch repair), mmu04130, mmu05012 (Parkinson's disease), mmu05014 (Amyotrophic lateral sclerosis), mmu05100 (Bacterial invasion of epithelial cells), mmu05214 (Glioma), mmu05222 (Small cell lung cancer), mmu00100 (Steroid biosynthesis), mmu00190 (Oxidative phosphorylation), mmu00260 (Glycine, serine and threonine metabolism), mmu00563 (Glycosylphosphatidylinositol(GPI)-anchor biosynthesis), mmu00630 (Glyoxylate and dicarboxylate metabolism)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
PAF/CPG	Environmental Information Processing, Immune System, Metabolism	mmu04310 (Wnt signaling pathway), mmu04622, mmu04623 (Cytosolic DNA-sensing pathway), mmu00230 (Purine metabolism), mmu00240 (Pyrimidine metabolism), mmu00260 (Glycine, serine and threonine metabolism), mmu00460 (Cyanoamino acid metabolism), mmu00511 (Other glycan degradation), mmu00670 (One carbon pool by folate), mmu00740 (Riboflavin metabolism)
TER/BAF	Cellular Processes, Environmental Information Processing, Genetic Information Processing, Metabolism	mmu04110 (Cell cycle), mmu04114 (Oocyte meiosis), mmu04115 (p53 signaling pathway), mmu04144 (Endocytosis), mmu04520 (Adherens junction), mmu04540 (Gap junction), mmu04810 (Regulation of actin cytoskeleton), mmu04310 (Wnt signaling pathway), mmu00970 (Aminoacyl-tRNA biosynthesis), mmu03018 (RNA degradation), mmu03040 (Spliceosome), mmu03410 (Base excision repair), mmu04120 (Ubiquitin mediated proteolysis), mmu00020 (TCA cycle), mmu00030 (Pentose phosphate pathway), mmu00230 (Purine metabolism), mmu00260 (Glycine, serine and threonine metabolism), mmu00460 (Cyanoamino acid metabolism)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
DIM/TGF	Environmental Information Processing, Genetic Information Processing, Human Diseases, Immune System, Metabolism, Organismal System	mmu04150 (mTOR signaling pathway), mmu00970 (Aminoacyl-tRNA biosynthesis), mmu03010 (Ribosome), mmu03020 (RNA polymerase), mmu03022, mmu03040 (Spliceosome), mmu03060 (Protein export), mmu03410 (Base excision repair), mmu03430 (Mismatch repair), mmu03440, mmu04130, mmu05010 (Alzheimer's disease), mmu05100 (Bacterial invasion of epithelial cells), mmu05210 (Colorectal cancer), mmu05212 (Pancreatic cancer), mmu04622, mmu04623 (Cytosolic DNA-sensing pathway), mmu04662 (B cell receptor signaling pathway), mmu00020 (TCA cycle), mmu00030 (Pentose phosphate pathway), mmu00100 (Steroid biosynthesis), mmu00190 (Oxidative phosphorylation), mmu00260 (Glycine, serine and threonine metabolism), mmu00510 (N-Glycan biosynthesis), mmu00563 (Glycosylphosphatidylinositol(GPI)-anchor biosynthesis), mmu04260 (Cardiac muscle contraction), mmu04910 (Insulin signaling pathway), mmu04962 (Vasopressin-regulated water reabsorption), mmu04966 (Collecting duct acid secretion)

Matched Ligands	Conserved KEGG Pathway Categories	Conserved KEGG Subpathway IDs
BLC/IGF	Cellular Processes, Organismal System	mmu04110 (Cell cycle), mmu04115 (p53 signaling pathway), mmu04722 (Neurotrophin signaling pathway), mmu04910 (Insulin signaling pathway)

Table A.3 Top matched ligands based on expression patterns in the consensus tree shown in figure 5 in the paper. The KEGG pathway categories correspond to the pathway categories highlighted in table 2 in the supplementary material and table 2 in the paper

BIBLIOGRAPHY

[1] T. Aittokallio and B. Schwikowski. Graph-based methods for analysing networks in cell biology. *Briefings in Bioinformatics*, 7(3):243, 2006.

[2] Alliance for Cell Signaling. The UCSD-Nature Signaling Gateway. http://www.signaling-gateway.org, August 2010.

[3] S.F. Altschul, T.L. Madden, A.A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3390, 1997.

[4] M. Ashburner, CA Ball, JA Blake, D. Botstein, H. Butler, JM Cherry, AP Davis, K. Dolinski, SS Dwight, JT Eppig, et al. Gene ontology: Tool for the unification of biology. The Gene Ontology Consortium. *Nature genetics*, 25(1):25, 2000.

[5] C. Auffray, S. Imbeaud, M. Roux-Rouquie, and L. Hood. From functional genomics to systems biology: concepts and practices. *C R Biol*, 326(10-11):879–92, Oct-Nov 2003.

[6] Ferhat Ay, Tamer Kahveci, and Valerie de Crecy-Lagard. Consistent alignment of metabolic pathways without abstraction. In *7th Annual International Conference on Computational Systems Bioinformatics*, 2008.

[7] Brett W. Bader and Tamara G. Kolda. Efficient matlab computations with sparse and factored tensors. *SIAM J. Scientific Computing*, 30(1):205–231, 2007.

[8] G. D. Bader, D. Betel, and C. W. Hogue. Bind: the biomolecular interaction network database. *Nucleic Acids Res*, 31(1):248–50, Jan 1 2003.

[9] J. S. Bader. Greedily building protein networks with confidence. *Bioinformatics*, 19(15):1869–74, Oct 12 2003.

[10] J. S. Bader, A. Chaudhuri, J. M. Rothberg, and J. Chant. Gaining confidence in high-throughput protein interaction networks. *Nat Biotechnol*, 22(1):78–85, Jan 2004.

[11] A. Bairoch, R. Apweiler, C.H. Wu, W.C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, et al. The Universal Protein Resource (UniProt). *Nucleic Acids Research*, 33:D154, 2005.

[12] M. Baitaluk, X. Qian, S. Godbole, A. Raval, A. Ray, and A. Gupta. Pathsys: integrating molecular interaction graphs for systems biology. *BMC Bioinformatics*, 7:55, 2006.

[13] S. Bandyopadhyay, R. Sharan, and T. Ideker. Systematic identification of functional orthologs based on protein network comparison. *Genome research*, 16(3):428–435, 2006.

[14] E. Banks, E. Nabieva, R. Peterson, and M. Singh. NetGrep: fast network schema searches in interactomes. *Genome Biology*, 9(9):R138, 2008.

[15] Eric Banks, Elena Nabieva, Bernard Chazelle, and Mona Singh. Organization of physical interactomes as uncovered by network schemas. *PLoS Comput Biol*, 4(10):e1000203, 10 2008.

[16] A.L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509, 1999.

[17] A.L. Barabasi and Z.N. Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

[18] D.A. Benson, I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, and D.L. Wheeler. GenBank: update. *Nucleic Acids Research*, 32(Database Issue):D23, 2004.

[19] A. Bernard and A. J. Hartemink. Informative structure priors: joint learning of dynamic regulatory networks from multiple types of data. *Pac Symp Biocomput*, 0:459–70, 2005.

[20] A. Beyer, S. Bandyopadhyay, and T. Ideker. Integrating physical and genetic maps: from genomes to interaction networks. *Nature Reviews Genetics*, 8(9):699–710, 2007.

[21] J.A. Blake, J.E. Richardson, C.J. Bult, J.A. Kadin, and J.T. Eppig. MGD: the mouse genome database. *Nucleic acids research*, 31(1):193, 2003.

[22] K.M. Borgwardt and H.P. Kriegel. Shortest-Path Kernels on Graphs. *Proceedings of the Fifth IEEE International Conference on Data Mining*, 1:74–81, 2005.

[23] K.M. Borgwardt, H.P. Kriegel, SVN Vishwanathan, and N.N. Schraudolph. Graph Kernels For Disease Outcome Prediction From Protein-Protein Interaction Networks. *Proceedings of the Pacific Symposium of Biocomputing*, 1:4, 2007.

[24] EI Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, JM Cherry, and G. Sherlock. GO:: TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics (Oxford, England)*, 20(18):3710, 2004.

[25] M.C. Brandon, M.T. Lott, K.C. Nguyen, S. Spolim, S.B. Navathe, P. Baldi, and D.C. Wallace. MITOMAP: a human mitochondrial genome database–2004 update. *Nucleic acids research*, 33(Database Issue):D611, 2005.

[26] S. Brasch, L. Linsen, and G. Fuellen. VANLO- Interactive visual exploration of aligned biological networks. *BMC bioinformatics*, 10(1):327, 2009.

[27] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, 1998.

[28] S. Bruckner, F. Huffner, R.M. Karp, R. Shamir, and R. Sharan. Topology-Free Querying of Protein Interaction Networks. In *Recomb*, 2009.

[29] F. J. Bruggeman and H. V. Westerhoff. The nature of systems biology. *Trends Microbiol*, 15(1):45–50, Jan 2007.

[30] L.W. Burrus and A.P. McMahon. Biochemical analysis of murine Wnt proteins reveals both shared and distinct properties. *Experimental cell research*, 220(2):363–373, 1995.

[31] S. L. Carter, C. M. Brechbuhler, M. Griffin, and A. T. Bond. Gene co-expression network topology provides a framework for molecular characterization of cellular state. *Bioinformatics*, 20(14):2242–50, Sep 22 2004.

[32] D.D. Chaplin. Overview of the immune response. *Journal of Allergy and Clinical Immunology*, 125(2):S3–S23, 2010.

[33] JM Cherry, C. Adler, C. Ball, SA Chervitz, SS Dwight, ET Hester, Y. Jia, G. Juvik, T. Roe, M. Schroeder, et al. SGD: Saccharomyces genome database. *Nucleic Acids Research*, 26(1):73, 1998.

[34] M.S. Cline, M. Smoot, E. Cerami, A. Kuchinsky, N. Landys, C. Workman, R. Christmas, I. Avila-Campilo, M. Creech, B. Gross, et al. Integration of biological networks and gene expression data using Cytoscape. *Nature protocols*, 2(10):2366–2382, 2007.

[35] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines*. Cambridge university press, 2000.

[36] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*. Cambridge Univ Pr, 2000.

[37] J.M. Dal Porto, S.B. Gauld, K.T. Merrell, D. Mills, A.E. Pugh-Bernard, and J. Cambier. B cell antigen receptor signaling 101. *Molecular immunology*, 41(6-7):599–613, 2004.

[38] H. de Jong. Modeling and simulation of genetic regulatory systems: a literature review. *J Comput Biol*, 9(1):67–103, 2002.

[39] A.L. DeFranco. Molecular aspects of B-lymphocyte activation. *Annual review of cell biology*, 3(1):143–178, 1987.

[40] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

[41] T. G. Dietterich. Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 1857:1–15, 2000.

[42] N. Dojer, A. Gambin, A. Mizera, B. Wilczynski, and J. Tiuryn. Applying dynamic bayesian networks to perturbed gene expression data. *BMC Bioinformatics*, 7:249, 2006.

[43] Q. Dong, S.D. Schlueter, and V. Brendel. PlantGDB, plant genome database and analysis tools. *Nucleic acids research*, 32(Database Issue):D354, 2004.

[44] J.F. Dufayard, L. Duret, S. Penel, M. Gouy, F. Rechenmann, and G. Perrière. Tree pattern matching in phylogenetic trees: automatic search for orthologs or paralogs in homologous gene sequence databases. *Bioinformatics*, 21(11):2596–2603, 2005.

[45] R. Edgar, M. Domrachev, and A.E. Lash. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, 30(1):207, 2002.

[46] B. Efron. The jackknife, the bootstrap and other resampling plans. In *CBMS-NSF regional conference series in applied mathematics*, volume 38. Siam, 1982.

[47] J.A. Eisen and M. Wu. Phylogenetic analysis and gene functional predictions: phylogenomics in action. *Theoretical population biology*, 61(4):481–488, 2002.

[48] B. Elliott, M. Kirac, A. Cakmak, G. Yavas, S. Mayes, E. Cheng, Y. Wang, C. Gupta, G. Ozsoyoglu, and Z. Meral Ozsoyoglu. PathCase: pathways database system. *Bioinformatics*, 24(21):2526, 2008.

[49] AJ Enright, S. Van Dongen, and CA Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic acids research*, 30(7):1575, 2002.

[50] Christos Faloutsos, Tamara G. Kolda, and Jimeng Sun. Mining large graphs and streams using matrix and tensor tools. In *SIGMOD '07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 1174–1174, New York, NY, USA, 2007. ACM.

[51] I. Farkas, H. Jeong, T. Vicsek, A.L. Barabasi, and ZN Oltvai. The topology of the transcription regulatory network in the yeast, Saccharomyces cerevisiae. *Physica A: Statistical Mechanics and its Applications*, 318(3-4):601–612, 2003.

[52] J. Felsenstein. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, 39(4):783–791, 1985.

[53] J. Felsenstein. PHYLIP (phylogeny inference package) version 3.6. *Distributed by the author. Department of Genome Sciences, University of Washington, Seattle*, 2005.

[54] AL Ferraz, A Ojeda, M Lpez-Bjar, LT Fernandes, A Castell, JM Folch, and M Prez-Enciso. Transcriptome architecture across tissues in the pig. *BMC Genomics*, 9:173, Apr 16 2008.

[55] M. Fiore, GN Chaldakov, and L. Aloe. Nerve growth factor as a signaling molecule for nerve cells and also for the neuroendocrine-immune systems. *Reviews in the neurosciences*, 20(2):133, 2009.

[56] J. Flannick, A. Novak, C.B. Do, B.S. Srinivasan, and S. Batzoglou. Automatic parameter learning for multiple network alignment. *Lecture Notes in Computer Science*, 4955:214–231, 2008.

[57] J. Flannick, A. Novak, B.S. Srinivasan, H.H. McAdams, and S. Batzoglou. Graemlin: General and robust alignment of multiple large interaction networks. *Genome Research*, 16(9):1169, 2006.

[58] P. Flicek, BL Aken, K. Beal, B. Ballester, M. Caccamo, Y. Chen, L. Clarke, G. Coates, F. Cunningham, T. Cutts, et al. Ensembl 2008. *Nucleic acids research*, 36(Database issue):D707, 2008.

[59] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using bayesian networks to analyze expression data. *J Comput Biol*, 7(3-4):601–20, 2000.

[60] Z. Fu, X. Chen, V. Vacic, P. Nan, Y. Zhong, and T. Jiang. MSOAR: A high-throughput ortholog assignment system based on genome rearrangement. *Journal of Computational Biology*, 14(9):1160–1175, 2007.

[61] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. WH Freeman & Co. New York, NY, USA, 1979.

[62] H. Ge, A.J.M. Walhout, and M. Vidal. Integrating 'omic' information: a bridge between genomics and systems biology. *Trends in Genetics*, 19(10):551–560, 2003.

[63] R. Gentleman, V. Carey, D. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, et al. Bioconductor: open software development for computational biology and bioinformatics. *Genome biology*, 5(10):R80, 2004.

[64] E. Glaab, J.M. Garibaldi, and N. Krasnogor. ArrayMining: a modular web-application for microarray analysis combining ensemble and consensus methods with cross-study normalization. *BMC bioinformatics*, 10(1):358, 2009.

[65] L. Goodstadt and C.P. Ponting. Phylogenetic reconstruction of orthology, paralogy, and conserved synteny for dog and human. *PLoS Comput Biol*, 2(9):e133, 2006.

[66] M.A. Hall and L.A. Smith. Feature selection for machine learning: comparing a correlation-based filter approach to the wrapper. In *Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference*, pages 235–239, 1999.

[67] J. D. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. Walhout, M. E. Cusick, F. P. Roth, and M. Vidal. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430(6995):88–93, Jul 1 2004.

[68] F. Harary. *Graph theory*. 1969.

[69] D.R. Hardoon and J. Shawe-Taylor. Decomposing the tensor kernel support vector machine for neuroscience data with structured labels. *Machine learning*, 79(1):29–46, 2010.

[70] L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–52, Dec 2 1999.

[71] N.E. Harwood and F.D. Batista. Early events in B cell activation. *Annual Review of Immunology*, 28:185–210, 2009.

[72] X. He, D. Cai, H. Liu, and J. Han. Image clustering with tensor representation. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 132–140. ACM, 2005.

[73] A.P. Heath and L.E. Kavraki. Computational challenges in systems biology. *Computer Science Review*, 3(1):1–17, 2009.

[74] AE Hirsh and HB Fraser. Protein dispensability and rate of evolution. *Nature*, 411(6841):1046–9, 2001.

[75] M.T. Holder, J. Sukumaran, and P.O. Lewis. A justification for reporting the majority-rule consensus tree in Bayesian phylogenetics. *Systematic biology*, 57(5):814, 2008.

[76] R.C. Hsueh and R.H. Scheuermann. Tyrosine kinase activation in the decision between growth, differentiation, and death responses initiated from the B cell antigen receptor. *Advances in Immunology*, 75:283–316, 2000.

[77] T. Ideker and R. Sharan. Protein networks in disease. *Genome Research*, 18(4):644, 2008.

[78] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc Natl Acad Sci U S A*, 98(8):4569–74, Apr 10 2001.

[79] H. Jeong, S. P. Mason, A. L. Barabasi, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–2, May 3 2001.

[80] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A. L. Barabasi. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–4, Oct 5 2000.

[81] M. Kalaev, V. Bafna, and R. Sharan. Fast and accurate alignment of multiple protein networks. *Lecture Notes in Computer Science*, 4955:246, 2008.

[82] M. Kalaev, M. Smoot, T. Ideker, and R. Sharan. NetworkBLAST: comparative analysis of protein networks. *Bioinformatics*, 24(4):594, 2008.

[83] M. Kanehisa, M. Araki, S. Goto, M. Hattori, M. Hirakawa, M. Itoh, T. Katayama, S. Kawashima, S. Okuda, T. Tokimatsu, et al. KEGG for linking genomes to life and the environment. *Nucleic Acids Research*, 36(Database issue):D480, 2008.

[84] M. Kanehisa, M. Araki, S. Goto, M. Hattori, M. Hirakawa, M. Itoh, T. Katayama, S. Kawashima, S. Okuda, T. Tokimatsu, and Y. Yamanishi. Kegg for linking genomes to life and the environment. *Nucleic Acids Res*, 36(Database issue):D480–4, Jan 2008.

[85] G. Karlebach and R. Shamir. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9:770–780, 2008.

[86] N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, 20(11):1746–58, Jul 22 2004.

[87] N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Topological generalizations of network motifs. *Phys Rev E Stat Nonlin Soft Matter Phys*, 70(3 Pt 1):031909, Sep 2004.

[88] B. P. Kelley, R. Sharan, R. Karp, E. T. Sittler, D. E. Root, B. R. Stockwell, and T. Ideker. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proc Natl Acad Sci*, 100:11394–9, 2003.

[89] B.P. Kelley, B. Yuan, F. Lewitter, R. Sharan, B.R. Stockwell, and T. Ideker. PathBLAST: a tool for alignment of protein interaction networks. *Nucleic Acids Research*, 32:W83, 2004.

[90] R. Khanin and E. Wit. How scale-free are biological networks. *J Comput Biol*, 13(3):810–8, Apr 2006.

[91] P. Kharchenko, G.M. Church, and D. Vitkup. Expression dynamics of a cellular metabolic network. *Molecular Systems Biology*, 1, 2005.

[92] M. Kirac and G. Ozsoyoglu. Protein Function Prediction Based on Patterns in Biological Networks. *Lecture Notes In Computer Science*, 4955:197, 2008.

[93] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[94] K. Klemm and S. Bornholdt. Topology of biological networks and reliability of information processing. *Proceedings of the National Academy of Sciences*, 102(51):18414–18419, 2005.

[95] Tamara G. Kolda and Jimeng Sun. Scalable tensor decompositions for multi-aspect data mining. In *ICDM*, pages 363–372. IEEE Computer Society, 2008.

[96] E. Koonin. Orthologs, paralogs and evolutionary genomics. *Annu. Rev. Genet*, 39:309–38, 2005.

[97] M. Koyuturk, Y. Kim, U. Topkara, S. Subramaniam, W. Szpankowski, and A. Grama. Pairwise alignment of protein interaction networks. *Journal of Computational Biology*, 13(2):182–199, 2006.

[98] O. Krishnadev, K. V. Brinda, and S. Vishveshwara. A graph spectral analysis of the structural similarity network of protein chains. *Proteins*, 61(1):152–63, Oct 1 2005.

[99] M. Krull, S. Pistor, N. Voss, A. Kel, I. Reuter, D. Kronenberg, H. Michael, K. Schwarzer, A. Potapov, C. Choi, O. Kel-Margoulis, and E. Wingender. Transpath: an information resource for storing and visualizing signaling pathways and their pathological aberrations. *Nucleic Acids Res*, 34(Database issue):D546–51, Jan 1 2006.

[100] M. Krzywinski, J. Schein, İ. Birol, J. Connors, R. Gascoyne, D. Horsman, S.J. Jones, and M.A. Marra. Circos: an information aesthetic for comparative genomics. *Genome research*, 19(9):1639, 2009.

[101] O. Kuchaiev, T. Milenković, V. Memišević, W. Hayes, and N. Pržulj. Topological network alignment uncovers biological function and phylogeny. *Journal of The Royal Society Interface*, 7(50):1341, 2010.

[102] O. Kuchaiev, A. Stevanovic, W. Hayes, and N. Przulj. GraphCruch 2: Software tool for network modeling, alignment and clustering. *BMC bioinformatics*, 12(1):24, 2011.

[103] S. Kullback and R.A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, pages 79–86, 1951.

[104] J. Lee, R. Sinkovits, D. Mock, E. Rab, J. Cai, P. Yang, B. Saunders, R. Hsueh, S. Choi, S. Subramaniam, et al. Components of the antigen processing and presentation pathway revealed by gene expression microarray analysis following B cell antigen receptor (BCR) stimulation. *BMC bioinformatics*, 7(1):237, 2006.

[105] J.J. Letterio and A.B. Roberts. REGULATION OF IMMUNE RESPONSES BY TGF-β*. *Annual review of immunology*, 16(1):137–161, 1998.

[106] L. Li, C.J. Stoeckert, and D.S. Roos. OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome research*, 13(9):2178–2189, 2003.

[107] C.S. Liao, K. Lu, M. Baym, R. Singh, and B. Berger. IsoRankN: spectral methods for global alignment of multiple protein networks. *Bioinformatics*, 25(12):i253, 2009.

[108] J. Lim, T. Hao, C. Shaw, A.J. Patel, G. Szabó, J.F. Rual, C.J. Fisk, N. Li, A. Smolyar, D.E. Hill, et al. A Protein–Protein Interaction Network for Human Inherited Ataxias and Disorders of Purkinje Cell Degeneration. *Cell*, 125(4):801–814, 2006.

[109] Rune Linding, Lars J. Jensen, Adrian Pasculescu, Marina Olhovsky, Karen Colwill, Peer Bork, Michael B. Yaffe, and Tony Pawson. Networkin: a resource for exploring cellular phosphorylation networks. *Nucl. Acids Res.*, 36:D695–699, 2008.

[110] D. Maglott, J. Ostell, K.D. Pruitt, and T. Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research*, 35(Database issue):D26, 2007.

[111] U. Manber. *Introduction to algorithms: a creative approach*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989.

[112] X. Mao, T. Cai, J.G. Olyarchuk, and L. Wei. Automated genome annotation and pathway identification using the KEGG Orthology (KO) as a controlled vocabulary. *Bioinformatics*, 21(19):3787–3793, 2005.

[113] G. Marsaglia, W.W. Tsang, and J. Wang. Evaluating Kolmogorov's distribution. *Journal of Statistical Software*, 8(18):1–4, 2003.

[114] L. R. Matthews, P. Vaglio, J. Reboul, H. Ge, B. P. Davis, J. Garrels, S. Vincent, and M. Vidal. Identification of potential interaction networks using sequence-based searches for conserved protein-protein interactions or "interologs". *Genome Res*, 11(12):2120–6, Dec 2001.

[115] T. Milenkoviæ and N. Pržulj. Uncovering Biological Network Function via Graphlet Degree Signatures. *Cancer Informatics*, 6:257, 2008.

[116] T. Milenković, J. Lai, and N. Pržulj. GraphCrunch: a tool for large network analyses. *BMC bioinformatics*, 9(1):70, 2008.

[117] T. Mitchell. *Machine Learning.* McGraw-Hill, Boston, MA, 1997.

[118] J. Murn, I. Mlinaric-Rascan, P. Vaigot, O. Alibert, V. Frouin, and X. Gidrol. A Myc-regulated transcriptional network controls B-cell fate in response to BCR triggering. *BMC genomics*, 10(1):323, 2009.

[119] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14: Proceedings of the 2002 [sic] Conference*, page 849. MIT Press, 2002.

[120] K.P. O'Brien, M. Remm, and E.L.L. Sonnhammer. Inparanoid: a comprehensive database of eukaryotic orthologs. *Nucleic acids research*, 33(Database Issue):D476, 2005.

[121] H. Ogata, S. Audic, V. Barbe, F. Artiguenave, P. E. Fournier, D. Raoult, and J. M. Claverie. Selfish dna in protein-coding genes of rickettsia. *Science*, 290(5490):347–50, Oct 13 2000.

[122] H. Ogata, W. Fujibuchi, S. Goto, and M. Kanehisa. A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic acids research*, 28(20):4021, 2000.

[123] J. O'Madadhain, D. Fisher, S. White, and Y. Boey. The JUNG (Java Universal Network/Graph) Framework. *University of California, Irvine, California*, 2003.

[124] R.D.M. Page. Tree View: An application to display phylogenetic trees on personal computers. *Computer applications in the biosciences: CABIOS*, 12(4):357, 1996.

[125] H. Parkinson, M. Kapushesky, N. Kolesnikov, G. Rustici, M. Shojatalab, N. Abeygunawardena, H. Berube, M. Dylag, I. Emam, A. Farne, et al. ArrayExpress update–from an archive of functional genomics experiments to the atlas of gene expression. *Nucleic acids research*, 37(Database issue):D868, 2009.

[126] H. Parkinson, M. Kapushesky, M. Shojatalab, N. Abeygunawardena, R. Coulson, A. Farne, E. Holloway, N. Kolesnykov, P. Lilja, and M. Lukk. ArrayExpress–a public database of microarray experiments and gene expression profiles. *Nucleic Acids Research*, 35(Database issue):D747, 2007.

[127] R.Y. Pinter, O. Rokhlenko, E. Yeger-Lotem, and M. Ziv-Ukelson. Alignment of metabolic pathways. *Bioinformatics*, 21(16):3401–3408, 2005.

[128] S. Pradervand, M.R. Maurya, and S. Subramaniam. Identification of signaling components required for the prediction of cytokine release in RAW 264.7 macrophages. *Genome biology*, 7(2):R11, 2006.

[129] N. Pržulj. From Topology to Phenotype in Protein–Protein Interaction Networks. *Network Science*, pages 31–49, 2010.

[130] J. Quackenbush. Microarray data normalization and transformation. *nature genetics*, 32:496–501, 2002.

[131] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. ISBN 3-900051-07-0.

[132] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. L. Barabasi. Hierarchical organization of modularity in metabolic networks. *Science*, 297(5586):1551–5, Aug 30 2002.

[133] M. Remm, C.E.V. Storm, and E.L.L. Sonnhammer. Automatic clustering of orthologs and in-paralogs from pairwise species comparisons. *Journal of molecular biology*, 314(5):1041–1052, 2001.

[134] J. Ross, I. Schreiber, and M.O. Vlad. *Determination of Complex Reaction Mechanisms: Analysis of Chemical, Biological, and Genetic Networks.* Oxford University Press, USA, 2006.

[135] T. Saitoh and S. Akira. Regulation of innate immune responses by autophagy-related proteins. *The Journal of cell biology*, 189(6):925, 2010.

[136] L. Salwinski, C.S. Miller, A.J. Smith, F.K. Pettit, J.U. Bowie, and D. Eisenberg. The database of interacting proteins: 2004 update. *Nucleic Acids Research*, 32(Database Issue):D449, 2004.

[137] J. Scott, T. Ideker, R.M. Karp, and R. Sharan. Efficient Algorithms for Detecting Signaling Pathways in Protein Interaction Networks. *Journal of Computational Biology*, 13(2):133–144, 2006.

[138] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*, 13(11):2498–504, Nov 2003.

[139] R. Sharan and T. Ideker. Modeling cellular machinery through biological network comparison. *Nature Biotechnology*, 24:427–433, 2006.

[140] T. Shlomi, D. Segal, E. Ruppin, and R. Sharan. QPath: a method for querying pathways in a protein-protein interaction network. *BMC Bioinformatics*, 7:199, 2006.

[141] M.J. Slakter. A comparison of the Pearson chi-square and Kolmogorov goodness-of-fit tests with respect to validity. *Journal of the American Statistical Association*, 60(311):854–858, 1965.

[142] B.S. Srinivasan, N.H. Shah, J.A. Flannick, E. Abeliuk, A.F. Novak, and S. Batzoglou. Current progress in network research: toward reference networks for key model organisms. *Briefings in Bioinformatics*, 8(5):318, 2007.

[143] C. Steinbeck, C. Hoppe, S. Kuhn, M. Floris, R. Guha, and E.L. Willighagen. Recent Developments of the Chemistry Development Kit (CDK)-An Open-Source Java Library for Chemo-and Bioinformatics. *Current Pharmaceutical Design*, 12(17):2111–2120, 2006.

[144] U. Stelzl, U. Worm, M. Lalowski, C. Haenig, F. H. Brembeck, H. Goehler, M. Stroedicke, M. Zenkner, A. Schoenherr, S. Koeppen, J. Timm, S. Mintzlaff, C. Abraham, N. Bock, S. Kietzmann, A. Goedde, E. Toksoz, A. Droege, S. Krobitsch, B. Korn, W. Birchmeier, H. Lehrach, and E. E. Wanker. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–68, Sep 23 2005.

[145] J.M. Stuart, E. Segal, D. Koller, and S.K. Kim. A Gene-Coexpression Network for Global Discovery of Conserved Genetic Modules. *Science*, 302(5643):249–255, 2003.

[146] A.I. Su, M.P. Cooke, K.A. Ching, Y. Hakak, J.R. Walker, T. Wiltshire, A.P. Orth, R.G. Vega, L.M. Sapinoso, A. Moqrich, et al. Large-scale analysis of the human and mouse transcriptomes. *Proceedings of the National Academy of Sciences*, 99(7):4465, 2002.

[147] Chris Soon Heng S. Tan, Bernd Bodenmiller, Adrian Pasculescu, Marko Jovanovic, Michael O. Hengartner, Claus Jørgensen, Gary D. Bader, Ruedi Aebersold, Tony Pawson, and Rune Linding. Comparative analysis reveals conserved protein phosphorylation networks implicated in multiple diseases. *Science signaling*, 2(81):ra39+, July 2009.

[148] R.L. Tatusov, M.Y. Galperin, D.A. Natale, and E.V. Koonin. The COG database: a tool for genome-scale analysis of protein functions and evolution. *Nucleic Acids Research*, 28(1):33, 2000.

[149] Wenhong Tian and Nagiza F. Samatova. Pairwise alignment of interaction networks by fast identification of maximal conserved patterns. *Proc. of the Pacific Symposium on Biocomputing*, 2009.

[150] A.K. Topaloglu, F. Reimann, M. Guclu, A.S. Yalin, L.D. Kotan, K.M. Porter, A. Serin, N.O. Mungan, J.R. Cook, M.N. Ozbek, et al. TAC3 and TACR3 mutations in familial hypogonadotropic hypogonadism reveal a key role for Neurokinin B in the central control of reproduction. *Nature genetics*, 41(3):354–358, 2008.

[151] F. Towfic, M. Heather West Greenlee, and V. Honavar. Detecting orthologous genes based on protein-protein interaction networks. *IEEE International Conference on Bioinformatics and Biomedicine proceedings*, 2009.

[152] F. Towfic, M.H.W. Greenlee, and V. Honavar. Aligning Biomolecular Networks Using Modular Graph Kernels. *Lecture Notes in Computer Science*, 5724:245–361, 2009.

[153] Fadi Towfic, M. Heather-West Greenlee, and Vasant Honavar. Aligning biomolecular networks using modular graph kernels. In *Algorithms in Bioinformatics*, volume 5724 of *Lecture Notes in Computer Science*, pages 345–361. Springer, 2009.

[154] Fadi Towfic, Susan VanderPlas, Casey A. Oliver, Oliver Couture, Christopher K. Tuggle, M. Heather West Greenlee, and Vasant Honavar. Detection of gene orthology from gene co-expression and protein interaction networks. *BMC Bioinformatics*, 11(S-3):7, 2010.

[155] V.G. Tusher, R. Tibshirani, and G. Chu. Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences*, 98(9):5116, 2001.

[156] P. Uetz, L. Giot, G. Cagney, T. A. Mansfield, R. S. Judson, J. R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P. Pochart, A. Qureshi-Emili, Y. Li, B. Godwin, D. Conover, T. Kalbfleisch, G. Vijayadamodar, M. Yang, M. Johnston, S. Fields, and J. M. Rothberg. A comprehensive analysis of protein-protein interactions in saccharomyces cerevisiae. *Nature*, 403(6770):623–7, Feb 10 2000.

[157] SVN Vishwanathan, K.M. Borgwardt, and N.N. Schraudolph. Fast Computation of Graph Kernels. *Technical report, NICTA*, 2006.

[158] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 2007.

[159] F.C. Vuaden, L.E.B. Savio, C.M.A. Bastos, M.R. Bogo, and C.D. Bonan. Adenosine A2A receptor agonist (CGS-21680) prevents endotoxin-induced effects on nucleotidase activities in mouse lymphocytes. *European Journal of Pharmacology*, 2010.

[160] A. J. Walhout. Unraveling transcription regulatory networks by protein-dna and protein-protein interaction mapping. *Genome Res*, 16(12):1445–54, Dec 2006.

[161] I. Wapinski, A. Pfeffer, N. Friedman, and A. Regev. Automatic genome-wide reconstruction of phylogenetic gene trees. *Bioinformatics*, 23(13):i549, 2007.

[162] D.R. White and S.P. Borgatti. Betweenness centrality measures for directed graphs. *Social Networks*, 16(4):335–346, 1994.

[163] S. White and P. Smyth. Algorithms for estimating relative importance in networks. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 266–275. ACM New York, NY, USA, 2003.

[164] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, second edition, 2005.

[165] S.L. Wong, L.V. Zhang, A.H.Y. Tong, Z. Li, D.S. Goldberg, O.D. King, G. Lesage, M. Vidal, B. Andrews, H. Bussey, et al. Combining biological networks to predict genetic interactions. *Proceedings of the National Academy of Sciences*, 101(44):15682–15687, 2004.

[166] P. Ye, B. Mariniello, F. Mantero, H. Shibata, and W.E. Rainey. G-protein-coupled receptors in aldosterone-producing adenomas: a potential cause of hyperaldosteronism. *Journal of Endocrinology*, 195(1):39, 2007.

[167] E. Yeger-Lotem, S. Sattath, N. Kashtan, S. Itzkovitz, R. Milo, R. Y. Pinter, U. Alon, and H. Margalit. Network motifs in integrated cellular networks of transcription-regulation and protein-protein interaction. *Proc Natl Acad Sci U S A*, 101(16):5934–9, Apr 20 2004.

[168] S. H. Yook, Z. N. Oltvai, and A. L. Barabasi. Functional and topological characterization of protein interaction networks. *Proteomics*, 4(4):928–42, Apr 2004.

[169] H. Yu, N.M. Luscombe, H.X. Lu, X. Zhu, Y. Xia, J.D.J. Han, N. Bertin, S. Chung, M. Vidal, and M. Gerstein. Annotation Transfer Between Genomes: Protein-Protein Interologs and Protein-DNA Regulogs. *Genome Research*, 14(6):1107–1118, 2004.

[170] H. Yu, X. Zhu, D. Greenbaum, J. Karro, and M. Gerstein. Topnet: a tool for comparing biological sub-networks, correlating protein properties with topological statistics. *Nucleic Acids Res*, 32(1):328–37, 2004.

[171] M. Zaslavskiy, F. Bach, and J.P. Vert. Global alignment of protein-protein interaction networks by graph matching methods. *Bioinformatics*, 25(12):i259, 2009.

[172] X. Zhou, M.C.J. Kao, and W.H. Wong. Transitive functional annotation by shortest-path analysis of gene expression data. *Proceedings of the National Academy of Sciences*, 99(20):12783–12788, 2002.

[173] X. Zhu, R. Hart, M.S. Chang, J.W. Kim, S.Y. Lee, Y.A. Cao, D. Mock, E. Ke, B. Saunders, A. Alexander, et al. Analysis of the major patterns of B cell gene expression changes in response to short-term stimulation with 33 single ligands. *The Journal of Immunology*, 173(12):7141, 2004.