CLUSTERING PROTEIN-PROTEIN INTERACTIONS BASED ON CONSERVED DOMAIN SIMILARITIES

A THESIS

SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Aslı Ayaz

August, 2004

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

_____

Assist. Prof. Dr. Uğur Doğrusöz (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

_____

Prof. Dr. Özgür Ulusoy

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

_____

Assist. Prof. Dr. Uygar Tazebay

Approved for the Institute of Engineering and Science:

_____

Prof. Dr. Mehmet B. Baray
Director of the Institute Engineering and Science

ABSTRACT

CLUSTERING PROTEIN-PROTEIN INTERACTIONS BASED ON CONSERVED DOMAIN SIMILARITIES

Aslı Ayaz

M.S. in Computer Engineering

Supervisor: Assist. Prof. Dr. Uğur Doğrusöz

August, 2004

Protein interactions govern most cellular processes, including signal transduction, transcriptional regulation and metabolism. *Saccharomyces ceravisae* is estimated to have 16,000 protein interactions. Appereantly only a small number of these interactions were formed ab initio (invention), rest of them were formed through gene duplications and exon shuffling (birth). Domains form functional units of a protein and are responsible for most of the interaction births, since they can be recombined and rearranged much more easily compared to innovation. Therefore groups of functionally similar, homologous interactions that evolved through births are expected to have a certain domain signature. Several high throughput techniques can detect interacting protein pairs, resulting in a rapidly growing corpus of protein interactions. Although there are several efforts for computationally integrating this data with literature and other high throughput data such as gene expression, annotation of this corpus is inadaquate for deriving interaction mechanism and outcome. Finding interaction homologies would allow us to annotate an unannotated interaction based on already annotated known interactions, or predict new ones. In this study we propose a probabilistic model for assigning interactions to homologous groups, according to their conserved domain similarities. Based on this model we have developed and implemented an Expectation-Maximization algorithm for finding the most likely grouping of an interaction set. We tested our algorithm with synthetic and real data, and showed that our initial results are very promising. Finally we propose several directions to improve this work.

ÖZET
PROTEİN-PROTEİN ETKİLEŞİMLERİNİN KORUNMUŞ BÖLGE BENZERLİĞİNE GÖRE ÖBEKLENMESİ

Aslı Ayaz

Bilgisayar Mühendisligi, Yüksek Lisans

Tez Yöneticisi: Yard. Doç. Dr. Uğur Doğrusöz

Ağustos, 2004

Protein protein etkileşimleri (PPE) sinyal iletimi, transkripsiyonel düzenleme ve metabolizma gibi pek çok hücresel işlemi yürütürler. *Saccharomyces ceravisae*'de 16,000 civarında PPE olduğu tahmin edilmektedir. Bu etkileşimlerin pek azının rastgele mutasyonlarla evrimleştiği (keşif), diğerlerinin ise gen çiftlenmesi ve exon değişimi gibi olaylarla oluştuğu (doğum) kabul edilmektedir. Bölgeler, proteinlerin işlevsel birimleridir, ve yeniden derlenip düzenlenebildikleri için doğumların çoğundan sorumludurlar. Dolayısıyla doğumlar yoluyla evrimleşmiş eşköklü etkileşimlerin ortak bölgelerden oluşan bir imzası olması beklenir. PPE'leri tespit eden bir kaç yüksek üretimli yöntem sayesinde, hızla artan miktarlarda PPE verisi elde edilmektedir. Bu verileri literatür ve gen ifadesi gibi diğer yüksek verimli verilerle tümleştirmek için çabalar bulunsa da, hali hazırdaki verilerle ilişkilendirilmiş bilgiler etkileşimin mekanizmasını ve işlevini anlamak için yetersizdir. Etkileşim eşköklülüğünün tespiti, yeni ya da bilinmeyen etkileşimlerin, eşköklü bilinen etkileşimler yoluyla tanımlanmasını sağlayabilir. Bu çalışmada bölge benzerliğini esas alarak etkileşimleri eşköklü gruplara atamak için olasılıksal bir model tanımlıyoruz. Bu modeli temel alarak, en olası öbeklemeyi bulmak için bir Beklenti-Maksimizasyon algoritması geliştirdik. Algoritmayı sentetik ve gerçek veriler üzerinde sınadık ve ilk sonuçların oldukça umut verici olduğunu gösterdik. Son olarak bu çalışmanın geliştirilebilmesi için bir kaç öneri sunduk.

Acknowledgement

I would like to express my gratitude to my supervisor Assist. Prof. Uğur Doğrusöz, for his guidance, valuable comments and incredible effort in the writing of this thesis. I learned many things from him not only during the thesis process but also in the development of our team project, PATIKA. It was a great pleasure and chance for me to work with him.

I would like to thank Prof. Özgür Ulusoy and Assist. Prof. Uygar Tazebay for reviewing the manuscript of this thesis.

I wish to thank Özgün Babur for his patience during our weekly meetings and his critical comments. I also wish to thank Ozan Gerdaneri, Zeynep Erson, Erhan Giral, and all other members of the PATIKA team. It was a great experience to be a member of such a friendly team. They have been another family for me.

It is not possible to thank Emek Demir enough. Without his support and contributions this thesis would not be possible. He was very patient during our endless discussions on the model and came up with bright ideas. He has always encouraged and guided me when I felt totally lost and thought it was impossible. This world is more beautiful with him.

Above all, I am very grateful for the love and support of my parents Nevin and Servet, and my sister Ödül. In every step of my life there is their love and trust.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

Life is a very complex mechanism and requires a strict regulation and synchronization of inter and intra-cellular processes. Every second, a cell must respond to a plethora of inputs, changing its internal state, activating proper processes and give back an appropriate output when necessary. All these processes and regulations are mediated by the interactions of molecules [1].

Proteins, with their sheer numbers and functional variety, are responsible for most of these tasks. From metabolic enzymes to receptors to transcription factors, they facilitate chemical reactions, create complex decision making mechanisms and transduce signals by regulating each other. Mutations that result in aberrant, malfunctioning proteins are responsible for most of the hereditary diseases and cancer. One of the most important goals of the proteomics era is to reveal and understand the complex network of protein-protein interactions (PPIs). Although the problem has multiple aspects, it can be roughly expressed in terms of four important questions: “What interacts with what?”, “under which conditions?”, “what is the mechanism of interaction?” and “what is the outcome?”

There are several challenges for this task. Perhaps the most important challenge, which is only recently being acknowledged by the scientific community, is the need for systems level analysis and modeling. It has been shown that like many other social and biological networks, PPI networks are small-world graphs, which implies that although it is possible to identify modules that act together, modules themselves are also highly coupled with each other. Therefore analysis of the functions of isolated proteins or even modules is limited to partial if not erroneous results. Another important challenge comes from the transient nature of protein-protein interactions. Two proteins might be interacting briefly, under a very specific condition making the interaction hard to detect,

and even harder to observe its mechanism and outcome. This has not been the case with genetic or protein structure studies, which concern most of the time relatively static and stable mechanisms.

Finding interacting pair of proteins is the most straightforward subproblem and is already being addressed by various experimental studies, including cross-linking, fluorescence energy transfer analysis, protease digestion and western blotting. More importantly, new high throughput techniques such as yeast two-hybrid and phage display allowed researchers to produce organism-wide interaction maps in the last several years. However one should always bear in mind that these techniques also produce a substantial amount of false positives and negatives compared with conventional techniques. There are ongoing efforts to increase reliability of these methods, and to complement them with other techniques to identify errors.

Despite of recent advances in fluorescence resonance energy transfer, X-ray crystallography, mass spectrometry and other state-of-the-art techniques, there is still no high throughput technique for identifying conditions and mechanisms of PPIs. This makes computational methods even more important. Integrating data on existing knowledge bases, computational biologists infer, model and test *in silico* systems, hopefully developing useful explanations on how complex biological systems operate.

Research shows that for eukaryotes, only a very small fraction of PPIs evolved *ab initio* [2]. Rest of them evolved through gene duplications and exon shuffling, reusing existing functional domains in the proteins over and over. The modularity of the domains allows rapid evolution of molecular networks. An example is SH2 domain [3], which is missing in yeast, but is present and widespread in every multicellular organism. Therefore SH2 must have been originated in one of the first multicellular organisms, possibly along with protein tyrosine kinases which are also missing in yeast. However SH3 domain is present in 28 yeast proteins, which also have human homologs. 11 of these homologs also contain SH2 domains, indicating that these proteins *acquired* the needed function instead of inventing them [4].

This idea points us to a notion of interaction homology and families, similar to gene families. Protein pairs interacting over homologous functional domains should be similar in function and mechanism. Finding these families would be of immense benefit, since it

would make computational annotation of a whole family possible once a member of the family is annotated. Similarly finding such families across different organisms would be very valuable for comparative proteomics.

In this study we aim to find such homologous interaction groupings. Using the domain information of the interacting pairs, we try to determine a set of clusters of interactions such that all interactions in a certain cluster has a common subset of domains, and overlaps between these clusters are minimal. We have developed a probabilistic model and used an Expectation-Maximization algorithm to find most likely clustering. Once such clusters are obtained they can be used for annotating protein interaction data.

Chapter 2 Theory and Background

2.1 Basics

Every cell in an organism is assumed to have a complete copy of the genome, specific to that organism. Large chains of DNA molecules form the genome, which can be modeled as strings of a four letter alphabet with A, T, C and G corresponding to different types of nucleotides. Each chain forms a helix with its complementary chain.

DNA encodes most of the functionality and the structure that forms the organism. This code is first converted to RNA molecules and then to protein molecules which are the usual executers of the code. Transmission of the code from DNA to RNA molecules is called transcription and interpretation of this code from RNA to protein is called translation. As a whole, this information flow is called the "Central Dogma".

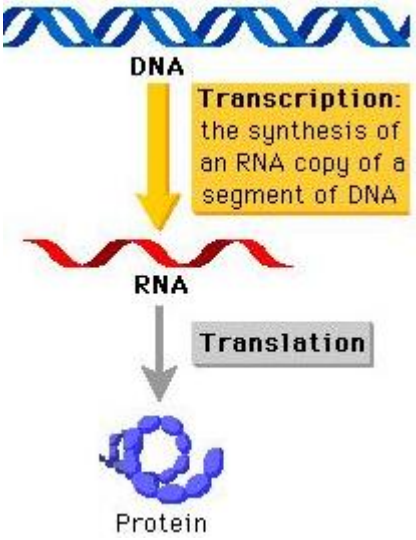

Figure 1: Central Dogma. RNA is synthesized from DNA by a process called transcription and protein is synthesized from RNA segment by a process called translation.

A protein is a macromolecule composed of one or more amino acid chains of specific order. Sequence information is transmitted from the DNA to the RNA and then to the proteins. Proteins are synthesized according to the nucleotide sequences of the gene coding that specific protein. Every triplet of nucleotides encodes one of the 20 amino acids and sequence of these nucleotide triplets determines amino acid order of a protein, which in turn determines the final protein structure.

Proteins have several functions in the cell and they are categorized according to their functions:

- Enzymes catalyze specific reactions via interaction to the reactants and increase the reaction rate. An example is phosphorylation of pyruvate by pyruvate kinase.

- Regulatory proteins regulate the metabolism of the cells. Hormones and transcription factors are some examples of this group of proteins. Hormones such as insulin regulates glucose intake to the cell, whereas transcription factor p53 activates transcription of p21 protein in the presence of DNA damage.

- Transport proteins, either binds to small molecules such as O_2 or fatty acids and carries them in the blood or they may help transport of molecules through the membranes. (e.g. glucose transporter).

- Storage proteins act as reservoirs of nutrients. Some examples are casein in milk as amino acid storage and ferritin as iron storage.

- Contractile and motile proteins, function in the movement of cells, or movement in a cell. For example actin and myosin allows muscle contractions whereas tubulin functions in the mitotic spindle formation.

- Structural proteins help the formation of the structure of the cell or the tissues. Cytoskeletal proteins form the cellular structure.

- Protective proteins such as immunoglobulins or antibodies function in the immune response.

Proteins might have multiple functions. Their globular structure, composition of multiple globular domains, allows them to perform different functions.

2.2 Protein Interactions

All the above functions are carried out by interactions of proteins with other molecules such as small molecules, nucleic acids and other proteins. In the last few years, genetic information accumulated in an exponential rate and genomes of many organisms have been sequenced. However the knowledge of expressed genes and their sequence do not tell us a lot about the mechanism of biological processes. Comprehensive information on protein interactions would be valuable in understanding of the cellular program. Among these interactions protein-protein interactions (PPIs) are fundamental to cellular processes. They are responsible for critical processes such as metabolism, replication, transcription and most of the regulatory events. An organism contains PPIs in the order of ten thousands related with the size of its genome. For example it is estimated that there are about five interaction partners per protein in yeast when the most highly connected proteins are

excluded and eight partners otherwise [5]. This estimate suggests 16,000 – 26,000 interaction pairs in yeast.

Conventionally PPIs are detected via various experimental techniques such as western blotting, cross-linking and protease digestion. Recent high-throughput experimental techniques such as Yeast Two-Hybrid (Y2H) or Mass Spectrometry (MS) allowed scientific community to produce large amounts of PPI data (Figure 2).

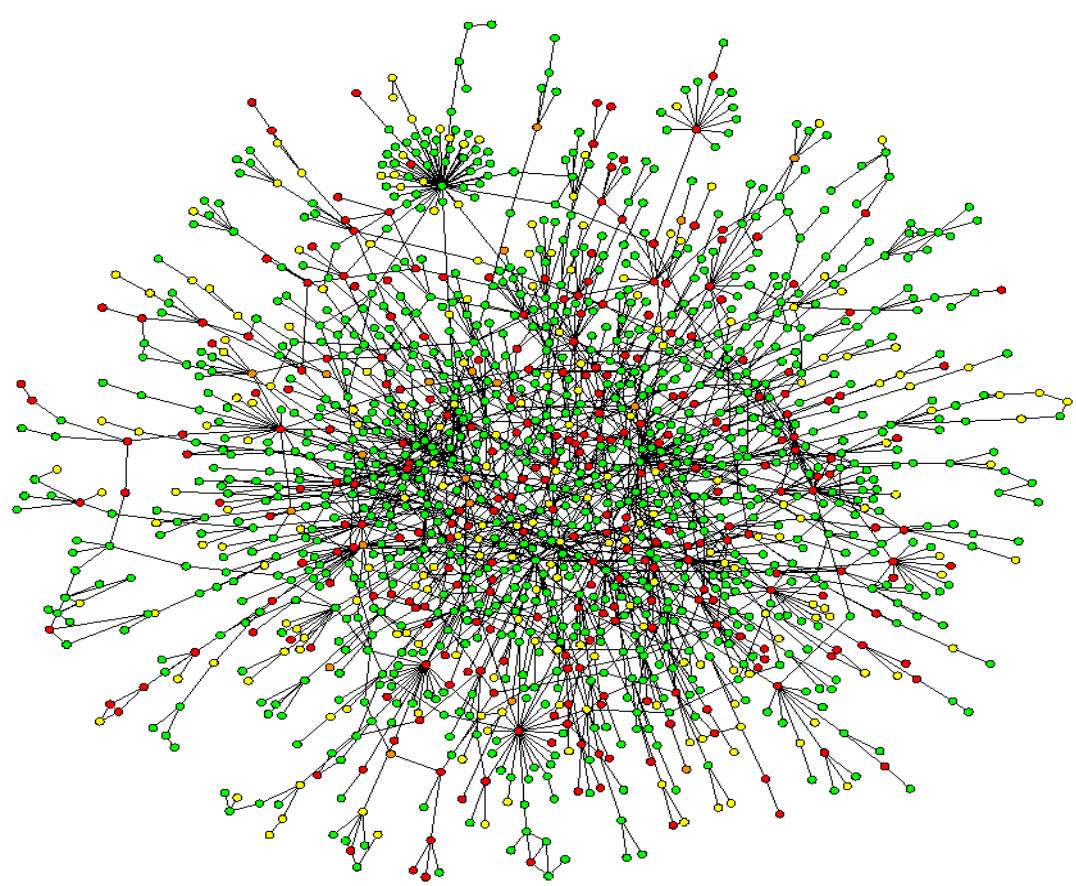

Figure 2: PPI map of 1870 proteins of yeast [6]. Each circle represents a different protein and each line indicates that the two end proteins are capable of binding to one another.

2.2.1 Yeast Two-Hybrid

Yeast two-hybrid (Y2H) [7]is a genetic assay to detect PPIs. This assay makes use of the process of regulation of gene expression in yeast. To activate expression of

a gene, a transcription activator protein is required to bind Upstream Activation Sequence (UAS) of a gene. That transcription activator protein has two domains with different functions. One is DNA binding domain (DBD) and the other is activation domain (AD). DBD of transcription activator binds to UAS and AD activates the expression afterwards (Figure 3).

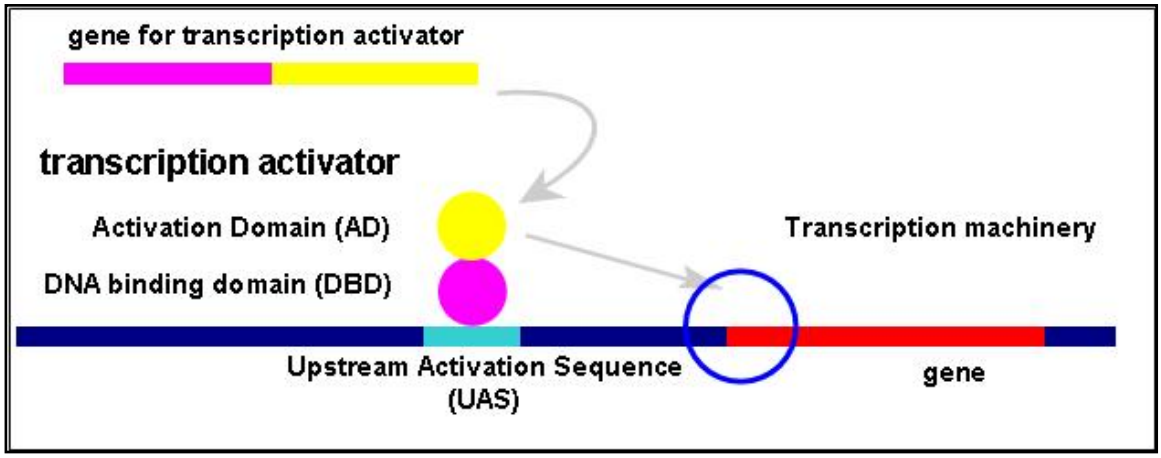

Figure 3: Regulation of gene expression in yeast.

In Y2H assay, artificial gene constructs are prepared to understand whether two proteins X and Y interact. First, the expressed gene is replaced by a reporter gene such as a fluorescent protein to observe the presence of gene expression easily. Then artificial constructs of two-hybrid proteins are prepared (X-DBD and Y-AD). If X and Y proteins interacts, DBD and AD will function properly and activate expression of the reporter gene; otherwise there will be no expression (Figure 4).

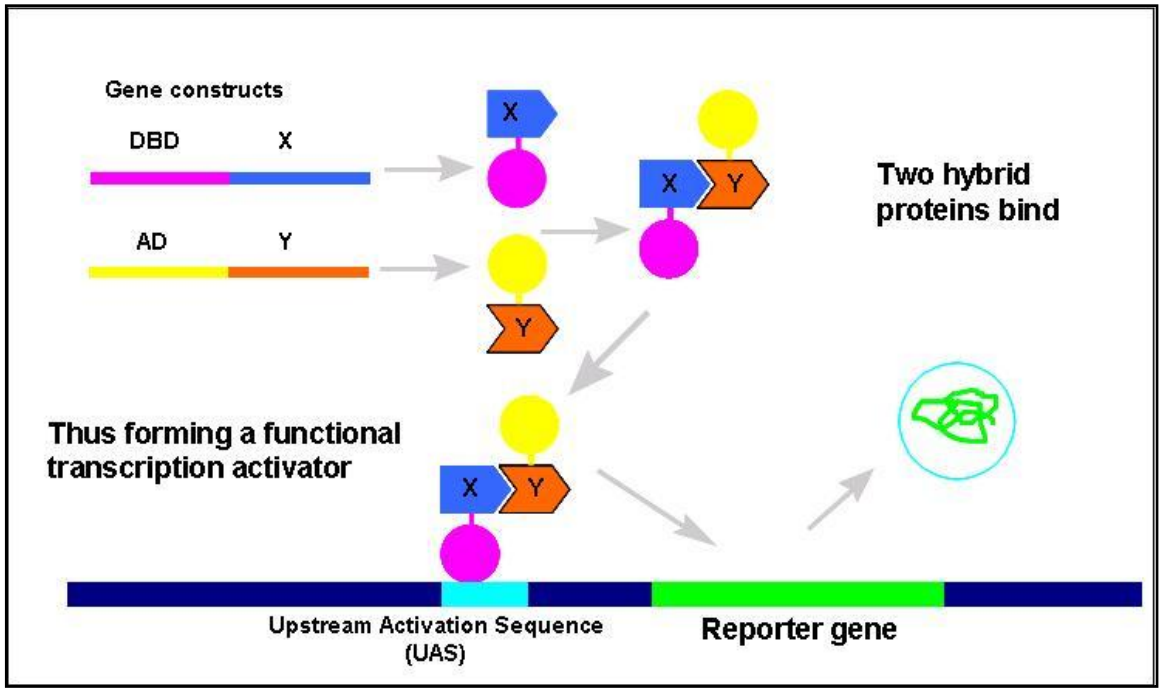

Figure 4: Y2H assay (modified from [8])

Preparing hybrid proteins associated with DBD and AD of transcription activator in large scales and assaying every pair of proteins in a yeast colony and observation of reporter gene allow us to detect organism wide PPIs.

However one should also bear in mind that Y2Hs are shown to produce a lot of false positives and negatives, due to several reasons including higher-than-normal target concentrations, misfolding due to artificial constructs or inhibitory binding topology. In a recent study [9] it has been shown that phage display and Y2H agree on only 95 out of 551 interactions detected by either one of the system. There are several ongoing efforts to increase reliability of such high throughput methods [10].

2.2.2 Mass Spectrometry

Mass spectrometry, also called mass spectroscopy, is an instrumental approach that allows for the mass measurement of molecules. The five basic parts of any mass spectrometer are: a vacuum system, a sample introduction device, an ionization source, a mass analyzer, and an ion detector. Combining these parts a mass spectrometer determines the molecular weight of chemical compounds by ionizing, separating, and measuring molecular ions according to their mass-to-charge ratio

(m/z). The ions are genes. No such message rated in the ionization source by inducing either the loss or the gain of a charge (e.g. electron ejection, protonation, or deprotonation). Once the ions are formed in the gas phase, they can be electrostatically directed into a mass analyzer, separated according to mass and finally detected. The result of ionization, ion separation, and detection is a mass spectrum that can provide molecular weight or even structural information. Traditionally use of MS was limited to analysis of small molecules. However several important advances in the field including Electrospray ionization [11] and soft laser desorption [12] allowed vaporization of large molecules such as proteins and their complexes [13][14][15].

2.2.3 X-ray Crystallography

X-ray crystallography exploits the fact that X-rays are diffracted by crystals [8]. Based on the diffraction pattern obtained from X-ray scattering off the periodic assembly of molecules or atoms in the crystal, the electron density can be reconstructed. A crystal of a protein complex can provide us with the partners of the complex and exact binding geometry. There is a substantial amount of protein complex structures in the Protein Data Bank [16].The problem with X-ray crystallography is to obtain the crystal itself. Although there are several methods for high throughput crystallization, none of them works universally and their data did not appear yet. Besides, transient interactions simply do not live long enough to crystallize and fail to be detected by X-ray crystallography.

2.2.4 Protein Interaction Databases

Data obtained from the above high-throughput techniques led to construction of a number of protein interaction databases.

A.0.1.1 BIND

BIND (Biomolecular Interaction Network Database) [17] has three groups of molecular associations: 1) molecular interactions, 2) molecular complexes and 3) pathways. Interactions are basic units of BIND and these interactions are not only

defined between proteins but between biological entities also including DNA, RNA, small molecules, molecular complexes, photons and unidentified entities. Molecular complexes are defined as a collection of two or more molecules to perform a single function. In BIND two or more interactions constitute a molecular complex. Pathways are defined as sequence of two or more interactions. Each BIND entry has to be supported by at least one publication in a peer-reviewed journal. Additional information such as complex topology and number of subunits supports molecular complex entries and pathway entries provides information about in which stage of the cell cycle this pathway is observed and whether this pathway is associated with a specific disease [18].

A.0.1.2 DIP

DIP (Database of Interacting Proteins) [19] stores experimentally determined interactions between proteins. DIP interaction data were curated manually by expert curators and also automatically by a computational approach which utilizes the information about the most reliable core interaction network of DIP [20].

A.0.1.3 MIPS CYGD

CYGD (Comprehensive Yeast Genome Database) contains 15488 annotated PPIs of *Saccharomyces cerevisiae*, including many high throughput experiment results.[21]

A.0.1.4 IntAct

IntAct [22] is an molecular interactions database that were formed by various European groups including EBI, MINT and SIB and MPI-MG. IntAct currently contains 27652 proteins and 36641 interactions.

2.3 Properties of Proteins

Data available in these databases are pairs of proteins or molecules determined to interact. We will be studying PPIs in this study since they carry out most of the regulatory function and execute the code stored in the genome of the cells. However

little or nothing is known about the nature or function of these interactions. But what we know most of the time is the properties of proteins. There are several databases storing available information about the proteins of several organisms. Swiss-Prot [23] is a curated protein sequence database. It provides high level of annotation of proteins such as the description of the function of a protein, its domains structure, post-translational modifications. This database also supports high level of integration with other databases. Protein Information Resource (PIR) supports a similar protein sequence database with high level of annotation [24].

In this study we aim to obtain more information about PPIs by using available data of interacting proteins. For the initial study we restricted our data sets to domain information of proteins, GO annotations and Swiss-Prot keywords.

2.3.1 Domains

Protein domains are structural or functional units of proteins and usually determined by evolutionarily conserved amino acid sequence modules. Proteins interact through their domains so the presence of certain domains in proteins brings the probability of interaction between those proteins. Also functions or structures of domains might provide information about the nature of the interaction. In Figure 5, domain architecture of pyruvate kinase protein can be observed.

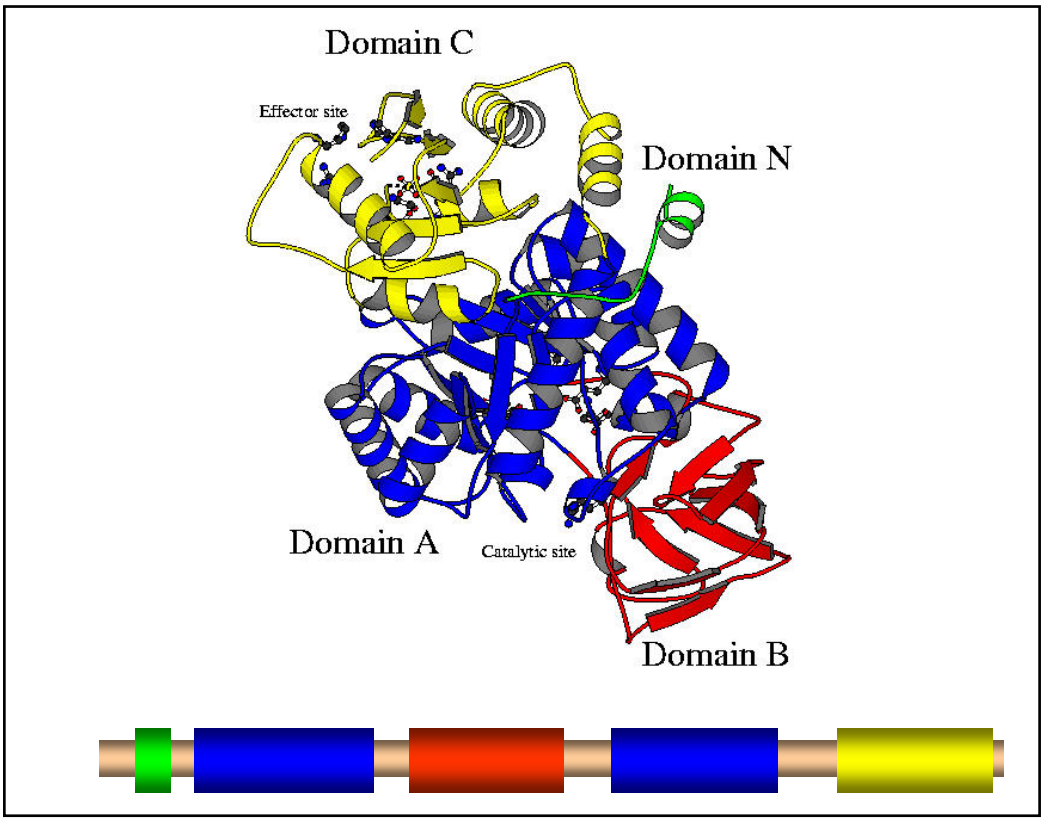

Figure 5: Crystal structure of one of the four subunits of pyruvate kinase [25].

For example growth factor receptors possess an extracellular domain for capturing the ligand, an intermembrane domain and finally a cytoplasmic protein-tyrosine kinase domain. When the ligand binds to the receptor, receptor forms a dimer through the intermembrane domain, autophosphorylates itself, and finally exposes docking sites for other cytoplasmic signaling proteins. These proteins, although diverse, always contain at least one Src homology 2 (SH2) domains, which is involved in binding to the receptor at specific binding sites. The human genome is estimated to contain 115 proteins containing SH2 domain [3].

SH2 is a prototype for a set of different interaction domains, such as SH3, PDZ, PH, KH or Puf repeat domains. Interaction domains control almost all cellular functions including signal transduction, protein trafficking, gene expression and chromatin organization [4].

In this study we referred to Protein families' (PFAM) database of alignments and Hidden Markov Models (HMMs) to obtain domain architecture of proteins.

PFAM is a large collection of multiple sequence alignments and HMMs covering many common protein domains [26]. There are two types of domains, PfamA and PfamB domains, in the database. PfamA domains are curated domains and have an assigned function, whereas PfamB domains are automatically generated domains using ProDom domains and do not have assigned functions. PfamA and PfamB are mutually exclusive. It is also possible to reach domain organizations of proteins by Pfam.

2.3.2 GO Annotations

Gene Ontology (GO) terms are the result of an ongoing effort of Gene Ontology Consortium to produce a controlled vocabulary to describe essential features of gene products of various organisms [27]. GO has three categorical roots

- molecular function,
- biological process and
- cellular component.

These three categories form a directed acyclic graph (DAG) independently but they have common terms and form a network when represented together. GO terms constitutes a common base for researchers to annotate proteins. Some of GO terms associated with p53 protein can be seen in Figure 6

P53 Protein:

GO:0005634 Cellular component: nucleus

GO:0003700 Molecular function: transcription factor activity

GO:0006281 Biological process: DNA repair

Figure 6: GO terms associated with tumor suppressor protein p53

2.3.3 Swiss-Prot Keywords

Swiss-Prot database has a flat set of keywords and uses them to annotate gene products in the database [23]. These keywords are based on structural, functional or other categorical properties of gene sequences. An example annotation can be seen in Figure 7.

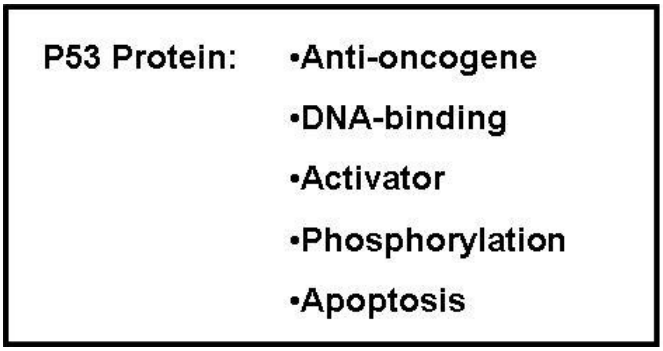

Figure 7: Swiss-Prot keyword annotation of p53 protein.

Chapter 3 Related Work

Currently there is significant data accumulation in molecular biology. This data is in the form of interaction data, gene and gene product annotations, gene expression data, etc. This accumulation gives rise to the need to extract useful information or to derive unavailable information using multiple sources of data.

There is a significant effort in the field to obtain useful information and to analyze these data. Some of these approaches are focused on below topics.

3.1 Protein Function Prediction

Elucidating sequences of genomes of many organisms, functional assignments of these sequences became one of the most important issues in molecular biology. There are several proteins whose functions are not known yet and there are several attempts to attack this problem.

Clare and King modified known machine learning and data mining algorithms such as C45 to learn rules about protein functions of yeast using multiple sources of data [28]. The data sets they used were sequence, phenotype, expression, homology, predicted secondary structure and MIPS functional classification data. They used the derived rules to assign functions to the proteins sequences with unknown function.

In a recent study of Deng et al, an algorithm based on Markov Random Field method is developed to assign protein function using physical and genetic interaction

networks of proteins [29]. They used GO terms in function assignment and they assigned GO terms with a probability representing the confidence of prediction, to proteins with unknown function. A similar study to predict protein functions according to GO categories was done by Jensen et al [30]. Sequence data of proteins with unknown functions is used as an input to the algorithm which has been trained with functionally categorized protein sequences before. Function prediction of a protein is not based on sequence similarity but instead the sequence derived features such as post-translational modification sites, protein sorting signals and other physical or chemical properties derived from amino acid sequences.

Clustering has been a widely used method to analyze gene expression data [31] [32]. But elucidation of roles of these clusters remained as a challenge to be taken. Lee et al [33] proposed a graph theoretical modeling of GO terms to interpret gene clusters. First they extracted common GO terms of a cluster of genes and then they found the representative interpretation of the cluster by traversing the GO tree structure they proposed.

3.2 Properties of a PPI Network

Several researchers studied global topology of PPI networks. *S. cerevisiae* PPI network is shown to have a highly heterogeneous degree distribution and scale free properties. Scale free graphs oppose regular graphs for their small diameter and highly connected neighborhoods compared to graph connectivity. Barabasi and Albert proposed a model for generating such graphs in which a graph is allowed to extend by adding a new vertex, and connecting this vertex to other vertices according to the “preferential attachment” rule, in which probability of connecting the vertex is a linear function of its degree [34]. It has been shown that most biological graphs fall into this category including PPI networks.

The domain reuse could be the reason for preferential attachment, as a protein with “popular domains” would be more likely to get attached to another protein created by gene duplication or obtained a new domain through exon shuffling.

3.3 PPI Inference

Even though it is possible to obtain PPI data in a high throughput fashion, it is an expensive and laborious procedure. There are many organisms without any interaction maps or with the maps, which are not complete. The reliability of the experimental techniques is also under discussion. So there are approaches to infer PPIs with computational methods either to support experimental interaction data or to complete interaction maps of organisms.

One approach used to infer PPIs is "interaction mining" which makes use of proteome similarity of closely related organisms. Given experimental interaction map of an organism and whole proteome sequence of the related organism, interaction map of the target organism is inferred using a learning algorithm based on computational statistical learning theory [35]. A previous study to infer the interaction map of an organism was done by Wojcik and Schachter in 2001 [36]. Interaction map of *E. coli* is predicted from interaction map of *H. pylory* and its associated protein domains. The inference method used the similarity searches together with clustering based on domain profiles and interaction patterns.

Another method to infer PPIs is protein interaction classification by unlikely protein profiles which is referred to as PICUPP [37]. This method is a statistical approach where the unusual profile pairs are determined if its occurrence in the interaction data is significantly unusual when compared to its occurrence in random protein profile pairing.

PICUPP method is performed independently with different protein profiles as Pfam domains, InterPro [38] signatures and Blocks [39] protein families using the DIP interaction data. After identifying the unusual protein profile pairs putative interacting proteins can be determined.

3.4 Domain Interaction Inference

Another approach using high throughput interaction data is trying to infer domain-domain interactions. Domain-domain interactions can be used to validate

PPIs, to obtain functional information to annotate PPIs or to infer interaction networks.

Ng et al developed a computational method to infer domain-domain interactions from interaction data, protein complexes and Rosetta Stone sequences [40][41]. They constructed a database of interacting domains (InterDom) with associated confidence levels for each putative pair of interacting domains.

Li et al used protein interaction data and 3D protein structure of complexes to derive interacting motif pairs [42].

These examples can be extended and there are other studies which do not fall in the above categories. For instance in a study of Segal et al [43] a probabilistic model is developed, which identifies molecular pathways using gene expression and protein interaction data. A pathway is defined to be a set of genes and gene products that function in a coordinated manner to accomplish a specific task.

In another study, Oyama et al tried to extract information on interactions using multiple sources of data available for interacting proteins [44]. Their approach was to discover association rules related to PPIs. Interaction data for the algorithm is created by combining the features of interacting left side protein (LSP) and right side protein (RSP). As a result they obtained rules like:

LSP: Localization = “Nuclear nucleoulus” &

Keyword = “nuclease”

⟶ RSP: Localization = “nuclear nucleoulus”

[14, 100.0%]

Rule means proteins having localization “nuclear nucleolus” and keyword “nuclease” interacts with proteins having localization “nuclear nucleolus”. The first number in brackets represents support and the second number represents confidence of this rule. They also discovered rules such as “an SH3 domain binds to a proline rich region”.

Chapter 4 Method

4.1 Problem Definition

If we assume that mechanistic basis of each interaction between two proteins is a particular evolutionary innovation (i.e. ab initio formation of a new interaction mechanism), then we can think of the interaction space as a rooted forest, where each root corresponds to an innovated interaction and the other non-root nodes correspond to evolutionary steps this interaction went through. Obviously we observe only a very small subset of this graph, as our PPI data is not complete even for the yeast, let alone a complete evolutionary account. We would like to find and group interactions that belong to the same tree.

In order to infer these groupings, we use a different but closely affiliated evolutionary relation: conserved domains. We assume that each tree has a rule, an ordered pair of set of domains, called relevant domains that must be satisfied by all interactions in the tree. In order to satisfy a rule, corresponding proteins of the interacting pair must have a superset of corresponding relevant domain set.

The problem is complicated by the fact that in eukaryotes, a protein takes part in an average of 5-8 interactions. Therefore not every domain of a protein is relevant to the interaction. For example, ERBB2 has 13 pfam domains (Figure 8), and is also known to be interacting with several proteins [45] including cyclin Bs. It interacts

with cyclin B's C-terminal domain via its protein kinase domain. Its other domains are involved in different interactions.

Figure 8: Domains of ERBB2 according to Pfam.

We assume that each interaction between two proteins exists for a particular mechanism, which can be represented as a specific domain signiture, so each interaction should be a member of a single cluster. Even though an interaction might occasionally satisfy rules of two or more clusters, the underlying cause of that interaction should be based on a certain conserved mechanism, preferably defined by a single cluster rule. We call the features of the protein that are relevant to this conserved mechanism active features for this interaction. (e.g. in ERBB2-cyclin B interaction protein kinase domain of ERBB2 is active whereas other domains are inactive.)

Given

- a set of features (domain features for this study) $D = \{d_1,..., d_m\}$,
- a set of proteins $P = \{p_1,..., p_t\}$, where every protein is defined by a set of features, $D(p_i) \subseteq D$,
- a set of PPIs $I = \{i_1,..., i_n\}$, where i_j is an ordered pair of proteins, $i_j \in P \times P$,

We would like to partition interaction set I to clusters such that interactions in the same cluster share the same conserved mechanism. We also aim to extract the rules which are composed of active features, describing interaction clusters. Formally we can model this problem as

to find a clustering $C = \{c_1,\ldots,c_k\}$, where $c_i \subseteq I$, $\bigcup_{i=1}^{k} c_i = I$ and $c_i \cap c_j = \varnothing$ for

$1 \leq i \neq j \leq k$, minimizing the total entropy function H:

$H=\sum_{c\in C}\sum_{d\in D} x_{d,c} \lg x_{d,c}$ where $x_{d,c}=\frac{\sum_{i\in c}\alpha_{d,i}}{|c|}$ and $\alpha_{d,i}$ is 1 if domain d is active in interaction i, 0 otherwise.

Obviously we do not know the α function and we will need some heuristics to estimate it.

4.2 Clustering Approach

Our approach to attack the above problem is to cluster PPIs with Expectation Maximization (EM) algorithm [46] and to assign descriptive rules to interaction clusters by observing cluster parameters.

In data mining it is not usually possible to describe the data by a single probabilistic or parametric model. Real life data is usually in the form of a mixture of different distributions and it is modeled as linear combination of multiple distribution functions. EM is one of the approaches used to solve such mixed models. Another property of such data is that it may not be possible to observe or determine all features of data instances. There may be missing features in some instances of data or a feature might not be observed at all. EM also performs well with unobserved features.

EM is a general iterative optimization algorithm, which maximizes a likelihood score function, given a probabilistic model with possibly missing data. Likelihood score function is a measure of how well the model fits and explains the data. In our problem, cluster labels are missing data and the probabilistic model is the mixture of different distribution functions of clusters. EM algorithm iterates between expectation (E) and maximization (M) steps until it satisfies some predetermined condition. In the expectation step, missing data is predicted using the computed parameters of the mixed model. In our problem we calculate membership probabilities of interactions in the E step. Parameters of the mixed model are computed in the M step using the calculated missing data of E step as well as observed features of the instances. These parameters are calculated to maximize the likelihood function. At the beginning of the algorithm either parameters of the model

or the missing data are assigned. This assignment might be random or some more intelligent assignment might be performed. Then algorithm starts from the step which would use these assignments. In our approach we prefer to assign missing data (membership probabilities) randomly and start from the maximization step which computes parameters using this random assignment.

4.2.1 Probabilistic Model

We have a set of interactions I = $\{i_1,..., i_n\}$. We assume that each interaction belongs to precisely one of k clusters. We represent this as an attribute of i such that i.C $\in \{1,\ldots, k\}$, where i.C variables are hidden variables of our model which we aim to determine in this study.

As given in the problem definition, each protein p is represented by its features. Each protein has a discrete-valued (0/1) m=|D| variables, '1' representing a present feature, '0' representing an absent one. Since interactions are ordered pairs of proteins, interaction i is represented by 2m variables, first m variable defining the features of the first interacting protein (protein-A), second m defining the features of the second one (protein-B). We use Naïve Bayes Model and we assume that these attributes are conditionally independent.

Equation 1 represents the model of observing interaction i as a member of cluster p given the parameters ω_p of cluster p, where $i.V_j$ represents the value of j^{th} attribute of interaction I and $\omega_{p,j}$ represents the parameter of cluster p for attribute j.

$$P_p(i \mid \omega_p) = \min\left(\prod_{j=1}^{m} P_p(i.V_j \mid \omega_{p,j}), \prod_{j=m+1}^{2m} P_p(i.V_j \mid \omega_{p,j}) \right) \quad (1)$$

Since the attributes are assumed to be conditionally independent, this model can be represented with the multiplication of individual probabilities of observing 1 for the j^{th} attribute in the p^{th} cluster. Denoting this probability ω_p, density function for the p^{th} cluster can be written as:

$$P_p(i.V_j \mid \omega_{p,j}) = \omega_{p,j}^{i.V_j}(1-\omega_{p,j})^{1-i.V_j}, \qquad 1 < p < k \quad (2)$$

The equation for observing interaction i in our data set can be expressed as the weighted sum of these cluster densities

$$P(i) = \sum_{p=1}^{k} \pi_p P(i \mid \omega_p) = \sum_{p=1}^{k} \pi_p . \min\left(\prod_{j=1}^{m} \omega_{p,j}^{i.V_j} (1-\omega_{p,j})^{1-i.V_j}, \prod_{j=m+1}^{2m} \omega_{p,j}^{i.V_j} (1-\omega_{p,j})^{1-i.V_j} \right) \quad (3)$$

where π_p represents the probability of cluster p ($0 \le \pi_p \le 1$ and $\sum_{p=1}^{k} \pi_p = 1$).

A.0.1.5 Expectation Step

In the expectation step of the EM algorithm we try to assign interactions to clusters. We do this assignment by computing the probabilities that an interaction i belongs to each cluster. Let $P(i.C = p)$ be the probability that interaction i belongs to cluster p. By Bayes rule and given fixed set of parameters ω and π, this probability can be written as:

$$P(i.C = p) = \frac{\pi_p . \min\left(\prod_{j=1}^{m} \omega_{p,j}^{i.V_j} (1-\omega_{p,j})^{1-i.V_j}, \prod_{j=m+1}^{2m} \omega_{p,j}^{i.V_j} (1-\omega_{p,j})^{1-i.V_j} \right)}{P(i)} \quad (4)$$

A.0.1.6 Maximization Step

In the Maximization step of EM algorithm we compute the new parameters which fit better to the cluster assignments done in the previous expectation step. New parameter matrix ω^{new} for interaction attributes are computed as follows:

$$\omega_{p,j}^{new} = \frac{\sum_{t=1}^{n} P(i_t.C = p) \cdot i_t.V_j}{\sum_{t=1}^{n} P(i_t.C = p)} \quad (5)$$

New cluster probability parameter vector π^{new} is computed as follows:

$$\pi_p^{new} = \frac{\sum_{t=1}^{n} P(i_t.C = p)}{\sum_{q=1}^{k} \sum_{t=1}^{n} P(i_t.C = q)} \quad (6)$$

We will iterate on E and M steps until the difference between new and old parameters is smaller than some predetermined threshold.

4.2.2 EM Algorithm

Basic idea of the algorithm and some definitions that will be used throughout this section are as follows.

We run EM algorithm multiple times to be able select the best clustering among multiple runs. At each EM run we iteratively perform E and M steps. In the E step we try to assign cluster membership probabilities of interactions which are represented by *hidden_clusters*. *hidden_clusters* is a 2D array of size *nof_interactions* x *nof_clusters*. In the M step we try to compute parameters of our mixed model, which are *weights* and *phi*. *weights* is a 2D array of size *nof_clusters* x *nof_features*, *phi* is an array of size *nof_clusters*. These two arrays correspond to ω and π of the probabilistic model, respectively. Maximization step also computes the *mean_weight_difference* value, which is the mean of the differences of the new values and the previous values of *weights* array.

EM iterates until this *mean_weight_difference* is smaller than a predetermined threshold which is 10^{-9} for our study.

Two arrays are introduced to decrease complexity of the algorithm. One is *feature_interaction_mapping* of size *nof_features* and each element *feature_interaction_mapping(j)* holds a list of interactions containing feature j. The other is *present_features* of size *nof_interactions*. *present_features(k)* holds the present features of interaction k.

algorithm RUN_EM
1) for i =1 to i= *nof_runs*
2) **call** DO_EM()
3) write output
4) write entropies

algorithm DO_EM

1) **call** INIT_CLUSTERS
2) *mean_weight_difference* : = **call** DO_MAXIMIZATION
3) **while** *mean_weight_difference* greater than *mean_weight_difference _threshold*
4) and *nof_iteration* smaller than *max_nof_iteration*
5) **call** DO_EXPECTATION
6) *mean_weight_difference* : = **call** DO_MAXIMIZATION
7) increase iteration by 1

algorithm INIT_CLUSTERS

1) **for** i = 1 to *nof_interactions*
2) **for** j = 1 to *nof_clusters*
3) *hidden_clusters(i,j)* randomly determine probability
4) normalize probabilities for each interaction

algorithm DO_MAXIMIZATION

1) **for** p = 1 to p= *nof_clusters*
2) **for** j = 1 to j= *nof_features*
3) compute *weights(p,j)* using

$$\frac{\sum hidden\_clusters(k,p) \quad [k \in feature\_interaction\_mapping(j)]}{\sum_{i=1}^{nof\_interactions} hidden\_clusters(i,p)}$$

4) compute *phi(p)* using

$$\frac{\sum_{i=1}^{nof\_interactions} hidden\_clusters(i,p)}{\sum_{q=1}^{nof\_clusters} \sum_{i=1}^{nof\_interactions} hidden\_clusters(i,q)}$$

5) **return** *mean_weight_difference*

algorithm DO_EXPECTATION

1) **for** k = 1 to k= *nof_interactions*
1) **for** p = 1 to p= *nof_clusters*
2) $proteinA = \prod_i weights(p,i) \cdot \prod_j (1 - weights(p,j))$

where i∈ *present_features(k)* and 1 ≤ i ≤ *nof_features/2*,

j ∉ *present_features(k)* and 1 ≤ j ≤ *nof_features/2*.

3) $proteinB = \prod_i weights(p,i) \cdot \prod_j (1 - weights(p,j))$

where i∈ *present_features(k)* and

(*nof_features* / 2 +1) ≤ i ≤ *nof_features*,

j ∉ *present_features(k)* and (*nof_features* / 2 +1) ≤ i ≤ *nof_features*.

4) *hidden_clusters(k,p) = phi(p)* . min (*proteinA, proteinB*)

5) normalize *hidden_clusters(k*)

4.3 Complexity Analysis

Let's denote the number of clusters with k, interaction data size with n and number of domain features with m.

Complexity of the expectation step:

In the expectation step, for each interaction we calculate Equation 4. A naïve approach takes Θ(kmn) steps as we have to iterate over all clusters and domains for each interaction. However we observe that domain features are very sparse. In order to take advantage of this fact we calculate, membership probability of a domainless protein for each cluster:

$$\rho_p{}^A = \prod_{j=1}^{m} (1 - \omega_{p,j}) \tag{7}$$

$$\rho_p{}^B = \prod_{j=m+1}^{2m} (1 - \omega_{p,j}) \tag{8}$$

We calculate this once per iteration and it takes Θ(km) time.

Now we can consider only present domains for each interaction as follows:

$$P(i.C = p) = \frac{\pi_p . \min\left(\rho_p^A \prod_{i.V_j=1 \wedge j \le m} \omega_{p,j} /(1-\omega_{p,j}), \rho_p^B \prod_{i.V_j=1 \wedge j > m} \omega_{p,j} /(1-\omega_{p,j}) \right)}{\sum_{p=1}^{p=k} \pi_p . \min\left(\rho_p^A \prod_{i.V_j=1 \wedge j \le m} \omega_{p,j} /(1-\omega_{p,j}), \rho_p^B \prod_{i.V_j=1 \wedge j > m} \omega_{p,j} /(1-\omega_{p,j}) \right)} \quad (9)$$

Let's denote the number of domain instances in all interactions with M *i.e.* the number of 1s in our sparse interaction-domain matrix. If we use appropriate data structures (such as a vector of linked lists instead of an array), then complexity of evaluating Equation 4 becomes $\Theta(km + kM)$. If we denote average number of domains per interaction with $\mu = M/n$, where $\mu << m$, expectation step complexity takes the final form of $\Theta(km + nk\mu)$.

Complexity of the maximization step:

In the maximization step, for each cluster we calculate Equation 5. Naively calculating Equation 5 takes $\Theta(knm)$ steps as we have to iterate over all interactions and domains for each cluster. Again we take advantage of sparseness property and for each domain we consider only interactions where this specific domain is found. We calculate, total of membership probabilities of all interactions for each cluster:

$$\sigma_p = \sum_{t=1}^{n} P(i_t.C = p) \quad (10)$$

in $\Theta(kn)$ steps. Then for each domain we calculate membership probabilities of interactions having this domain.

$$\omega_{p,j}^{new} = \frac{\sum_{i_t.V_j=1} P(i_t.C = p)}{\sigma_p} \quad (11)$$

If we use appropriate data structures (such as a vector of linked lists instead of an array), then our complexity becomes $\Theta(kn + kM)$. Our maximization step complexity takes the final form of $\Theta(kn + kn\mu) = \Theta(kn\mu)$.

Overall complexity:

We iterate over alternating E and M steps for e epochs. Thus the complexity becomes:

$\Theta(e(km+kn\mu))$

where μ is scale free and can be considered as a constant for each organism; thus we finally have $\Theta(ek(m+n))$.

There is no hard limit on the number of epochs needed for convergence. However since EM is a hill-climbing algorithm, it is guaranteed to converge. For hypothetic and real sample clustering, so far we never needed more than 100 epochs. For the entire real data set available, clustering into 40 clusters can be achieved in the order of 100 epochs. However for large number of clusters such as 400, 5000 epochs was not sufficient for the convergence of the algorithm. Optimizing cluster number and mean weight difference threshold might solve the convergence problem.

Chapter 5 Results

To test the performance of the algorithm we initially performed our studies on hypothetic data where we do the cluster assignments of interactions ourselves. So we could compare the results of our clustering algorithm with the actual assignments.

5.1 Hypothetic Data

We prepare the hypothetic data for given number of interactions (n), number of features of an interaction (2m) and number of proteins (t).

First we determine the cluster rules, which are common active features of interacting proteins within a cluster. We do this by assigning features to be active features with certain probability. This probability value is decided according to the properties of the real data. We expect to see about 2-6 active features per interaction. This expectation is based on the fact that yeast has ~5 domains per protein. Proteins have multiple functions and different domains or combinations of domains serve different functions and mediate different interactions. So a protein can allocate 1-3 domains to a single interaction on average. This results in 2-6 active features in an interaction on average.

In this study we determined the number of active features as 4 on average, ensuring the presence of at least 2 active features, one on protein-A, the other on protein-B side of an interaction rule. So first we determined one active feature for

each protein description of a rule and then assigned activity to the other features with probability

(4(average) -2(ensured)) / 2m (feature size) = 1/m.

Rule assignment corresponds to determination of ω matrix of the EM algorithm.

After cluster rule assignment, cluster probability assignment is done by assigning random values to each cluster and normalizing afterwards. This property corresponds to π vector.

Interactions are created after determining parameters of clusters. For each interaction its cluster assignment is determined according to cluster probability vector and two proteins (protein A and B) are randomly selected to participate in that interaction. At the beginning, all values of m features defining a protein are 0. When a protein is selected to be protein-A of an interaction, its features are updated according to the assigned cluster rule and active features among first m features of the rule vector are updated to be present in protein-A. Active features among remaining half of the rule vector are updated to be present in protein-B.

This process is performed until all interactions are assigned to some cluster and their interacting protein pairs are determined. For every interaction protein features are updated according to cluster rules.

This model treats interactions as ordered pairs, for example we expect all kinases on the left position and all their targets on the right. Obviously this is not the case with the real data. As a workaround, we introduce two ordered pairs (a,b) and (b,a) in order to represent unordered pair {a,b}. As a side effect we expect symmetric clusters with a perfect clustering. At the end we obtain a hypothetic data set of 2n interactions of 2k symmetric clusters.

5.2 Real Data

We obtained our interaction data set using BIND data. BIND has multiple types of interactions such as protein-DNA and protein-RNA as well as PPI. In addition it is not a single organism database. We extracted 5817 PPIs of Saccharomyces cerevisiae

from BIND data. These 5817 interactions are among 3620 proteins. We prepared a relational database of interactions and populated this database with the information available for proteins in public databases. We annotated the proteins with Pfam domains, GO terms and Swiss-Prot keywords. For this study we use only Pfam domain compositions of proteins as the feature set describing a protein.

Interaction data is written on a text file, which would be input to the clustering algorithm. Each interaction constitutes a single line of two times the feature set size of characters in data file, first feature set signifying protein-A, second signifying protein-B. Each character is either '1' or '0' corresponding to presence or absence of features respectively.

5.3 Clustering Evaluation

We have three measures to evaluate clustering and the clusters. They are all based on statistical entropy measures [47].

5.3.1 Statistical Entropy

Our clustering algorithm computes a 2D array of probabilities called *hiddenClusters*. Rows correspond to interactions and columns correspond to clusters. So the element at position (i,j), *hiddenClusters(i,j)*, defines the probability of interaction i to belong to cluster j. For an interaction the sum of probabilities is 1.

In our statistical entropy measure we assign interactions to its most probable cluster. Then for a cluster we measure the entropy of cluster c_p as

$$Entropy(c_p) = -\sum_{i=1}^{k} x_i \log x_i ,$$

where k is the number of hypothetic clusters and x_i is the ratio of number of interactions which belong to hypothetic cluster h_i to the number of elements of cluster c_p:

$$x_i = \frac{|\, h_i \cap c_p \,|}{|\, c_p \,|}$$

5.3.2 Probabilistic Entropy

With this measure, different from the measure above we do not assign interactions to the clusters with the highest probability; instead we use probabilistic values:

$$ProbabilisticEntropy(c_p) = -\sum_{i=1}^{k} x_i \log x_i \ ,$$

where c_p is the cluster, of which we measure the entropy, k is the number of hypothetic clusters and x_i is calculated using

$$x_i = \frac{\sum_{u \in h_i} \left(u \cdot hiddenClusters(u, p)\right)}{\sum_{v \in c_p} \left(v \cdot hiddenClusters(v, p)\right)} .$$

5.3.3 Unsupervised Entropy

The above measures can only be used in the presence of information of hypothetic or real cluster information. But we need to have a measure to evaluate the clusters when we do not know the real clusters which would be the case for our real data. So we developed an entropy measure for an unsupervised clustering.

We computed entropies of each cluster c_p, as the total of multiplication of each interaction probability to be in that specific cluster with the entropy of that interaction:

$$UnsupervisedEntropy(c_p) = \sum_{v} \left(hiddenClusters(v, p) \cdot E(v)\right),$$

where entropy of an interaction is computed from probability distribution of that interaction using

$$E(v) = -\sum_{i=1}^{k} hiddenClusters(v, i) \cdot \log hiddenClusters(v, i) .$$

5.4 Hypothetic Data Results

We prepared hypothetic data of different numbers of interactions, features, proteins and clusters. Then we clustered the hypothetic interactions with EM algorithm for multiple runs and compared the total entropies to choose the best clustering with the lowest entropy.

Our first clustering results are with 400 interactions belonging to 20 clusters. Interactions were among 1000 proteins with 500 features representing them. EM clustering algorithm is run 10 times and the total entropy measures of clusters for each run are shown in Table 1. According to total statistical entropy and total probabilistic entropy measures last run scores lowest, which suggests that run 10 gives the best clustering among the 10 runs. Figure 9 shows the distributions of the members of the clusters obtained as the result of last EM run. Members of a cluster are determined by assigning an interaction to its highest probable cluster. Each bar represents what percent of the members of that cluster originates from which hypothetic cluster. Colors represent hypothetic clusters.

	Total Ent	**Total Prob Ent**	**Total Unsuper Ent**
RUN 1	13.991	14.208	16.061
RUN 2	15.692	15.867	30.272
RUN 3	15.178	15.421	45.341
RUN 4	16.201	16.596	53.442
RUN 5	13.249	13.568	35.325
RUN 6	15.663	16.241	65.079
RUN 7	13.957	14.150	22.875
RUN 8	14.880	15.227	45.344
RUN 9	14.729	14.876	15.980
RUN 10	12.778	13.041	24.236

Table 1: Total entropy values calculated for each run of the algorithm. Total statistical entropy and total probabilistic entropy show similar pattern, whereas the unsupervised entropy behaves differently. According to first two entropy measures 10^{th} run gives the best clustering.

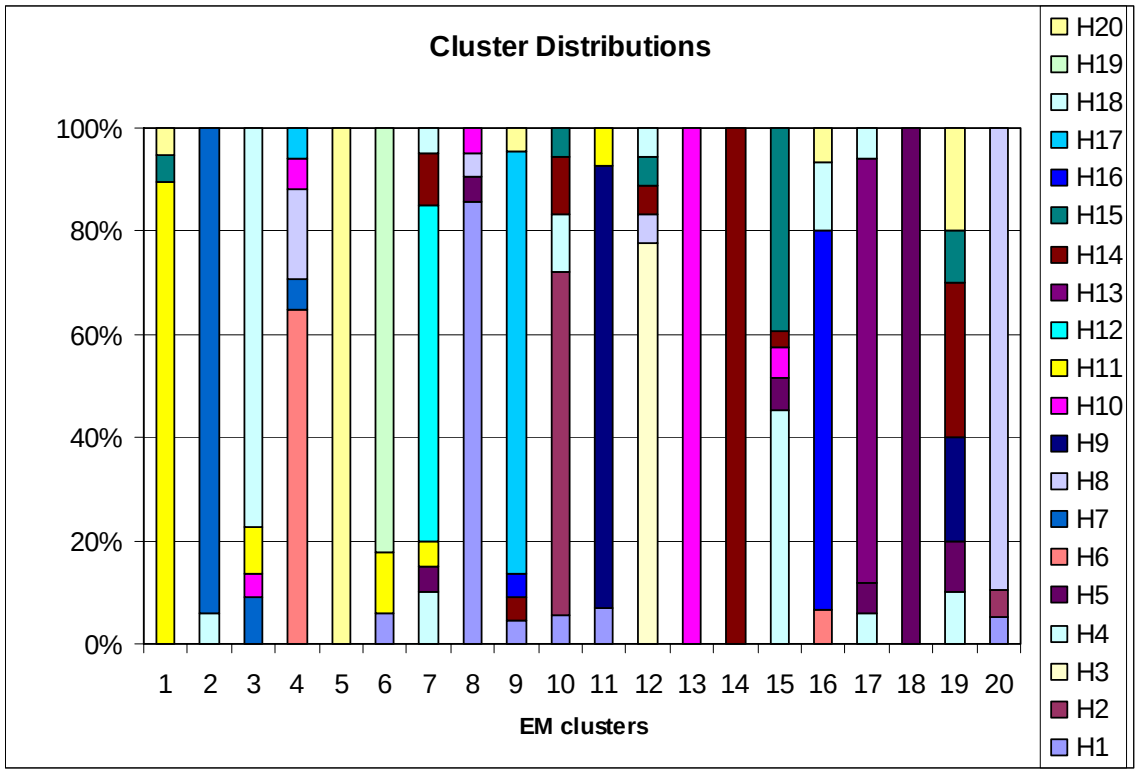

Figure 9: Cluster distribution of hypothetic data of 400 symmetric interactions (A-B and B-A) of 1000 proteins. Each protein is defined by 500 features. Interactions are created according to 20 cluster rules without introducing any error to the rules. This figure represents cluster distributions of clusters obtained by EM. Each bar represents percentage of members coming from hypothetic clusters and colors represent hypothetic clusters. Color hypothetic cluster mapping is given with the legend on the right.

	Stat Ent	Prob Ent	Unsuper Ent
EM 1	0.409459	0.413097	0.49244826
EM 2	0.223718	0.240408	0.0302038
EM 3	0.775714	0.780758	0.17848597
EM 4	1.087761	1.092503	0.15379087
EM 5	0	1.27E-05	3.03897891
EM 6	0.578325	0.554874	0.64984492
EM 7	1.189886	1.210222	0.19090528
EM 8	0.567061	0.665109	1.26900196
EM 9	0.726193	0.726246	0.02112411
EM 10	1.079735	1.079735	1.46E-14
EM 11	0.509137	0.523422	1.08281205
EM 12	0.837772	0.837772	8.05E-25
EM 13	0	4.12E-09	2.39814848
EM 14	0	1.43E-13	3.03764265
EM 15	1.171123	1.26529	5.59577996
EM 16	0.857174	0.862877	0.00266242
EM 17	0.659872	0.659505	0.03115513
EM 18	0	1.84E-11	3.61538726
EM 19	1.695743	1.721635	2.01557814
EM 20	0.409459	0.407506	0.43181102

Table 2: Entropy measures. Three different entropy measures of each clusters obtained at the 10^{th} run of the clustering algorithm. Statistical entropy values and probabilistic entropies give similar results. Cluster 5, 13, 14, 18 have the lowest entropy values according to first two measures.

As seen in Table 1 and Table 2, unsupervised entropy measure shows an unpredicted behavior. This may be the result of a wrong assumption about the properties of clusters. We will not be using that measure to evaluate clustering in this study.

When we observe Figure 9 and Table 2, clusters 5, 13, 14 and 18 have homogenous composition and thus the lowest entropy measures. In these clusters we expect to observe similar or same rules to corresponding hypothetic clusters. Matching cluster rules can be seen at Table 4. 18 of the hypothetic cluster rules are identified with quite high weights. Even the cluster compositions are not homogenous in the sense of hypothetic clusters, the rules are determined with very high rate.

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20	Total
EM1	0	0	0	0	0	0	0	0	0	0	17	0	0	0	1	0	0	0	0	1	**19**
EM2	0	0	0	1	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	**17**
EM3	0	0	0	0	0	0	2	0	0	1	2	0	0	0	0	0	0	17	0	0	**22**
EM4	0	0	0	0	0	11	1	3	0	1	0	0	0	0	0	0	1	0	0	0	**17**
EM5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	**39**
EM6	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	14	0	**17**
EM7	0	0	0	2	1	0	0	0	0	0	1	13	0	2	0	0	0	1	0	0	**20**
EM8	18	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	**21**
EM9	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	18	0	0	1	**22**
EM10	1	12	0	2	0	0	0	0	0	0	0	0	0	2	1	0	0	0	0	0	**18**
EM11	1	0	0	0	0	0	0	0	12	0	1	0	0	0	0	0	0	0	0	0	**14**
EM12	0	0	14	0	0	0	0	1	0	0	0	0	0	1	1	0	0	1	0	0	**18**
EM13	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	**39**
EM14	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	**12**
EM15	0	0	0	15	2	0	0	0	0	2	0	0	0	1	13	0	0	0	0	0	**33**
EM16	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	11	0	2	0	1	**15**
EM17	0	0	0	1	1	0	0	0	0	0	0	0	14	0	0	0	0	1	0	0	**17**
EM18	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**11**
EM19	0	0	0	1	1	0	0	0	2	0	0	0	0	3	1	0	0	0	0	2	**10**
EM20	1	1	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	**19**
total	**23**	**13**	**14**	**22**	**17**	**12**	**19**	**22**	**14**	**44**	**23**	**13**	**14**	**22**	**17**	**12**	**19**	**22**	**14**	**44**	**400**

Table 3 Distributions of members of clusters. Each row corresponds to one of the resulting clusters of EM algorithm and each column corresponds to one hypothetic cluster. In a row how many members of a cluster originates from each hypothetic cluster can be observed. Last row demonstrates sizes of hypothetic clusters and the last column demonstrates sizes of EM clusters.

H cluster	EM cluster	Rule Domains	H weight	EM weight
20	5	162	1	1
		312	1	1
		394	1	1
10	13	312	1	1
		394	1	1
		162	1	1
14	14	92	1	1
		169	1	1
5	18	487	1	1
		55	1	1
11	1	251	1	0.89
		148	1	0.95
		453	1	0.95
7	2	472	1	0.94
		111	1	0.99
		155	1	0.99
		311	1	0.99
18	3	240	1	0.82
		360	1	0.86
		394	1	0.86
		456	1	0.86
6	4	95	1	0.76
		51	1	0.76
		304	1	0.76
		482	1	0.76
19	6	17	1	0.95
		167	1	0.95
		195	1	0.84
12	7	349	1	0.65
		403	1	0.65
		280	1	0.98
		907	1	0.98
1	8	148	1	0.9
		453	1	0.9
		251	1	0.84
17	9	111	1	1
		155	1	1
		311	1	1
		472	1	0.82
2	10	280	1	1
		407	1	1
		349	1	0.67
		403	1	0.67
9	11	195	1	0.85
		17	1	1
		167	1	1
3	12	185	1	1
		268	1	1
		425	1	1
		58	1	0.78
		155	1	0.78
		229	1	0.78
16	16	51	1	0.86
		304	1	0.86
		482	1	0.86
		95	1	0.86
13	17	58	1	0.82
		155	1	0.82
		229	1	0.82
		185	1	1
		268	1	1
		425	1	1
8	20	360	1	1
		394	1	1
		456	1	1
		240	1	0.9

Table 4: Matching cluster rules of hypothetic and EM clusters. Lightly shaded regions represent protein-A related part of the rule, darkly shaded regions represent protein-B part of an interaction. All hypothetic rule weights are '1' because we did not introduce error to this data set. 18 of the 20 cluster rules could be determined with very high weight values.

Results for other hypothetic data sets which are prepared with different parameters can be seen in Appendix.

5.5 Real Data Results

Before clustering whole data we prepared a sample data set of real interactions. Since we do not know the real clusters of whole interaction set, we wanted to test the algorithm on a small set of real interactions which we selected according to some certain interaction rules. Then we performed clustering of whole interaction data.

5.5.1 Clustering of Real Sample Data

This interaction set is composed of 188 interactions (A-B and B-A) of 14 clusters. We chose cluster rules by first searching for the mostly supported domain pairs, one in protein-A one in protein-B of an interaction. We verified interaction of these domain pairs from either iPfam [48] interacting domain pair entries, which are obtained from 3D protein structures, or from Pfam annotations of domains or from literature. After determining an interacting domain pair we queried the interactions containing the domain pair and formed a cluster accordingly. Cluster rules we determined and the sizes of real clusters can be observed in Table 5.

	Protein-A		Protein -B		
Cluster Rule	**domain**	**pfam id**	**domain**	**pfam id**	**nof interactions**
1	LSM	PF01423	LSM	PF01423	21
2	SH3	PF00018	SH3	PF00018	20
3	SH3	PF00018	WH2	PF02205	17
4	SNO	PF01174	SOR_SNZ Pfam-B_54868	PF01680 PB054868	13
5	SH3	PF00018	protein kinase domain	PF00069	10
6	Y_phosphatase2	PF03162	Y_phosphatase2	PF03162	7
7	CKS	PF01111	Cyclin_C Cyclin_N	PF02984 PF00134	6
8	LSM	PF01423	LSM	PF01423	21
9	SH3	PF00018	SH3	PF00018	20
10	WH2	PF02205	SH3	PF00018	17
11	SOR_SNZ Pfam-B_54868	PF01680 PB054868	SNO	PF01174	13
12	protein kinase domain	PF00069	SH3	PF00018	10
13	Y_phosphatase2	PF03162	Y_phosphatase2	PF03162	7
14	Cyclin_C Cyclin_N	PF02984 PF00134	CKS	PF01111	6

Table 5: Cluster rules and sizes of real sample data. First seven rules are forward rules and the remaining rules are obtained by reversing the first seven rules. Protein-A part of the rules are lightly shaded and the protein-B parts are darkly shaded.

We ran the EM algorithm for 50 times. 3rd run gave the lowest value for both statistical entropy and probabilistic entropy. Cluster distributions of the clustering results can be observed in Figure 10 and in Table 6.

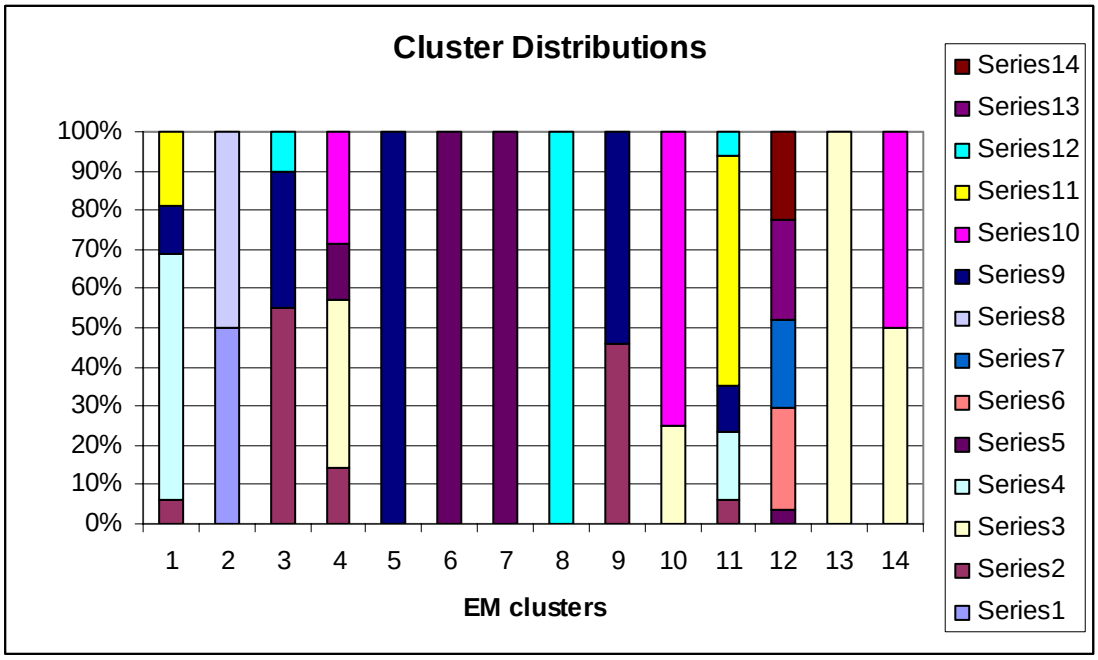

Figure 10: Cluster distributions of sample real data of 188 interactions.

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	Total
EM1	0	1	0	10	0	0	0	0	2	0	3	0	0	0	**16**
EM2	21	0	0	0	0	0	0	21	0	0	0	0	0	0	**42**
EM3	0	11	0	0	0	0	0	0	7	0	0	2	0	0	**20**
EM4	0	1	3	0	1	0	0	0	0	2	0	0	0	0	**7**
EM5	0	0	0	0	0	0	0	0	2	0	0	0	0	0	**2**
EM6	0	0	0	0	3	0	0	0	0	0	0	0	0	0	**3**
EM7	0	0	0	0	5	0	0	0	0	0	0	0	0	0	**5**
EM8	0	0	0	0	0	0	0	0	0	0	0	7	0	0	**7**
EM9	0	6	0	0	0	0	0	0	7	0	0	0	0	0	**13**
EM10	0	0	3	0	0	0	0	0	0	9	0	0	0	0	**12**
EM11	0	1	0	3	0	0	0	0	2	0	10	1	0	0	**17**
EM12	0	0	0	0	1	7	6	0	0	0	0	0	7	6	**27**
EM13	0	0	5	0	0	0	0	0	0	0	0	0	0	0	**5**
EM14	0	0	6	0	0	0	0	0	0	6	0	0	0	0	**12**
Total	**21**	**20**	**17**	**13**	**10**	**7**	**6**	**21**	**20**	**17**	**13**	**10**	**7**	**6**	**188**

Table 6: Cluster member distributions of sample real data clustering. Rows correspond to clusters obtained as result of EM clustering and columns correspond to real clusters.

When the rules of the clusters, obtained as a result of the EM clustering (Table 7), are observed and compared with the real cluster rules, it can be seen that most of the rules are captured by the algorithm.

- EM2 captured rule R1 and R8 which is the interaction of LSM and LSM domains. It was our mistake to represent these interactions in two clusters in the real data. But clustering algorithm collected all these kind of interactions into one cluster.

- Rules R2 and R9, which are again the rules of interaction of the same domain SH3, are captured by EM3, EM5 and EM9 with some additional domains whose participation to the interactions can be examined.

- Rule R3 is captured by EM13. But there are extra domains on protein-B site of the captured rule. These are PF00568 ‘WH1’ domain and PB000008 not annotated pfamB domain. Since we created our clusters according to some interacting domain pairs these pairs might not be the only domains mediating that cluster of interactions. So we can say that WH1 and the pfamB domains might be putative interaction partners of SH3 domain. The reverse rule R8 is captured by EM10.

- EM1 captured rule of R4 and the reverse rule which is the rule of R11 captured by EM11.

- Rule R5 is captured by clusters EM6 and EM7. But there are some other domains complementing the R5. The reverse rule R12 is captured by EM8.

- Rule R6 and its reverse rule R13, which is the same, are captured by EM12 but the weights are low in EM12 since the composition of EM12 is not homogenous.

- Rule R7 and its reverse rule R14 are not captured by any of the EM cluster rules. But all of their instances are collected in EM12 with other interactions.

–

EM cluster	Rule Domains	EM weight
1	PF01174	0.81
	PF01680	0.81
	PB054868	0.81
2	PF01423	1
	PF01423	1
3	PF00018	0.9
	PF00063	0.5
	PF06017	0.5
	PF00018	1
4	PF02205	0.71
	PB063121	0.71
	PB068476	0.71
	PB070621	0.71
	PF00018	0.86
5	PF00018	1
	PB027744	0.5
	PF00018	1
	PF02809	1
	PF00790	1
	PB100632	1
6	PF00241	0.67
	PF00018	1
	PB083483	0.67
	PF00069	1
7	PF00018	1
	PF00168	0.6
	PF00130	0.6
	PF02185	0.6
	PF00069	1
	PF00433	0.6
8	PF00069	1
	PF00018	1
9	PF00612	0.54
	PF00063	0.85
	PF06017	0.85
	PF00018	1
	PF00018	1
10	PF00568	1
	PF02205	1
	PB000008	1
	PF00018	1
11	PF01680	0.76
	PB054868	0.76
	PF01174	0.76
12	PF03162	0.52
	PF03162	0.52
13	PF00018	1
	PF00568	1
	PF02205	1
	PB000008	1
14	PF00018	1
	PF00568	0.59
	PF02205	1
	PB000008	0.59

Table 7: Rules of clusters obtained by EM.

We can say that all rules, except two of them (R7 and R14), are captured by the clusters of EM clustering. But the distributions within clusters are usually not homogenous. This shows us that even when we capture correct parameters in expectation step we might not be able to assign correct interactions to correct clusters to correct clusters in maximization step.

5.5.2 Clustering of Entire Data Set

We clustered entire real data set which is composed of 5817 interactions. Before starting clustering we added reverse interaction data to our data set obtaining total of ~12,000 interactions. We clustered them into 400 clusters in a single run with 5000 iterations. The only criteria for the analysis of the clustering results are the rule weights. Some of the resulting cluster rules, with high rule weights can be observed in Table 8.

Pkinase	Pkinase
SH3_1	WH1 WH2
Myosin_head Myosin_tail_2 SH3_1	Drf_GBD FH2
SOR/SNZ family Pfam-B_54868	SNO
XPG I-region XPG N-terminal domain	Pumilio-family RNA binding repeat RNA recognition motif
Pkinase	AMPKBI
Arm Importin beta binding domain	Glyceraldehyde 3-phosphate dehydrogenase, C-terminal domain Glyceraldehyde 3-phosphate dehydrogenase, NAD binding domain
Armadillo/beta-catenin-like repeat Importin beta binding domain	bZipTranscriptionfactor

Table 8: Some interaction cluster rules obtained as a result of entire real data set clustering.

Some of the rules are same as the rules we used to extract sample real data such as "SH3 – WH1, WH2" interaction rule and SOR/SNZ family, PfamB_54868 – SNO" interaction rule. A comprehensive analysis of the results might identify new putative interaction rules.

Chapter 6 Implementation

6.1 PATIKA

PATIKA is an integrated, multi-user environment to build, store, query, manipulate and visualize network of complex cellular events to facilitate effective analysis of biological data [49]. Cellular events are modeled on a well defined comprehensive ontology in PATIKA [50]. PATIKApro, which is currently being developed, can model biological data at two levels:

State-transition level: This level models actors of the cellular events as states and the events as transitions. Basic components of this level ontology can be seen in Figure 11.

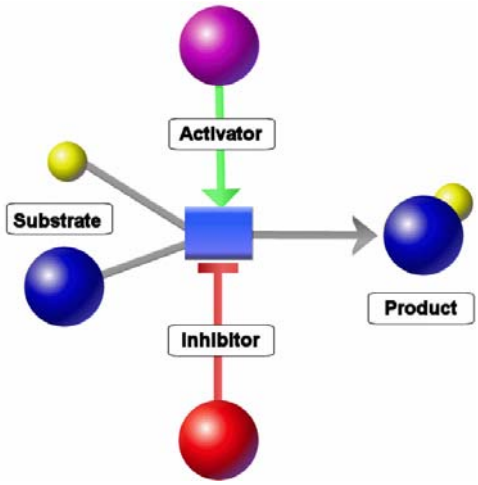

Figure 11: Basics of state-transition level ontology.

Bioentity level: Most of the time states have a common path of synthesis, and they have very similar chemical structures. For example phosphorylated, mdm2-bound and native states of p53 proteins are all different states and different functions but they are very similar in structure. PATIKA groups these states and models them as single biological entity called bioentity. It is also very common to group these states in pathway drawings.

6.2 Bioentity Graphs

Most of the available biological data do not have state-transition level of detail. PPI data is an example of this type of data. We often know that two proteins interact, but we do not know which state of proteins interact. They might have different states such as phosphorylated, aberration, nuclear or cytoplasmic state. We also might not know the mechanism of the interaction; in other words, the type of the transition. The reason for an interaction may be chemical modification, cleavage or allosteric change of one interaction partner by the other. Another type of o information which is frequently missing is whether this interaction is activated or inhibited by another entity.

But still this incomplete data is valuable and needs to be represented. Bioentity graphs of PATIKA allow us to model and represent this type of data.

PATIKA has three types of bioentities:

- Chemical bioentities are biological entities representing small molecules, ions, amino acids etc.

- Physical bioentities represent biological entities corresponding to physical events such as UV radiation, mechanical stress or heat.

- Genetic bioentities correspond to molecules derived from genetic material which are DNA, RNA molecules and proteins.

Different types of interactions among bioentities are modeled in PATIKA. These are transcriptional regulation (activation or inhibition), protein-protein interaction and generic interaction. (Figure 12)

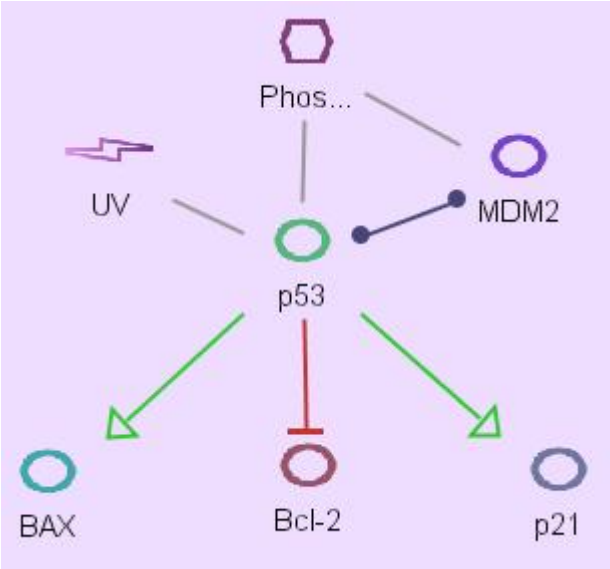

Figure 12: Different type of bioentities and bioentity interactions types of PATIKA. Nodes correspond to bioentities and the edges correspond to bioentity interactions.

It is also possible to query bioentities or subgraphs of bioentity network in PATIKApro.

Representation of custom data is also possible. In our study we associated cluster assignments of PPIs as custom data and represented them with different colors on the bioentity graph (Figure 13). Bioentity graph of PATIKA make the analysis easier since it allows us to see properties of proteins interactively with an object inspector. For example, the properties of protein Myo5 are displayed on the inspector window on the right of Figure 13.

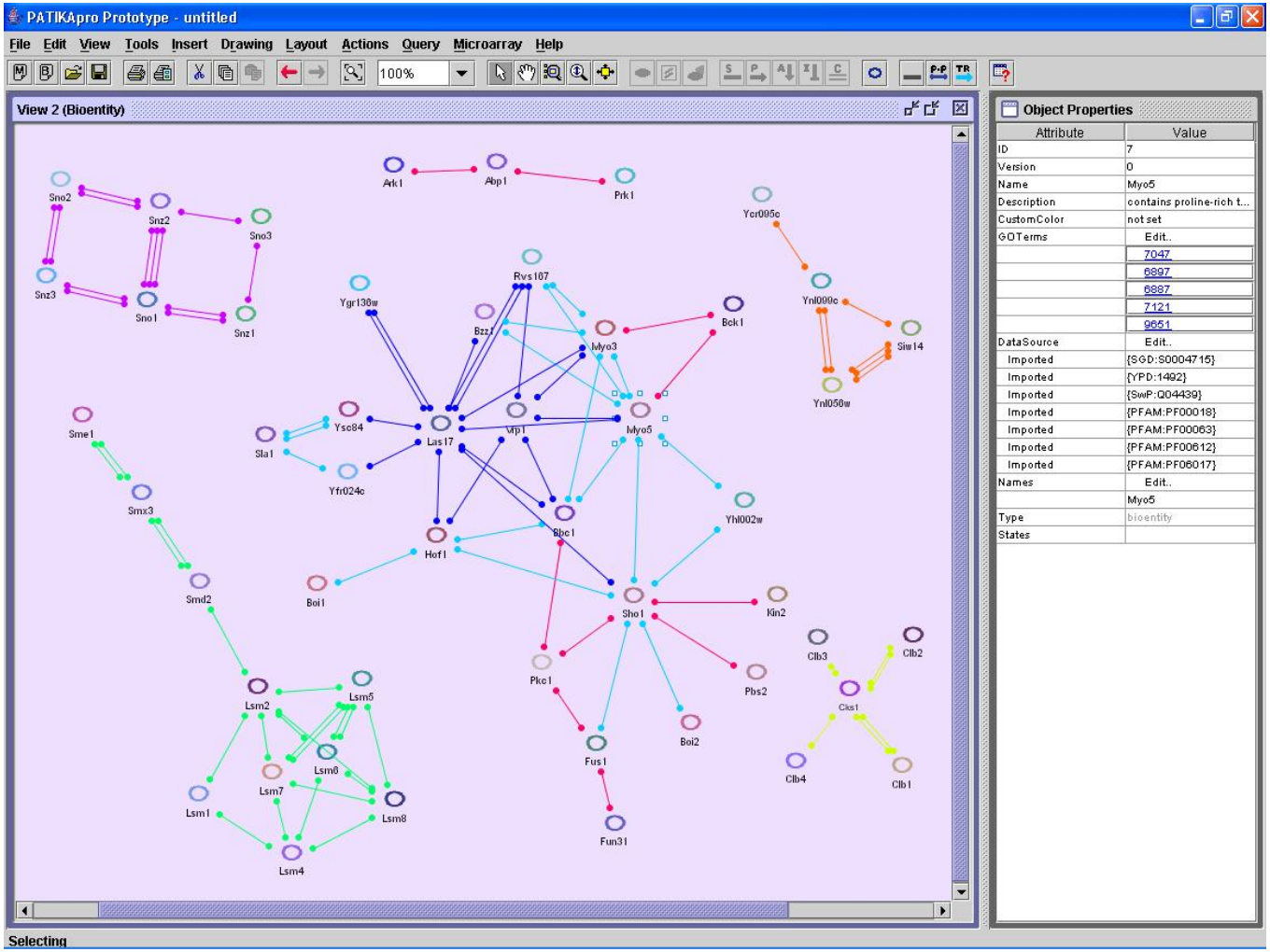

Figure 13: Bioentity graph of sample real PPI data with their assigned clusters. Different colors represent different clusters.

Chapter 7 Conclusion and Discussion

PPIs are fundamental to cellular processes and PPI data is available in very large amounts, thanks to the recent high throughput techniques. Even though there is a large amount of interaction data available, little is known about their mechanisms or roles in a cell.

In this study we have developed methods to gain information about the interactions by clustering them into interaction groups, with the assumption that within a group, interactions would probably share evolutionarily conserved mechanism. We built our approach around the idea that, domains form the structural frame of PPIs and they are evolutionarily conserved with genetic processes such as exon shuffling. So we aimed to obtain cluster rules in the form of two sets of domains, one needs to be present in one protein of an interacting pair and the other in the other protein. Analysis of domain set pairs allows us to gain information on interaction mechanism and functions.

We applied a probabilistic approach to model interaction clusters and used the EM algorithm to solve the clustering problem. According to our hypothetic data results, and the sample real data it is observed that cluster rules can be captured pretty well. Even when we have the exact matching rule, interaction composition of a cluster might not be homogenous. This condition gets worse with increasing number of interactions. This might mean that even if we can come up with good parameters in maximization step, we might have problems in the assignment of cluster membership probabilities of interactions in the expectation step. Increase in the entropy measures with increasing

number of interactions might be the result of increased effect of incorrect assignments in the E step.

Our probabilistic model penalizes interactions with high number of features while assigning them to clusters. This might be one aspect of low performance of the E step. Probabilistic approach might be modified to overcome this problem.

Another approach to improve the results might be by using more sophisticated initial assignments of cluster membership probabilities. Random initialization might lead the algorithm to get stuck at local minima. If we could prepare some seed clusters, EM clustering results might be improved. We might pre-cluster interactions manually according to findings in the literature or we might even start from rule assignments instead of membership assignments. We could also initialize rules according to the results of other data mining approaches such as association rule mining.

Not all domains interact with other domains. Indeed a major group of domains recognize and bind smaller peptide sequences, called protein motifs. Considering motifs as a feature of interactions would allow a much more precise clustering. *InterPro BindingSite* [51] entries and PRINTS fingerprints [52] could be the most suitable data sources for such information.

Clustering can also be improved by including annotations of proteins such as GO terms or Swiss-Prot keywords in feature sets. This might enhance clustering and enrich the information content of the rules.

Miscellaneous other ideas for improvements can be listed as future work as follows:

- New strategies to evaluate real data clusters would be invaluable for the analysis of resulting clusters. Investigation of statistics of clusters might help developing such an evaluation measure.

- New interactions can be classified and annotated according to existing cluster rules.

- Well annotated interactions of clusters can be used to generate detailed interaction mechanism templates of clusters. These templates can be used to derive mechanistic details of other interactions of the same cluster.

- Clustering studies can be extended to other organisms and the rules obtained with *S. cerevisiae* can be used as seed clusters for clustering of interactions of other organisms since the mechanisms are conserved through evolution.

References

[1] G.L. Miklos, G.M. Rubin. The role of the genome project in determining gene function: insights from model organisms. *Cell,* **1996***, 86*, 521-9

[2] G. Karev,Y. Wolf, A. Rzhetsky, F. Berezovskaya, E. Koonin. Birth and death of protein domains: A simple model of evolution explains power law behavior. *BMC Evol Biol,* **2002***, 2*, 18

[3] T. Pawson, G.D. Gish, P. Nash. SH2 domains, interaction modules and cellular wiring. *Trends Cell Biol,* **2001***, 11*, 504-11

[4] T. Pawson, P. Nash. Assembly of cell regulatory systems through protein interaction domains. *Science,* **2003***, 300*, 445-52

[5] A. Grigoriev. On the number of protein-protein interactions in the yeast proteome. *Nucleic Acids Res,* **2003***, 31*, 4157-61

[6] H. Jeong, S.P. Mason, A.L. Barabasi, Z.N. Oltvai. Lethality and centrality in protein networks. *Nature,* **2001***, 411*, 41-2

[7] M. Fromont-Racine, J. Rain, P. Legrain. Toward a functional analysis of the yeast genome through exhaustive twohybrid screens. *Nat. Genet*., **1997**, *16*, 277–282.

[8] http://www.celanphy.sci.kun.nl/Bruce web/slide shows.htm

[9] A.H.Y. Tong, B. Drees, G. Nardelli, G.D. Bader, B. Brannetti, L. Castagnoli, M. Evangelista, S. Ferracuti, B. Nelson, S. Paoluzi, M. Quondam, A. Zucconi, C.W.V. Hogue, S. Fields, C. Boone, G. Cesareni. A combined experimental and

computational strategy to define protein interaction networks for peptide recognition modules. *Science,* **2002***, 295*, 321-4

[10] D.J. Reiss, B. Schwikowski. Predicting protein-peptide interactions via a network-based motif sampler. *Bioinformatics,* **2004***, 20 Suppl 1*, I274-I282

[11] J.B. Fenn, M. Mann, C.K. Meng, S.F. Wong, C.M. Whitehouse. Electrospray ionization for mass spectrometry of large biomolecules. *Science,* **1989***, 246*, 64-71

[12] K. Tanaka, H. Waki, Y. Ido, S. Akita, Y. Yoshida, T. Yoshida. Protein and polymer analyses of up to m/z 100,000 by laser ionization time-of-flight mass spectrometry. *Rapid Commun. Mass Spectrom.***,** *2*, 151-153.

[13] A.C. Gavin, M. Bosche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J.M. Rick, A.M. Michon, C.M. Cruciat, M. Remor, C. Hofert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M.A. Heurtier, R.R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer, G. Superti-Furga. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature,* **2002***, 415*, 141-7

[14] Y. Ho, A. Gruhler, A. Heilbut, G.D. Bader, L. Moore, S.L. Adams, A. Millar, P. Taylor, K. Bennett, K. Boutilier, L. Yang, C. Wolting, I. Donaldson, S. Schandorff, J. Shewnarane, M. Vo, J. Taggart, M. Goudreault, B. Muskat, C. Alfarano, D. Dewar, Z. Lin, K. Michalickova, A.R. Willems, H. Sassi, P.A. Nielsen, K.J. Rasmussen, J.R. Andersen, L.E. Johansen, L. Hansen, H. Jespersen, A. Podtelejnikov, E. Nielsen, J. Crawford, V. Poulsen, B.D. Sorensen, J. Matthiesen, R.C. Hendrickson, F. Gleeson, T. Pawson, M.F. Moran, D. Durocher, M. Mann, C.W.V. Hogue, D. Figeys, M. Tyers. Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry. *Nature,* **2002***, 415*, 180-3

[15] R.J. Deshaies, J.H. Seol, W.H. McDonald, G. Cope, S. Lyapina, A. Shevchenko, A. Shevchenko, R. Verma, J.R. Yates. Charting the protein complexome in yeast by mass spectrometry. *Mol Cell Proteomics,* **2002***, 1*, 3-10

[16] www.rcsb.org/pdb

[17] G.D. Bader, D. Betel, C.W.V. Hogue. BIND: the Biomolecular Interaction Network Database. *Nucleic Acids Res,* **2003***, 31*, 248-50

[18] http://www.blueprint.org/bind

[19] L. Salwinski, C.S. Miller, A.J. Smith, F.K. Pettit, J.U. Bowie, D. Eisenberg. The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res,* **2004***, 32 Database issue*, D449-51

[20] http://dip.doe-mbi.ucla.edu/

[21] http://mips.gsf.de/proj/yeast/CYGD/interaction/

[22] H. Hermjakob, L. Montecchi-Palazzi, C. Lewington, S. Mudali, S. Kerrien, S. Orchard, M. Vingron, B. Roechert, P. Roepstorff, A. Valencia, H. Margalit, J. Armstrong, A. Bairoch, G. Cesareni, D. Sherman, R. Apweiler. IntAct: an open source molecular interaction database. *Nucleic Acids Res,* **2004***, 32 Database issue*, D452-5

[23] B. Boeckmann, A. Bairoch, R. Apweiler, M.C. Blatter, A. Estreicher, E. Gasteiger, M.J. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, M. Schneider. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res,* **2003***, 31*, 365-70

[24] C.H. Wu, L.S.L. Yeh, H. Huang, L. Arminski, J. Castro-Alvear, Y. Chen, Z. Hu, P. Kourtesis, R.S. Ledley, B.E. Suzek, C.R. Vinayaka, J. Zhang, W.C. Barker. The Protein Information Resource. *Nucleic Acids Res,* **2003***, 31*, 345-7

[25] http://www.icp.ucl.ac.be/annrep98/trop/trop6.html

[26] A. Bateman, L. Coin, R. Durbin, R.D. Finn, V. Hollich, S. Griffiths-Jones, A. Khanna, M. Marshall, S. Moxon, E.L.L. Sonnhammer, D.J. Studholme, C. Yeats, S.R. Eddy. The Pfam protein families database. *Nucleic Acids Res,* **2004***, 32 Database issue*, D138-41

[27] M. Ashburner, C.A. Ball, J.A. Blake, D. Botstein, H. Butler, J.M. Cherry, A.P. Davis, K. Dolinski, S.S. Dwight, J.T. Eppig, M.A. Harris, D.P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J.C. Matese, J.E. Richardson, M. Ringwald, G.M.

Rubin, G. Sherlock. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, **2000**, *25*, 25-9

[28] A. Clare, R.D. King. Predicting gene function in Saccharomyces cerevisiae. *Bioinformatics,* **2003***, 19 Suppl 2*, II42-II49

[29] M. Deng, Z. Tu, F. Sun, T. Chen. Mapping Gene Ontology to proteins based on protein-protein interaction data. *Bioinformatics,* **2004***, 20*, 895-902

[30] L.J. Jensen, R. Gupta, H.H. Staerfeldt, S. Brunak. Prediction of human protein function according to Gene Ontology categories. *Bioinformatics,* **2003***, 19*, 635-42

[31] A. Ben-Dor, R. Shamir, Z. Yakhini. Clustering gene expression patterns. *J Comput Biol*, **1999**, *6*, 281-97

[32] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci U S A*, **1998**, *95*, 14863-8

[33] S.G. Lee, J.U. Hur, Y.S. Kim. A graph-theoretic modeling on GO space for biological interpretation of gene clusters. *Bioinformatics,* **2004***, 20*, 381-8

[34] A.L. Barabasi, and Albert, Emergence of scaling in random networks. *Science*, **1999**, *286*, 509-12

[35] J.R. Bock, D.A. Gough. Whole-proteome interaction mining. *Bioinformatics,* **2003**, *19*, 125-34

[36] J. Wojcik, V. Schachter. Protein-protein interaction map inference using interacting domain profile pairs. *Bioinformatics,* **2001***, 17 Suppl 1*, S296-305

[37] B.H. Park , G. Ostrouchov, G.X. Yu, A. Geist, A. Gorin, N.F. Samatova. PICUPP: Protein Interaction Classification by Unlikely Profile Pair Submitted to *IEEE Bioinformatics Conference*. **2003**, (http://www.genomes2life.org/publications/PICUPP.pdf)

[38] N.J. Mulder, R. Apweiler, T.K. Attwood, A. Bairoch, D. Barrell, A. Bateman, D. Binns, M. Biswas, P. Bradley, P. Bork, P. Bucher, R.R. Copley, E. Courcelle, U. Das, R. Durbin, L. Falquet, W. Fleischmann, S. Griffiths-Jones, D. Haft, N. Harte, N. Hulo, D. Kahn, A. Kanapin, M. Krestyaninova, R. Lopez, I. Letunic, D. Lonsdale, V. Silventoinen, S.E. Orchard, M. Pagni, D. Peyruc, C.P. Ponting, J.D. Selengut, F. Servant, C.J.A. Sigrist, R. Vaughan, E.M. Zdobnov. The InterPro

Database, 2003 brings increased coverage and new features. *Nucleic Acids Res,* **2003***, 31*, 315-8

[39] J.G. Henikoff, E.A. Greene, S. Pietrokovski, S. Henikoff. Increased coverage of protein families with the blocks database servers. *Nucleic Acids Res,* **2000***, 28*, 228-30

[40] S.K. Ng, Z. Zhang, S.H. Tan, K. Lin. InterDom: a database of putative interacting protein domains for validating predicted protein interactions and complexes. *Nucleic Acids Res,* **2003***, 31*, 251-4

[41] S.K. Ng, Z. Zhang, S.H. Tan. Integrative approach for computationally inferring protein domain interactions. *Bioinformatics,* **2003***, 19*, 923-9

[42] H. Li, J. Li, S.H. Tan, S.K. Ng. Discovery of binding motif pairs from protein complex structural data and protein interaction sequence data. *Pac Symp Biocomput,* **2004**, 312-23

[43] E. Segal, H. Wang, D. Koller. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics,* **2003***, 19 Suppl 1*, i264-71

[44] T. Oyama, K. Kitano, K. Satou, T. Ito. Extraction of knowledge on protein-protein interaction by association rule discovery. *Bioinformatics,* **2002***, 18*, 705-14

[45] M. Tan, T. Jing, K.H. Lan, C.L. Neal, P. Li, S. Lee, D. Fang, Y. Nagata, J. Liu, R. Arlinghaus, M.C. Hung, D. Yu. Phosphorylation on tyrosine-15 of p34 (Cdc2) by ErbB2 inhibits p34 (Cdc2) activation and is involved in resistance to taxol-induced apoptosis. *Mol Cell*, **2002**, *9*, 993-1004

[46] A.P. Dempster, N.M. Laird, D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Stat. Soc. B*, **1977**, *39*, 1–39

[47] F. Heylighen, C. Joslyn. Cybernetics and Second Order Cybernetics, in: R.A. Meyers (ed.), Encyclopedia of Physical Science & Technology , (Academic Press, New York), **2000,** *4*, 155-170

[48] R.D. Finn, A. Bateman. iPfam: Visualisation of protein-protein interactions at domains and amino acid resolutions (in preparation).

[49] E. Demir, O. Babur, U. Dogrusoz, A. Gursoy, G. Nisanci, R. Cetin-Atalay, M. Ozturk. PATIKA: An integrated visual environment for collaborative construction and analysis of cellular pathways. *Bioinformatics,* **2002***, 18*, 996-1003

[50] E. Demir, O. Babur, U. Dogrusoz, A. Gursoy, A. Ayaz, G. Gulesir, G. Nisanci, R. Cetin-Atalay. An ontology for collaborative construction and analysis of cellular pathways. *Bioinformatics,* **2004***, 20*, 349-56

[51] http://www.ebi.ac.uk/interpro/entryList/Binding_site.html

[52] http://bioinf.man.ac.uk/dbbrowser/PRINTS/

Appendix A Hypothetic Data Clustering

This section provides some example clustering of hypothetic data prepared by different parameters. Parameters denoted as below:

Number of interactions : 2n,

Number of clusters : k,

Number of atributes : 2m

Number of proteins : t

A.1 Example 1

Clustering of hypothetic data prepared with parameters:

2n = 1000, k = 20, 2m = 1000, t = 1000.

	Total Ent	Total Prob Ent
RUN 1	29.859	30.310
RUN 2	32.171	32.861
RUN 3	29.830	30.462
RUN 4	30.865	31.425
RUN 5	31.664	31.872
RUN 6	28.443	28.522
RUN 7	29.852	30.279
RUN 8	30.232	30.373
RUN 9	27.960	28.080
RUN 10	29.590	30.396

Table 9: Total entropies for each run. 9th run has the lowest entropy.

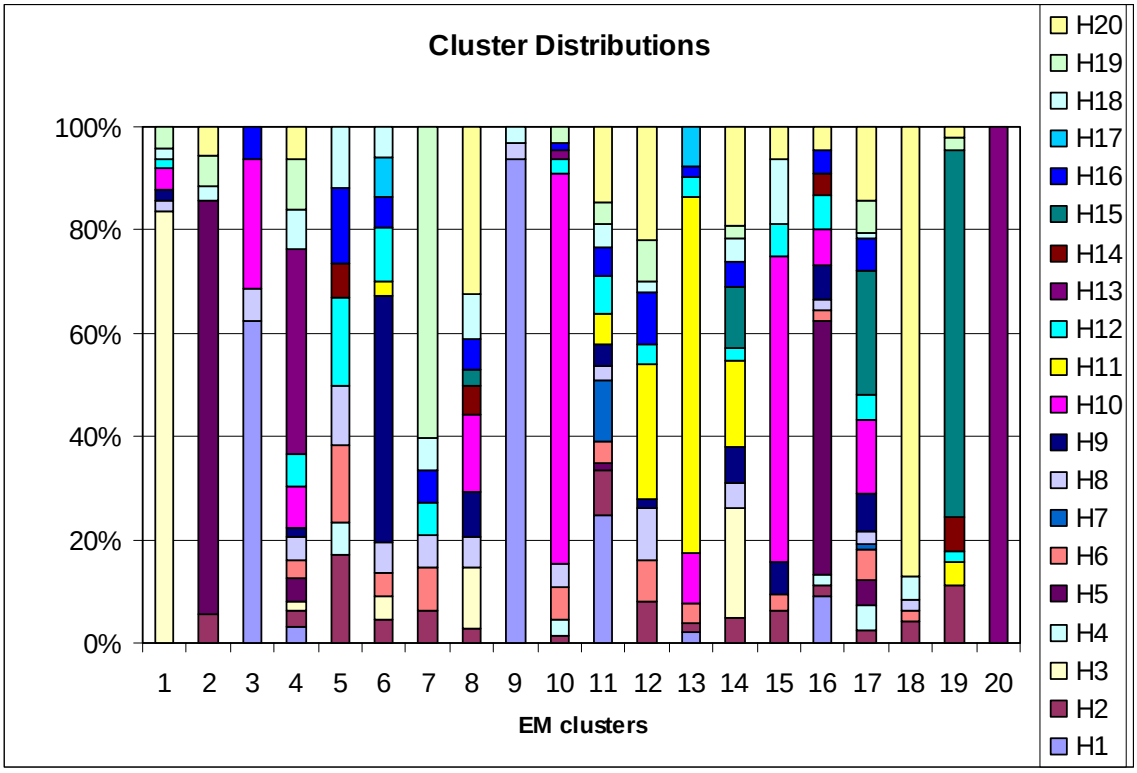

Figure 14: Cluster distributions.

	Stat Ent	Prob Ent
EM 1	0.728	0.792
EM 2	0.771	0.853
EM 3	0.987	1.020
EM 4	2.093	2.055
EM 5	2.023	2.028
EM 6	1.809	1.834
EM 7	1.378	1.388
EM 8	2.035	2.022
EM 9	0.277	0.383
EM 10	1.041	1.071
EM 11	2.317	2.252
EM 12	2.035	2.073
EM 13	1.156	1.129
EM 14	2.144	2.121
EM 15	1.371	1.273
EM 16	1.860	1.879
EM 17	2.324	2.318
EM 18	0.552	0.489
EM 19	1.059	1.100
EM 20	0.000	0.000

Table 10: Entropy measures of clusters.

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20	Total
EM1	0	0	41	0	0	0	0	1	1	2	0	1	0	0	0	0	0	1	2	0	**49**
EM2	0	2	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	**35**
EM3	10	0	0	0	0	0	0	1	0	4	0	0	0	0	0	1	0	0	0	0	**16**
EM4	2	2	1	0	3	2	0	3	1	5	0	4	25	0	0	0	0	5	6	4	**63**
EM5	0	16	0	6	0	14	0	11	0	0	0	16	0	6	0	14	0	11	0	0	**94**
EM6	0	3	3	0	0	3	0	4	32	0	2	7	0	0	0	4	5	4	0	0	**67**
EM7	0	3	0	0	0	4	0	3	0	0	0	3	0	0	0	3	0	3	29	0	**48**
EM8	0	1	4	0	0	0	0	2	3	5	0	0	0	2	1	2	0	3	0	11	**34**
EM9	30	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	**32**
EM10	0	1	0	2	0	4	0	3	0	49	0	2	1	0	0	1	0	0	2	0	**65**
EM11	17	6	0	0	1	3	8	2	3	0	4	5	0	0	0	4	0	3	3	10	**69**
EM12	0	4	0	0	0	4	0	5	1	0	13	2	0	0	0	5	0	1	4	11	**50**
EM13	1	1	0	0	0	2	0	0	0	5	36	2	0	0	0	1	4	0	0	0	**52**
EM14	0	2	9	0	0	0	0	2	3	0	7	1	0	0	5	2	0	2	1	8	**42**
EM15	0	2	0	0	0	1	0	0	2	19	0	2	0	0	0	0	0	4	0	2	**32**
EM16	4	1	0	1	22	1	0	1	3	3	0	3	0	2	0	2	0	0	0	2	**45**
EM17	0	2	0	4	4	5	1	2	6	12	0	4	0	0	20	5	0	1	5	12	**83**
EM18	0	2	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	2	0	41	**47**
EM19	0	5	0	0	0	0	0	0	0	0	2	1	0	3	32	0	0	0	1	1	**45**
EM20	0	0	0	0	0	0	0	0	0	0	0	0	32	0	0	0	0	0	0	0	**32**
total	**64**	**53**	**58**	**13**	**58**	**44**	**9**	**42**	**55**	**104**	**64**	**53**	**58**	**13**	**58**	**44**	**9**	**42**	**55**	**104**	**1000**

Table 11: Distributions of members of clusters.

H cluster	EM cluster	Rule Domains	H weight	EM weight
1	9	360	1	0.92
		442	1	0.92
		123	1	0.96
		155	1	0.96
		232	1	0.96
3	1	38	1	0.95
		211	1	0.98
		422	1	0.95
		3	1	0.87
		138	1	0.87
		284	1	0.87
		406	1	0.87
5	2,16	193	1	0.86 - 0.54
		337	1	0.86 - 0.54
		431	1	0.86 - 0.54
		299	1	0.91 - 0.98
		326	1	0.91 - 0.98
		387	1	0.91 - 0.98
		436	1	0.91 - 0.98
9	6	211	1	0.77
		423	1	0.64
		91	1	0.6
		434	1	0.6
10	10,15	105	1	0.82 - 0.75
		179	1	0.98 - 0.91
		197	1	0.98 - 0.91
		357	1	0.98 - 0.91
11	13	123	1	0.98
		155	1	0.98
		232	1	0.98
		360	1	0.7
		442	1	0.7
13	4,20	3	1	1 - 1
		138	1	1 -1
		284	1	1 - 1
		406	1	1 - 1
		38	1	0 - 1
		211	1	0.57 - 1
		422	1	0 - 1
15	17,19	299	1	0.55 - 0.89
		326	1	0.55 - 0.89
		387	1	0.55 - 0.89
		436	1	0.55 - 0.89
		193	1	0.57 - 0.84
		337	1	0.57 - 0.84
		431	1	0.57 - 0.84
19	7	91	1	0.74
		434	1	0.74
		211	1	0.86
		423	1	0.86
20		179	1	0.84 - 1
		197	1	0.84 - 1
		357	1	0.84 - 1
		105	1	0.96 - 0.84

Table 12: Matching hypothetic and EM cluster rules.

A.2 Example 2

Clustering of hypothetic data prepared with parameters:

2n = 1000, k = 20, 2m = 1000, t = 2500.

We increased the number of proteins in a ratio same with the 400 interactions and 1000 proteins to see the how the clustering is effected when we increased the number of interactions and the number of proetins while keeping the ratio same.

	Total Ent	Total Ent	Prob
RUN 1	21.826	22.043	
RUN 2	20.251	20.430	
RUN 3	18.810	18.979	
RUN 4	20.756	20.741	
RUN 5	22.606	22.927	
RUN 6	19.925	20.199	
RUN 7	18.787	19.061	
RUN 8	20.673	20.912	
RUN 9	19.394	19.769	
RUN 10	19.634	19.970	

Table 13: Total entropies

	Stat Ent	Prob Ent
EM 1	2.051	2.035
EM 2	1.876	1.838
EM 3	1.097	1.097
EM 4	0.445	0.450
EM 5	1.034	1.040
EM 6	1.655	1.621
EM 7	1.032	1.019
EM 8	0.534	0.551
EM 9	0.399	0.432
EM 10	1.119	1.130
EM 11	0.812	0.875
EM 12	0.391	0.409
EM 13	0.602	0.631
EM 14	0.107	0.166
EM 15	0.445	0.404
EM 16	1.824	1.824
EM 17	0.682	0.714
EM 18	1.273	1.286
EM 19	0.646	0.648
EM 20	0.787	0.808

Table 14: Entropy values of each cluster of run 3.

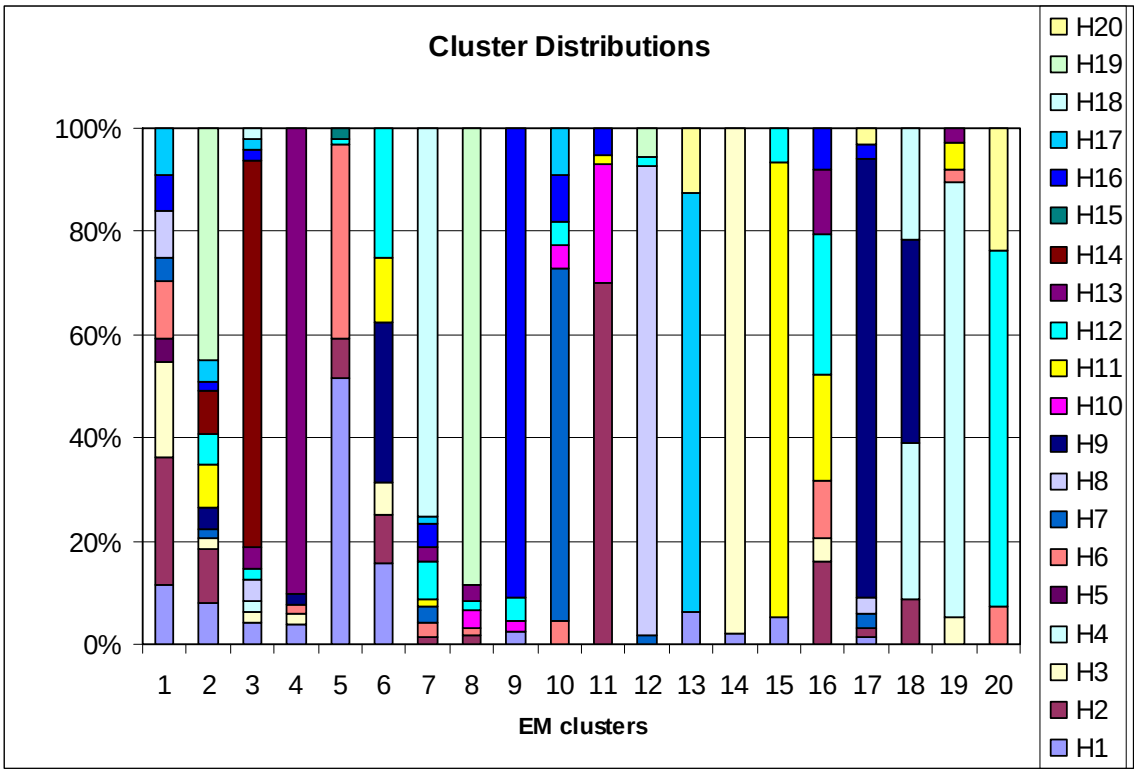

Figure 15: Cluster distributions.

	H1	H2	H3	H4	H5	H6	H7	H8	H9	H10	H11	H12	H13	H14	H15	H16	H17	H18	H19	H20	Total
EM1	5	11	8	0	2	5	2	4	0	0	0	0	0	0	0	3	4	0	0	0	**44**
EM2	4	5	1	0	0	0	1	0	2	0	4	3	0	4	0	1	2	0	22	0	**49**
EM3	2	0	1	1	0	0	0	2	0	0	0	1	2	36	0	1	1	1	0	0	**48**
EM4	2	0	1	0	0	1	0	0	1	0	0	0	47	0	0	0	0	0	0	0	**52**
EM5	52	8	0	0	0	38	0	0	0	0	0	1	0	0	2	0	0	0	0	0	**101**
EM6	5	3	2	0	0	0	0	0	10	0	4	8	0	0	0	0	0	0	0	0	**32**
EM7	0	1	0	0	0	2	2	0	0	0	1	5	2	0	0	3	1	52	0	0	**69**
EM8	0	1	0	0	0	1	0	0	0	2	0	1	2	0	0	0	0	0	54	0	**61**
EM9	1	0	0	0	0	0	0	0	0	1	0	2	0	0	0	40	0	0	0	0	**44**
EM10	0	0	0	0	0	1	15	0	0	1	0	1	0	0	0	2	2	0	0	0	**22**
EM11	0	40	0	0	0	0	0	0	0	13	1	0	0	0	0	3	0	0	0	0	**57**
EM12	0	0	0	0	0	0	1	50	0	0	0	1	0	0	0	0	0	0	3	0	**55**
EM13	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	2	**16**
EM14	1	0	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**45**
EM15	3	0	0	0	0	0	0	0	0	0	52	4	0	0	0	0	0	0	0	0	**59**
EM16	0	10	3	0	0	7	0	0	0	0	13	17	8	0	0	5	0	0	0	0	**63**
EM17	1	1	0	0	0	0	2	2	57	0	0	0	0	0	0	2	0	0	0	2	**67**
EM18	0	2	0	7	0	0	0	0	9	0	0	0	0	0	0	0	0	5	0	0	**23**
EM19	0	0	2	32	0	1	0	0	0	0	2	0	1	0	0	0	0	0	0	0	**38**
EM20	0	0	0	0	0	4	0	0	0	0	0	38	0	0	0	0	0	0	0	13	**55**
total	**77**	**82**	**62**	**40**	**2**	**60**	**23**	**58**	**79**	**17**	**77**	**82**	**62**	**40**	**2**	**60**	**23**	**58**	**79**	**17**	**1000**

Table 15: Distributions of members of clusters.

H cluster	EM cluster	Rule Domains	H weight	EM weight
1	5	6	1	0.61
		24	1	0.61
		196	1	0.53
2	11	103	1	0.73
		200	1	0.73
		242	1	0.7
3	14	189	1	1
		204	1	1
		347	1	1
		92	1	0.97
		340	1	0.97
		474	1	0.97
4	19	26	1	1
		124	1	1
		206	1	1
		318	1	1
		379	1	1
		113	1	0.84
		350	1	0.84
7		190	1	0.77
		329	1	0.77
		32	1	0.81
		277	1	0.81
8	12	165	1	0.92
		497	1	0.92
		28	1	0.98
		60	1	0.98
		335	1	0.98
		458	1	0.98
9	17	25	1	0.94
		90	1	0.94
		342	1	0.94
		391	1	0.94
		31	1	0.9
		144	1	0.9
		402	1	0.9
11	15	196	1	0.9
		6	1	1
		24	1	1
12	20	242		0.71
		103	1	0.72
		200	1	0.72
13	4	92	1	0.98
		340	1	0.98
		474	1	0.98
		189	1	0.92
		204	1	0.92
		347	1	0.92
14	3	113	1	0.75
		350	1	0.75
		26	1	0.98
		124	1	0.98
		206	1	0.98
		318	1	0.98
		379	1	0.98
16	9	67	1	0.97
		442	1	0.97
		116	1	0.9
17	13	32	1	0.86
		277	1	0.86
		190	1	0.8
		329	1	0.8
18		28	1	0.99
		60	1	0.99
		335	1	0.99
		458	1	0.99
		165	1	0.78
		497	1	0.78
19		31	1	0.88
		144	1	0.88
		402	1	0.88
		25	1	1
		90	1	1
		342	1	1
		391	1	1

Table 16: Matching hypothetic and EM cluster rules.