# Exploring the Function and Evolution of Proteins Using Domain Families

Adam James Reid

Research Department of Structural and Molecular Biology
University College London

A thesis submitted to University College London for the degree of Doctor of Philosophy

January 2009

# Declaration

I, Adam James Reid, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Adam James Reid

January 2009

# Abstract

Proteins are frequently composed of multiple domains which fold independently. These are often evolutionarily distinct units which can be adapted and reused in other proteins. The classification of protein domains into evolutionary families facilitates the study of their evolution and function. In this thesis such classifications are used firstly to examine methods for identifying evolutionary relationships (homology) between protein domains. Secondly a specific approach for predicting their function is developed. Lastly they are used in studying the evolution of protein complexes.

Tools for identifying evolutionary relationships between proteins are central to computational biology. They aid in classifying families of proteins, giving clues about the function of proteins and the study of molecular evolution. The first chapter of this thesis concerns the effectiveness of cutting edge methods in identifying evolutionary relationships between protein domains.

The identification of evolutionary relationships between proteins can give clues as to their function. The second chapter of this thesis concerns the development of a method to identify proteins involved in the same biological process. This method is based on the concept of domain fusion whereby pairs of proteins from one organism with a concerted function are sometimes found fused into single proteins in a different organism. Using protein domain classifications it is possible to identify these relationships.

Most proteins do not act in isolation but carry out their function by binding to other proteins in complexes; little is understood about the evolution of such complexes. In the third chapter of this thesis the evolution of complexes is examined in two representative model organisms using protein domain families. In this work, protein domain superfamilies allow distantly related parts of complexes to be identified in order to determine how homologous units are reused.

This work was generously supported by the Biotechnology and Biological Sciences Research Council.

# Acknowledgements

Firstly, many, many thanks to Professor Christine Orengo for allowing me to join the group, guiding me through my PhD and helping me to produce a good body of work. It has been an honour to work in such a prestigious group.

Thanks go to Orengo and Martin group members past and present who have helped me out with problems of all sorts and provided an excellent working environment: Mark Dibley, Oliver Redfern, Timothy Dallman, Ian Sillitoe, Sarah Addou, Michael Maibaum, Russell Marsden, Tony Lewis, Alison Cuff, Andy Clegg, Jon Lees, Benoit Dessailly, David Lee, Stathis Sideris, Stephano Lise, Phil Carter, Robert Rentzsch, James Perkins, Lisa McMillan, Abhinandan Raghavan, Anya Baresic, Jakob Hurst and Ilhem Diboun. Thanks to Michael Wright for his top notch administration. Many thanks to Jesse Oldershaw, Tom Knight, Jahid Ahmed, Donovan Binns and Duncan McKenzie, members of the IT team past and present for keeping our farms and other machines running.

Special thanks to Ian Sillitoe for, amongst other things, his dedication to CATH and his awesome script for generating figures based on SSAP alignments. Tim Dallman for inviting me to join his awesome band (Party in Hiroshima!). Oliver Redfern for long, interesting chats about science and helping revise my thesis. Corin Yeats for important early guidance in my work. Many thanks to Juan Antonio Garcia Ranea for a great deal of advice and guidance on analyses, statistics and biology generally.

Thanks to David Jones, chair of my thesis committee and Andrew Martin, member of my thesis committee for guidance on my work.

Many thanks and kind regards to Alison Cranage for helping me through from start to finish, when it was fun and easy and when it got really hard.

Unending thanks to my parents Martyn and Teresa Reid for giving me an amazing start and funding me well into my twenties.

To anyone I may have forgotten, I apologise.

A final thanks again to the BBSRC for funding this work and continuing to fund basic life sciences research in the UK.

# List of Abbreviations

| | |
|---|---|
| BDBH | Bi-Directional Best Hit |
| BLAST | Basic Local Alignment Search Tool |
| BLOSUM | BLOcks of Amino Acid SUbstitution Matrix |
| CATH | Class Architecture Topology Homology |
| CODA | Co-Occurrence of Domains Analysis |
| COGS | Cluster of Orthologous Groups |
| COMPASS | COmparison of Multiple Protein sequence Alignments with assessment of Statistical Significance |
| CSA | Catalytic Site Atlas |
| DAG | Directed Acyclic Graph |
| DDI | Domain-Domain Interaction |
| DFT | Domain Fusion Template |
| DNA | DeoxyriboNucleic Acid |
| EC | Enzyme Commission |
| EM | Expectation Maximisation |
| EPQ | Errors Per Query |
| EVD | Extreme Value Distribution |
| FDR | False Discovery Rate |
| FP | False Positive |
| FSSP | Fold classification based on Structure-Structure alignment of Proteins |
| GO | Gene Ontology |
| GOSS | GO Semantic Similarity |
| GTD | Genomic Threading Database |
| HMM | Hidden Markov Model |
| IPI | International Protein Index |

| | |
|---|---|
| LAMA | Local Alignment of Multiple-Alignments |
| MDA | Multi-Domain Architecture |
| N-W | Needleman-Wunsch |
| OPHID | Online Predicted Human Interactions Database |
| PAM | Percent Accepted Mutation |
| PDB | Protein DataBank |
| PFP | Protein Function Prediction |
| PIN | Protein Interaction Network |
| PPI | Protein-Protein Interaction |
| PPV | Positive Predictive Value |
| PQS | Protein Quaternary Structure |
| PRC | PRofile Comparer |
| PSI-BLAST | Position-Specific Iterated BLAST |
| PSSM | Position Specific Scoring Matrix |
| PTM | Post-Translational Modification |
| RMSD | Root Mean Square Deviation |
| SAM | Sequence Alignment and Modelling system |
| SAS | Structural Alignment Score |
| SCOP | Structural Classification Of Proteins |
| SMART | Simple Modular Architecture Research Tool |
| SSAP | Sequential Structure Alignment Program |
| STRING | Search Tool for the Retrieval of INteracting Genes/proteins |
| SVM | Support Vector Machine |
| S-W | Smith-Waterman |
| SYSTERS | SYSTEmatic Re-Searching |
| TAP-MS | Tandem Affinity Purification linked to Mass Spectrometry |
| TIGR | The Institute for Genome Research |
| TP | True Positive |

| | |
|---|---|
| Y2H | Yeast Two-Hybrid |

# Table of Contents

# List of Figures

# List of Tables

# List of Equations

# Chapter 1 Introduction

## 1.1. Molecular Biology as an Information Science

The principle common factor of all living beings is a nucleic acid genome. Excepting some viruses, this is always DeoxyriboNucleic Acid (DNA). The genome contains the hereditary information which is passed between generations and describes how organisms are to be built and maintained. In theory an organism can be entirely described based on its genome and environmental background. Genes are units of the genome and most of these encode proteins, the principal effectors of the genetic program. It was determined in the last century that genes follow a three letter code, where three nucleic acids are interpreted as a single amino acid, the smallest subunit of proteins (CRICK et al., 2009). While there are four different subunits (the nucleic acids guanine, cytosine, thymine, adenine) used in DNA, there are 20 subunits used in proteins. Of the 64 possible 3-letter codes in DNA, several alternative triplets usually correspond to the same amino acid and some also provide meta-data, defining the beginning and end of genes. Other, longer codes also exist in regions of DNA which do not encode proteins, but are related to the activity of genes. There are signals for which parts of the gene are used to build proteins (alternative splicing), how many copies are produced (expression levels) and whether the gene is accessible (DNA packaging).

The DNA code is not directly transliterated into the protein code. DNA is first transcribed into RNA (Figure 1.1). There are many types of RNA with different functions, but it is messenger RNA (mRNA) that is transcribed from genes and subsequently translated into protein. The RNA code is essentially

the same as the DNA code, except that the nucleic acid uracil is substituted for thymine.

The sequence of amino acids in a protein is thought to specify both its structure and function, at least in the context of the cell. Thus the sequence of nucleic acids in genes also specifies the structure and function of proteins. This concept is at the heart of biology as an information science.

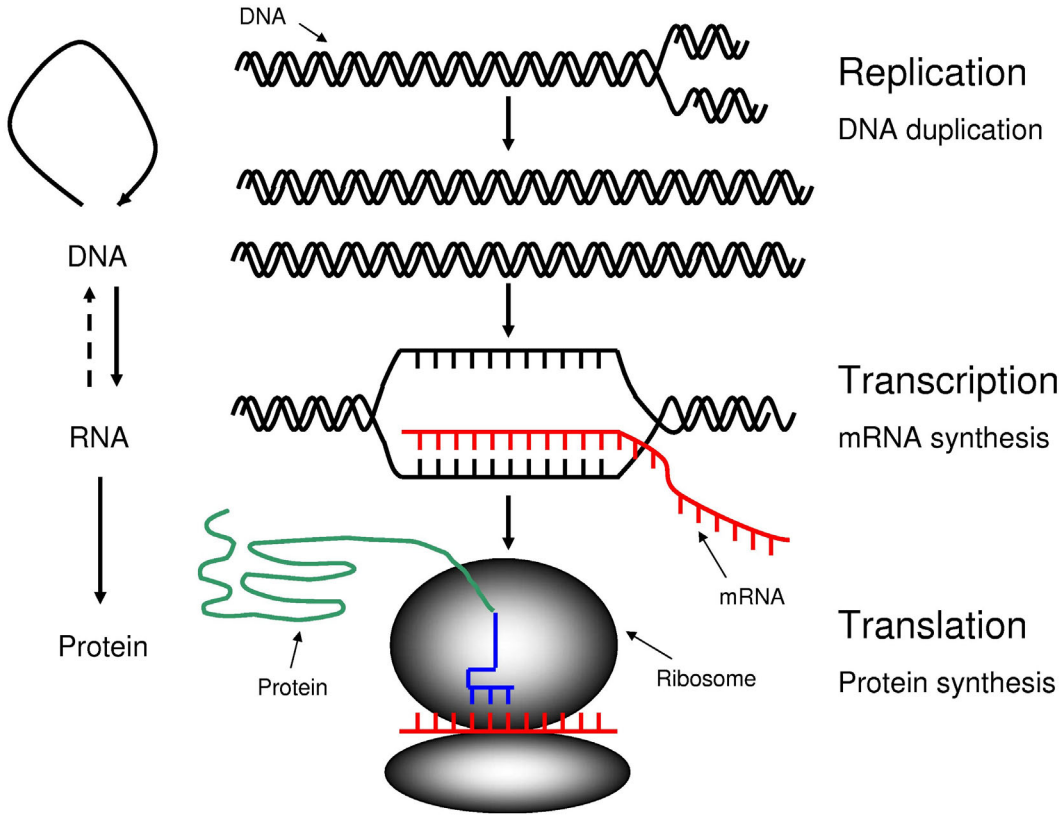



Figure 1.1 The central dogma of molecular biology.

Advancements in DNA sequencing beginning with the Sanger method (Sanger et al., 1977) have, in recent years, allowed the entire genomes of many organisms have been sequenced. The first cellular organism sequenced was *Haemophilus influenzae* in 1995 (Fleischmann et al., 1995) with various viruses having been sequenced prior to this. There are now over 700 completed genomes (664 bacteria, 53 archaea, and 62 eukaryotes according to http://www.ebi.ac.uk/integr8). It is relatively trivial to determine where the protein-coding genes are in bacterial genomes and although less so, certainly possible in eukaryotic genomes. The genome thus contains sufficient information to determine much of the proteome (all the proteins encoded by the genome).

Defining variations in the proteins which come from single genes due to alternative splicing has proved harder (Tress et al., 2007). Genes and proteins can also be identified using mRNA, the mass sequencing of which allows us to determine those regions of the genome which are transcribed. The determination of mRNA expression levels also provides a clue as to how much of a gene product is required by the cell. Furthermore, genes that have similar expression patterns tend to have similar roles in the cell (Page et al., 2007). There is much data on the three-dimensional structure of proteins, which contains detailed information of their function. Increasingly there is a focus on determining which proteins interact and several high-throughput approaches allow us to observe this on a large scale (Walhout and Vidal, 2001). This has become important due to an increasing awareness that the number and variety of genes in an organism is not sufficient to explain its biological complexity (Szathmary et al., 2001).

This wealth of data has brought us to a far greater understanding of the complexity that leads to functioning organisms. Figure 1.3 Shows the increase in nucleotide sequence, protein sequence and protein structure data over the last 20 years. It is clear that computational biology will only increase in importance as more and more data becomes available. A greater integration between computational and experimental biology is also

necessary in order that appropriate data is generated, hypotheses can be formed from *in silico* analyses and then tested experimentally.

## 1.2. Proteins

This thesis is principally concerned with proteins. Although the information contained in a protein sequence is also in the gene sequence, protein sequences are more useful for many studies. Proteins, with 20 elements to the code rather than the four present in DNA, are more useful for recognising distant similarities between genes which are related in their evolutionary history and function. As gene sequences diverge the similarities more quickly become no different than expected by chance, whereas protein sequences retain meaning due to redundancy in the genetic code and functional equivalence due to overlapping properties between amino acids (Figure 1.4). Thus functional information is more accessible in the protein sequence.

Protein structure can be described at several levels (Figure 1.6). The primary structure or sequence is simply the order of amino acids from the amino (N) terminus to the carboxyl (C) terminus. As primary structure consists of a limited range of amino acids, so secondary structure consists of a limited range of forms. Protein chains primarily fold into either alpha helices or beta strands, connected by random coil (or loop) regions. Additional structures include beta-turns, $3_{10}$ helices and π-helices. Alpha helices may wind around each other to form coiled-coils and beta strands may line up to form beta sheets (in parallel or anti-parallel orientations).

The tertiary structure of a protein is the three-dimensional (3D) arrangement of secondary structures and is often termed the fold. The quaternary structure of a protein takes into account multiple chains. Chains may interact in order to form stable complexes or for example, one protein may chemically modify another.

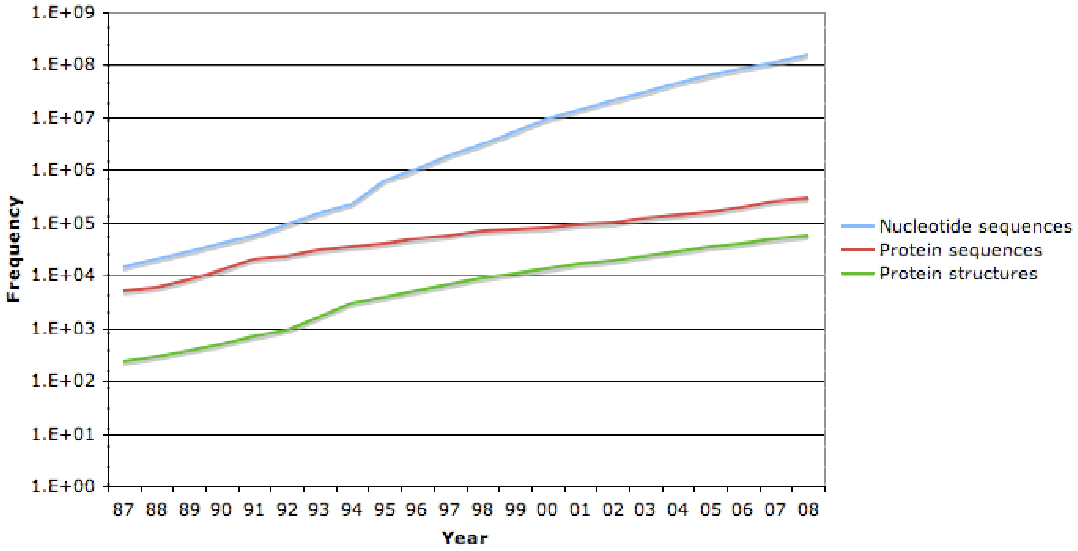



Figure 1.3 Growth of biological data.

Nucleotide sequence numbers were taken from European Molecular Biology Laboratory nucleotide database, protein sequence numbers from UniProt and protein structure numbers from Protein DataBank.

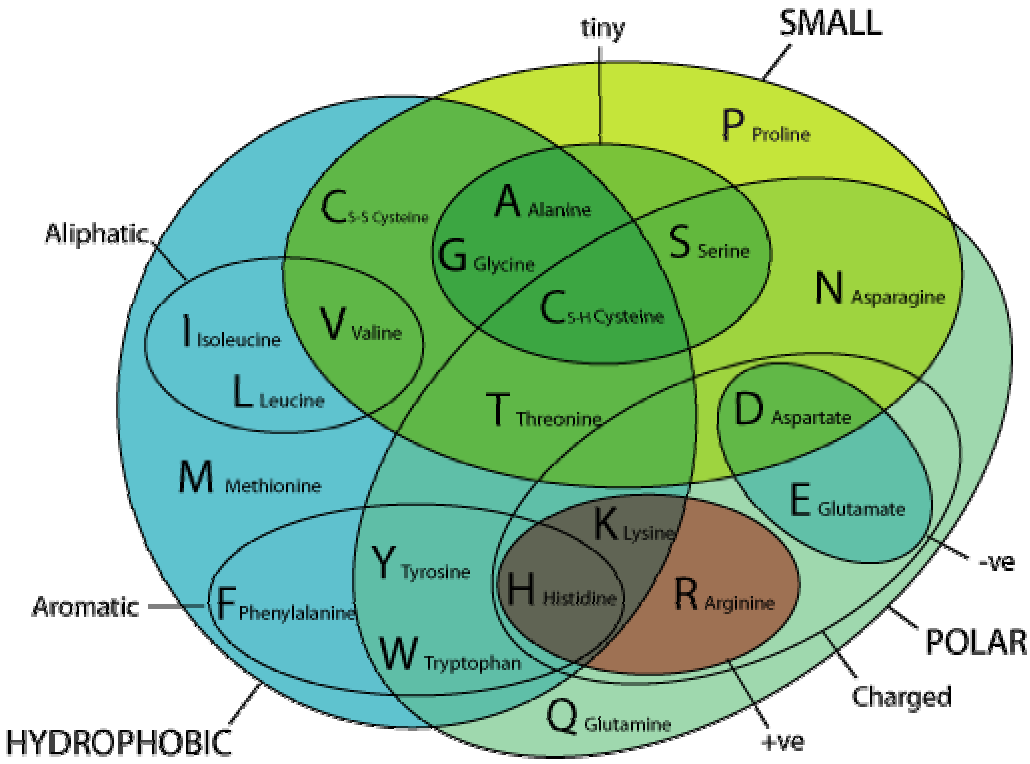



Figure 1.4 Properties of the amino acids (Taylor, 1986).

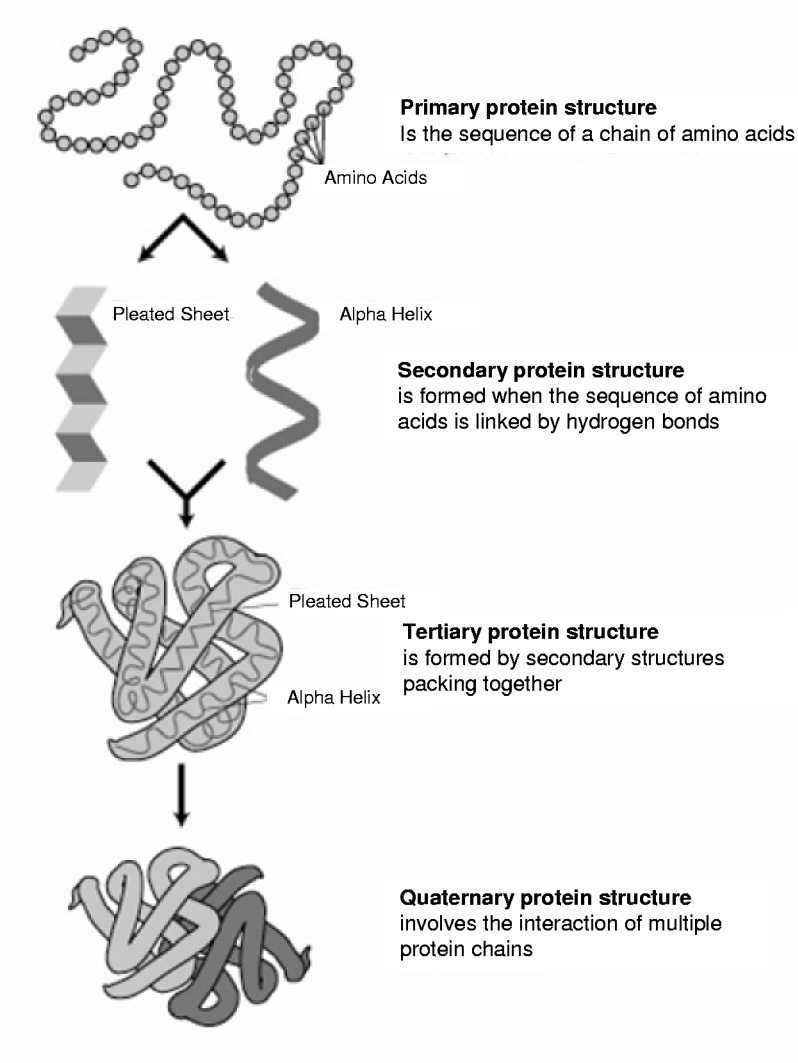



Figure 1.6 The four levels of protein structure.

This image was taken from the National Human Genome Research Institute's Talking Glossary of Genetics.

The tertiary structure of a protein can be divided into domains. Domains are distinct evolutionary units of tertiary structure and often assume distinct, independently folding units (Orengo and Thornton, 2005). They can perform distinct elements of a protein's function such as ligand binding or a particular catalytic step. There are exceptions however, such as where an active site occurs between two domains. Up to 80% of proteins in eukaryotes and 60% in prokaryotes are predicted to consist of multiple domains (Apic et al., 2001b).

It is thought that the amino acid sequence of a protein determines the tertiary structure and that the tertiary structure in turn determines the protein's function. However due to the complexity of this relationship no principle has yet been discovered to accurately predict tertiary structure or function from the amino acid sequence alone.

### 1.2.1. Protein Sequence Resources

Protein sequence data is available from several resources. UniProtKB is an extensive resource which resulted from the integration of three pre-existing resources, Swiss-Prot, TrEMBL and the Protein Information Resource (PIR) (The UniProt Consortium, 2008). UniProtKB consists of two principal parts. The part derived from Swiss-Prot contains manually-curated records with highly accurate protein sequences. The part derived from TrEMBL contains theoretical translations of gene sequences from the EMBL nucleotide sequence database (Kulikova et al., 2007). RefSeq (Pruitt et al., 2005) also provides a non-redundant set of protein sequences from diverse organisms, however fewer of its records are curated than for UniProtKB.

### 1.2.2. Protein Structure Resources

The principal source of data on protein structures in the Protein DataBank (PDB; Berman et al., 2007). Like UniProtKB this resource is a consortium of several other databases seeking to standardise quality and distribution of

data. Structures derived from X-ray crystallography and NMR experiments are routinely deposited in the PDB.

## 1.3. Detecting Evolutionary Relationships between Proteins

### 1.3.1. Homology, Orthology and Paralogy

There are many examples of genes which are clearly related by duplication from a common ancestor. However, for most pairs of genes, there is no evidence that they are related. It is not clear how many times genes have evolved independently and we expect to have lost any observable similarity between the most distantly related genes. Gene duplication is a common process shaping genomes and has been especially frequent in the larger eukaryotic genomes (Ohno, 1970). Genes resulting from the duplication of a common ancestral gene are termed homologues. Orthologues are homologues in different species, which derive from a single gene in the common ancestor of those species. Orthologues often share the same function in different species, although this is not always the case. Paralogues are homologues which derive from a gene duplication within a genome, rather than through speciation (Fitch, 1970). Analogous proteins have similar sequences or structures but they are not derived from a common ancestor, their similarity arising by convergent evolution. It is unclear how common analogous proteins are (Krishna and Grishin, 2004).

Homologous proteins are more likely to share similar structural and functional properties than non-homologues. Therefore, recognising homology between genes or proteins allows us to infer common structural and functional properties. The result of this is that given a well characterised protein, the properties of its homologues can be predicted in many cases.

### 1.3.2. Homologue Detection

The inference that two protein domains share a common evolutionary ancestor is one line of evidence for shared function. Such evidence allows the function of well characterised proteins to be inherited to proteins of unknown function. The ability to detect similarity between proteins and infer

homology is central to bioinformatics. In general, homology is inferred by detecting degrees of similarity between proteins either in their sequence or structure. Similarity scores of proteins known to be related and those known to be unrelated are determined and a score cut-off is chosen to maximise the separation between the two groups. Homology or the lack of it can then be inferred where relationships are not known.

The key factors influencing the development of homology detection methods have been their ability to distinguish homologues from non-homologues, their sensitivity to accurately detect more distant homologues and the need to increase the speed of algorithms to cope with an increasing volume of data.

In general, sequence-based homologue detection methods involve the alignment of a sequence to each of a library of sequences from within which one wishes to identify the likely homologues. The alignment of gene and protein sequences is based on the idea that duplicated genes diverge by substitution, deletion and insertion of nucleic or amino acids. Each aligned residue represents an evolutionarily conserved position, between which there may be gaps representing indels (either insertions or deletions). The sequence identity between two sequences may be calculated by determining the percentage of identical, aligned residues. A more sophisticated scoring function is also used to determine the similarity of aligned sequences. The score is increased where there are conserved residues or where substitutions are for amino acids with similar properties and penalised where there are substitutions for dissimilar amino acids or gaps in the alignment. A score cut-off can be determined by benchmarking the method so that on one side of the cut-off one can say that two sequences are probably homologous and on the other side probably not homologous, with a known error. The major approaches to aligning sequences and scoring the similarity between them are discussed below in sections 1.3.3 to 1.3.6. Section 1.3.7 describes how protein structures can be used to determine homologous relationships. Firstly some background concepts are introduced.

### *1.3.2.1. Substitution Matrices*

Some mutations between amino acids are more favourable than others. That is, properties of structure and function are more conserved when an amino acid is substituted for a similar amino acid rather than one with very different properties. This premise is very useful when scoring sequence alignments. Two amino acids with similar properties in equivalent positions of two aligned proteins ought to receive a better score than two such amino acids with very different properties. For example, lysine and arginine are both polar and positively charged (see Figure 2). A mutation between these is therefore less likely to affect stability or function of a protein than a mutation between two less similar amino acids. Substitution matrices consist of a score (positive or negative) between each pair of amino acids.

Point Accepted Mutation (PAM) substitution matrices are based on empirically observed substitutions. Margaret Dayhoff and co-workers (M.O.Dayhoff et al., 1978) generated alignments of close evolutionary relatives (>85% sequence identity) and calculated the frequency of substitutions between equivalent residues. The probabilities of each substitution were normalised to an evolutionary rate of 1 mutation every 100 residues (PAM1). The matrix can easily be transformed to represent other evolutionary rates to account for expected mutation rate and time of divergence.

BLOcks Substitution Matrices (BLOSUM) were generated from regions of locally aligned sequences in the BLOCKS database (Henikoff and Henikoff, 1992). Proteins, clustered at different sequence identities, were used to calculate substitution rates representing different evolutionary distances. The BLOSUM50 matrix uses clusters at 50% sequence identity for instance. These matrices have been shown to perform better in detecting homologous proteins than PAM matrices (Henikoff and Henikoff, 1993).

### *1.3.2.2. E-values*

E-values are used by many sequence comparison methods when searching a query sequence against a database to find homologues. They give an

estimate of the number of errors to be expected for a particular score. An E-value of one for a match between a model and a sequence means that one random match should be expected among sequences with that score or better in a database of a certain size. The E-value is dependent on database size because in a large database one would expect more high scoring random matches than in a small one. E-values are calculated using the observation that the scores of random matches produced by sequence comparison methods approximate an Extreme Value Distribution (EVD; Durbin et al., 1998). For ungapped alignments it is understood precisely how scores follow this distribution, but for gapped alignments it is necessary to fit empirical data to an EVD (Altschul et al., 1997). The E-value formula is shown in Equation 1.1.

$$E = Kmne^{-\lambda S}$$

Equation 1.1 E-value formula

In Equation 1.1, $m$ is the length of the query sequence, $n$ is the combined length of the sequences in the database and $S$ is the score of the match. $K$ and $\lambda$ are parameters of the EVD to which random matches have been fitted for a particular sequence comparison method.

### 1.3.3. Single Sequence Methods

The first methods for homology detection were based on the comparison of pairs of sequences with dynamic programming and are still in frequent use today. Needleman and Wunsch (1970) produced the first of these methods which compares two amino acid sequences from end to end. Necessarily aligning two sequences from end to end is termed global alignment. In 1982 Gotoh (1982) introduced a more efficient version, which is more commonly used, although it is still referred to as the Needleman & Wunsch (NW)

algorithm. A substitution matrix is used to assign scores for each pair of residues between the sequences.

Smith and Waterman (1981) extended the NW method by allowing local alignment between sequences. Rather than force sequences to align globally this produces the highest scoring alignment amongst subsequences of two proteins.

Dynamic programming algorithms are widely used in bioinformatics as they can be used to efficiently find an optimal solution to alignment problems. Here dynamic programming is described in detail in terms of the alignment of two protein sequences using the NW algorithm and pictorially in Figure 1.8.

For two protein sequences A and B, a matrix is constructed such that each element relates to a pair of residues *i,j* where $i \in$ A and $j \in$ B. The matrix is populated with values from a substitution matrix, representing the likelihood of mutation between residues *i,j*. Another matrix of equal dimensions is created and beginning in the bottom right-hand corner the elements are populated using the function S(*i, j*). This function uses the score from the first matrix for that element added to the maximum of either the element (*i*+1, *j*+1), (*i*, *j*+1) or (*i*+1, *j*). The first of these terms represents an alignment of residues *i* and *j*, whereas the last two terms represent a gap in the alignment and a gap penalty G is used to reduce the score. Once this matrix has been populated, the highest scoring, optimal path is determined in the traceback step.

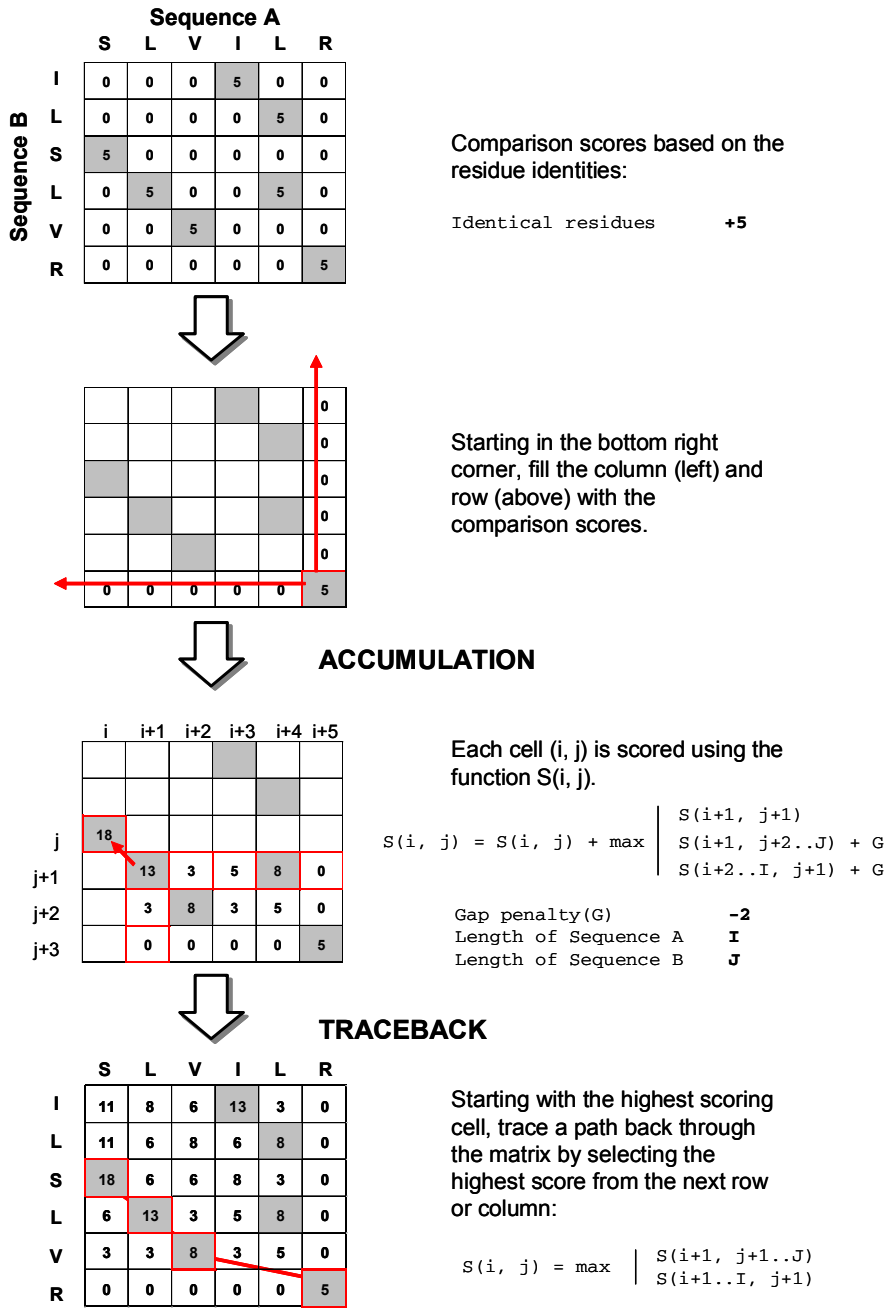



Figure 1.8 The Needleman-Wunsch dynamic programming algorithm.

Each residue pair in sequences A and B is scored for similarity and these scores are used to populate a matrix. For simplicity in this example, a score of +5 is given to identical residues, rather than using a substitution matrix. The accumulation step populates another matrix using the

function *S(i,j)*. The final traceback step identifies the highest scoring path.

Dynamic programming methods will find the optimal global (NW) or local alignment (Smith & Waterman; SW) for two sequences given the substitution matrix, but are relatively slow. With increases in sequence database size it has become impractical to use these methods when searching for homologues. This has lead to the development of heuristic methods which reduce the search space by quickly excluding sequences which are unlikely to produce a good score. They are not, however, guaranteed to find the optimal alignment. The most popular methods in this category are BLAST (Altschul et al., 1990) and FASTA (Pearson and Lipman, 1988).

BLAST initially makes a list of amino acid subsequences (words) of a certain length (three by default) that are present in the query sequence and that produce a score higher than a threshold. It then searches a database of sequences for these words and, finding one, tries to extend an ungapped match in both directions to attain a maximal scoring extension. This reduces the search space considerably over NW and SW, allowing one to search a query sequence against a database of millions of sequences in a few seconds. Subsequently the authors produced a version of BLAST allowing gapped alignments (Altschul et al., 1997).

### 1.3.4. Profile Methods

Profile methods compare a single protein sequence against an alignment of known homologues and determine the similarity between them. There tend to be positions in an alignment of homologues where amino acids are highly conserved (i.e. present in the vast majority of sequences). Putative homologues are likely to have the same amino acid conserved at these positions and the score ought to reflect this by penalising alternative residues. Other positions in the alignment may be more variable and thus the score for a putative homologue should not be greatly affected by variation at these positions. It has been found that methods using multiple sequences detect three times as many remote homologues as pairwise methods (Park et al., 1998).

The most popular profile method is PSI-BLAST (Altschul et al., 1997). PSI-BLAST takes a single sequence and performs an iterated BLAST against a database. In the first iteration close homologues of the sequence are found and used to build a profile. The profile represents the likelihood of observing each amino acid at each conserved position. This profile is then searched against the database to pull in more distant relatives from which a new profile is built and yet more distant homologues can be detected.

## 1.3.5. Profile Hidden Markov Models

Profile Hidden Markov Models (profile-HMMs or simply HMMs) (Hughey and Krogh, 1996; Krogh *et al.*, 1994; Eddy, 1996) can be considered a more formal approach to the profile methodology with the key incorporation of position-specific gap penalties. Similar models have been used for trans-membrane helix prediction (e.g. TMHMM; Krogh et al., 2001) and gene prediction (e.g. GeneWise; Birney and Durbin, 2000). The two commonly used implementations for homologue detection are SAM (Karplus et al., 1998) and HMMer (Eddy, 1998). HMMs are generally used to model protein domains as their power to detect remote relationship is reduced when considering multi-domain proteins. HMMs are used extensively in Chapter 2 and the predictions they provide of protein domains are used throughout Chapters 3 and 4; they are therefore discussed here in some detail.

### *1.3.5.1. HMM Architecture*

The schema of an example HMM is shown in Figure 1.10. Each arrow and labelled box in the diagram has one or more parameters which are calculated based on the multiple alignment of the sequence family to be modelled and background probabilities based on proteins in general. Each conserved column in a domain alignment is represented by a match state (M in Figure 1.10). Match states are one of two types of emission state and emit amino acid residue symbols according to a probability distribution. From a match state, the model can pass into either the next match state, a delete state (D) or an insert (I) state. The probability of passing from one state to another state is

termed the transition probability. Figure 1.10 shows the plan 7 HMM architecture as implemented in HMMer. SAM implements a slightly more complicated version known as plan 9, which includes transitions directly between insert and delete states. Insert states are the second form of emitting state and also the only type of state with a transition back to itself. Insert states model inserted sequence between match states. Delete states are silent and do not emit any residues. These allow the model to skip a match state and reflect the situation where a member of the protein domain family has undergone the deletion of a residue.

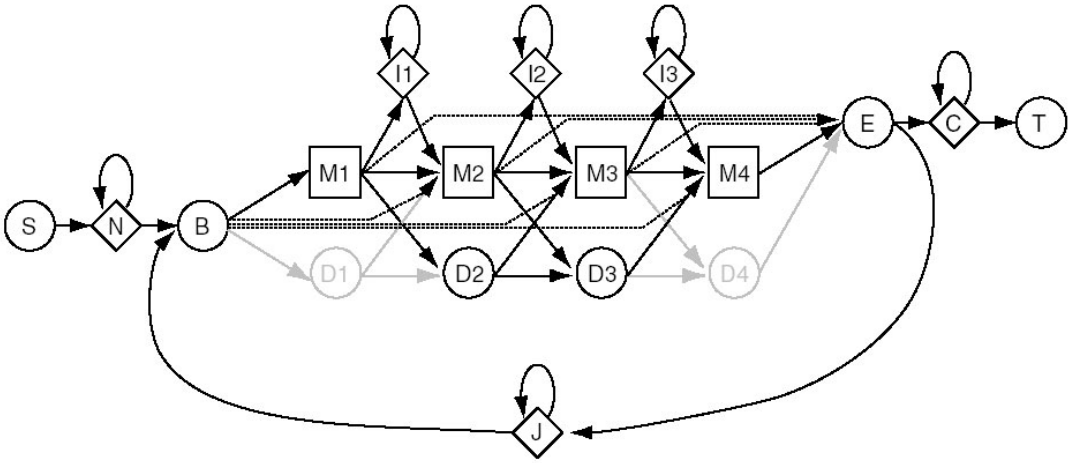



Figure 1.10 Plan 7 HMM architecture as implemented in HMMer (Eddy, 1998).

Profile HMMs of this form are used to model protein domain families. Match states (squares) represent conserved positions in a domain and emit amino acid symbols based on a probability distribution derived from the domain family sequence alignment. Insert states (diamonds I1, I2 & I3) model insertions between match states. Delete states (circles D1, D2, D3 & D4) model the deletion of a conserved position (i.e. allow a match state to be skipped). See text for further details.

The S state is trivial, with the transition probability into the model being one. This is a feature of the implementation rather than the protein family being modelled. The B state allows for transition into the first M or D state, the probability is one for the M transition in the case of global alignment. The E state has transition probabilities specifying the likelihood of repeated matches to the whole model versus exiting the model. The J state allows inserted residues between multiple matches of the model to a sequence (representing duplicated domains) by modelling inserted residues between matches and a path back to the start of the model. N and C states allow insertion of residues at either end of the model to achieve local scoring.

### *1.3.5.2. Model Parameterisation*

The emission and transition probabilities of an HMM are parameterised in such a way as to make it the model which is most likely to have produced the training sequences. The trained model is then used to assess the likelihood that a sequence of interest has been emitted by the model.

It is generally the case that homologous protein domains have a core of conserved tertiary structure with more variable regions occurring in the loops between secondary structures (Reeves et al., 2006). In a multiple sequence alignment this is visualised as regions of conservation with intervening indels. Match states are generally created for each position in the alignment where the majority of the sequences have a residue. Unaligned regions (where the majority of sequences have gaps) contribute to the probability of transition into an insert state from the previous match state. The more sequences that have inserted residues, the more likely this transition will be. For match states where some sequences have a deletion, more or less of these deletions will raise or lower the probability of a transition to the delete state which causes the model to skip that match state. The fewer deletions and insertions there are, the higher the probability that the model will pass from one match state to the next. The transition probabilities from any particular state must add up to one. It is necessary

that every possible transition has a non-zero probability in order that unseen variation in true homologues is not excluded.

Emission probabilities give the likelihood of emission for each amino acid from each emitting state. For match states, these are based on the observed counts of each amino acid in each match state of the alignment, with each match state having a different set of probabilities. Zero probabilities for amino acids must be avoided and there are several methods for this. The most successful method is to use Dirichlet mixtures (Sjolander et al., 1996). These take account of the properties of observed amino acids and upweight emission probabilities for similar types of amino acids. If the predominant amino acid observed for a match state is small, hydrophobic and therefore probably buried in the protein, a large hydrophilic residue is unlikely to be substituted as this would disrupt the fold of the protein. Dirichlet mixtures may also be used to model emission probabilities for insert states.

It is unwise to give equal weight to every sequence in the alignment used to parameterise the model, as this tends to lead to more common sequences dominating the probabilities in the model (Karchin and Hughey, 1998). Sequences which are more similar to each other are down-weighted, allowing more divergent sequences to express their features. The result is a more general model which is better at detecting divergent homologues.

#### *1.3.5.3. Creating Domain Family Alignments for Model Parameterisation*

In order to implement profile methods it is necessary to construct an alignment of homologous sequences. One approach is to derive an initial set of seed sequences from any suitable source e.g. literature or database searches. A seed alignment is then created using iterative pairwise alignment and may be manually curated to reflect knowledge of structurally and functionally conserved residues. HMMs are built from the seed alignment and used to find further homologues for a full alignment.

Alternatively a more automated approach may be used, such as that implemented in the SAM-t2k program (Karplus et al., 1998). This procedure requires as input only one representative seed sequence, although existing alignments or multiple homologues may also be used. Given a single seed sequence and a non-redundant database of protein sequences, BLAST is used with a permissive E-value cut-off of 100 to reduce the database to sequences which are similar to the seed sequence although by no means necessarily homologous. An iterative HMM procedure is then used to align progressively more distant homologues using progressively higher E-value cut-offs.

#### *1.3.5.4. Scoring Sequences against HMMs*

As for other homologue detection algorithms, a score is required which represents how well a sequence and an HMM match. This can be achieved using either the Viterbi or forward dynamic programming algorithms, which are related to the Needleman-Wunsch and Smith-Waterman algorithms. Viterbi is faster, but slightly less accurate than forward. Viterbi calculates the most probable path of a sequence through the model, whereas the forward algorithm calculates the sum of the probabilities of all possible paths through the model. These algorithms give the probability that the model would produce the query sequence (Durbin et al., 1998).

Null models are used in HMM scoring in order to account for the fact that some sequences have an amino acid composition which is close to the background frequency. In such cases a sequence may score highly by finding a path through a model of non-homologues due to the background frequencies assigned to the emitting states. Therefore each sequence scored against a model is also scored against a null model, which represents a random match in some way. Similar scores for both the real and null models suggest a random match to the real model. A significantly higher score for the real model versus the null model represents a good match. Null models can be randomly generated based on background amino acid frequencies (Karplus et al., 1998). However a reverse null model uses the same set of

states and probabilities as the real model, but in the reverse orientation. This preserves the sequence composition and has been shown to give increased performance over using background probabilities (Karplus et al., 2005).

To obtain E-values from HMM matches it may be necessary to calibrate them. Calibration involves estimating the parameters of the Extreme Value Distribution (EVD) representing the distribution of errors for a particular model. To estimate these parameters the model is scored against a set of random sequences and the distribution of the resulting scores is fitted to an EVD (Durbin et al., 1998). The parameters of this fitted distribution are used to calculate the E-values for the scores produced by the model.

HMMs allow several modes of scoring depending on whether they should match the whole of a query sequence or just part of it (global/global and global/local scoring respectively). They may also allow for scoring part of a model against part of a query sequence (local/local scoring). Global/local scoring is the most useful for identifying domains, although local/local allows matches to domain fragments which may result from incorrect gene predictions.

## 1.3.6. Profile-Profile Methods

Profile-profile methods compare two profiles rather than a sequence and a profile. The advantage of profile-profile technologies is that they allow the detection of yet more remote homology than sequence-profile methods (Soding, 2005). Both sides of the comparison include variations that are known in those sequence families and this information allows high scoring matches to be produced between more distantly related families.

### *1.3.6.1. Profile Comparison*

Profile comparison methods are an extension of the sequence-profile concept where, instead of aligning a sequence to a profile, two sequence profiles are aligned. Example implementations include LAMA (Pietrokovski, 1996), COMPASS (Sadreyev and Grishin, 2003) and prof_sim (Yona and Levitt, 2002).

COMPASS allows the gapped alignment of sequence profiles and introduced the estimation of E-values to profile-profile comparison. Numerical profiles are generated from a sequence alignment, counting the frequency of each amino acid (or gap) at each position. Sequences are weighted to prevent common sequences from dominating the profile and columns with many gaps are excluded. The log-odds scoring used for PSI-BLAST was generalised by Sadreyev and Grishin to the log-sum-of-odds, which allows scoring between two profiles. E-values for COMPASS are calculated empirically as described for BLAST.

### *1.3.6.2. HMM Comparison*

HMM comparison (HMM-HMM) follows on from sequence-HMM comparison and profile comparison, aligning and scoring two profile-HMMs. This has been implemented in the PRC (Madera, 2006) and HHSearch (Soding, 2005) programs. These two approaches are closely related in their treatments of HMM alignment and scoring. Detailed discussion of PRC follows.

The general approach taken to compare two HMMs is to calculate the joint emission probability. In simple terms: do they give similar scores to the same proteins? PRC approaches this by aligning two profile-HMMs in the form of a pair-HMM (Figure 1.12), allowing a score to be derived using the Viterbi algorithm. Each state of the pair-HMM corresponds to pairs of domain family HMM states (matches M, inserts I and deletes D) and a transition in the pair HMM models simultaneous transitions in both domain family HMMs.

Figure 1.12 shows the PRC pair-HMM which is used to model the alignment of two profile-HMMs *i*, *j*. Note that there are states $B_LB_L$ and $E_LE_L$ where both models begin and end, respectively. The $M_iM_j$ state models the situation where the domain family HMMs have aligned match states, $D_iM_j$ and $M_iD_j$ where a delete state is aligned to a match state, $I_iM_j$ and $M_iI_j$ where an insert state is aligned to a match state. Transition probabilities between pair-HMM states are the product of the corresponding transition states in

each of the individual domain family HMMs. Similarly, emission probabilities for emitting states ($M_iM_j$, $I_iM_j$, *and* $M_iI_j$) are calculated using the product of the corresponding emission vectors in the domain family HMMs.

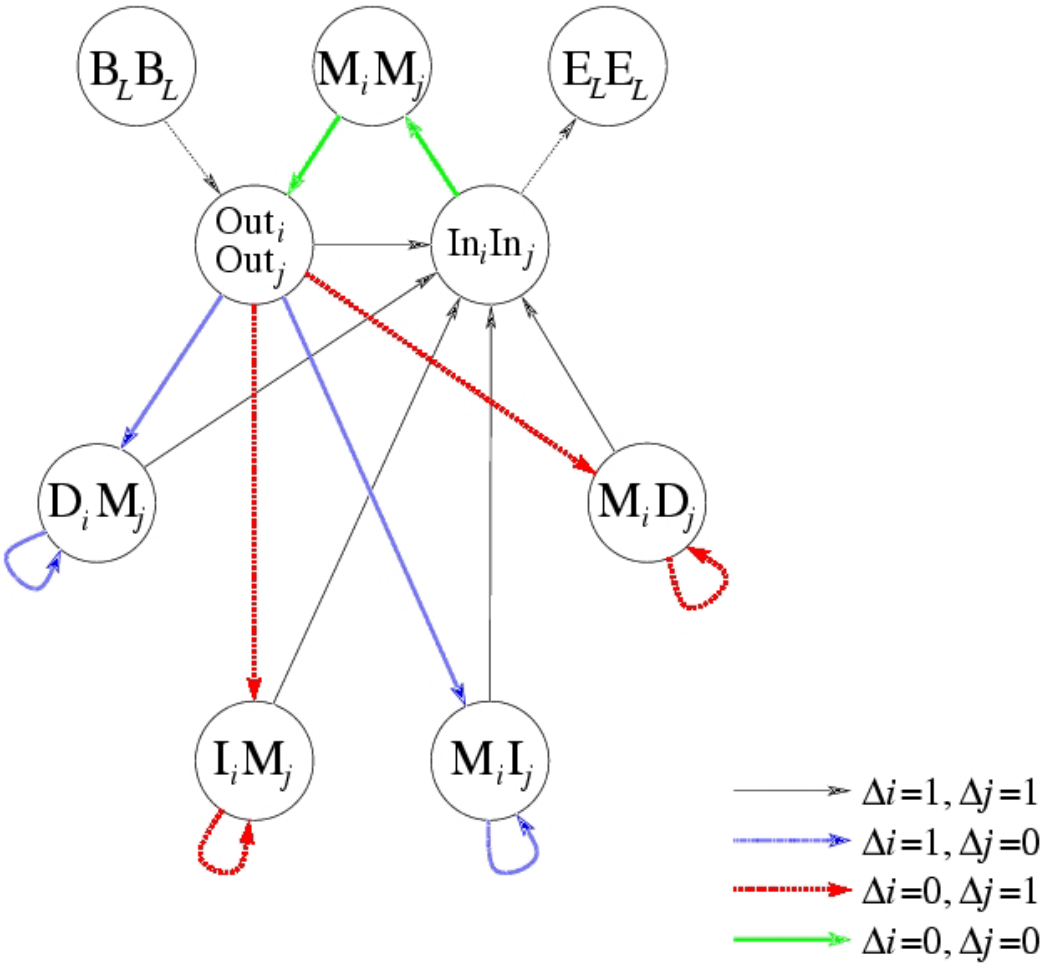



Figure 1.12 PRC's pair-Hidden Markov Model.

The pair-HMM models the alignment of two profile-HMMs and allows the alignment to be scored using the Viterbi algorithm. The $M_iM_j$ state represents aligned match states from the two HMMs. $D_iM_j$ and $M_iD_j$ represent a deletion state aligned to a match state. The $I_iM_j$ and $M_iI_j$ states represent insert states aligned to match states. The differently coloured arrows indicate which profile-HMM should advance a match state during a particular pair-HMM transition. This figure was reproduced from Madera (2006). See text for further details.

As with HMMer's plan 7 architecture, PRC makes assumptions about allowed transitions for simplicity and speed. The pair-HMM architecture shown in Figure 1.12 is by no means the only one possible. Note that each transition must be to either the same state or via the $M_iM_j$ state which could be avoided by a more complex architecture. InIn and OutOut are essentially part of the $M_iM_j$ state, but allow for local scoring by providing routes in and out of the model.

A score for the alignment is produced as for alignment of a sequence to a HMM, by applying the Viterbi algorithm and a null model. The PRC null model takes into account both the effect of length-dependence and low complexity sequences that have residue frequencies close to the background.

### 1.3.7. Structure-Based Homology Detection

Even when there is no detectable sequence similarity between proteins, similarities in their 3D structure can often be observed (Chothia and Lesk, 1986). This means that when their structures are available, it is generally easier to detect similarities between homologous proteins by comparing their structures. In fact it has been shown that for even very remote homologues with <20% sequence identity, at least 50% of the structure remains conserved (Reeves et al., 2006).

In order to determine the similarity between two structures they are aligned in three dimensions. This is achieved in two steps. Firstly the similarity between residues and/or secondary structural features of both proteins is determined and secondly an alignment is sought to maximise the score of aligned positions. Once structures are superposed their similarity is usually quantified using the Root Mean Square Deviation (RMSD). This is the square root of the average squared distance between equivalent atoms, as in Equation 1.2 below.

$$RMSD = \sqrt{\frac{1}{N}\sum_{i=1}^{i=N}\delta_i^2}$$

Equation 1.2 Root Mean Square Deviation formula

In Equation 1.2, $\delta$ is the distance between $N$ pairs of equivalent atoms $i$.

Popular structural alignment programs include SSM (Krissinel and Henrick, 2004) and GRATH (Harrison et al., 2002), which use secondary structure and SSAP (Taylor and Orengo, 1989), DALI (Holm and Sander, 1993), CE (Shindyalov and Bourne, 1998), STRUCTAL (Subbiah et al., 1993) and LSQMAN (Kleywegt, 1996), which are based on residue comparisons. The relative effectiveness of structural comparison methods for homology detection has been examined by Kolodny et al. (Kolodny et al., 2005) and Redfern et al. (Redfern et al., 2007). In this thesis SSAP is used as it has been shown by these authors to be among the best methods. In Chapter 2 SSAP is used to measure structural similarity for improving benchmarks of sequence-based homologue detection methods.

#### *1.3.7.1. Sequential Structure Alignment Program*

Sequential Structure Alignment Program (SSAP) uses double dynamic programming to find the optimal alignment for two protein structures. The algorithm is shown graphically in Figure 1.14. Initially, residue views are defined for each Cβ atom in each of the two protein structures. The Cβ atom is the first carbon atom in an amino acid side chain. A residue view is the set of vectors from one Cβ atom to all other Cβ atoms in the protein structure. For any pair of residues between the two proteins, their residue views can then be compared to determine their similarity. A residue-level score matrix is constructed for each pair of residues with similar accessibility and torsional angles between the proteins. These matrices are then populated with scores based on the similarity of each pair of vectors in the residue view. Dynamic programming is used to determine the highest scoring path through each residue pair matrix. The top 20 pairs of residues which score

above a threshold are added to the summary score matrix and another round of dynamic programming is used to find the optimal path through this matrix. The SSAP score is calculated using the similarity of the aligned residue views normalised by the size of the largest protein.

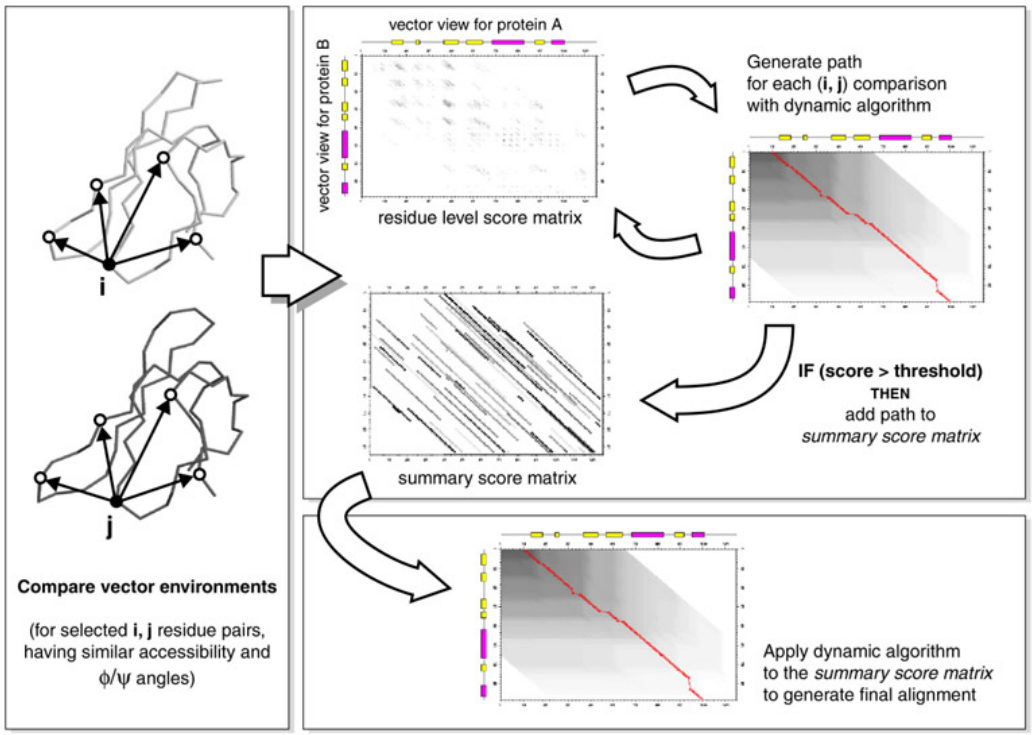



Figure 1.14 Flowchart of the SSAP algorithm.

Vector environments are compared between pairs of potentially equivalent residues in each protein. A residue level score matrix is constructed for each pair and optimal paths are calculated by dynamic programming. High scoring paths are then added to the summary score matrix. Dynamic programming is applied to the summary matrix to generate the alignment of the two structures.

Certain scoring schemes have been shown to give improved separation between homologues and non-homologues over the use of native structural comparison scores (Kolodny et al., 2005). The SAS score (Subbiah et al., 1993) for instance, is based on the RMSD produced by an alignment method, normalised by the number of residues aligned. The SAS score is used in this thesis, rather than the native SSAP score. Equation 1.3 shows how the score is calculated; $N_A$ is the number of aligned residues.

$$SAS = \frac{100 \times RMSD}{N_A}$$

Equation 1.3 SAS score for structural comparison.

### *1.3.7.2. Threading*

Threading is used to determine whether a particular protein sequence is compatible with a known structure. It does not primarily exploit evolutionary information to recognise structural domains in sequences, instead it determines how well a sequence fits a particular fold (Jones et al., 1992). The quality of the fit is determined from a distribution of observed inter-residue distances. Because very different sequences can form similar structures, this approach allows very distant, homologous relationships to be recognised. Full structural threading requires large amounts of computing power and it has been shown that using a more heuristic approach as implemented in GenTHREADER (McGuffin and Jones, 2003) and 3D-PSSM (Kelley et al., 2000) produces similar results (Cherkasov and Jones, 2004).

## 1.3.8. Algorithms for Clustering Proteins

Clustering algorithms are useful for identifying distinct groups of homologous proteins based on their similarity. Given a similarity matrix (of sequence identities or E-values, for instance) for a set of proteins or domains, these algorithms determine clusters where members tends to be more similar to each other than to members of other clusters. Some clustering algorithms

require a parameter determining a similarity value that co-cluster members should satisfy although others require a defined number of clusters.

Clustering of protein domain sequences is used in Chapter 2 to generate benchmarking datasets and Chapter 4 to identify proteins complexes.

#### *1.3.8.1. Single and Multi-Linkage Hierarchical Clustering*

In hierarchical clustering, elements are initially in single member clusters. Clusters are then merged based on the distance (similarity) between the clusters. If the distance is less than a pre-determined cut-off, then the clusters are merged. In the single-linkage approach the distance between two clusters is calculated as the distance between the closest elements in those clusters. This can lead to a phenomenon known as chaining, whereby elements which are less similar than the cut-off end up in the same cluster.

Multi-linkage clustering will only allow the merging of clusters where all elements in the clusters are at least as similar as the cut-off specifies. The drawback of this method is that it can be too conservative as many members between the clusters may be similar enough to be in the same cluster.

#### *1.3.8.2. Markov Cluster Algorithm*

The Markov CLuster algorithm (MCL; Enright et al., 2002) uses a weighted graph (or similarity matrix) to determine clusters based on simulated flow in the graph (Van Dongen, 2000). The size of clusters is controlled by a term called the inflation parameter, rather than a similarity cut-off. This algorithm has been used for clustering proteins into families (Enright et al., 2003) and also for defining modules of interacting proteins in protein-protein interaction networks (Brohee and van Helden, 2006).

# 1.4. Protein Domain Classification Resources

Several resources have been developed to classify proteins into evolutionary families in order to understand the relationship between sequence, structure and function. Different evolutionary families evolve in different ways and perform different functions. Classifying protein domains into families therefore aids the study of protein evolution and function. The different types of resource and their approaches are discussed below.

## 1.4.1. Sequence-Based Classifications

Sequence-based domain classifications require detectable sequence similarity to identify evolutionary relationships. They are able to define less remote relatives than structural classifications (see 1.4.2) but have the advantage that there is much more sequence data than structural data. This comparative wealth allows sequence-based classifications to provide more domain annotations per gene than those based on structure (Marsden and Orengo, 2008).

### *1.4.1.1. Automated Sequence-Based Protein Domain Classifications*

Automated sequence-based classifications attempt to derive a complete set of domain families based on evolutionary conservation. Examples include ADDA (Heger et al., 2005), CLUP (Liu and Rost, 2004), Everest (Portugaly et al., 2007) and ProDom (Bru et al., 2005). ProDom uses PSI-BLAST (described in 1.3.3) recursively to generate a set of domain families. Given a database of protein sequences, the shortest is queried against a sequence database using PSI-BLAST to create a domain family. Sequence regions present in this family are removed from the database and the next shortest sequence is used for another round of PSI-BLAST. This process is iterated until the database is empty. ProDom has begun to use manual annotation to adjust domain boundaries, so it is no longer a purely exhaustive approach.

#### *1.4.1.2. Curated Sequence-Based Protein Domain Classifications*

Curated sequence-based classifications are based on the curation of automatically defined families. A variety of expert knowledge and peripheral resources can be used to improve domain predictions. Structural data for instance may be used as a guide where possible. Examples of this type of classification are Pfam (Finn et al., 2008), SMART (Letunic et al., 2006), PRINTS (Attwood et al., 2003) and BLOCKS (Henikoff and Henikoff, 1992).

Pfam is a database of curated multiple alignments of protein domain families with associated HMMs allowing users to determine the Pfam domain content of proteins. It was originally based on families defined by ProDom. The families often encompass a smaller region of sequence space than CATH or SCOP superfamilies due to reduced power in detecting distant relationships; however they are generally more specific in terms of function. Manual curation has produced an accurate and well trusted set of families, but is time consuming. Pfam-B is an exhaustive, automated supplement to the manually curated Pfam-A. It provides extra coverage and a starting point for manual curation.

### 1.4.2. Structure-Based Classifications

It has been shown that between homologous domains, tertiary structure is generally more conserved than sequence (Figure 1.16). Therefore, where sequence identity between domains is low, a common tertiary structure may allow an evolutionary relationship to be determined. Structural classifications of protein domains thus tend to produce larger families containing more remote homologues than sequence-based classifications. They are limited however by the diversity of known protein structures which is significantly less than that of known sequences (~$5.4 \times 10^4$ structures in the PDB, ~$6 \times 10^6$ sequences in RefSeq as of November 2008). The most popular structure-based classifications are CATH (Greene et al., 2007), SCOP (Andreeva et al., 2008) and FSSP (Holm and Sander, 1994). CATH is used extensively throughout this thesis.

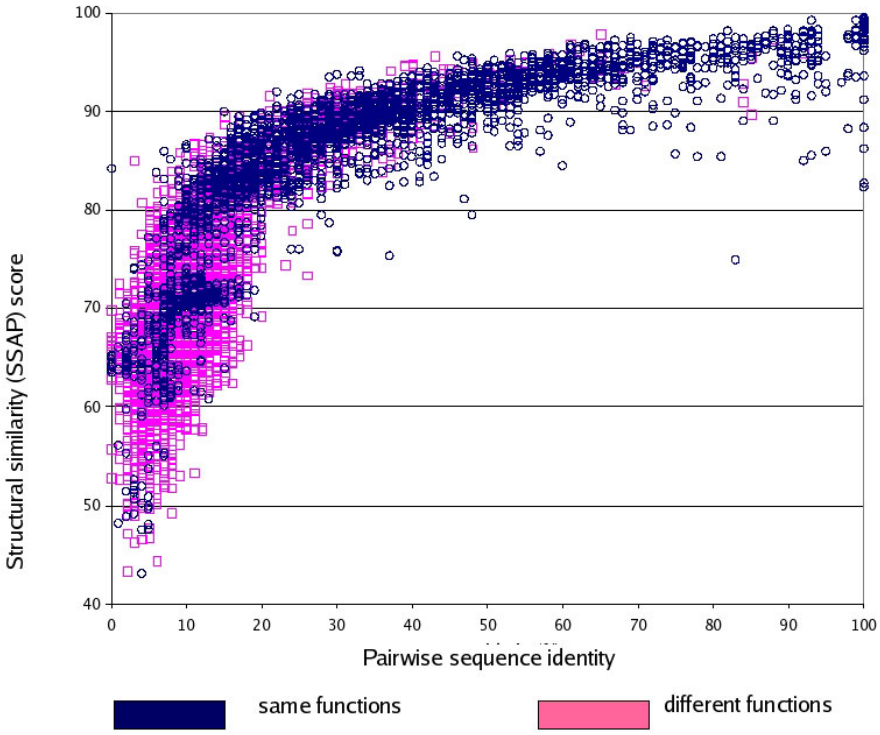



Figure 1.16 Relationship between the conservation of sequence and structure.

The sequence identity and structural similarity of all pairs of domains in CATH are shown coloured by whether the pair share the same function (blue) or not (pink). As sequence identity between a pair falls, structural similarity (measured by SSAP score) falls much more slowly, until ~20% sequence identity, where it begins to rapidly fall off. Note that above a sequence identity of 60%, two sequences are highly likely to share the same function, but below this level the relationship is more complex. This figure was taken from Reeves et al. (2006).

### *1.4.2.1. CATH*

CATH (Greene et al., 2007) is a hierarchical classification of protein domains produced by Orengo and colleagues at UCL. There are four principal levels to the hierarchy, denoted by the eponymous letters C, A, T and H.

1. The Class (C) level broadly categorizes domains according to their general secondary structure: mostly α, mostly β, α/β and few secondary structures.
2. The Architecture (A) level contains domains which have a similar spatial arrangement of secondary structures.
3. The Topology or fold (T) level groups protein domains together whose secondary structures are connected in the same way.
4. The Homologous Superfamily (H) level brings together domains which have sufficient structural, sequence and functional similarity to suggest they share a common ancestor.

Note that the first two levels of the hierarchy are phenetic, having nothing to say about the evolutionary relationship between domains in the same group (May, 1999). The T level groups domains which may be homologous or analogous. The H level groups homologous domains based on structural similarity, sequence similarity and evidence of common function. Examples of levels C, A and T are shown in Figure 1.18.

Below the H level are 4 sequence family levels (S, O, L and I) clustered with successively higher sequence identity cut-offs (35%, 60%, 95%, 100% respectively) and an 80% overlap cut-off. The leaves of the hierarchy (D) are individual domains. Each node of the classification has a representative domain and the S level representatives (S-reps) are useful for sequenced based work as will be shown in Chapter 2.

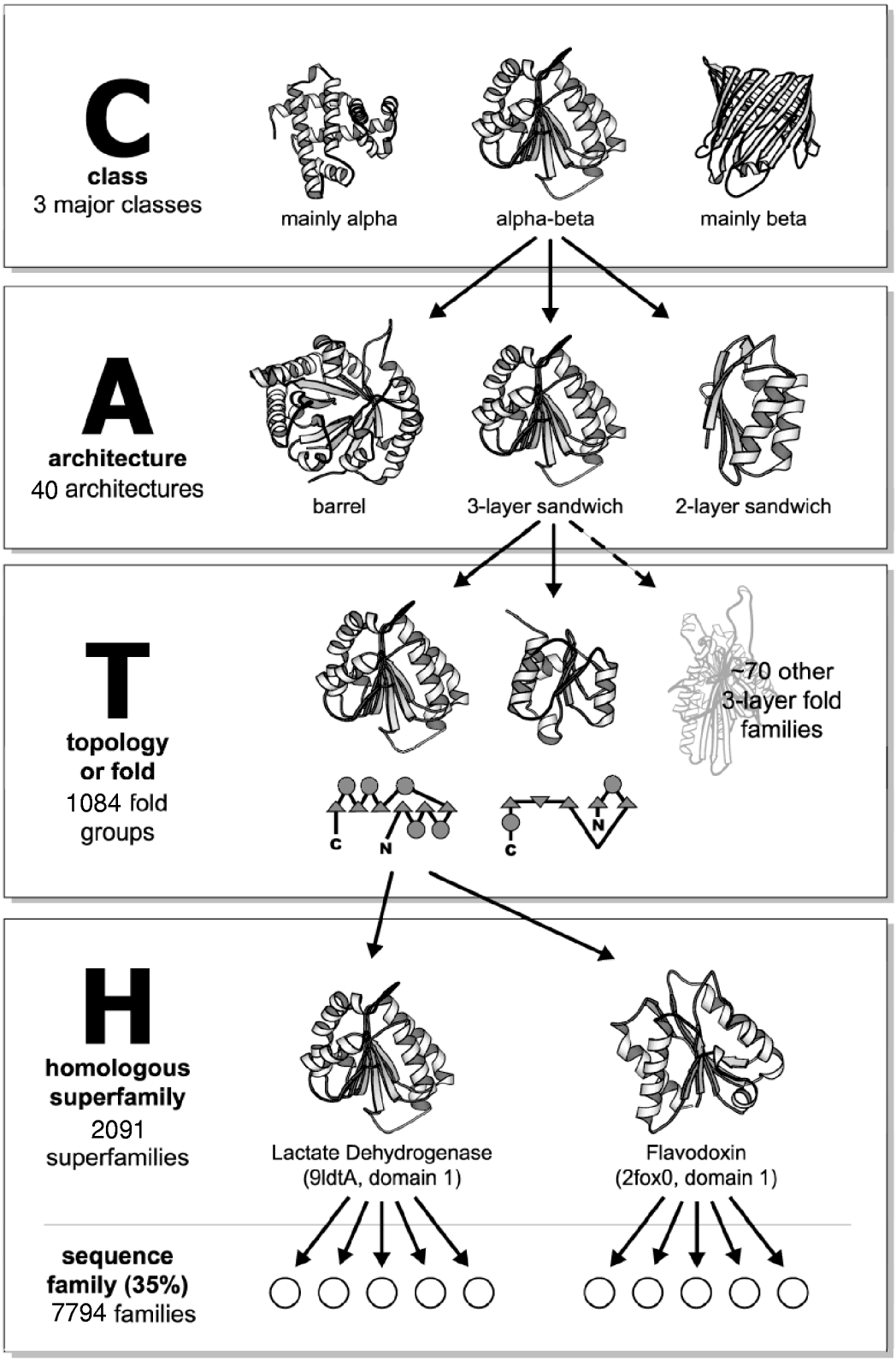



Figure 1.18 The CATH hierarchy organises protein domain structures into groups based on their structural similarity.

Numbers are based on CATH v3.1.0. This figure was created by E. Sideris.

A particular node in the hierarchy is referenced using a number for each level above and including that node. For example there is a superfamily of serine proteases domains denoted 3.40.50.200. Here the class is 3 (α/β), the architecture is 40 (3-layer αβα sandwich), the topology is 50 (Rossman fold) and the superfamily is 200 (serine protease).

Where possible, multi-domain chains are decomposed into individual domains using the CATHEDRAL (Redfern et al., 2007) algorithm to identify homologues of pre-existing CATH entries. When no identifiable homologues exist in the CATH database manual inspection is used to identify independent structural units.

### *1.4.2.2. SCOP*

Structural Classification Of Proteins (SCOP) is a classification of protein domain structures similar to CATH (Andreeva et al., 2004). SCOP has three major levels in its classification each equivalent to a level of the CATH hierarchy. The class level has nodes for mostly α proteins and mostly β proteins, but splits the α/β class of CATH into those with intercalated α and β structure (α/β) and those where α and β structure is largely separated (α+β). At the fold level, proteins have largely the same secondary structures and the same topology, but may not be evolutionarily related. The superfamily level groups domains which have structural and functional similarity suggestive of a common evolutionary origin. The family level groups domains that are clearly related and generally have a sequence identity of >30%. There is no equivalent to the CATH architecture level.

An important distinction between CATH and SCOP is that SCOP will not separate structural domains unless their homologues have been observed separately in different protein chains. Thus SCOP domains more closely represent independent evolutionary units whereas CATH domains more closely resemble independently folding structural units.

## 1.4.3. Identifying Domains in Protein Sequences

Given a domain family classification, sequences with unknown domain architecture can be annotated, generally using HMMs. This allows the elucidation of the frequency of particular domain superfamilies and, by extension, functions within different species (Lee et al., 2005). It also enables estimation of how many protein structures still need to be determined experimentally and which uncharacterised sequences might represent suitable targets for structural genomics projects (Marsden *et al.*, 2006; Marsden *et al.*, 2007). These classifications may be based on sequence or structural data and many of the approaches are combined in the InterPro resource (Mulder et al., 2007).

### *1.4.3.1. Pfam*

The Pfam resource both identifies sequence-based domain families, as discussed and provides a library of HMMs to annotate sequences. Pfam provides curated E-value cut-offs for each HMM which allow for very accurate predictions. Additionally there are two types of model for each family; models which best detect complete domains and those which are optimised to detect fragmented domains.

### *1.4.3.2. Gene3D*

Gene3D (Yeats et al., 2008) is a resource produced by the CATH group which maps CATH superfamilies onto all known protein sequences. HMMs, generated using SAM-T2K (Karplus et al., 1998) and based on S-level representatives in CATH, are used to predict domains. Multi-domain architectures (MDAs) are resolved using the DomainFinder protocol (Buchan et al., 2002). Gene3D also incorporates domain assignments from Pfam as well as functional data from the GO ontologies (Ashburner et al., 2000) and FunCat (Ruepp et al., 2004), pathway data from KEGG (Kanehisa et al., 2006) and protein-protein interaction data from IntAct (Kerrien et al., 2007), MINT (Chatr-aryamontri et al., 2007) and BIND (Bader et al., 2003).

#### *1.4.3.3. Superfamily*

Superfamily (Wilson et al., 2007) is similar to Gene3D, but based on SCOP rather than CATH. It uses multiple HMMs to represent each superfamily of proteins and allows annotation of genomes with SCOP domains.

#### *1.4.3.4. Genomics Threading Database*

The Genomic Threading Database (GTD; McGuffin et al., 2004) uses GenTHREADER (McGuffin and Jones, 2003) to obtain fold-level annotation for complete genomes. GenTHREADER involves a threading-based approach to structure prediction in combination with PSI-BLAST and secondary structure prediction. GTD allows keyword searches using PDB and SCOP codes, gene identifiers and descriptions as well as BLAST searches. Useful summary statistics on fold coverage of the genomes are provided.

#### *1.4.3.5. 3D-Genomics*

This resource from the Sternberg group at Imperial College (Fleming et al., 2004) provides SCOP and Pfam domain annotation, secondary structure predictions and sequence features such as low complexity regions, coiled-coils and transmembrane helices for completed genomes. It also allows determination of homologous features between genomes on the fly using BLAST. Another useful feature is the ability to perform genome comparisons based on statistics of domain features.

#### *1.4.3.6. InterPro*

InterPro (Mulder et al., 2007) brings together data from many different domain annotation resources, allowing users to compare their predictions. It includes both Gene3D (CATH) and SUPERFAMILY (SCOP) structural domain predictions as well as Pfam, Prosite, SMART, Panther, PRINTS, ProDom, TIGR sequence domains/motifs. It produces a consensus of these where possible as InterPro domains.

### 1.4.4. Whole-Chain Protein Classifications

This thesis concerns the use of domains to identify evolutionary relationships between proteins; however, relationships between proteins are often

classified independently of their domain architectures. There are many resources which classify whole proteins, rather than domains, into families. These include SYSTERS (Meinel et al., 2005), ClustR (Petryszak et al., 2005), Protomap (Yona et al., 2000), COGS (Tatusov et al., 2003) ProtoNet (Kaplan et al., 2005) and TRIBES (Enright et al., 2003).

## 1.5. Evolution of Protein Domain Families

In studying protein domain families many details of their evolution have been revealed. The size distribution of protein domain families follows a power law (Apic et al., 2001b). That is, the distribution of CATH domains in either the PDB or in genomes reveals that a small number of superfamilies occur many times whereas most superfamilies occur very few times (Orengo and Thornton, 2005). This is shown graphically in Figure 1.20. It has also been shown that highly expanded families are very diverse in their functions (Todd et al., 2001). These findings suggest that certain arrangements of protein structure are particularly useful in biology, whereas others have relatively niche roles.

Protein domains tend to exist in multi-domain chains. It has been shown that while a few superfamilies are found in proteins with many different partner superfamilies, most superfamilies are found with very few other superfamilies (Basu et al., 2008; Vogel et al., 2004), another example of the power law. Superfamilies which occur with many different superfamily members are termed promiscuous. The most promiscuous superfamilies also tend to be the largest and include the cofactor binding P-loop nucleotide triphosphate hydrolase domains which binds ATP and GTP and the NADP(P)-binding Rossmann domains (Vogel et al., 2005). These co-factors are involved in energy transfer (ATP, GTP) information processing (ATP, GTP), transcription (ATP, GTP), DNA synthesis/replication (ATP, NADP) and lipid biosynthesis (NADP) amongst others.

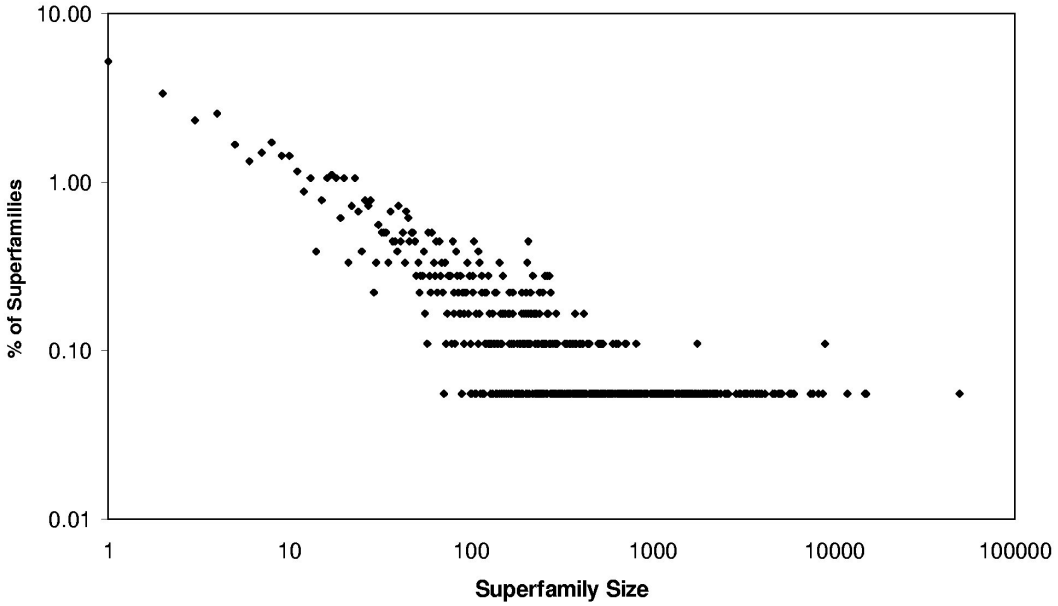



Figure 1.20 Power-law distribution of CATH protein domain families in Gene3D version 6.

Superfamily size is the number of members in the superfamily.

## 1.6. Protein Function Classifications

Early descriptions of protein function in databases were in the form of unstructured keyword fields. Accurate descriptions of protein function are complex and it has been found necessary to formalise them in order that different researchers can easily use the same terms to describe the same features. The most commonly used classifications are presented here. The Gene Ontology (GO; Harris et al., 2004) is used extensively in Chapters 3 and 4 to determine the functional similarity of proteins. FunCat (Ruepp et al., 2004) is used in Chapter 4 to classify protein complexes.

### 1.6.1. Gene Ontology

Ontologies describe entities and the relationships between them. They have come to be used in various aspects of biology. By rigorously defining entities and the relationships between them, the way in which biology is described is rationalised and more easily understood by computers. The Gene Ontology (GO) describes terms relating to protein function in three separate ontologies. The three ontologies define different aspects of protein function. This division in itself helps us to understand how we think about protein function. The biological process ontology concerns terms relating to a biological objective in which the gene product is involved. General biological process terms include *cell growth and maintenance*, whereas *pyrimidine metabolism* is a more specific term, associated with fewer gene products. The molecular function ontology describes what a gene product does without specifying where or when the event occurs. *Lyase* and *ligand* are examples of broad terms, while *adenylate cyclase* and *Toll receptor ligand* are more specific terms. The third ontology is entitled cellular component and concerns the location in the cell where the gene product is active. Examples of this aspect of function include *nucleoplasm* and *replication fork*.

The GO ontologies are formally described as Directed Acyclic Graphs (DAGs); they are hierarchical structures where a node may have both multiple children and multiple parents. The nodes in these graphs are

functional terms and the edges are parent-child relationships. The root node of each ontology bears the name of the ontology itself, i.e. biological process. This is the most general term in each ontology. Terms become more specific towards the leaf nodes, although there is no set depth to each DAG and no absolute measure of how specific the terms are at any depth.

In order to use GO it is necessary to apply ontology terms to proteins. Initially three genome annotation consortia (Flybase (Tweedie et al., 2008), Mouse Genome Informatics (Bult et al., 2008) & Saccharomyces Genome Database (Hirschman et al., 2006)) each annotated the genomes of their organism of interest as part of the combined GO consortium (Ashburner et al., 2000). Subsequently many more organism-specific databases have become involved, extending annotation to more genomes as well as the UniProt protein sequence database (Gene Ontology Consortium, 2006). Annotation is available from individual members of the GO consortium such as Saccharomyces Genome Database. One particular member is, however, perhaps the most useful. The Gene Ontology Annotation database (GOA) is a member of the GO consortium which aims to annotate UniProt and the International Protein Index (IPI) with GO terms (Camon et al., 2004). A GO annotation is an instance of a node in one of the ontologies, associated with a gene or gene product and an evidence code. Evidence codes (

Table 1.1) describe the approach used to annotate a gene product.

| Code | Long description |
|---|---|
| IMP | Inferred from mutant phenotype |
| IGI | Inferred from genetic interaction |
| IPI | Inferred from physical interaction |
| ISS | Inferred from sequence or structural similarity |
| IDA | Inferred from direct assay |
| IEP | Inferred from expression pattern |
| IEA | Inferred from electronic annotation |
| TAS | Traceable author statement |
| NAS | Non-traceable author statement |
| ND | No biological data available |
| RCA | Inferred from reviewed computational analysis |
| IC | Inferred by curator |

Table 1.1 Gene Ontology evidence codes.

### 1.6.2. FunCat

FunCat (Ruepp et al., 2004) was initially created as a hierarchical controlled vocabulary for use in annotating the *Saccharomyces cerevisiae* genome. It has since been extended to include terms suitable for prokaryotes and multicellular eukaryotes. The hierarchical structure of FunCat is more similar to EC (see 1.6.3) than to GO, however the focus of FunCat is on classifying proteins in terms of their biological process rather than their enzymatic or molecular function. Currently relatively few genomes have been manually annotated with FunCat terms, these include: *S. cerevisiae*, *A. thaliana*, *H. sapiens*, *N. crassa*, H. *pylori*, *L. innocua*, *L. monocytogenes*, *B. subtilis*, *T. acidophylum*. Table 1.2 shows the terms from the top level of the Funcat hierarchy which is used to annotate protein complexes in Chapter 4.

### 1.6.3. Enzyme Commission

The Enzyme Commission (EC) classification is concerned with enzymatic function and provides a hierarchy of terms describing it (Webb, 1992). The first of four levels describes the reaction class: 1, oxidoreductase; 2, transferase; 3, hydrolase; 4, lyase; 5, isomerase; 6, ligase. The nature of the second and third levels depends somewhat on the first; however in general these describe the actor and acceptor molecular groups involved in the enzymatic reaction. The fourth level indicates the substrate specificity.

| | |
|---|---|
| Metabolism | |
| 01 | Metabolism |
| 02 | Energy |
| 04 | Storage protein |
| Information pathways | |
| 10 | Cell cycle and DNA processing |
| 11 | Transcription |
| 12 | Protein synthesis |
| 14 | Protein fate (folding, modification and destination) |
| 16 | Protein with binding function or cofactor requirement (structural or catalytic) |
| 18 | Protein activity regulation |
| Transport | |
| 20 | Cellular transport, transport facilitation and transport routes |
| Perception and response to stimuli | |
| 30 | Cellular communication/signal transduction mechanism |
| 32 | Cell rescue, defence and virulence |
| 34 | Interaction with the cellular environment |
| 36 | Interaction with the environment (systemic) |
| 38 | Transposable elements, viral and plasmid proteins |

| Developmental processes | |
|---|---|
| 40 | Cell fate |
| 41 | Development (systemic) |
| 42 | Biogenesis of cellular components |
| 43 | Cell type differentiation |
| 45 | Tissue differentiation |
| 47 | Organ differentiation |
| Localization | |
| 70 | Subcellular localization |
| 73 | Cell type localization |
| 75 | Tissue localization |
| 77 | Organ localization |

Table 1.2 Level 1 of the Funcat hierarchy.

The numbered codes are shown in the left-hand column with category headings. The right-hand column gives descriptions for the different categories.

### 1.6.4. Measuring Functional Similarity

Functional similarity can be quantified in various ways. Early examples directly compared terms in the relatively unstructured keyword description of entries from the Swiss-Prot protein sequence database (Devos and Valencia, 2000). The Jaccard coefficient has been used to compare multiple terms by assessing the overlap between two sets (Marcotte and Marcotte, 2002). It is calculated as the intersection of those sets divided by their union. It does not however take advantage of the fact that some non-identical terms are more similar than others.

Semantic similarity (Resnik, 1999) allows for a more subtle comparison and has been extensively applied to the GO hierarchy. Semantic similarity as implemented by Lord et al. (Lord et al., 2003), for instance, uses the information content of the shared parent of two terms. The frequency of each term from a particular GO division (e.g. Biological Process) is determined from a corpus of terms (those assigned to a particular genome for instance). Each child term implicitly invokes its parent and therefore parents inherit the number of occurrences of their child terms. The root node thus has a frequency of one as it is implied by every term in the hierarchy. The probability of a particular term $t$ is then its total number of occurrences divided by the number of times any term occurs. The probability for a pair of terms $t_1$,$t_2$ is that of their closest shared parent. The semantic similarity is then calculated using Equation 1.4.

$$sim(t_1, t_2) = -\ln p_{ms}(t_1, t_2)$$

Equation 1.4 Semantic similarity formula

The semantic similarity between two terms is the negative log of the *probability of the minimum subsumer* ($p_{ms}$) for those terms. The *minimum*

*subsumer* is the shared parent with the minimum probability. Note that GO terms may share multiple parents through different paths.

Frequently, two proteins are annotated with multiple terms from the same GO ontology. In these cases it may be necessary to resolve a similarity value. In this thesis the maximum similarity between any two terms, between the proteins is taken.

It should be noted that functional annotation is generally applied at the protein chain level. It is therefore difficult to discuss the function of protein domains without assuming that terms at the chain level apply to all the domains within that chain. This is not problematic when considering biological process functions; however it can become a problem when molecular function is the focus as domains in the same chain may have very different molecular functions.

The measure of semantic similarity described here is used in Chapters 3 and 4 to determine the functional similarity between proteins based on their GO terms.

## 1.7. Prediction of Protein Function

Experimental determination of protein function involves biochemical and genetic techniques which are accurate but slow. There are insufficient resources to directly characterise every protein in every organism. With the advent of large-scale genome sequencing and bioinformatic techniques it has become possible to predict the function of many proteins computationally (Holt et al., 2002). This can be achieved by inferring information through evolutionary relationships or through prediction methods which exploit characteristics of the sequence or genomic context. In Chapter 3 a method is developed for predicting pairs of proteins involved in common biological processes.

### 1.7.1. Definition of Protein Function

What does it mean to identify the function of a protein? Following the Gene Ontology consortium, protein function can be divided into molecular function, biological process and cellular location. On the one hand it may be most interesting to know that a protein is a protein kinase (its molecular function). However, there are many protein kinases involved in a large variety of cellular processes and so to know that it is involved in a particular developmental signalling pathway (its biological process) for instance, may be more useful in a particular study. Some processes also occur in multiple locations, for instance transcription occurs in mitochondria as well as in the nucleus of eukaryotic organisms. Each type of function can also be described in more or less general terms, e.g. a protein kinase vs. a tyrosine kinase.

Homologues may diverge in molecular function while remaining involved in the same biological process or vice versa. In some metabolic pathways, duplicates of terminal proteins have been recruited to perform an extra step, metabolising a substrate which has become scarce (e.g. ligases in peptidoglycan biosynthesis; Diaz-Mejia et al., 2007). This process is however thought to be rare (Rison et al., 2002). In this case the biological process has remained the same, while the molecular function has changed. In other cases

homologues may perform the same enzymatic step in different tissues or pathways. Here the biological process has changed and the molecular function has remained the same. Functional properties can be predicted based on sequence or structural similarity (Martin *et al.*, 2004; Lee *et al.*, 2007; Porter *et al.*, 2004) as well as a variety of other approaches discussed below.

### 1.7.2. Homology-Based Methods for Predicting Protein Function

Given a protein of unknown function, the most common approach is to find a close homologue using sequence comparison (e.g. BLAST) and to use it to transfer functional annotation. If no homologue can be found with BLAST profile methods such as PSI-BLAST (Altschul et al., 1997) and HMMer (Eddy, 1996) can be used to find more remote homologues. The more remote the homologue however, the less likely that the two proteins perform a similar function (Todd et al., 2001). This relationship has been extensively studied for enzymes (Rost, 2002; Todd *et al.*, 2002; Tian and Skolnick, 2003). For example, Tian & Skolnik (2003) showed that 60% sequence identity is required between enzymes to have a 90% chance of correctly transferring function between them.

More advanced approaches such as GOtcha (Martin et al., 2004), ConFunc (Wass and Sternberg, 2008) and PFP (Hawkins et al., 2008) integrate multiple BLAST hits to assign GO terms to proteins of unknown function.

Motif-based methods identify small functional motifs which can be based in sequence or structure. Prosite (Hulo et al., 2006) is a library of sequence motifs associated with protein function. TEMPURA (Najmanovich et al., 2005) uses experimentally-verified catalytic sites described in the Catalytic Site Atlas (Porter et al., 2004) to discover potential sites of catalysis in protein structures of unknown function.

### 1.7.3. Function Prediction Using Protein-Protein Interactions

The function of proteins can be inferred using datasets of known Protein-Protein Interactions (PPIs). A simple approach transfers annotation to a protein using the most commonly occurring function of its neighbours in a Protein-protein Interaction Network (PIN; Schwikowski et al., 2000). It is also possible to inherit PPIs between homologues, revealing clues about the biological processes in which protein of unknown function are involved. Although Mika & Rost (2006) found that this can only be done at very high sequence identities, it is has been attempted with some success (Yu et al., 2004).

### 1.7.4. Inferring Functional Associations through Gene Expression Analysis

Gene products involved in the same complex or pathway commonly have similar expression patterns (Grigoriev, 2001). Complexes form in the cells of particular tissues at particular stages of development, or at particular points in the cell-cycle, for instance, with a certain stoichiometry. For complexes to function efficiently their components should be expressed at the same time. Microarray datasets over a given time course or across different tissues have allowed the determination of proteins with a correlated expression profile which are involved in common processes or protein complexes (Jansen et al., 2002).

### 1.7.5. Inferring Functional Associations Using Genome Context Methods

Genome context methods exploit the availability of complete genome sequences and aspects of their evolution to predict groups of proteins which are involved in related biological processes. Gene neighbourhood methods (Dandekar et al., 1998) exploit the fact that interacting or functionally related genes are often close to each other on chromosomes. In bacteria, interacting genes are often located in operons, where genes reside next to each other and

are co-transcribed. Even in eukaryotes, interacting, co-regulated genes often cluster in the genome (Teichmann and Babu, 2002).

Phylogenetic profile methods (Pellegrini *et al.*, 1999; Ranea *et al.*, 2007) are based on the supposition that pairs of proteins which are both present or both absent in the same subset of organisms are functionally related. It is assumed that both are required for some particular function and that one or other alone confers no selective advantage.

The gene fusion method, variations of which have been termed Rosetta Stone (Enright et al., 1999) or domain fusion, lead on from gene neighbour methods. It represents a more robust way of finding protein-protein interactions and functional linkages (Enright and Ouzounis, 2001). Chapter 3 concerns the development of a novel approach to domain fusion and this approach, as well as the background behind it, is discussed in detail in that chapter.

## 1.7.6. Resources of Genome Context Data

### *1.7.6.1. STRING*

Search Tool for the Retrieval of Interacting Genes/proteins (STRING; von Mering et al., 2007) is a resource from the Bork group which provides an integration of experimentally derived and predicted protein-protein interactions and functional associations. Genome context methods such as conserved neighbourhood, gene fusion and phylogenetic profiling are combined with co-expression analyses and experimentally-determined PPIs using an integrated scoring scheme.

Importantly, much of the functional inference used in STRING is based on orthology. Orthologues are more likely to play the same role in different organisms than non-orthologues, resulting in more accurate annotation.

### *1.7.6.2. Prolinks*

Prolinks (Bowers et al., 2004) is a resource developed by the Eisenberg group based firmly on the idea of functional linkages rather than PPIs and is therefore in contrast to STRING. Phylogenetic profiling, gene clusters, gene

neighbourhood and gene fusion methods are represented as well as text mining. Prolinks uses combinatorial probabilities to determine whether the proteins are linked by chance, enabling a ranking of their results.

### *1.7.6.3. FusionDB*

FusionDB (Suhre and Claverie, 2004) is based solely on the gene fusion method. Annotations are inherited to orthologues, increasing coverage. Several statistical measures of the fusions are provided, largely based on the alignments between query, target and fusion proteins.

### *1.7.6.4. Predictome*

Predictome (Mellor et al., 2002) is a resource from the DeLisi group which combines phylogenetic profiling, chromosomal proximity, domain fusion and experimentally derived protein-protein interaction data. Much like FusionDB and STRING, orthologous relationships are used to inherit annotations.

## 1.8. Protein-Protein Interaction Networks and Complexes

Protein-protein Interaction Networks (PINs) have become an important focus of molecular biology. With increasing numbers of complete genomes it has become clear that the number and variety of genes is not sufficient to explain organismal complexity (Stumpf et al., 2008). It is thought that understanding the interactions involved in PINs as well as transcriptional and metabolic networks will bring us closer to understanding the complexity we see in biology. Correspondingly there have been many recent efforts to produce datasets describing which proteins interact.

Protein-Protein Interaction (PPI) data is often considered as a graph or network. Graphs consist of nodes or vertices, connected by links or edges. In the case of PINs, the proteins are most often modelled as the nodes, with edges representing interactions between proteins. This mathematical formalism is useful both as a visual representation but also because graphs have been studied for many years, beginning with Euler (1741), and there are a range of mathematical tools for analysing them. The edges in PINs often have associated weights representing various attributes such as confidence in an interaction (Pereira-Leal et al., 2004; von Mering et al., 2003).

The clustering coefficient of a graph is a measure of how well connected it is. PPI graphs have a high clustering coefficient compared to random graphs which suggests there is a signal which might relate to protein complexes or other functional groupings (Hartwell et al., 1999). There is, however, some debate as to whether this is really the case. It has been argued that the observed clustering in PPI networks does not relate to complexes or other functional groupings but is merely an artefact (Wang and Zhang, 2007). It is further argued that various other properties of these (admittedly incomplete) networks are in fact qualitatively different from those of the true network due to sampling bias (de Silva et al., 2006). On the other hand, it has

been shown that PINs can be decomposed to accurately identify known complexes (Brohee and van Helden, 2006).

In Chapter 4 PINs are used to identify protein complexes in the prokaryote *E. coli* and the single-celled eukaryote *Saccharomyces cerevisiae*. CATH domain superfamily annotations are used to study their evolution.

### 1.8.1. Experimental Approaches to Determine Protein-Protein Interactions

Perhaps the highest quality dataset of protein-protein interactions is that found in the Protein Quaternary Structure database (PQS; Henrick and Thornton, 1998). The PQS is based on crystallographic data from the PDB. The relationships between different PDB chains are computationally adjusted to better represent true quaternary structure. It provides great detail on the residues involved in interactions; however, it has very low coverage of genomes and is probably very biased towards stable interactions.

Much of the PPI data that has become available has been produced using Tandem Affinity Purification Mass Spectrometry (TAP-MS; Puig et al., 2001) and Yeast-2-Hybrid (Y2H; Fields and Song, 1989) experiments but it can also derive from low-throughput techniques, literature mining, genome-context associations and others. TAP-MS is the current state of the art high throughput approach for determining protein complexes. This procedure uses individual proteins as bait to fish for interacting proteins, as well as further proteins which bind to the direct interactors. The components of these complexes are then identified by mass spectrometry. The Y2H method determines whether individual bait and prey proteins interact by hybridising them to reporter proteins in a yeast system.

### 1.8.2. Resources of Protein Interaction Data

Experimental PPI datasets are available from several sources. IntAct contains data from 8576 distinct experiments and publications for $>10^5$ interactions, mostly from yeast, human, fly and *E. coli* largely based on Y2H and two-hybrid array methods (Kerrien et al., 2007). MINT also contains $>10^5$

interactions mostly from Y2H and TAP experiments (Chatr-aryamontri et al., 2007). DIP contains $5.7x10^4$ interactions from $6.4x10^4$ experiments, principally in fly and yeast (Salwinski et al., 2004).

Several databases contain domain-domain interactions based on CATH, SCOP or Pfam domains in structural databases (Jefferson et al., 2007; Stein et al., 2005; Finn et al., 2005). However these account for only a small proportion of known PPIs (Schuster-Bockler and Bateman, 2007).

Several resources compile data on predicted interactions. Online Predicted Human Interaction Database (OPHID; Brown and Jurisica, 2005) provides predicted interactions for humans, and STRING (von Mering et al., 2007) provides integrated sources of this data for many species. Human Protein-protein Interaction Prediction (PIPs; McDowall et al., 2008) contains a large number of predicted interactions for human which are integrated to identify the most likely interactions.

### 1.8.3. Resources of Protein Complex Data

Several resources provide data on complexes as opposed to protein-protein interactions although there is not necessarily a clear distinction in some cases. Data from TAP experiments for instance can be considered as PPI data or complex data.

The Munich Information centre for Protein Sequences (MIPS; Mewes et al., 2008) provides a manually curated set of complexes for *Saccharomyces cerevisiae* and EcoCyc (Karp et al., 2007) provides a similar dataset for *Escherichia coli*. 3D complex (Levy et al., 2006) provides a database of complexes for various species derived from the PQS database.

## 1.9. Overview of Thesis

In this thesis protein domain families are used to explore the function and evolution of proteins. Structural domains from CATH are exploited to identify the best approaches to determine homologous relationships between proteins (Chapter 2). The annotation derived from the use of these methods is subsequently employed to develop a method to identify functional relationships between proteins, based on the domain fusion hypothesis , using Pfam domain families (Chapter 3). Lastly, CATH domain family annotations are employed to identify differences in the evolution of protein complexes between a prokaryote and a eukaryote (Chapter 4).

### 1.9.1. Chapter 2

In recent years several novel methods for detecting evolutionary relationships between protein domain families have been developed (Madera, 2006; Soding, 2005; Sadreyev and Grishin, 2003). Whereas previous approaches have compared single sequence to families, these profile-profile methods compare two families. The result is that more distant evolutionary relationships can be detected. In fact it was shown that such methods detect evolutionarily relevant similarities between families which are classified as non-homologous in structural databases (Soding, 2005). Benchmarking approaches for methods of homologue detection utilise such evolutionary relationships and therefore a novel modification to these benchmarks is introduced in Chapter 2. Furthermore, the relative performance of cutting edge methods is established.

After establishing the relative performance of each method, a consensus approach is introduced to integrate the different methods and improve accuracy in predicting homologous relationships.

### 1.9.2. Chapter 3

Genome context methods allow the prediction of functional associations between non-homologous proteins. They have been shown to be useful in

identifying proteins involved in common pathways and complexes as well as direct interactions (Huynen et al., 2000). One of these methods exploits the fact that two interacting proteins, encoded by separate genes, sometimes have orthologues in another species which are fused into a single gene. The detection of such gene fusion events thus allows the identification of a functional link between the separately encoded genes (Marcotte *et al.*, 1999; Enright *et al.*, 1999). Such methods have been shown to be accurate in prokaryotes and simple eukaryotes, however higher eukaryotes have much larger gene families which can lead to many incorrect predictions (Marcotte and Marcotte, 2002).

In Chapter 3 a new method named Co-Occurrence of Domains Analysis (CODA) is introduced with the aim of accurately identifying functional associations between proteins, using gene fusion, in the human genome.

### 1.9.3. Chapter 4

Currently, little is known about the evolution of protein complexes. This is largely due to a paucity of data describing such complexes. Much of the work so far has concerned only *S. cerevisiae* where there is far more data available than for other species. In Chapter 4, protein complex datasets are created for *S. cerevisiae* and *E. coli* to determine whether differences exist in the evolution of their protein complexes.

# Chapter 2 Benchmarking Sequence-Based Methods of Remote Homologue Detection

## 2.1. Introduction

### 2.1.1. Sequence-Based Methods of Remote Homologue Detection

The identification of remote homologues is a central problem in bioinformatics. New tools to accomplish this task appear frequently and it is essential that are they rigorously benchmarked against a range of other software, under varying conditions. Benchmarking is crucial both to determine the best tool for a particular job and to determine a discriminating E-value threshold.

Brenner et al. (1998) showed that sequence-sequence (often termed *pairwise*) methods such as BLAST can detect most relationships between proteins with >30% sequence identity. Park et al. (1998) showed that profile-sequence methods of remote homology detection could find three times as many homologues as sequence-sequence methods at sequence identities below 30%. These profile-sequence methods, including HMMer (Eddy, 1996), SAM (Karplus et al., 1998) and PSI-BLAST (Altschul et al., 1997) have become widely used for detecting remote homologues. More recently, profile-profile methods have been introduced which use a profile to search a database of profiles. These exploit evolutionary information in both the query and the target and thus more remote relationships can be detected. Such methods

include COMPASS (Sadreyev and Grishin, 2003), prof_sim (Yona and Levitt, 2002), LAMA (Pietrokovski, 1996), PRC (Madera, 2008) and HHSearch (Soding, 2005). Of these methods, COMPASS, HHSearch and PRC are examined in this chapter. COMPASS aligns profiles against profiles, whereas HHSearch and PRC align HMMs. All three use a log-sum-of-odds score; a generalisation of the log-odds score used by sequence-profile methods. Optimum alignments are determined by COMPASS using the Smith-Waterman algorithm, whereas HHSearch uses the Viterbi algorithm and PRC can use either the Viterbi or forward algorithms. All three methods use distribution fitting to calculate E-values, for this HHSearch requires a calibration step.

There has not previously been a comprehensive benchmark across commonly used sequence-sequence, sequence-profile and profile-profile methods. One study (Ohlson *et al.*, 2004) considered Smith-Waterman, PSI-BLAST and several profile-profile approaches but did not assess those freely available for use by researchers. It is important to determine the relative performance of the newest methods in specific remote homology detection tasks in order to make an informed choice of which methods should be used for different tasks.

### 2.1.2. Benchmarking Sequence-Based Methods of Remote Homologue Detection

The fundamental requirement of a benchmark for remote homology detection is a gold standard dataset of known evolutionary relationships between protein domains. Previously, 3D structure comparison has been shown to detect more distant evolutionary relationships than sequence comparison (Chothia and Lesk, 1986). Thus to date, classifications of domain structure have been exploited in benchmarking sequence-based remote homology detection methods. SCOP (Murzin et al., 1995), FSSP (Holm and Sander, 1994) and CATH (Greene et al., 2007) have all been used in this context by, for example, Park et al. (1998), Sadreyev & Grishin (2003) and Sillitoe *et al.* (2005) respectively. Bateman & Finn (2007) used Pfam clans

(Finn et al., 2006), which are based on a mixture of sequence and structural evidence.

Surprisingly, poor performance has been reported for profile-profile methods using benchmarks based solely on the SCOP structural classification (Soding, 2005). On close inspection, it was shown that this was due to potentially homologous domains which had been classified as unrelated. This occurred largely due to a previous lack of evidence for homology between these domains. Two domains are homologous if they have descended from a single ancestral domain. Over time homologues tend to diverge in both sequence and structure. In general, it is easier to find similarity in their structures than in their sequences, particularly when they have diverged greatly. However, structural similarity alone is not sufficient to guarantee homology as there may be physical constraints on folding and limited topologies available. Therefore, it is necessary to build up several lines of evidence to describe domains as homologous. CATH or SCOP initially group structurally similar domains into fold groups. Within fold groups, homologous relationships are subsequently recognized using one or more elements of independent evidence. For example, statistically significant sequence similarity or experimental verification of functional similarity (e.g. from the literature, bound ligands or identification of common catalytic residues).

Recent analyses of CATH have shown that homologues can diverge considerably in their structures (Reeves et al., 2006) and in some families a 5-fold or more variation in size is observed between extremely distant relatives. It is now apparent that the most highly sensitive profile-profile methods are detecting significant sequence patterns suggestive of homology between domains which are highly structurally divergent (Soding, 2005). In these cases, any structural similarity would fall below the stringent automatic thresholds currently used for classifying homologues in CATH and might be missed on manual inspection. These thresholds on structural similarity were determined empirically based on earlier analyses of homologous families,

but the more distant relationships recently revealed by profile-profile methods (Sadreyev et al., 2007) are prompting a re-examination of these thresholds. Consequently, the status of these putative homologues is uncertain and they should therefore not be considered as definitively non-homologous when benchmarking methods for remote homology detection.

β-propellers, for example, are thought to have evolved by duplication of β-sheet blades and inheritance of blades between structures (Jawad and Paoli, 2002). Their evolution is therefore somewhat unusual as structural variation can occur in the core of the domain making superposition of relatives difficult. Gough et al. (2001) identified such examples in SCOP by examining false positives produced by SAM and provide a modified SCOP-based benchmark (http://sup-fam.mrclmb.cam.ac.uk/SUPERFAMILY/ruleset_1.65\.html). Soding showed that such examples could be accounted for automatically by using a combination of sequence and structural evidence which is more powerful than taking either measure in isolation (Soding, 2005). He used a measure for local structural similarity to avoid similar problems in SCOP when benchmarking HHSearch. False positive hits were excluded by using a score cut-off from the MaxSub structural comparison algorithm (Siew et al., 2000). The score cut-off was chosen such that it represented significant structural similarity, but was not optimised for the task.

In addition to the choice of structural classification used for benchmarking, test sets of varying difficulty affect the separation of remote homologue detection methods. At high sequence identities sequence-sequence and profile-sequence methods detect a more similar number of true relationships than at lower sequence identities. Park et al. (1998) and Sillitoe et al. (2005) used <40 and <35% non-redundant (nr) test sets respectively for benchmarking profile-sequence methods. Soding (2005) used a <20% nr set of query sequences to benchmark the same type of method. Casbon and Saqi (2006) used a SCOP dataset where homologous pairs had <10% sequence identity in benchmarking HHSearch. Furthermore, benchmarking methods

differ in their definition of false positives. Yona and Levitt (2002) consider matches between members of the same SCOP/CATH superfamily to be true and anything else false. Park et al. (1998), Sillitoe et al. (2005) and Soding (2005) consider matches between members of the same fold but differing superfamily to be ambiguous and therefore discounted. Sadreyev and Grishin (2003) used a definition based on FSSP Z-scores. Muller et al. (1999) consider fold matches to be false in benchmarking PSI-BLAST for genome annotation. Aside from Muller et al. these benchmarks all focus on a general remote homology detection task, i.e. distinguishing homologues from non-homologues. However, remote homologue detection is frequently used to annotate genomes. To date, there has been no work comparing the abilities of different methods to annotate genomes with structural domains (Muller et al. give no comparator for PSI-BLAST).

### 2.1.3. Aims

To date, there has been no coherent benchmark encompassing the wide range of methods now available for sequence-based remote homology detection. Although Bateman and Finn (2007) benchmarked a range of profile-profile methods using Pfam clans, there is no comparison with profile-sequence methods and Pfam clans provide a relatively sparse test set compared to CATH or SCOP. The various benchmarks described in the literature differ significantly and are not easily comparable. Here, the relative performance of seven methods from among the sequence-sequence, profile-sequence and profile-profile classes is explored in greater depth than previous work. This chapter aims to measure the effect of datasets and benchmarking strategies when assessing the performance of homologue recognition methods. Two types of benchmark are performed, reflecting different applications of remote homology detection. The *allpos* benchmark models the general task of separating homologues from non-homologues whilst the *tophit* benchmark captures the task of annotating proteomes with structural domains.

In order to account for the erroneous high error rate of profile-profile methods, a heuristic filter is introduced to exclude false positives with low E-values and high structural similarity. This approach is compared to a set of exceptions defined by expert analysis.

Park et al. (1998) have shown that different homologue detection methods identify similar sets of homologues and dissimilar sets of false positives. This makes intuitive sense in two ways. Firstly, the methods have been trained to detect the same types of homologues, while false hits are the result of imperfections of the particular approach or implementation. Secondly, there are more false positives to choose from. It has been shown that in some cases taking the union of the results from two sequence-sequence methods can give improved coverage (Webber and Barton, 2003).

## 2.2. Methods

### 2.2.1. Datasets for Benchmarking Homologue Recognition Methods

The broadest dataset (nr35) consists of representative sequences from CATH v3.0.0. Initially, all CATH sequences were clustered into families at 35% sequence identity with an 80% residue overlap cut-off, by directed multi-linkage clustering. Cluster representatives were chosen by picking the member with the highest resolved structure and with a length closest to the average. This set contains many pairwise relationships which are trivial examples of remote homologues in the sense that they can be detected by sequence-sequence methods (i.e. BLAST). To better examine the ability of the most sensitive methods, a dataset clustered at 10% sequence identity (nr10), was created from the nr35 set. For this dataset a 60% overlap cut-off was used to take account of the greater length diversity between homologues at low sequence identity. Cluster representatives for the nr10 dataset were selected by greatest length. Both datasets were filtered to exclude sequences with no superfamily partner.

### 2.2.2. Profile and Model Building

Each sequence in the nr35 dataset was used as a seed for the SAM3.4 target2k program (Karplus et al., 1998) to build a Hidden Markov Model (HMM) representing the superfamily of that seed. This procedure initially performs a BLAST on the GenBank non-redundant database of all known protein sequences (at max E-value 400) to produce a reduced-size database within which to detect homologues. An iterative HMM procedure then builds an alignment, using more and more relaxed E-values, to an E-value of 0.005. These alignments were used for benchmarking PSI-BLAST. The alignments were filtered to remove positions aligned to gaps in the seed sequences before being used to build COMPASS profiles using mk_compass_db (part of the COMPASS software), as recommended by Ruslan Sadreyev (*personal*

*communication*). The SAM3.4 program w0.5 was used to build HMMs from the original alignments and these were used for benchmarking SAM. The SAM models were converted to HMMer format using Martin Madera's *convert.pl* script (http://www.mrc-lmb.cam.ac.uk/genomes/julian/convert/convert.html) and calibrated with 1000 random sequences using the *hmmcalibrate* program, which is part of the HMMer package. These HMMer models were used in benchmarking both HMMer and HHSearch. PRC was benchmarked using the SAM models converted to PRC format using the *convert_to_prc* program (provided with PRC).

### 2.2.3. Benchmarking Procedure

Both datasets (nr35 and nr10) were scanned all against all using each method. In the case of BLAST, sequences were scanned against sequences, for SAM/HMMer this was HMMs against sequences, for COMPASS profiles against profiles, for PRC/HHSearch HMMs against HMMs. For PSI-BLAST, profiles were scanned against sequences, allowing up to 20 iterations. HHSearch was used without structural information.

Local or local-local scoring was used throughout. Although domains are being compared in the benchmark, performance in annotating genomes and detection of very remote relationships within families was being assayed, for which local scoring was most appropriate. As suggested by Madera & Gough (2002) other parameters were defaults, such that the relatively inexperienced user can achieve the same performance. Equally, the methods presented here are those most easily available for download from the World Wide Web.

The rules of the benchmark were based on the CATH domain structure classification. CATH classifies protein domain structures, principally into Topological groups (T level) and Homologous Superfamilies (H level). Superfamilies consist of domains that are thought to be homologous using several lines of evidence, i.e. common structure, similar sequences and/or

functional similarity. If a method of remote homology detection matches two members of the same superfamily it was considered to have recognised a true relationship. Matches between members of different superfamilies that were within the same topology were treated as ambiguous and were excluded. Matches between superfamilies that were not within the same topology were counted as false hits. The benchmark involved an all against all search within each dataset and it was therefore necessary to exclude trivial matches between a query and itself, or the profile/model for which it was the seed.

Two different rules for counting true hits were implemented. The *allpos* rule included every true, non-trivial relationship. This rule reflects how well a method can separate homologues from non-homologues. The *tophit* rule included only the best-scoring, non-trivial hit for each query. This rule simulated the case of genome annotation in which only the best scoring hit is generally considered.

### 2.2.4. Exceptions to the Rule

A set of expert-curated exceptions to the CATH superfamily classification was determined by examining the structural superpositions, sequence alignments, functional annotation and literature relating to all false positives (matches between members of different topologies) incurred by PRC on the nr35 dataset up to an E-value of 0.1. Valid exceptions were those determined by this manual validation to be either true homologues, or fold matches, despite current CATH classification. Reasons for such apparent misclassifications are discussed in the results section. Incorporating these curated exceptions directly into the benchmark could however unfairly bias the results in favour of PRC. Therefore the effectiveness of a heuristic approach which could accurately reproduce these exceptions and also be applied *de novo* to any homologue detection method was explored.

The structural alignment program SSAP (see 1.3.7.1), scored with the SAS score (Equation 1.3), was benchmarked against CATH in the same way

as described for the sequence-based methods, using the nr35 dataset and the *allpos* rule. The SAS (Subbiah et al., 1993) score was varied to determine a cut-off which produced good agreement with the curated exceptions.

### 2.2.5. Coverage versus Error Plots

For each method, hits from the all against all scan were sorted by E-value and for successive 10-fold E-value cut-offs, the coverage and error rate were plotted. This is somewhat like the traditional ROC curve, which plots the proportions of true and false positives. Here however, error rate (or Errors Per Query, EPQ) is the number of false positives divided by the total number of false positives and true positives for a certain E-value. This gives a more intuitive reading of the results than plotting the proportion of total errors in the dataset, or the raw number of false positives. For instance, at 0.05 EPQ the results comprise 5% false positives and 95% true positives. Coverage, on the y axis, shows the proportion of true positives found for a particular E-value. Using the *allpos* rule this was calculated using the total number of pairwise homologues in the dataset. For the *tophit* rule the number of queries was used, as in this case, a maximum of one true positive could be found per query.

### 2.2.6. Combining Different Methods to Increase Specificity

A simple approach was employed to combine the results of multiple methods. All against all hits up to an E-value of 10 for two or more methods were compared. Hits were discarded unless all methods agreed on that hit. For those remaining hits, the Combined E-Value (CEV) for each was calculated as in Equation 2.1.

$$CEV = 10^{\frac{\sum \log E_{1..n}}{n}}$$

Equation 2.1 Combined E-value

In Equation 2.1 $E_{1..n}$ are the E-values from each method for a specific pairwise hit and $n$ is the number of methods.

All possible permutations of methods (excluding BLAST) were subjected to this analysis, using different datasets and the *allpos* rule. Only permutations of profile-sequence methods were used with the *tophit* rule as profile-profile methods are on the whole too computer-intensive for annotating whole genomes and profiles are not nececssarily available for genomic sequences.

# 2.3. Results

## 2.3.1. A Heuristic Rule to Improve Benchmarking of Sequence-Based Methods of Remote Homologue Detection

Profile-profile, sequence-based methods for remote homologue detection have previously been found to detect relationships between domains which are classified in different superfamilies in SCOP but are potentially homologous on close inspection (Soding, 2005). In these cases, homology may be confirmed by identifying structural similarity or evidence of functional similarity in combination with the significant sequence similarity already detected. Therefore it was necessary to create a general, heuristic rule which would allow these ambiguous relationships to be identified and excluded from a benchmark of sequence-based methods of remote homology detection. The validity of a heuristic method based on structural comparison scores was explored, similar to Soding's approach of using a MaxSub cut-off (Soding, 2005) but benchmarked against manually defined examples.

For all domain pair matches (identified by PRC on the nr35 dataset up to an E-value of 0.1) involving different CATH topologies, manually curated exceptions to the CATH classification were determined by examining structural superpositions, sequence alignments, functional annotations, catalytic residues and the literature. Those pairs with several lines of evidence to suggest a common ancestor were considered exceptions. The majority of these fell into six classes which are shown in Table 2.2. Several other examples were revealed as errors in CATH and were reclassified. The aim was to make a fairer benchmark by excluding these putative homologues. For the heuristic rule, the SSAP structural comparison algorithm was used (Orengo and Taylor, 1996) with the SAS score (See Chapter 1; Subbiah et al., 1993) to score structural alignments. The SAS score has proven to be a better discriminator at the fold level than native structural comparison scores (Kolodny et al., 2005).

| Exception class | Frequency of pairs |
|---|---|
| FAD/NAD-binding domain (3.50.50) vs. Rossman fold (3.40.50) | 75.1% (1674) |
| Neuraminidase (2.120.10) vs. Methylamine Dehydrogenase (2.130.10) | 12.5% (279) |
| Methanol Dehydrogenase (2.140.10) vs. Methylamine Dehydrogenase (2.130.10) | 4.3% (96) |
| PCNA (3.70.10) vs. Leucine-rich repeat (3.80.10) | 1.3% (30) |
| Neuraminidase (2.120.10 )vs. Methanol Dehydrogenase (2.140.10) | 0.8% (17) |
| Tachylectin-2 (2.115.10) vs. Neuraminidase (2.120.10) | 0.2% (4) |
| Total | 100.0% (2100) |

Table 2.1 Classes of curated exceptions for PRC on nr35 dataset at E-value cut-off of 0.01.

Percentages are the proportions of curated exceptions falling in that class. The CATH codes of each class are shown in brackets. The percentages are based on the curated exceptions.

Figure 2.1 shows how accurately the manually curated exceptions were captured by using a heuristic based on SSAP structural alignment and SAS scores. A SAS score of 8 gave a coverage of 0.86 of the curated exceptions with 0.12 EPQ. Lower thresholds resulted in much poorer coverage of the manually curated exceptions while higher thresholds caused a rapid increase in errors with relatively little gain in coverage. The errors are explored in detail below.

Figure 2.3 shows that the performance of PRC when excluding false positives with a SAS score of 8, 9 or 10 was very similar to that achieved using the manually curated exceptions rule. SAS8 is, however, the most appropriate rule since it was less error prone than SAS9 with no significant loss of coverage. Although SAS9 achieved closer performance to the curated exceptions in a benchmark, fitting less closely to PRC should reduce the bias incurred by benchmarking the exceptions solely on this method.

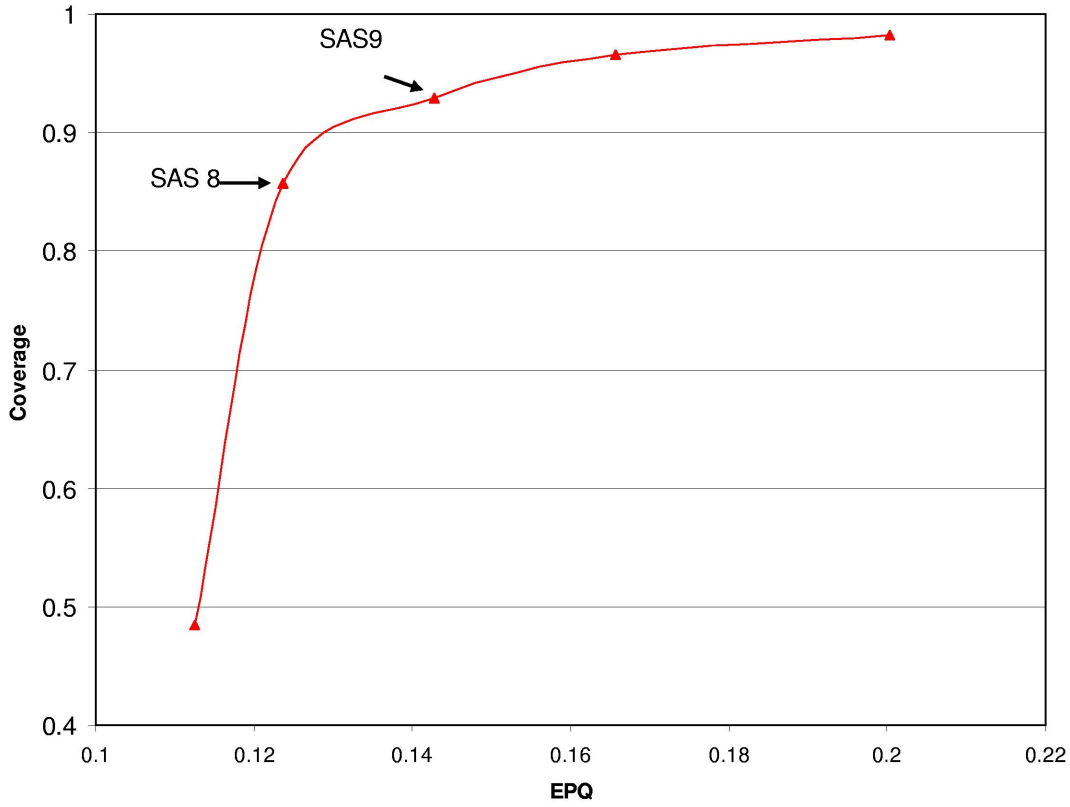



Figure 2.1 Accuracy in reproducing manually curated exceptions using heuristic rule with varying SAS score.

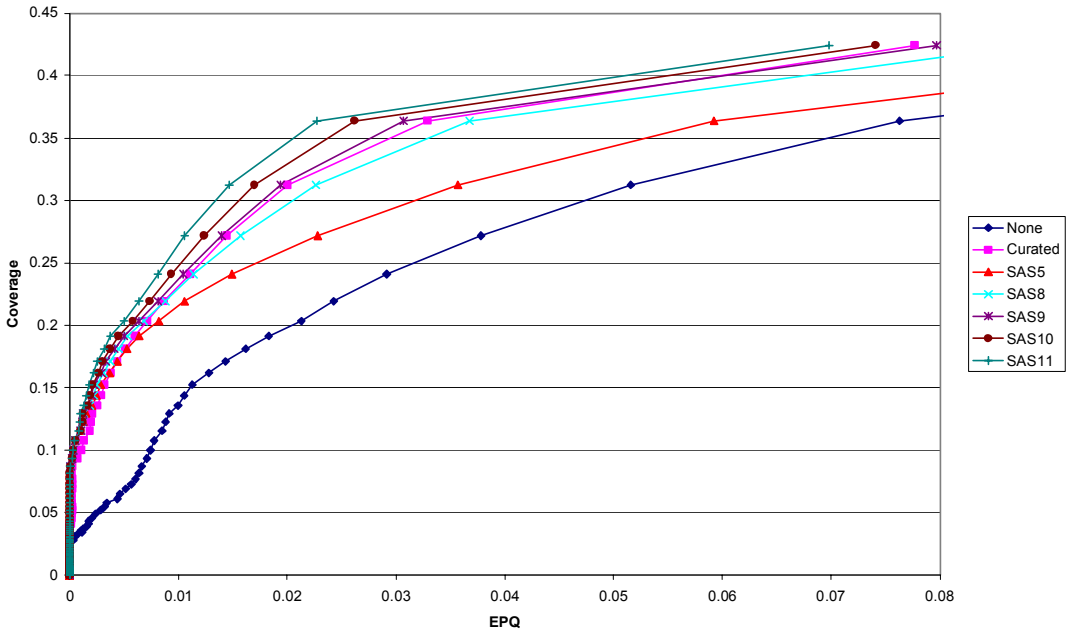



Figure 2.3 Performance of PRC assessed with no exceptions, using the manually curated exceptions or using the heuristic rule (at different SAS thresholds, with no overlap threshold).

This benchmark was performed using the nr35 dataset and the *allpos* rule.

Table 2.2 shows in detail how the *SAS8* exceptions heuristic closely reproduced the curated exceptions. By far the most common matches between putative homologues, for both curated and *SAS8* exceptions, were between the FAD/NAD(P) binding domain fold (3.50.50) and the Rossman fold (3.40.50), an example of which is shown in Figure 2.5a. These folds are from the CATH ββα and αβα sandwich architectures respectively. Previous analyses have suggested that these may be very remote homologues (Harrison et al., 2002) and there is evidently common structure. The SAS8 heuristic captures all of the β-propeller exceptions (2.115, 2.120, 2.130 and 2.140 architectures, e.g. Figure 2.5b) and all αβ box/horseshoe exceptions (3.70 and 3.80 architectures). The αβ box/horseshoe exceptions were due to misclassification in CATH and this has since been rectified. 87.6% of the heuristic exceptions were accounted for by the curated exceptions. Several smaller classes of exception were noted during manual curation, however these were either re-classified in CATH or appeared at E-values >= 0.01.

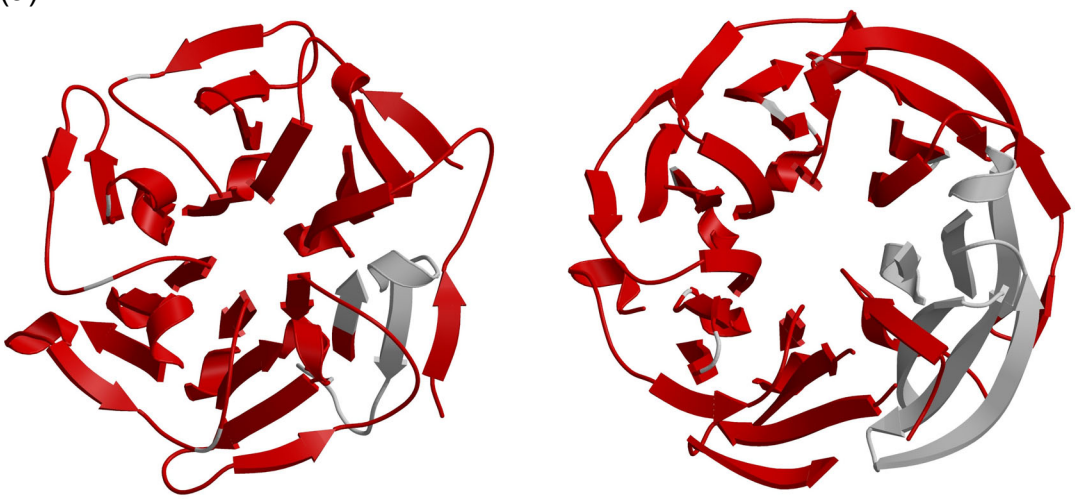
Several exceptions identified by the SAS8 rule were not recognised by manual curation. The three most frequent classes, comprising around half of these errors are discussed here in detail. The Aminoglycoside 3'-phosphotransferase (3.90.1200) vs. Phosphotransferase (1.10.510) class is the first of these. Both topologies comprise a single superfamily, each implicated in protein kinase activity. Only two members of the 3.90.1200.10 superfamily were in the nr35 dataset and one of these (1nd4A01) is in the process of being reclassified in CATH. The other (2bkkC02) did show some local structural similarity to members of the 1.10.510.10 superfamily following superposition (see Figure 2.5c). For the best match (with 1wbsA02), the SAS score was 6.37 and the PRC E-value is 1.7e-5. The topologies are clearly different however and these are therefore not valid exceptions.

| Exception class | SAS8 exceptions as percentage of curated exceptions |
|---|---|
| FAD/NAD-binding domain (3.50.50) vs. Rossman fold (3.40.50) | 81.9% (1371) |
| Neuraminidase (2.120.10) vs. Methylamine Dehydrogenase (2.130.10) | 100.0% (279) |
| Methanol Dehydrogenase (2.140.10) vs. Methylamine Dehydrogenase (2.130.10) | 100.0% (96) |
| PCNA (3.70.10) vs. Leucine-rich repeat (3.80.10) | 100.0% (30) |
| Neuraminidase (2.120.10 )vs. Methanol Dehydrogenase (2.140.10) | 100.0% (17) |
| Tachylectin-2 (2.115.10) vs. Neuraminidase (2.120.10) | 100.0% (4) |
| Total | 86.0% (1797) |

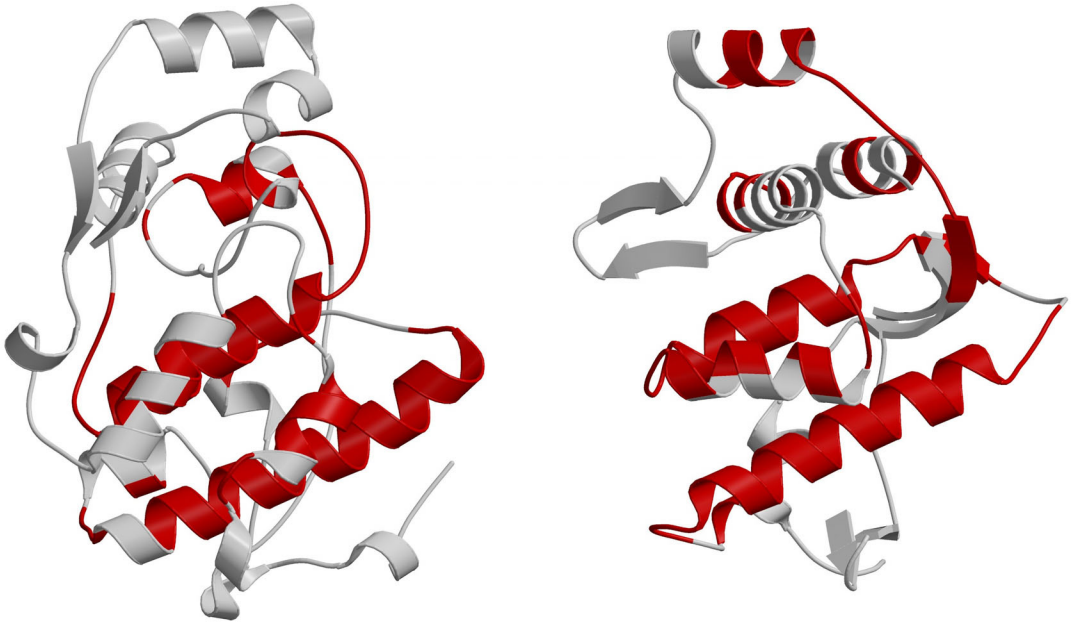
Table 2.2 Classes SAS8 exceptions as percentage of curated exceptions.

The CATH codes of each class are shown in brackets. The percentages are based on the curated exceptions, e.g. 81.9% of curated exceptions for the FAD/NAD-binding domain vs. Rossman fold class were identified using the SAS8 rule. Several small classes of SAS8 exceptions are not shown here, but are discussed in the text.

(a)



(b)



(c)

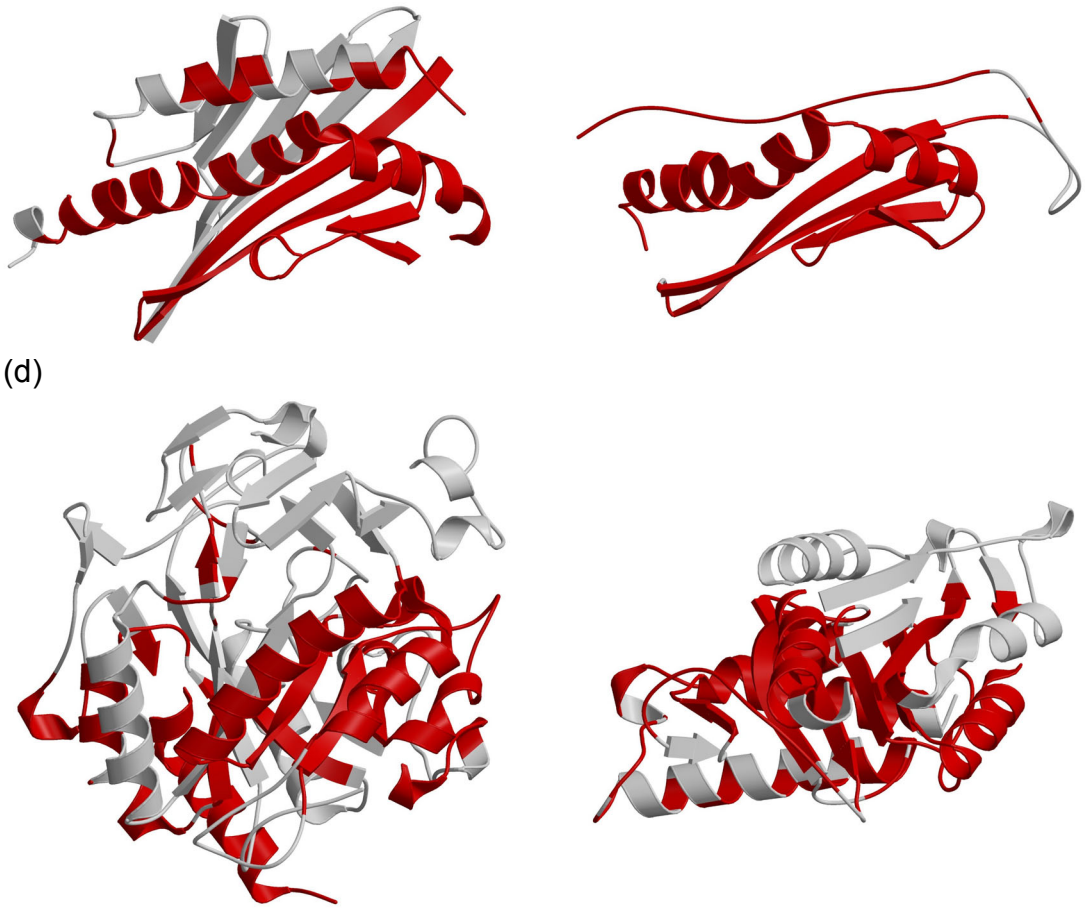



Figure 2.5 Examples of exceptions identified using the SAS8 rule.

(a), (b) and (d) represent putative pairs of homologues whereas (c) and (e) show no structural evidence for homology. Regions shown in red are equivalent positions identified by SSAP. (a) Examples of the ββα and αβα sandwich architectures: 1sezA01 (3.50.50.60.37) and 1gteA03 (3.40.50.720.7), PRC E-value = 2.5e-31, SAS = 2.49. (b) 6-bladed and 7-bladed propellers: 1rwiA00 (2.120.10.30.6) and 1l0qA01 (2.130.10.10.19), PRC E-value = 1.1e-39, SAS = 1.98. (c) Aminoglycoside 3`-phosphatase and phosphotransferase: 1wbsA02 (1.10.510.10.9) and 2bkkC02 (3.90.1200.10.1), PRC E-value = 1.7e-5, SAS = 6.37. (d) Class I and Class II MHC: 1kcgC00 (3.30.500.10.8) and 1ktdB01 (3.10.320.10.5), PRC E-value = 2.6e-08, SAS = 3.17.

(e) TIM barrel and Aspartate aminotransferase: 1p3wA02 (3.40.640.10.24) and 1hx0A01 (3.20.20.80.16), PRC E-value = 0.00051, SAS = 7.96.

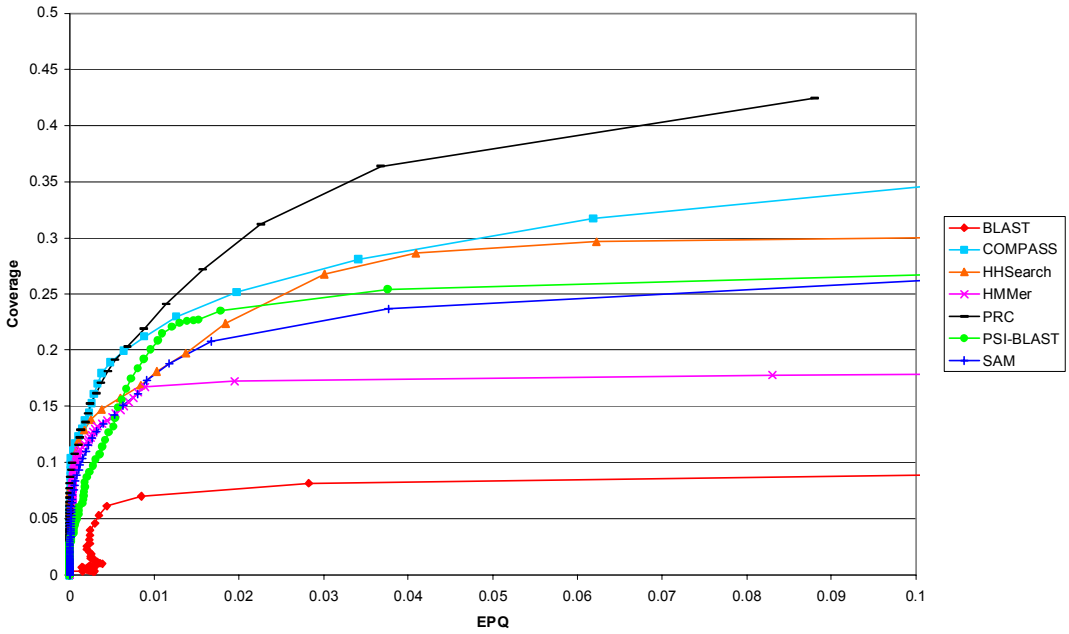
The Class I MHC (3.30.500) and Class II MHC (3.10.320) folds each contain only one superfamily. Most of the domains within each fold hit the other fold below an E-value of 0.01. The full crystal structures (e.g. 1ktd vs. 1kcg) show these folds are highly similar, with very good superposition. However half the domain for the Class II MHC examples is provided by a different chain (see Figure 2.5d). There is homology between the CATH domains, but the functional domain has been split between two chains in one case. This is therefore a very reasonable exception which escaped manual classification.

Matches between the TIM Barrel (3.20.20) and Aspartate Aminotransferase (3.40.640) folds only occur at E-values >0.0005. The best match by PRC (1hx0A01 vs. 1p3wA02, E-value 0.00051) only just makes the SAS score cut-off of 8 with 7.965 and has an RMSD of 15.93. On superposition, members of these superfamilies have no apparent structural similarity other than αβ motifs (see Figure 2.5e). Both superfamilies are large and these matches are probably genuine false positives.

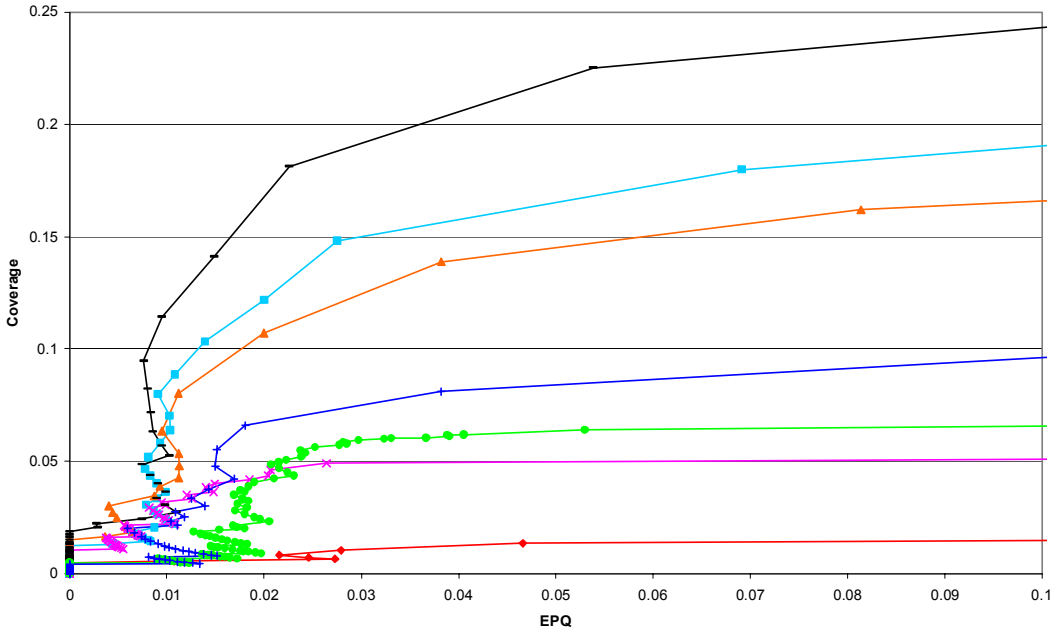
## 2.3.2. Detecting Remote Homologues

### *2.3.2.1. Distinguishing Homologues From Non-Homologues (allpos Rule)*

Seven methods (BLAST, PSI-BLAST, HMMer, SAM, HHSearch, COMPASS and PRC) were benchmarked with each dataset (nr35 and nr10) using the *allpos* scoring rule which captures the ability of the methods to distinguish all homologues from all non-homologues. The *SAS8* exceptions rule was used here and throughout the rest of the chapter to exclude matches between different CATH folds with low E-values and SAS scores of less than 8. Figure 2.7 shows the coverage vs. error plots for these benchmarks on nr35 (a) and nr10 (b) datasets, while Table 2.3 gives selected coverage values for varying error rates.




(a)




(b)

Figure 2.7 Performance of all methods using the *allpos* and SAS8 rules on the nr35 (a) and nr10 (b) datasets.

Note that runs were performed to an E-value of 10 and in some cases an E-value of 10 is reached at an EPQ of <0.1.

| Data-set | EPQ | Coverage | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | BLAST | COM-PASS | HH-Search | HMMer | PRC | PSI-BLAST | SAM |
| nr35 | 0.01 | 7.2 | 21.8 | 18.0 | 16.9 | 22.9 | 20.4 | 17.8 |
| | 0.05 | 8.6 | 30.4 | 29.3 | 17.6 | 38.8 | 25.9 | 24.4 |
| | 0.10 | 9.2 | 34.8 | 30.2 | 17.9 | >43.2 | 27.1 | 26.4 |
| nr10 | 0.01 | 0.5 | 8.2 | 7.5 | 3.0 | 11.8 | 0.5 | 2.1 |
| | 0.05 | 1.2 | 17.0 | 14.7 | 5.1 | 22.1 | 6.4 | 8.8 |
| | 0.10 | 1.5 | 19.1 | 16.8 | 5.2 | 25.2 | 6.6 | 10.0 |

Table 2.3 Percent coverage for each method at 0.01, 0.05 and 0.1 EPQ, using the allpos rule.

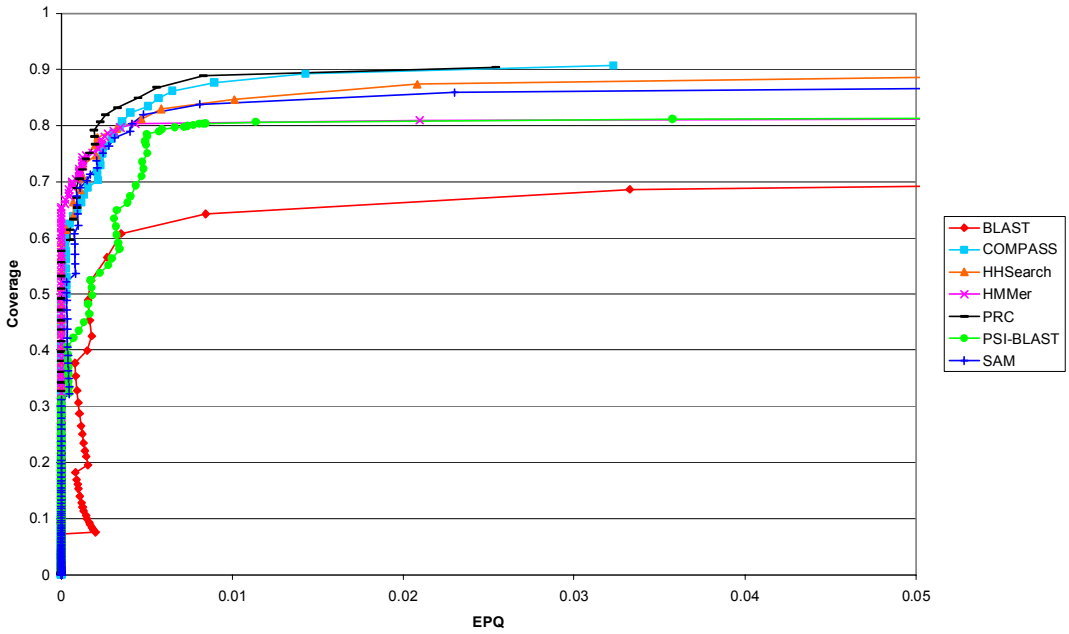
For each EPQ value, the maximum coverage obtained is plotted. '>' means that the maximum E-value of 10 had been passed and this was the last value.

It has been noted previously in similar benchmarks that profile-sequence methods achieve up to three times more coverage than BLAST for a fixed error rate when considering homologous pairs with sequence identities <30% (Park et al., 1998). This was confirmed here with PSI-BLAST (25.9%) and SAM (24.4%) achieving three times greater coverage than BLAST (8.6%) on the nr35 dataset. All profile-profile methods are better than all profile-sequence methods on all datasets at an error rate of 0.05 EPQ. For low error rates (e.g. 0.01 EPQ) PSI-BLAST and SAM achieve similar performance to profile-profile methods on the nr35 dataset. On all datasets PRC is the best method, performing almost 2.5 times better than the best profile-sequence method at 0.05 EPQ on the difficult nr10 dataset (22.1% coverage vs. 8.8% for SAM). This almost equals the increase in performance seen for profile-sequence methods over BLAST, although only on very remote homologues (<10% sequence identity).

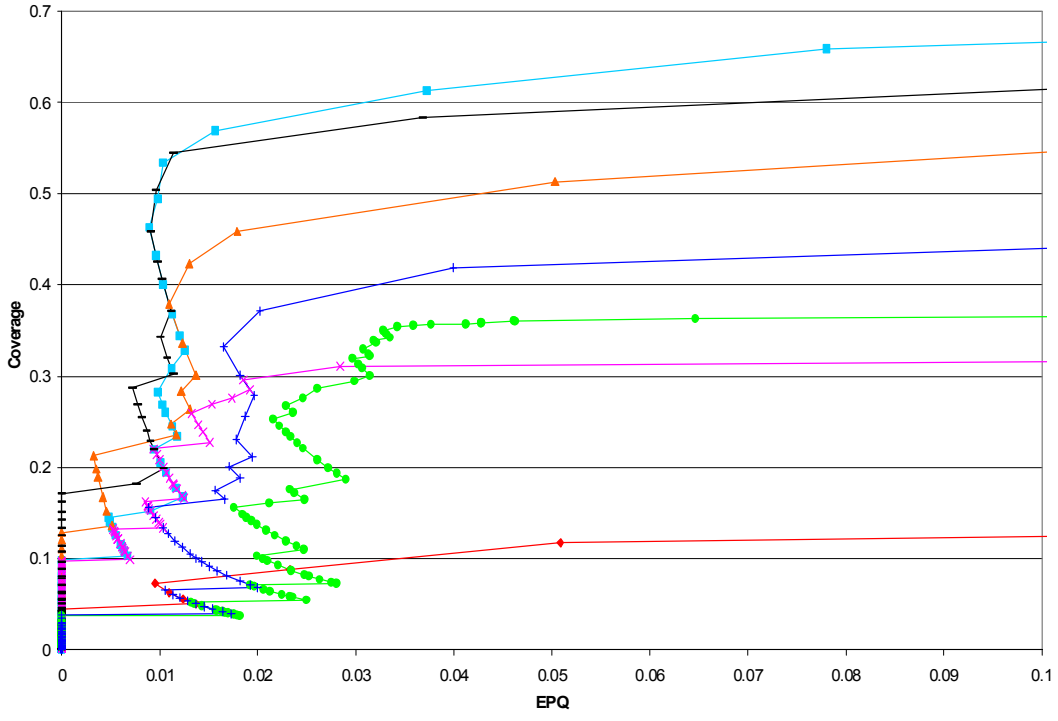
#### *2.3.2.2. Annotating Genomes (tophit Rule)*

All seven methods (BLAST, PSI-BLAST, HMMer, SAM, HHSearch, COMPASS and PRC) were benchmarked with each dataset (nr35 and nr10) using the *tophit* rule, which only scores the first true positive for each query, modelling the annotation of genomes. Figure 2.9 shows the coverage vs. error plots for these benchmarks, while
Table 2.4 gives selected coverage values for varying error rates.

On the nr35 dataset (Figure 2.9a), profile-profile methods slightly outperformed profile-sequence methods. The best profile-profile method was PRC which achieved 4.8% greater coverage than SAM (the best profile-sequence) at 0.01 EPQ (COMPASS had almost equal coverage to PRC). Interestingly PSI-BLAST performed as well as HMMer (81.4% and 81.2% respectively at 0.05 EPQ). BLAST's performance of 70% coverage at 0.05 EPQ shows that this was a relatively easy dataset. In fact, with the *tophit* rule, it was relatively easy for all methods to get high coverage because only the nearest neighbour needed to be identified.




(a)




(b)

Figure 2.9 Performance of all methods using the *tophit* and SAS8 rules on nr35 (a) and nr10 (b) datasets.

Note that runs were performed to an E-value of 10 and in some cases an E-value of 10 is reached at an EPQ of <0.1.

| Data-set | EPQ | Coverage | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | BLAST | COM-PASS | HH-Search | HMMer | PRC | PSI-BLAST | SAM |
| nr35 | 0.01 | 65.1 | 88.1 | 84.7 | 80.7 | 89.1 | 80.5 | 84.3 |
| | 0.05 | 70.0 | >91.0 | 88.9 | 81.2 | 91.0 | 81.4 | 86.7 |
| | 0.10 | 71.5 | >91.0 | 89.7 | 81.5 | >91.0 | 81.9 | >86.7 |
| Nr10 | 0.01 | 4.7 | 53.0 | 23.4 | 21.7 | 52.8 | 3.8 | 16.1 |
| | 0.05 | 10.3 | 63.7 | 51.3 | 31.7 | 60.1 | 36.1 | 42.5 |
| | 0.10 | 12.7 | 66.7 | 55.5 | 31.9 | 62.1 | 36.4 | 45.0 |

Table 2.4 Percent coverage for each method at 0.01, 0.05 and 0.1 EPQ, using the tophit rule.

Where an EPQ relates to multiple E-values, the coverage at the highest E-value is shown. '>' means that the results have reached an E-value of 10 and have run out.

The superior sensitivity of profile-profile methods was much clearer on the more difficult dataset (nr10, Figure 2.9b). COMPASS achieved ~30% greater coverage than HMMer for very remote homologues (nr10) at 0.01 EPQ and ~20% greater coverage than SAM at 0.05 EPQ. COMPASS outperformed PRC on these more difficult datasets in contrast to the benchmark using the *allpos* rule, where PRC was always the superior method.

Overall, COMPASS was the best performing method and all profile-profile methods outperformed all profile-sequence methods. However, for the nr35 dataset SAM performed almost as well as the profile-profile methods at 0.02 EPQ. In practice it is too computationally expensive to use profile-profile methods to annotate genomes. It would be necessary to build profiles for each genomic sequence. However, for a subset which does not score well with profile-sequence methodologies it may be worthwhile using COMPASS or PRC. COMPASS produced a >20% increase in coverage for the most remote homologues, at 0.05 EPQ over the closest profile-sequence method (SAM). For the bulk of genome annotation however, SAM is the best choice of method.

Interestingly, PSI-BLAST performed better than HMMer using both the *tophit* rule and the *allpos* rule. This has, to my knowledge, not been reported before in the literature. PSI-BLAST even performed better than SAM at 0.05 EPQ on the nr35 dataset with the *allpos* rule. The reason may have been the way in which PSI-BLAST was used. Generally PSI-BLAST is used to build a profile with the query sequence. In this case however, the profile was built using target2k. This means that PSI-BLAST had the advantage of HMM technology in building what may have been a more powerful profile than can be built by PSI-BLAST itself.

### 2.3.3. Determining Reliable E-Value Thresholds for Remote Homologue Detection

When applying homologue detection methods it is common to set an E-value (or score) cut-off, above (or below) which hits will be ignored. The determination of this cut-off is frequently the reason for benchmarking a

method. The role of an E-value is to provide an estimate of how many erroneous hits are likely to be found for a given score cut-off and database size. How does it relate to EPQ and how does it differ between *allpos* and *tophit* benchmarks?

Table 2.5 shows that, for the nr35 dataset, at 0.01 EPQ E-values varied significantly between methods. BLAST had the highest whereas PSI-BLAST E-values were the lowest. In fact, different methods differed by several order of magnitude at this error rate. Apart from BLAST, all methods appear to have underestimated the true error rate. This result suggests caution with in a literal reading of E-values at low error rates. Both COMPASS and HMMer grossly under-predict true error rates over most EPQ values.

E-value cut-offs for genome annotation (established using the *tophit* rule) were higher for the same error rate than for separating homologues and non-homologues. Table 2.6 shows that BLAST, SAM and HMMer produce very similar E-values, using *tophit*, to those produced using *allpos*. Whereas for the other methods, the E-values are shifted positively for the same error rates. This is likely to be because less false positives appear when only the best hit is recorded. It is important to consider the application for which remote homology detection is being used before choosing an E-value cut-off. When annotating genomes using an E-value cut-off based on an *allpos* style benchmark, coverage will be unnecessarily low.

| | 0.01 EPQ | 0.05 EPQ | 0.1 EPQ |
|---|---|---|---|
| BLAST | 0.018 | 0.21 | 0.48 |
| PSI-BLAST | 4.0e-10 | 0.16 | 0.47 |
| HMMer | 2.3e-07 | 4.3e-06 | 1.5e-05 |
| SAM | 0.000239 | 0.164 | 0.553 |
| COMPASS | 2.38e-07 | 0.000431 | 0.00658 |
| HHSearch | 8.2e-05 | 4.5 | 36.0 |
| PRC | 3.4e-05 | 0.27 | 1.4 |

Table 2.5 E-value cut-offs for empirically determined error rates on the nr35 dataset using *allpos* rule.

| | 0.01 EPQ | 0.05 EPQ | 0.1 EPQ |
|---|---|---|---|
| BLAST | 0.017 | 0.19 | 0.51 |
| PSI-BLAST | 0.005 | 0.18 | 0.61 |
| HMMer | 4.3e-07 | 3.2e-06 | 1.4e-05 |
| SAM | 0.0161 | 0.391 | - |
| COMPASS | 0.0207 | 3.16 | - |
| HHSearch | 1.0 | 48.0 | 360.0 |
| PRC | 0.16 | 3.2 | - |

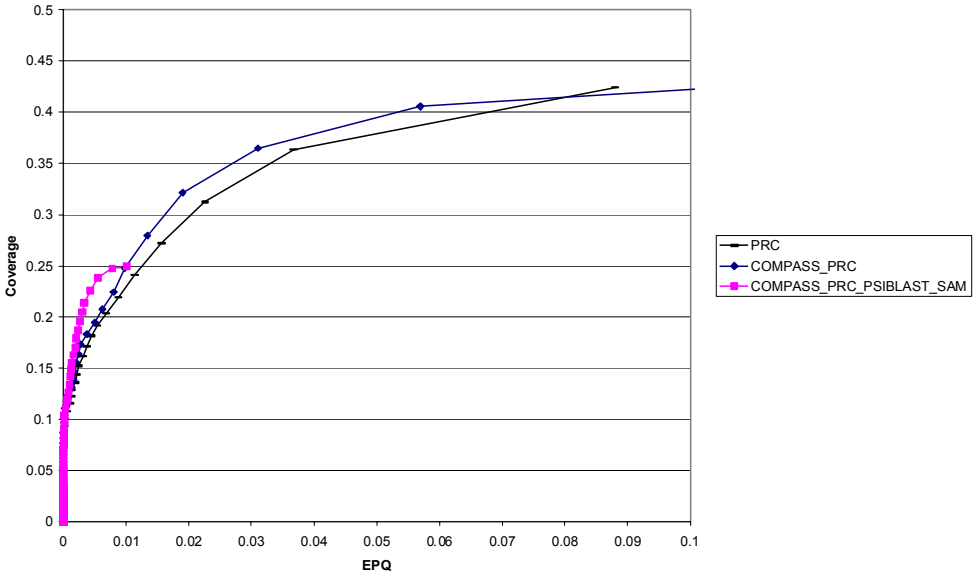
Table 2.6 E-value cut-offs for empirically determined error rates using *tophit* rule on the nr35 dataset.

### 2.3.4. Combining Methods Improves Performance by Excluding False Positives

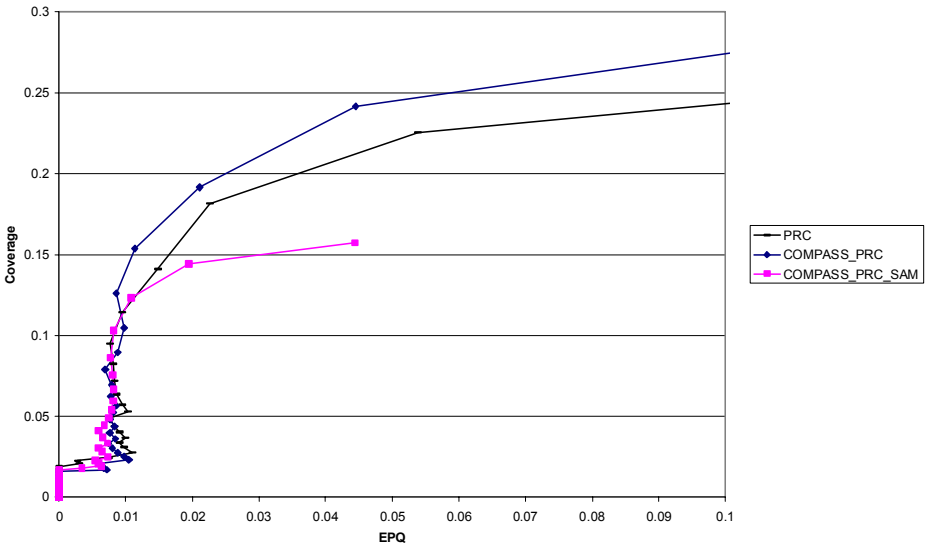
Seeking to improve performance by filtering out false positives, the effect of combining different methods of remote homologue detection was explored. Figure 2.11, (a) and (b), show the best performing combined methods using the *allpos* rule, on nr35 and nr10 datasets respectively. The best performing combination was COMPASS and PRC, which gave an increase of 2-3% over the best single method on both datasets over a large range of error rates.

Figure 2.11, (c) and (d), shows benchmarks on the nr35 and nr10 datasets respectively using the *tophit* rule. All the profile-sequence methods are shown individually and in combination. Profile-profile methods are excluded as they are not practical for large-scale genome annotation. The results show that a significant increase in coverage of 10% was achieved at an error rate of 0.01 EPQ using PSI-BLAST combined with SAM on very remote homologues (nr10). A combination of HMMer and SAM performed best at 0.01 EPQ, but produced a much more modest increase on the nr35 dataset. At higher error rates (>0.02EPQ) SAM was again the best performer on both datasets. This was because the combination of methods allowed for an increase in specificity, but was not expected to increase sensitivity.

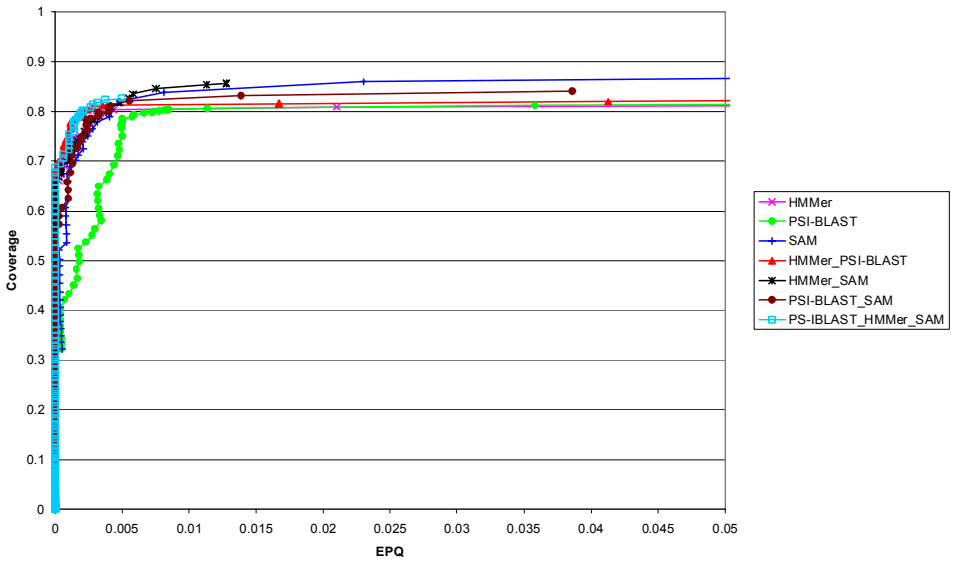
For both *tophit* and *allpos*, the best combined methods were the two best performing single methods. Additionally, they may complement each other because they were based on different technologies (PSSMs and HMMs). In the *tophit* case this was PSI-BLAST and SAM, in the *allpos* case this was COMPASS and PRC.


0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
Coverage
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
EPQ
PRC
COMPASS_PRC
COMPASS_PRC_PSIBLAST_SAM

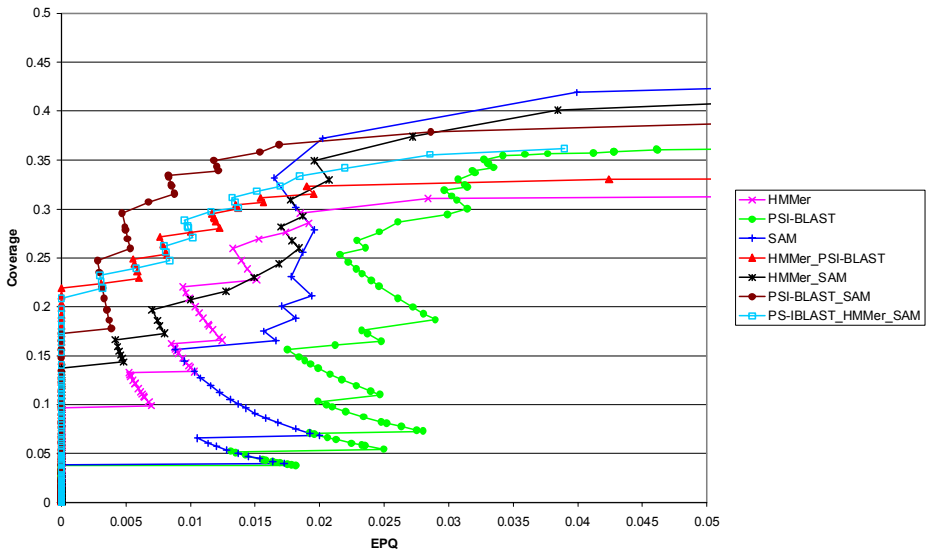
(a)


0.3
0.25
0.2
0.15
0.1
0.05
0
Coverage
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
EPQ
PRC
COMPASS_PRC
COMPASS_PRC_SAM

(b)




(c)




(d)

Figure 2.11 Combining methods to improve specificity.

The best performing single and combined methods (at 0.01, 0.05 and 0.1 EPQ) for (a) nr35 *allpos* and (b) for nr10 *allpos* are shown. (c) Shows all single and combined methods for nr35 *tophit* and (d) for nr10 *tophit*. Note that runs were performed to an E-value of 10 and in some cases an E-value of 10 is reached at an EPQ of <0.1.

## 2.4. Discussion

### 2.4.1. Heuristic Exceptions Rule

The need to make exceptions to structural classifications in benchmarking remote homology detection methods reflects a shift in our abilities to detect homology from sequence. Until recently, structural similarity has been a reliable gold standard. It now appears that some relatives diverge structurally, whilst a sequence signal can be detected by the most sensitive profile-profile methods. An effective solution to this problem, the SAS8 rule, has been presented. Using a SAS score cut-off for a SSAP structural alignment, a reliable benchmark can be produced for these very sensitive profile-profile methods. In the future, structural classifications such as CATH and SCOP will clearly benefit from using these methods to detect more remote homologues.

Shortly after work based on this chapter was published (Reid et al., 2007), Qi and co-workers published an alternative solution to this problem (Qi et al., 2007). They used a Support Vector Machine (SVM) trained on both sequence and structural similarity scores between SCOP domains from different classes and those from the same superfamily to classify previously ambiguous relationships between domains. Rather than remove ambiguous relationships from the dataset as was the aim in this work, their aim was to include as many ambiguous relationships as possible by explicitly classifying them as homologous or non-homologous. In the benchmarks most comparable to those presented here, they conversely found that HHSearch had improved performance over COMPASS. In fact it seems that the relatively low performance for HHSearch presented in this chapter may have been caused by an error in the HHSearch code which was subsequently fixed (Johannes Soding, *personal communication*).

### 2.4.2. The Importance of Benchmarking for Application

In benchmarking methods of remote homologue detection it is important to bear in mind the task for which they will be used. Different rules for counting true and false positives reflect whether one is interested in annotating genomes (only the closest homologue is required) or whether a score cut-off is needed to determine whether individual domain pairs are homologous (separation of all homologues from non-homologues). Methods perform more or less well at different tasks and this knowledge is invaluable for ensuring that results are reliable. E-value cut-offs for some methods are very different for different applications.

### 2.4.3. Relative Performance of Methods

Having established the SAS8 exception rule and suitable benchmarks to model both genome annotation and the simple scoring of homologues, PRC was shown to be the best method for distinguishing homologues and non-homologues. In fact, for distant homologues (<10% sequence identity) PRC is 2.5 times better than the best profile-sequence method at separating homologues from non-homologues. Profile-profile methods are greatly increasing our ability to recognise remote homologues.

PSI-BLAST performed surprisingly well in the benchmarks presented. This may be due to the way in which it was used. Profiles were built using the SAM T2K program and then used in up to 20 iterations of PSI-BLAST. The use of SAM T2K brings an element of HMM technology to PSI-BLAST.

COMPASS was shown to be the best method overall for annotating genomes at low sequence identities (<10%). However, at sequence identities of <35% PRC is equally effective and there is only ~5% increase in coverage over the best profile sequence method (SAM). It is not possible to use profile-profile methods for annotating whole genomes as for the genomic sequence, a profile is lacking.

### 2.4.4. Combining Methods Improves Performance

When determining relatives for a protein of interest, an investigator may compare the results of multiple methods to increase the likelihood of a correct assignment. In this chapter, an automated approach was described using the combined results of multiple methods to improve assignment. When annotating genomes, combining profile-sequence methods gave a large increase in performance of 10% at a 1% error rate. This increase was achieved by combining SAM and PSI-BLAST.

### 2.4.5. Future Work

The approach presented here to combine methods of remote homologue detection and improve performance was a simple one. There is scope for integrating scores from such methods in a more advanced framework. Weighting different methods would perhaps be the first improvement.

# Chapter 3 Developing CODA to Predict Functional Associations between Proteins

## 3.1. Introduction

### 3.1.1. Gene and Domain Fusion Detection Methodologies

In the post-genomic era it has become clear that the parts list of genomes is insufficient to explain organismal complexity. Research is shifting towards understanding organisms as systems of interacting parts. Many new approaches are being developed to identify the relationships between these parts in terms of interactions and functional associations. Domain, or gene fusion is one of several genome context methods which can be used to predict functional associations between pairs of proteins (Marcotte *et al.*, 1999; Enright *et al.*, 1999). Genome context methods allow inheritance of functional information between non-homologous proteins. They are thus an orthogonal approach to homology-based methods of function prediction. In addition, they can predict networks of proteins involved in common complexes and pathways (von Mering et al., 2007).

Gene fusion is an evolutionary process whereby initially separate genes become fused into a single open reading frame which is expressed as a multi-domain protein chain. Perhaps the most compelling argument for the evolutionary role of fusions is that in eukaryotic evolution, as cells increased in size, fusions were selected for to maintain the relative concentrations of

interacting proteins without increasing the absolute amount of protein produced (Enright et al., 1999). It has also been proposed along similar lines that fusions have been favoured due to a decrease in diffusion rates in eukaryotic cells caused by obstacles such as the cytoskeleton (Yanai et al., 2001). Fusions have frequently been found in prokaryotes however (Enright and Ouzounis, 2001), suggesting that these arguments cannot account for all fusions.

Bioinformatic approaches which identify fusion events in order to predict functional associations use either whole protein sequence comparison or domain family assignments. These are known as gene fusion and domain fusion respectively. Table 3.1 shows various approaches to gene/domain fusion which have appeared in the literature.

| Authors | Fusion detection method | All homologues/Orthologues-only | Scoring |
|---|---|---|---|
| Marcotte et al. (1999) | Gene fusion (BLAST) and domain fusion (ProDom) pooled. | All homologues – 5% most promiscuous domains removed | None |
| Enright et al. (1999) | Gene fusion (BLAST and S-W) | All homologues | S-W based Z-scores |
| Snel et al. (2000) | Gene fusion (S-W) | Orthologue-only (bidirectional best hit) | None |
| Enright & Ouzounis (2001) | Gene fusion (BLAST, component overlap <10%) | All homologues (although component and composite proteins clustered) | None |
| Yanai et al. (2001) | Gene fusion (BLAST) | Orthologue-only (one link between each COG) | None |
| Marcotte & Marcotte, (2002) | Gene fusion (BLAST) | All homologues | Probability of observing fusion and uncertainty due to large families |
| Truong & Ikura (2003) | Domain fusion (Pfam domains) | All homologues (promiscuous domains removed) | None |
| Bowers et al. (2004) | Gene fusion (BLAST) | All homologues | Probability of observing fusion |
| CODA (this chapter) | Domain fusion (Pfam domains) | All homologues | Frequency of homologues in query and individual target genomes |

Table 3.1 Overview of gene/domain fusion implementations for predicting functional associations.

'All homologues vs. orthologues-only' specifies whether the approach identifies functional similarity for all the

homologues of the fusion protein or only those thought to be orthologous to it.

The most common approach is gene fusion using BLAST (Altschul et al., 1997), frequently in combination with the Smith-Waterman (1981) algorithm, to detect triplets of proteins. In this scheme two proteins from a single genome (query proteins) which are both predicted to be homologous to a third protein in a different genome (fusion protein), but are not homologous to each other are identified as functionally associated (i.e. take part in a common biological process). Proteins which are truly related by a fusion event may contain homologous domains, however it is generally not useful to link query proteins through homologous domains as they are less likely to be involved in the same biological process than if linked through non-homologous domains (Enright and Ouzounis, 2001). This is commonly a problem with promiscuous domain families. Furthermore it is advantageous to exclude such homologous examples when benchmarking the performance of the method as such associations can be identified more easily by homology based approaches.

Promiscuous domain families are found in many different proteins, fused to many different partner domain families (Apic et al., 2001a). The protein kinase family Pkinase (Pfam code: PF00069) from the Pfam protein family database (Finn et al., 2008) is one of the most promiscuous in Nature. It comprises largely eukaryotic protein kinases involved in diverse biological processes. It is found fused to >250 different Pfam families in a variety of organisms. The result of this is noise in the domain fusion analysis through functionally misinformative fusions. Any protein containing members of the Pkinase family can be linked to every other protein which contains one of the >250 domains to which Pkinase is found fused.

Domain fusion uses domain-based descriptions of sequences (e.g. Pfam) rather than direct sequence comparison. In this case a fusion event is identified where two proteins in one genome contain distinct domains that are found fused together in another genome. Again, promiscuous domains can cause erroneous associations between functionally unrelated proteins. For domain fusion approaches, proteins containing highly promiscuous

domains can be explicitly excluded from the results in order to improve accuracy in detecting functional relationships (Marcotte et al., 1999).

Another problem affecting both the gene and domain fusion approaches is that of large gene/domain families. In domain fusion for instance, if a relative of domain family A is found fused to a relative from domain family B, all proteins containing domains from A are potentially associated to all those containing domains from B within any particular genome. If families A and B are large, then there are many possible functionally associated pairs. In large families, it is unlikely that all members will be involved in the same biological process (Marcotte and Marcotte, 2002). Figure 3.1 illustrates the problems encountered in detecting functional relationships with gene/domain fusion.

There are two ways of coping with this 'paralogue problem' which have appeared in the literature. The first is to accept only those pairs of query proteins which are thought to be orthologous to the fusion protein (Snel et al., 2000). This has been achieved by using bi-directional best hit orthologues (Snel *et al.*, 2000; Kummerfeld and Teichmann, 2005). The results show high accuracy, although relatively few functional relationships are determined – a maximum of one per fusion protein in any particular genome (Huynen et al., 2000). The second approach is to apply a scoring scheme which takes account of the size of families and the uncertainty about which pairs are orthologous; we expect some paralogues to take part in the same biological processes (Marcotte and Marcotte, 2002). Therefore, this approach allows more predictions to be made, although presumably at a lower accuracy than the orthologue-only approach. No assessment has been published of the relative performance of these two approaches, or any different implementations of gene/domain fusion.

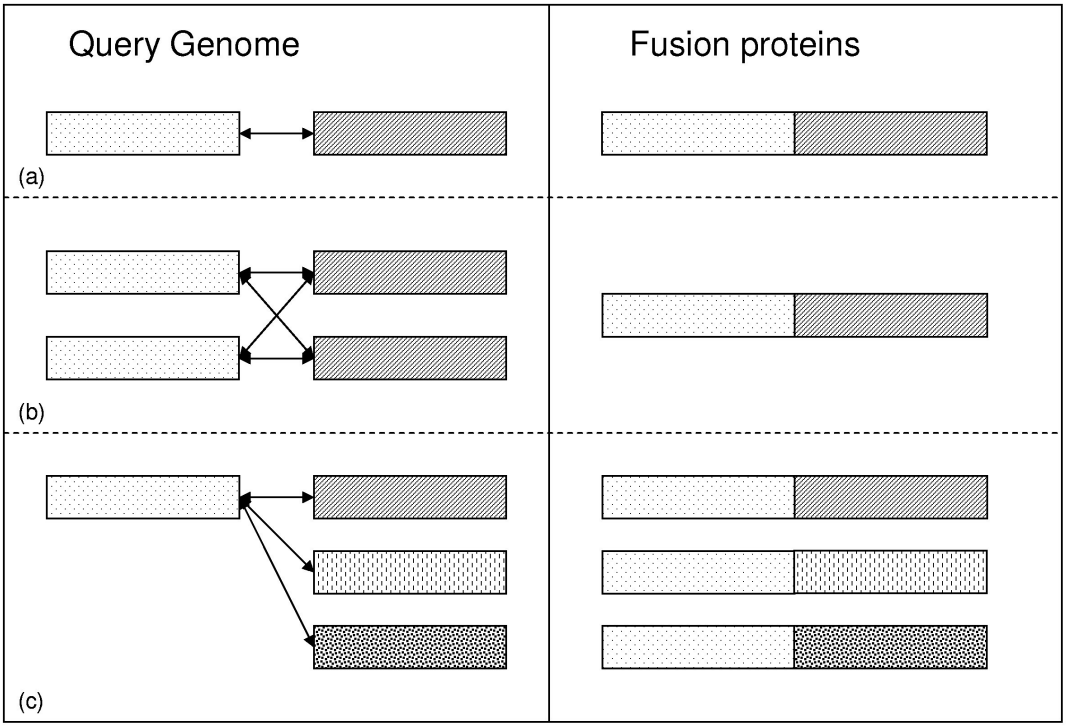



Figure 3.1 Problems encountered in detecting gene/domain fusions.

Boxes with the same pattern represent homologous domains, arrows represent possible functional linkages. (a) shows the simple case where the query genome contains only one pair of proteins which can be linked using a particular fusion protein. (b) shows an example of the problem of large domain families, where increasing numbers of homologues result in greater uncertainty as to which might be orthologous to the fusion protein. There is therefore decreasing certainty as to whether any particular pair of homologues shares a similar function. (c) shows an example of where a promiscuous domain, one fused to many other domains, causes uncertainty about relevant functional linkages. Promiscuous domain families tend to be involved in a variety of different processes and are

therefore unreliable for use in identifying functional relationships through gene/domain fusions.

### 3.1.2. Aims

The aim of this chapter was to develop a domain fusion approach which was able to accurately detect functional relationships in higher eukaryotic genomes. Co-Occurrence of Domains Analysis (CODA), the method introduced in this chapter, uses the domain fusion approach and implements a novel score to cope with the problem of large families.

CODA is compared against two existing implementations of gene fusion and one of domain fusion. This allows an analysis of the relative performance of different approaches.

## 3.2. Methods

### 3.2.1. Gene3D Multi-Domain Architecture Datasets

Co-Occurrence of Domains Analysis (CODA) requires Multi-Domain Architectures (MDAs) of proteins for complete genomes. An MDA is a symbolic representation of the predicted domains for a protein. The order and frequency of domains in a protein is not considered by CODA and so discontinuous domains can simply be collapsed. Gene3D (Yeats et al., 2008) is an ideal source of this data as it contains protein sequences for all complete genomes with predictions for CATH (Greene et al., 2007) and Pfam (Finn et al., 2008) domains as well as functional annotations including GO (Harris et al., 2004).

Several alternative MDA datasets were generated, each for all 527 complete genomes (50 eukaryotes, 438 eubacteria and 39 archaea) contained in Gene3D v5. Individual datasets were created using only CATH domains, only Pfam domains or a combination of the two in order to test which was more effective in representing proteins in domain fusion analysis. Annotation for both domain types was retrieved from Gene3D. The datasets which included both CATH and Pfam domains were generated in two ways. The CATH-Pfam dataset had CATH domains assigned first, while Pfam-CATH had Pfam domains assigned first. Each example of the second type of domains was added if the overlap between it and the already assigned domains was no greater than 30% in both directions. The initial set of CATH domains did not overlap with each other, nor did the Pfam domains. This resulted in 4 different datasets – CATH, Pfam, CATH-Pfam and Pfam-CATH.

### 3.2.2. Prolinks, STRING and Truong Datasets

In order to compare CODA against the other methods, it was necessary to recreate the datasets used to generate their results. The reason for this is that their methods are not available for use on arbitrary datasets and it was

therefore necessary to run CODA on the datasets used by those methods in order to give a fair comparison. For instance, it has recently been shown that the performance of gene fusion methods is particularly sensitive to the number of genomes available in which to search for fusions (Kamburov et al., 2007). For STRING and Prolinks, descriptions of the sequences used were available from the respective webservers. STRING provided a file for download containing all sequences used in their analyses (http://string.embl.de/newstring_download/protein.sequences.v7.1.fa.gz). Prolinks provided a file for download containing all GI numbers, but not the sequences themselves (http://mysql5.mbi.ucla.edu/public/reference_files/geneIDS_to_GInum.txt ). It was necessary to obtain these sequences independently from the NCBI FTP site (ftp://ftp.ncbi.nih.gov/genbank/), although a small number were no longer available and it was necessary to acquire them directly from the Prolinks website HTML. The Truong dataset was Swiss-Prot release 39 combined with TrEMBL release 17. The Swiss -Prot release was retrieved from the EBI FTP server (ftp://ftp.ebi.ac.uk/pub/databases/swissprot/sw_old_releases), while TrEMBL release 17 was kindly provided by the PANDA group at the European Bioinformatics Institute.

All STRING and Prolinks sequences were scanned with Pfam HMMs using the same *pfam_scan.pl* protocol used for Gene3D (Yeats et al., 2008). Details of these datasets, including Pfam coverage is shown in Table 3.2.

| Resource | Genomes | Total dataset coverage by Pfam domains | Yeast Pfam coverage | Human Pfam coverage |
|---|---|---|---|---|
| STRING v7 | 373 | 71% (1074952/1513782) | 64% (4245/6680) | 74% (16371/22218) |
| Prolinks v2.0 | 168 | 73% (429173/590444) | 73% (4195/5761) | 74% (17266/23213) |
| Truong dataset | 210 | 50% (128332/257962) | 44% (2935/6690) | n/a |

Table 3.2 Coverage of STRING, Prolinks and Truong datasets with Pfam domains.

Coverage was calculated as the percentage of proteins with at least one domain. Raw numbers are shown in brackets.

Truong-fusion and CODA both use Pfam domains. Using the most recent Pfam annotation would therefore provide CODA with more information than was available to Truong-fusion. Therefore Pfam domain annotation for the Truong dataset was retrieved from the aforementioned Swiss-Prot and TrEMBL records. The STRING and Prolinks datasets comprise protein sequences from completed genomes. The Truong dataset however gave no information of which genomes in the dataset were complete and it is difficult to determine which genomes were completed at this time. Therefore those proteins from species which currently remain unsequenced were removed. The result is that some incomplete genomes will remain and this may reduce the performance of CODA which was designed to use complete genome information to accurately score its results.

### 3.2.3. A Benchmark for Functional Similarity Using Gene Ontology Terms

The aim of the CODA method is to identify pairs of proteins which are involved in similar biological processes. In order to benchmark CODA it was therefore necessary to determine the functional similarity between an arbitrary pair of proteins. The Gene Ontology (GO) is well suited to this and has commonly been used for this purpose (e.g. Ranea et al., 2007). GO clearly separates biological process from molecular function annotation and there is a growing literature based on different approaches to measuring the similarity between GO terms. One of the most popular of these approaches is GO Semantic Similarity (GOSS) (Resnik, 1999). This method uses statistics from the corpus of terms assigned to a particular genome and the information content of the shared parent for two terms to determine their similarity (described in detail in 1.6.4). An in-house implementation of the Resnik method, as described by Lord et al. (2003) was used.

The corpus of terms used in calculating functional similarities between proteins was varied according to whether the benchmark was performed in yeast or human and whether the dataset was Gene3D, STRING, Prolinks or Truong. The coverage of each of these datasets by relevant GO terms is

shown in Table 3.3. For each pair of putative functionally associated proteins, all biological process GO terms relating to these proteins were extracted from Gene3D. Those terms with evidence type 'Inferred from Electronic Annotation (IEA)', 'No biological Data available (ND)' and 'Inferred from Genomic Context (IGC)' were removed. Excluding IGC annotations is particularly important to avoid the circularity of benchmarking a method using results derived from similar methods. GOSS was used to calculate the similarity between each term, between each pair of proteins. The GOSS score between two proteins A and B was taken as the maximum GOSS score between any pair of terms, one from A, one from B.

| Dataset | Yeast | Human |
|---|---|---|
| Gene3D v5 | 75% (4203/5586) | 18% (6192/34888) |
| STRING v7 | 67% (4447/6680) | 22% (4861/22218) |
| Prolinks v2.0 | 76% (4385/5761) | 23% (4980/23213) |
| Truong dataset | 58% (3885/6690) | n/a |

Table 3.3 Percentages of proteins from yeast and human genomes which had at least one relevant GO term in each dataset.

In this benchmark false positives could not be directly determined as many proteins were unannotated or annotated with relatively non-specific GO terms. Therefore, instead of precision, enrichment was calculated based on the number of positive hits expected by chance. It was necessary to determine a GOSS score cut-off which was unlikely to be exceeded by a score between randomly associated proteins. Protein pairs identified by a method, which exceed this score, were considered true positive hits. Figure 3.3 shows the distribution of GOSS scores in the yeast genome. The figure shows that only ~3% (260754) of GOSS scores were ≥ 4. Considering all protein pairs in yeast (i.e. including those with no appropriate GO terms), the likelihood of a score ≥ 4 is 0.0167. Therefore if a gene fusion method picked 50 protein pairs, we would expect to see 0.835 (50 x 0.0167) significant pairs by random chance. Therefore if 10 of the pairs, predicted by the method, had a GOSS score ≥ 4, the prediction method has performed 11.97 (10 / 0.835, observed true positives divided by expected positives) times better than expected by chance, this value is the enrichment. The distribution of GOSS scores for the human genome was very similar to the yeast genome, although 93.7% of pairs did not have a GOSS score. For the human genome ~3% (1010288) of GOSS scores were ≥ 4. For both human and yeast datasets, GOSS scores of 4 and above were sufficiently rare that they were unlikely to be picked by chance ($p < 0.05$). This was true for the STRING and Prolinks datasets as well as the Gene3D dataset. The proportion of expected positives was varied appropriately for each dataset, taking into account the frequency of GOSS values ≥ 4 expected by chance. The frequency of expected significant GOSS scores for each dataset is presented in Appendix A.

The Benchmark plots (e.g. Figure 3.5) were generated by calculating the enrichment (observed true positives / expected positives) and the number of hits (observed true positives) for successive cut-offs of the different method's native scores.

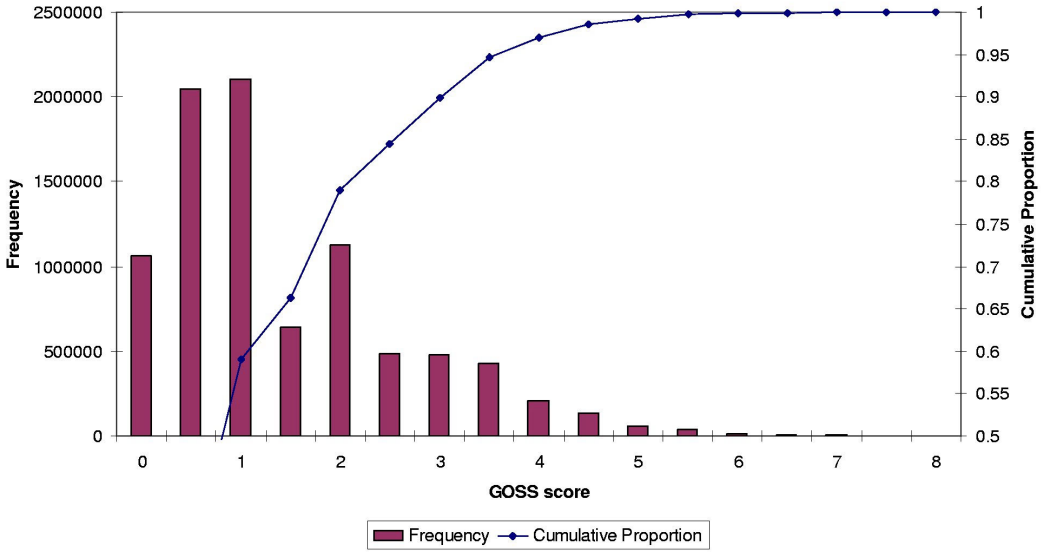



Figure 3.3 Distribution of biological process GOSS scores between yeast proteins in the Gene3D dataset.

Proteins without appropriate GO terms were excluded. GOSS score bins are lower bounded by the previous value and upper bounded by less than the stated value, thus the 2.5 bin contains values ≥ 2 and < 2.5. The red bars represent the frequency and the blue line represents the cumulative proportion of GOSS scores which have less than the stated value.

### 3.2.4. The CODA Score

Co-Occurrence of Domains Analysis (CODA) uses a Multi-Domain Architecture (MDA) representation of proteins in complete genomes (target genomes) to discover pairs of proteins involved in common biological processes within a complete genome of interest (the query genome). It is a novel approach in the domain fusion idiom using a new scoring method.

For a pair of proteins $i = (p,q)$ in a query genome $g$. P is the set of domains in protein $p$. $a \in$ P denotes that protein $p$ contains a domain of superfamily $a$. $J$ is the set of domain pairs $j = (a,b)$ where $a \in$ P, $b \in$ Q. In other words $J$ consists of all the distinct pairs of domains between proteins $p$ and $q$. It is also required that P ∩ Q = {}, as the two proteins must not share any domains of the same superfamily. Each superfamily was only counted once per protein.

To determine a fusion event we require that a target genome $t$ (one other than the query genome) contains a protein $s$ where $a \in S$ and $b \in S$, i.e. domains which are separated in the query genome are found fused in the target genome. The set $T$ comprises those genomes other than $g$ which contain such proteins $s$. For a domain pair $j$ in genome $g$, the fusion score $C_j$ is taken as a maximum over all genomes in $T$ (Equation 3.1).

$$C_j = \max_{t=1}^{|T|}\left(\frac{1}{n_{g_A} + n_{t_A}} + \frac{1}{n_{g_B} + n_{t_B}}\right)$$

Equation 3.1 CODA score for a particular pair of domain superfamilies $j$ in genome $g$.

In Equation 3.1 $|T|$ is the number of elements of set $T$ (i.e. the number of target genomes), $n_{g_A}$ and $n_{g_B}$ are the frequencies of domain superfamily $A$ and domain superfamily $B$ respectively in genome g, $n_{t_A}$ and $n_{tB}$ are the frequencies of domain superfamilies $A$ and $B$ respectively in genome $t$.

For a protein pair *i*, in query genome *g*, the maximum $C_j$ is taken over all possible domain pairs *j* (Equation 3.2).

$$C_i = \max_{j=1}^{|J|}\left(C_j\right)$$

Equation 3.2 CODA score for a pair of query proteins *i* in genome *g*.

In Equation 3.2 |J| is the number of elements in set J (i.e. distinct domain pairs). Thus $C_i$ is the CODA score for proteins *p*,*q* (pair *i*); the best (highest) score over all domain pairs between the proteins and over potential fusion proteins in all genomes *T*. The important novel aspect of this score is that it takes the maximum score amongst all the genomes whereas other methods do not consider target genomes individually. The score was chosen to reflect the uncertainty that fused domains and their unfused relatives are orthologues. The highest (best) possible score (one) is returned when there is only one example of each domain family in the query genome and one fused protein in a target genome, with no other domain homologues. In this case it is highly likely that the query protein domains are orthologous to the target protein.

### 3.2.5. CATH Subfamilies for CODA

CATH domains showed poor performance relative to Pfam domains in detecting functional relationships between proteins using CODA. This could have been due to low coverage of CATH domains relative to Pfam or because CATH has larger families causing low scores for many hits. CATH superfamilies were clustered at varying sequence identity cut-offs (30, 35, 40, 50, 60, 70, 80, 90, 95 and 100%) using an in-house implementation of directed multi-linkage clustering. Sequence identities were determined using BLAST with default parameters. The domain counts used in the CODA score were then adjusted using these clusters. Let us say that there are two proteins in yeast, each with one domain. The first protein contains domain *a* and the

second domain *b*. A protein is found in *E. coli* which is a fusion of these two domains – *a´b´*. Let us say that *a* and *a´* are in the same 50% cluster but not the same 60% cluster, i.e. they share ~50% sequence identity. The counts for $n_{g_A}$ in the CODA score (Equation 3.1) then only include the number of members of the 50% cluster containing *a* that belong to yeast. $n_{t_A}$ becomes the number of members of that 50% cluster which belong to *E.coli*. Likewise, if *b* and *b´* are in the same 70% cluster but not the same 80% clusters, then the counts are taken from that 70% cluster.

## 3.2.6. Details of Other Fusion Approaches Used in This Work

### *3.2.6.1. STRING-Fusion*

The STRING-fusion method (von Mering et al., 2007) applies the Smith-Waterman algorithm (Smith and Waterman, 1981) to align sequences from complete genomes and orthologues are determined between genomes using bi-directional best hits. A fusion is identified where a gene in one genome has two orthologues in another genome which do not overlap with each other when aligned to the fused protein. The fusions are scored by counting the number of fusion events and normalising by the number of species which contain fusion proteins (Snel et al., 2000)

### *3.2.6.2. Prolinks-Fusion*

All protein coding sequences from a genome of interest are aligned to a non-redundant database using BLAST. Fusions are identified where two non-homologous proteins align over at least 70% of their sequences to different regions of a third protein (Bowers et al., 2004). To cope with large domain families, a score based on the hypergeometric function is applied (Equation 3.3).

$$P(k' \mid n, m, N) = \frac{\binom{n}{k}\binom{N-n}{m-k}}{\binom{N}{m}}$$

Equation 3.3 Prolinks score.

Equation 3.3 applies to two proteins *A* and *B*, where $k'$ is the number of fusion proteins in a sequence database, $n$ is the number homologues of protein A, $m$ is the number of homologues of protein *B* and $N$ the total number of sequences in the database. This function calculates the likelihood that a pair of query proteins is orthologous to a fusion protein, given the number of fusion proteins and query protein homologues.

### *3.2.6.3. Truong-Fusion*

Truong & Ikura (2003) applied the domain fusion approach using Pfam domains. They identified Domain Fusion Templates (DFTs), pairs of non-homologous Pfam domains which occur in the same protein chain in a genome other than the genome of interest. They then found protein pairs in the query genome which were linked by a DFT. In order to avoid false positives, results which were identified using the same domain pairs at least 10 times are excluded. For example, in their analysis of human, the RasGAP and SH3 domains were used to link 72 different pairs of proteins and these 72 pairs were excluded. No scoring was applied to the results.

# 3.3. Results

## 3.3.1. Performance of CODA

### *3.3.1.1. Alternative Multi-Domain Architecture Representations*

The first step in implementing CODA was to generate multi-domain architecture (MDA) datasets to represent the genomes. Four different domain-based datasets were produced using domain assignments from the Gene3D v5 database (Yeats et al., 2008). These contained either CATH domains (CATH-MDA), Pfam domains (Pfam-MDA) or a combination of the two (CATH-Pfam-MDA with CATH taking precedence and Pfam-CATH-MDA with Pfam taking precedence).

Table 3.4 shows that Pfam had better coverage of the genomes than CATH, but also that CATH and Pfam were complementary, giving greater coverage when used together than with either resource alone. The effectiveness of these different genome representations in creating functional linkages between proteins using CODA is explored below.

Whereas many gene fusion methods use BLAST scores between whole proteins to determine fusions, CODA uses domain pairs. Therefore multi-domain assignments based on different domain classifications are likely to result in differences in the performance of CODA. In previous work sequence-based domain families such as those of ProDom (Bru et al., 2005) and Pfam (Finn et al., 2008) have been used to detect domain fusions for prediction of functional associations (Enright *et al.*, 1999; Truong and Ikura, 2003). Structural domain superfamilies have been used to explore the evolution of fusions (Kummerfeld and Teichmann, 2005) but not to detect functional relationships. The relative effectiveness of the two types of domain family in predicting functional relationships has not previously been explored.

| Dataset | Total proteins | Yeast coverage (of 5586 distinct protein sequences) | Human coverage (of 34888) |
|---|---|---|---|
| CATH | 821801 | 38% (2130) | 40% (13831) |
| Pfam | 1423060 | 73% (4050) | 65% (22736) |
| CATH-Pfam | 1495200 | 76% (4226) | 68%(23679) |
| Pfam-CATH | 1495200 | 76% (4226) | 68% (23679) |

Table 3.4 Size of datasets and genome coverage with different Multi-Domain Architecture (MDA) types.

Coverage is calculated as the percentage of proteins which have at least one domain. The CATH-Pfam and Pfam-CATH datasets therefore appear identical, although their domain assignments are not.

Figure 3.5 shows enrichment against number of hits obtained by CODA using different MDA datasets. Enrichment is a measure of accuracy: the number of true positives divided by the number of positives expected by chance given the number of hits (see 3.2.3). An enrichment of 10 was chosen as an example cutoff to reflect a moderate accuracy, although a range of enrichments are examined.

At an enrichment of 10 CODA performed best using the Pfam-CATH dataset and found 1791 hits; using Pfam it found 1663 hits, CATH-Pfam 792 and CATH 296. At higher enrichment (e.g. 15), the Pfam dataset was optimal, with CODA finding ~500 hits.

Datasets based principally on CATH domains (CATH-Pfam and CATH) performed less well than those based on Pfam domains. This may be because CATH superfamilies tend to be broader than Pfam families, including more functional subfamilies. This could result in generally reduced scores for hits involving these larger families. In order to determine whether this was the case, sequence-based subfamilies were created for each CATH superfamily as described in 3.2.5. Using these subfamilies resulted in higher scores where homologous domains between the query and target genomes were more similar than to other members of that superfamily. However, Figure 3.7 shows that this did not improve the performance of CODA when using CATH domains. It seems therefore that the reduced performance of CATH relative to Pfam was related more to lower coverage of genomes than to the size and functional specificity of the families. Pfam MDA datasets were chosen over Pfam-CATH due to a similar performance at moderate enrichment and superior performance at higher enrichment. CODA should be used with a score cut-off of 0.56 to achieve an enrichment of 10 on this dataset and 0.65 for an enrichment of 15.

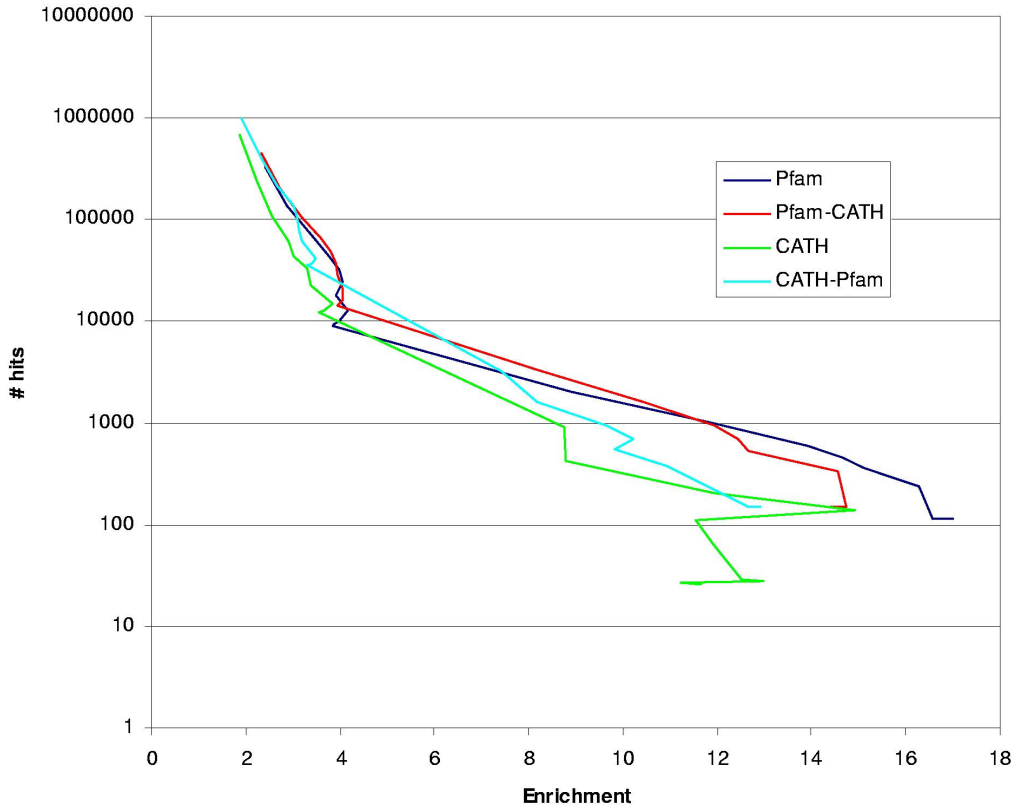



Figure 3.5 Comparative performance of Pfam, Pfam-CATH, CATH and CATH-Pfam MDA datasets on the yeast genome.

Enrichment is the ratio of true positives achieved by CODA to the number expected by chance. Curves in this and subsequent figures were plotted at intervals of 0.05 of the CODA score.

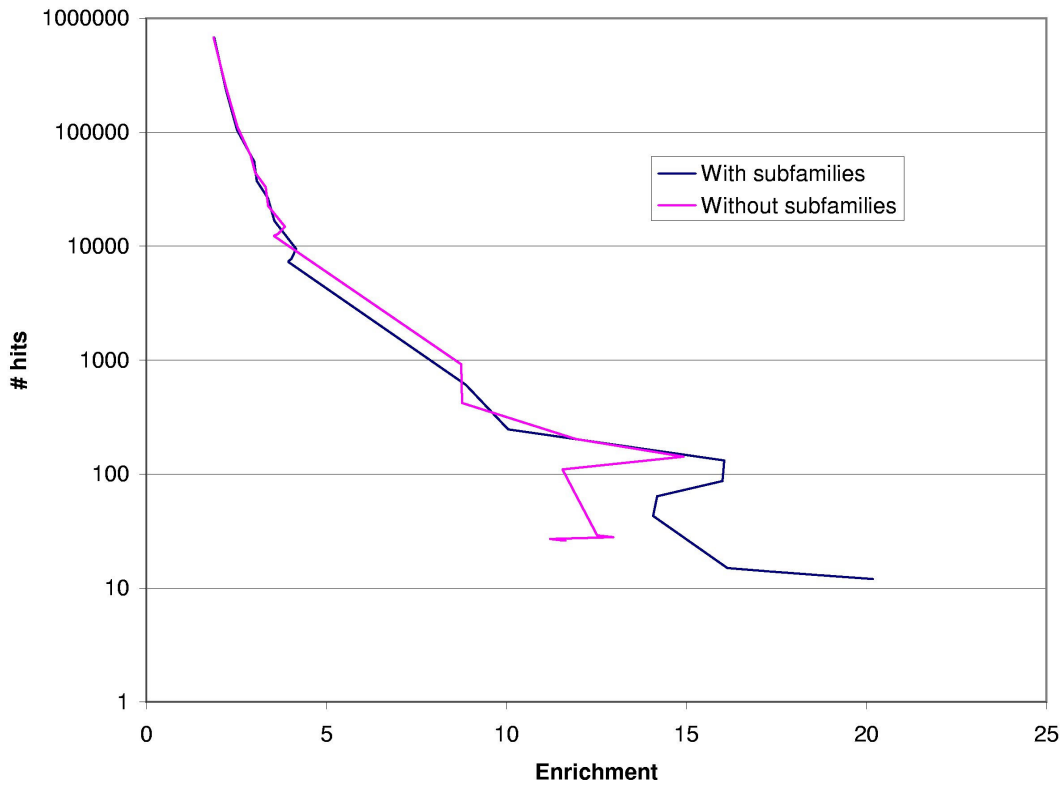



Figure 3.7 Performance of CODA on yeast Gene3D dataset using CATH domains, with and without sequence subfamilies.

### *3.3.1.2. CODA is Insensitive to Promiscuous Domains*

Gene/domain fusion methods are liable to detect many false positives due to promiscuous domains and large homologous gene/domain families (Marcotte and Marcotte, 2002). Large domain families with many homologues are not very common in organisms with small genomes, but in larger eukaryotic genomes there are many such families. This problem is tackled by CODA in two ways. Firstly, the CODA score takes account of the size of domain families and gives lower scores where there are many homologues of the domains involved in the fusion. Secondly, unlike other score-based fusion methods, in CODA, the final score for a pair of proteins in the query genome is the best score out of all possible fusion proteins detected in all the genomes screened. Other methods calculate a single score summing over all genomes (Marcotte and Marcotte, 2002; Bowers *et al.*, 2004). The CODA score therefore penalises larger families which is advantageous due to the problem of paralogues discussed earlier. Additionally, as large families tend also to be promiscuous, the scoring method should also penalise promiscuity.

Figure 3.9 shows that when results involving promiscuous domains were removed there was little change in performance except at the highest and lowest enrichments. Here a promiscuous domain family is described as one which co-occurs with more than 50 other domain families. The CODA method therefore copes well with promiscuous domains, finding a greater number of hits for an enrichment of 10 when promiscuous domains were present (1663) compared to when they were removed (1494).

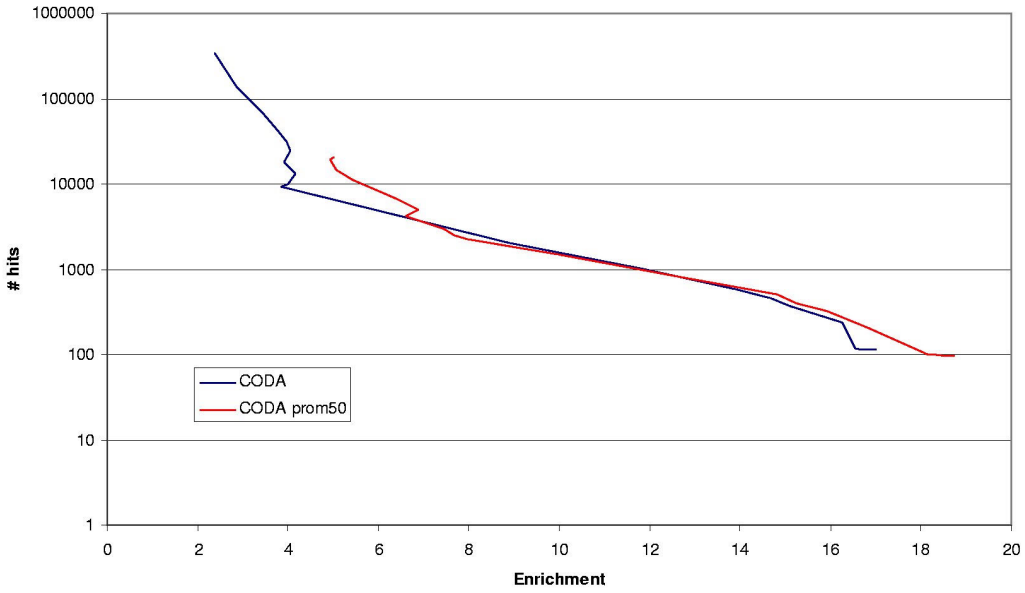



Figure 3.9 CODA with and without promiscuity filter (prom50).

The promiscuity filter removes all results involving a domain that is known to occur in protein chains with 50 or more different domain families, across all genomes.

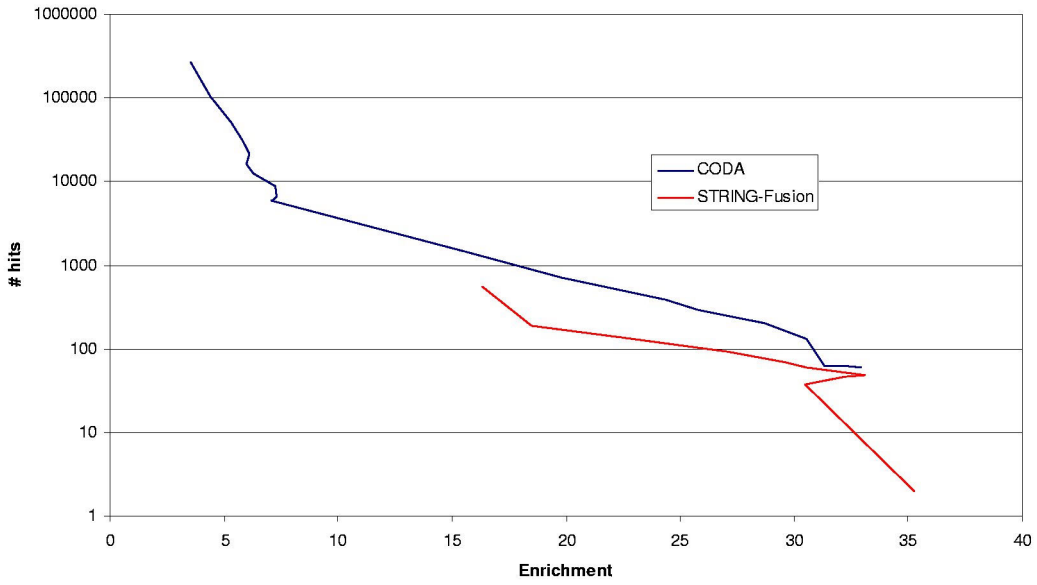
### 3.3.2. Comparison of CODA with Prolinks-Fusion, STRING-Fusion and Truong-Fusion in Yeast

It is important to determine how well CODA performs relative to other comparable (i.e. gene/domain fusion) methods. It has been unclear from the literature what the relative effectiveness of different methods is. Several resources provide data from such methods. These include STRING (von Mering et al., 2007), Prolinks (Bowers et al., 2004) and the Domain Fusion Database (Truong and Ikura, 2003). Results from Predictome (Mellor et al., 2002) and FusionDB (Suhre and Claverie, 2004) were not available for download.

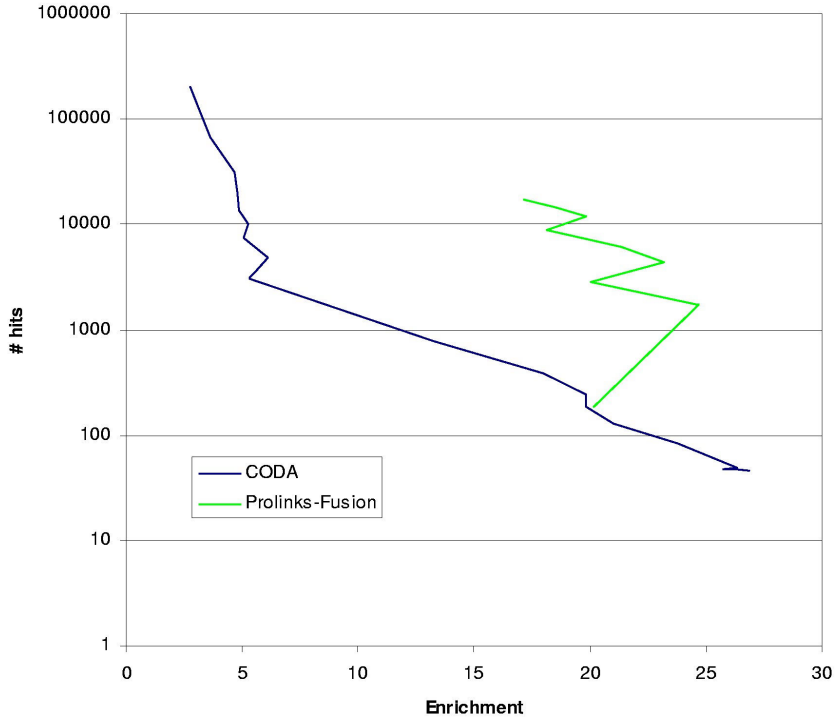
None of the three methods could be run on an arbitrary set of genomes/sequences, only results based on specific datasets were available. Therefore, in order to produce a fair benchmark it was necessary to use only those sequences which had been used to produce the results provided by the respective webservers. New MDA datasets were generated from the sequences provided by these resources (see 3.2.2) so that no extra information was available to CODA either in the query genome or the reference genomes. This also meant that it was not possible to directly compare all three methods. CODA was compared to STRING-Fusion on the STRING sequence set to Prolinks-Fusion on the Prolinks sequence set and to Truong-fusion on the Truong sequence set.

#### *3.3.2.1. Relative Performance of CODA and Other Methods*

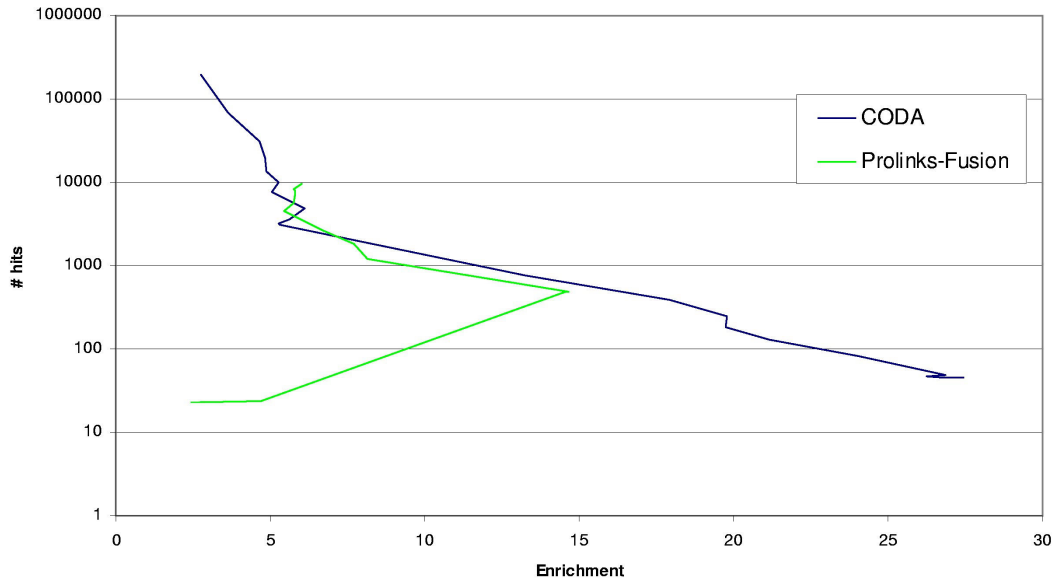
Figure 3.11a shows that CODA outperformed STRING-Fusion at almost all levels of enrichment. STRING-Fusion considers only pairs of proteins thought to be orthologous to fusion proteins and so had a relatively small maximum number of hits, 548. This was at an enrichment of 16.3. For a similar enrichment, CODA found 1549 hits. CODA found 2246 hits for an enrichment of 10.


1000000
100000
10000
1000
100
10
1
# hits
0
5
10
15
20
25
30
35
40
Enrichment
CODA
STRING-Fusion

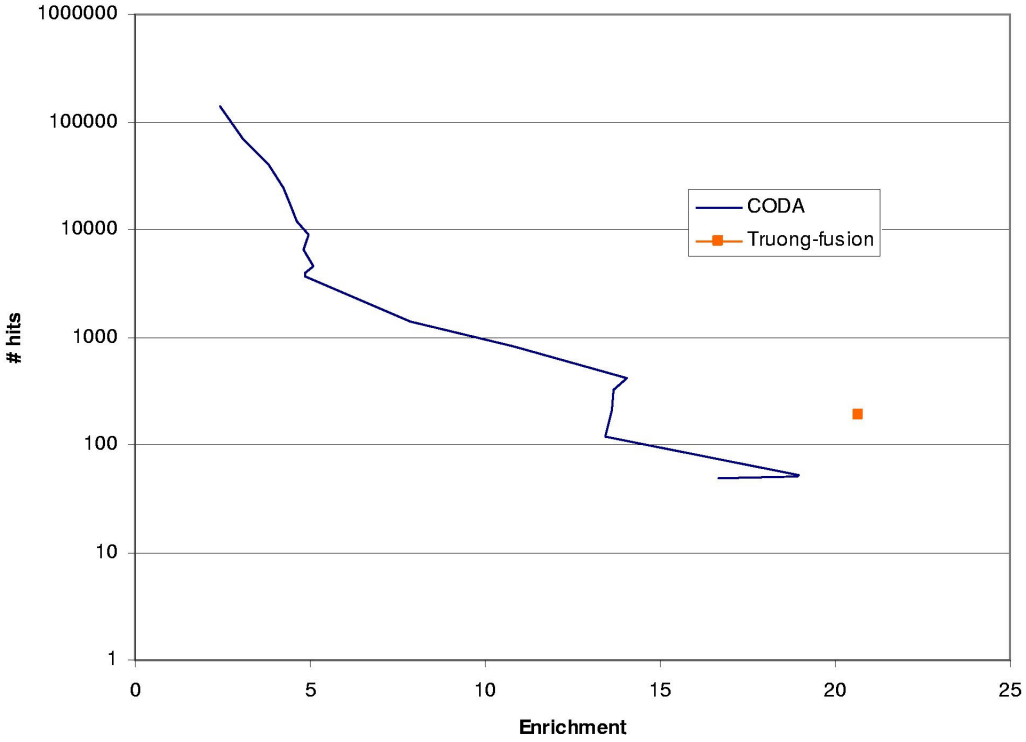
(a)


1000000
100000
10000
1000
100
10
1
# hits
0
5
10
15
20
25
30
Enrichment
CODA
Prolinks-Fusion

(b)




(c)




(d)

Figure 3.11 Performance of CODA relative to the other methods.

(a) performance of CODA (blue) and STRING-fusion (red) methods on the STRING dataset, using yeast as query. (b) relative performance of CODA (blue) and Prolinks-fusion (green) using Prolinks dataset with yeast as query. (c) relative performance of CODA (blue) and Prolinks-Fusion (green) using Prolinks dataset with yeast as query with all results involving homologous pairs removed (BLAST E-value <1e-6). (d) relative performance of CODA (blue) and Truong-fusion (orange) using Truong dataset with yeast as query.

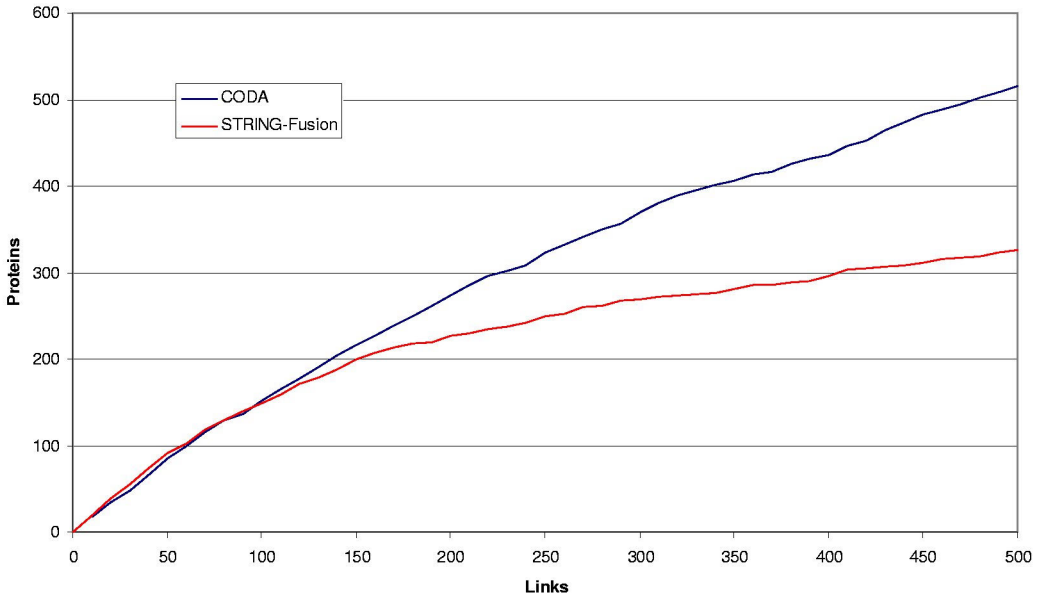
Figure 3.11b shows that Prolinks-fusion far outperformed CODA. For an enrichment of 10 CODA found 1312 protein pairs while Prolinks-fusion found 17361 pairs (all its results) for a higher enrichment of 17. Figure 3.11c shows that the improved performance of Prolinks over CODA was due to a large number of links between homologues. In fact when homologous pairs were removed from the results of both methods (pairs with BLAST E-value <= 1e-6), CODA found 1306 protein pairs for an enrichment of 10, while Prolinks-fusion found only 1021. Note that CODA explicitly excludes pairs with homologous domains.

Figure 3.11d shows the results for CODA against Truong-fusion. There was no score provided for results from Truong-fusion and so there is only one point on the plot referring to the complete set of 189 pairs of proteins identified by the method. Compared to CODA, Truong-fusion is more accurate for the number of hits it produces, with an enrichment of 21 for 189 hits. CODA found 52 hits for an enrichment of 19 and was able to find 1023 hits for an enrichment of 10.

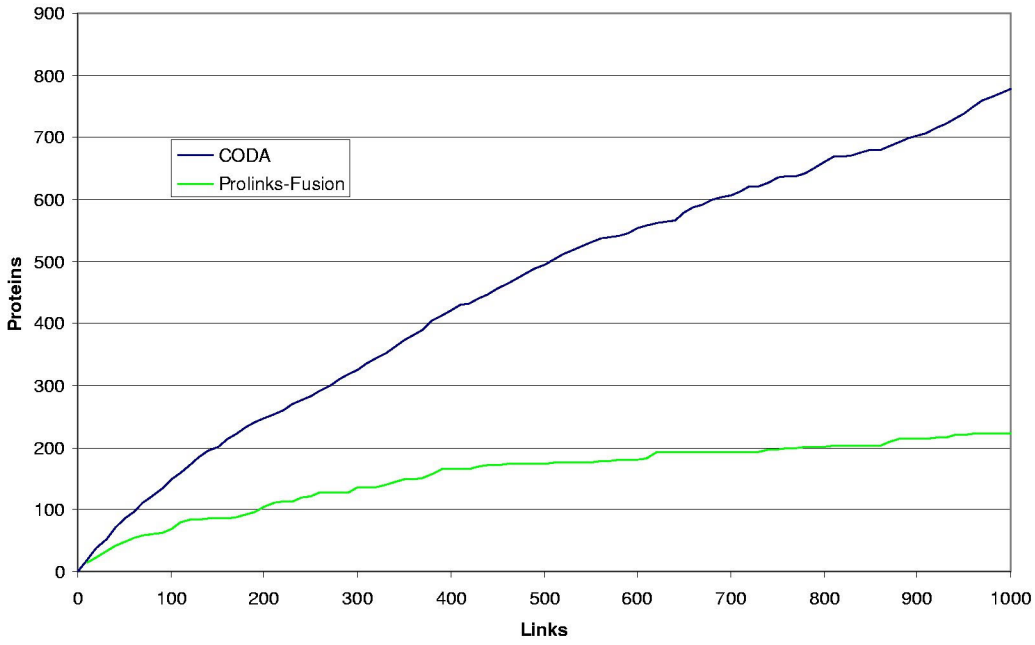
#### *3.3.2.2. Domain Fusion Methods Find Functional Associations for More Proteins than Gene Fusion Methods*

Do fusion methods tend to find many links between few proteins, or few links between many proteins? In order to examine the number of links vs. proteins produced by the methods the first 500 top scoring hits from CODA were taken for comparison with the top 500 from STRING-fusion as STRING-fusion only produced ~500 hits (Figure 3.13a). For comparison between CODA and Prolinks-fusion the first 1000 hits were taken (Figure 3.13b). Truong-fusion produced only 189 hits and so these were compared to the top-scoring 189 hits from CODA (Figure 3.13c). The results show that in all cases CODA had a roughly 1:1 relationship between new links and proteins. For each novel link, on average, one of the proteins had not been seen before. Both Prolinks-fusion and STRING-fusion introduced fewer novel proteins for each link. Truong-fusion however behaved similarly to CODA, suggesting

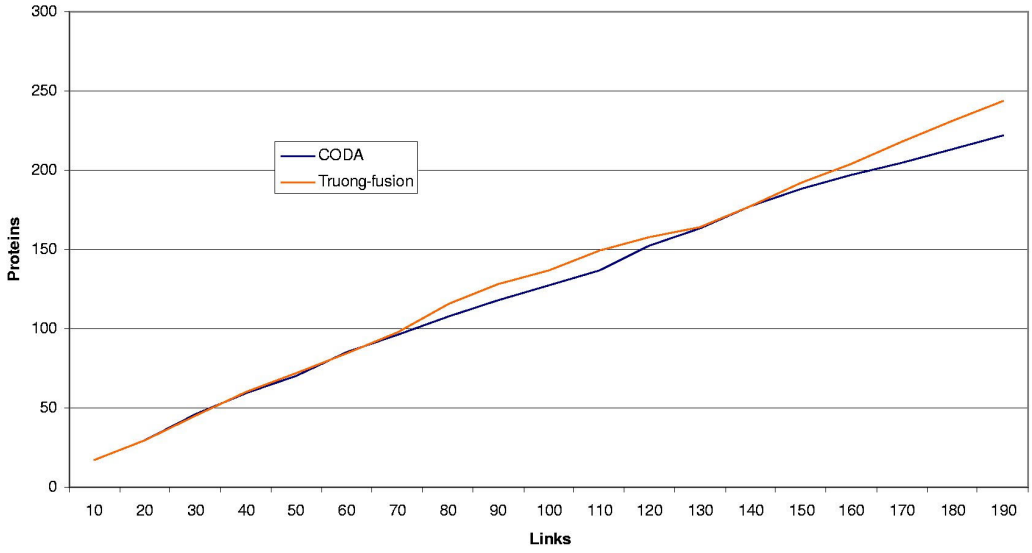
that this behaviour may be a feature of domain fusion methods. It seems therefore that for a given query protein, gene fusion methods provide more links to other proteins and thus increase the probability that there will be functional information available to annotate the query protein. This could be particularly important for query proteins from genomes with a low coverage of functional annotation. Where annotation is more frequent, domain fusion methods may provide a greater increase in coverage by identifying associations for more proteins. Ultimately this suggests that gene and domain fusion methods are complementary and should be used together.


600
500
400
300
200
100
0
Proteins
CODA
STRING-Fusion
0
50
100
150
200
250
300
350
400
450
500
Links

(a)


900
800
700
600
500
400
300
200
100
0
Proteins
CODA
Prolinks-Fusion
0
100
200
300
400
500
600
700
800
900
1000
Links

(b)




(c)

Figure 3.13 Relationship between number of links and proteins.

(a) CODA (blue) and STRING-fusion (red), (b) CODA (blue) and Prolinks-fusion (green), (c) CODA (blue) and Truong-fusion (orange).

### 3.3.2.3. Overlap Between the Results of Different Methods

There was only a small overlap between CODA and the gene fusion methods (STRING-fusion and Prolinks-fusion) in the identity of proteins for which functional links were identified (Figure 3.15a). There was a larger overlap between CODA and Truong-fusion as might be expected from their more similar methodologies. In terms of the specific pairwise associations found the overlap was much smaller however (Figure 3.15b). Out of 500 links CODA and STRING-fusion shared only 54, and of the proteins found shared only 97. CODA and Prolinks-fusion shared only 4 of the 1000 links and 26 of the proteins. Despite their similar methodologies, CODA and Truong-fusion do not find any of the same links amongst the first 189 hits. These results further indicate that there is potential for integrating different methods of gene and domain fusion to increase prediction power in determining proteins involved in common biological processes.

### 3.3.2.4. Assessment of Performance in the Human Genome

In previous work the analysis of gene fusion for function prediction has been largely limited to prokaryotes and yeast. The reason for this is that in higher eukaryotes, many gene/domain families have expanded resulting in increased noise in the fusion signal. So far in this work results have been presented in *S. cerevisiae*, a eukaryote with a small genome. How might CODA and the other methods fare given a much larger genome with large homologous domain families? Using a very different genome such as human also provides an independent validation of CODA's performance.

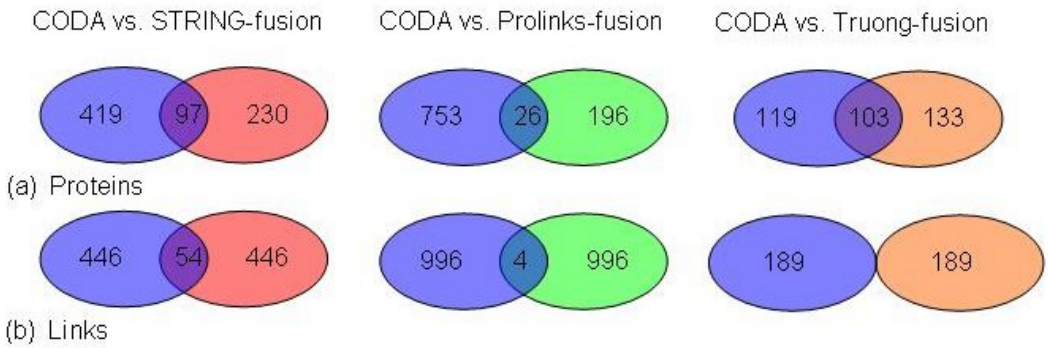


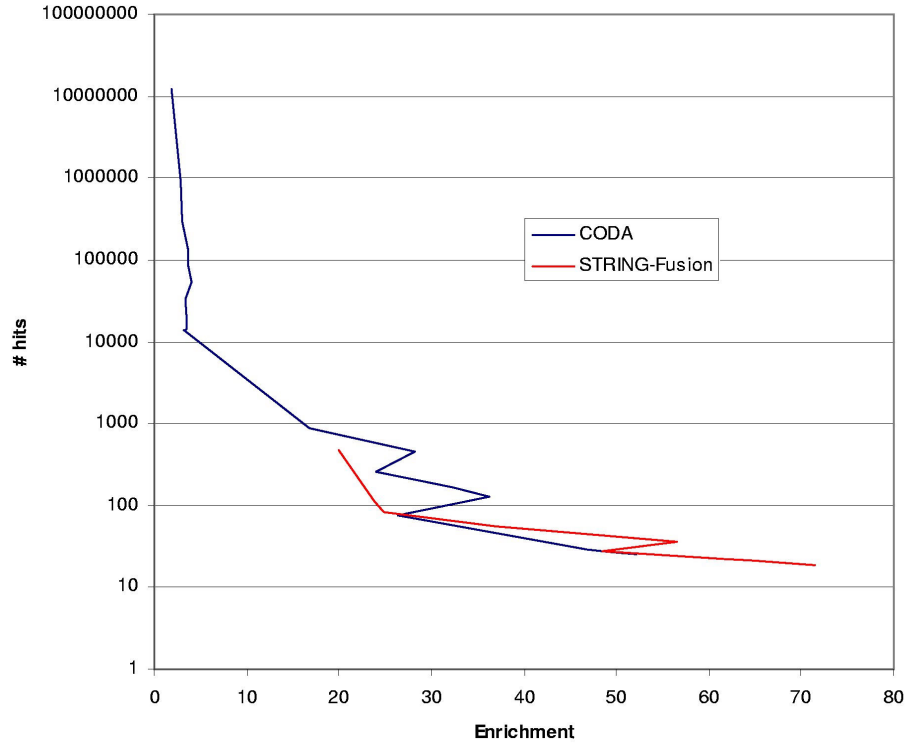

Figure 3.15 Overlap in proteins and linked pairs of proteins identified by fusions.

Data is shown for the top scoring 500 hits for CODA and STRING-fusion, the first 1000 hits for CODA and Prolinks-fusion and the first 189 hits for CODA and Truong-fusion. CODA is represented by blue ellipses, STRING-fusion by red and Prolinks-fusion by green and Truong-fusion by orange.

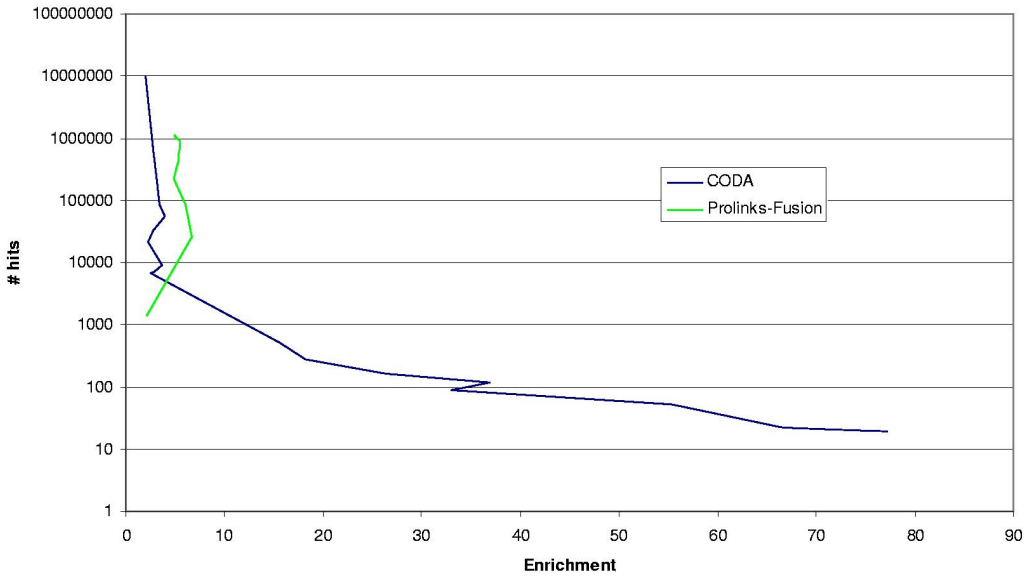
Figure 3.17a shows that STRING-fusion and CODA performed well despite the increased problems of promiscuity and large gene/domain families in the human genome. CODA outperformed STRING-fusion, finding 3932 hits at an enrichment of 10. STRING-fusion found a maximum of 561 hits for an enrichment of ~20; at this enrichment CODA found 1118 hits. STRING-fusion was able to achieve the highest enrichment of the two methods, finding 20 hits for an enrichment of 70. As might be expected, CODA discovered a greater number of protein pairs in human than in yeast (for the same enrichment), there being more functional links to discover in this organism.

Prolinks-fusion did not maintain its performance on the human genome (Figure 3.17b). CODA found 1611 protein pairs for an enrichment of 10, while Prolinks-fusion found none. The greatest enrichment that Prolinks-fusion achieved in human was 6.7, although it did find >25000 pairs at this level. At higher levels of enrichment CODA was able to find ~100 hits for an enrichment of >30. Note that CODA found fewer hits in the Prolinks dataset than the STRING dataset as the Prolinks dataset was somewhat smaller (see Table 3.2).

Results from the Truong-fusion method had been collected using Swiss-Prot release 39 and TrEMBL release 17. These datasets were released in 2001 at which point the human genome was not complete. CODA requires complete genomes for accurate scoring and therefore it was not possible to compare CODA against Truong-fusion for human. Truong-fusion was benchmarked alone however and found 235 associations between human proteins for an enrichment of 28.




(a)




(b)

Figure 3.17 Performance of CODA relative to other methods on the human genome. (a) CODA vs. STRING-fusion. (b) CODA vs. Prolinks-fusion.

### 3.3.3. Applying CODA to Identify Novel Associations Between Proteins

Annotations from the OMIM (Online Mendelian Inheritance in Man) database (McKusick, 1998) were extracted from Gene3D for those proteins identified by CODA in human. Only those links identified by CODA with a score of 0.56 or greater were included. This score cut-off was found to represent an enrichment of 10 for both yeast and human datasets. Uncharacterised proteins which were linked directly to proteins involved in human disease were identified.

#### *3.3.3.1. A Protein Predicted to be Involved in Depression*

Several proteins involved in mental disorders were found to be associated with Q6NZ37 (UniProt Id) using CODA. Tryptophan 5-hydroxylase 2 (TPH2; UniProt: Q8IWU9) is known to be involved in major depressive disorder (MIM: 608516) and is directly involved in the biosynthesis of serotonin from L-tryptophan. Another associate of Q6NZ37, Tryptophan 5-hydroxylase 1 (TPH; MIM:191060) has been shown to be involved in suicidal behaviour, thought to be related to depression (Bellivier et al., 2004). Several other associates of Q6NZ37 are known or thought to be involved in serotonin biosynthesis. Additional associates Sialic Acid Synthase (NANS; Q9NR45) and Quinolinate Phosphoribosyltransferase (QPRT; Q96G22) are known to be involved in brain function. Sialic acid is linked with development of neural tissues during embryogenesis (Hoffman and Edelman, 1983) and quinolate levels in human brain are thought to be involved in the pathogenesis of neurological disorders (MIM: 606248). Quinolate metabolism also feeds into serotonin metabolism. Searches within STRING, Prolinks and Truong data gave no associations for this protein.

#### *3.3.3.2. A Protein Associated with DNA Replication and Disease*

Another example of a functionally coherent network of interactions identified by CODA centred on DNA ligase 1. Mutations in this gene have

been linked with rare cases of multi-symptomatic disease (Barnes et al., 1992). A protein of unknown function, Q96LW4, was linked to DNA ligase 1, suggesting that it may also be involved in multi-symptomatic disease.

CODA also identified a previously known relationship between DNA ligase 1 and DNA primase. The primary role of DNA ligase 1 is in joining Okazaki fragments during lagging strand DNA replication. DNA polymerase is only able to synthesise strands in a 5` to 3` fashion, however the lagging strand must be synthesised 3` to 5`. This is accomplished by discontinuous 5` to 3` extension. A primase enzyme synthesises an RNA primer which is then extended 5` to 3` by DNA polymerase creating Okazaki fragments. These are subsequently joined at the phosphate backbone by DNA ligase I. CODA found a link between DNA ligase I (LIG1) and DNA primase small subunit (PRIM1); their concerted role in DNA replication is clear from the above explanation. Two DNA ligase III (LIG3) enzymes are also linked to PRIM1 by CODA; LIG3 is involved in DNA base excision repair, a process related to DNA replication.

Although the association between DNA ligase and DNA primase is already well established, this example shows the ability of CODA to identify the role of proteins in biological processes. Not least, we have also found a potential role in DNA replication for a currently uncharacterised human protein. Searches within STRING, Prolinks and Truong data gave no associations for this protein.

### 3.3.4. Additional Functional Coverage Produced by CODA

The amount of additional functional coverage of the human genome that could be generated by CODA was determined. CODA found 1453 high confidence (CODA score >=0.56) associations between 900 human proteins using the Gene3D dataset. Of these 900 proteins, 664 could already be annotated with a GO biological process term using annotation from Gene3D v5, allowing all evidence types. Of the remaining 236 unannotated proteins, 107 could be annotated by transferring high quality GO annotation

(experimental evidence and author statements) using the associations established by CODA. Although this is a small number of proteins in terms of the whole human genome, these proteins have not been annotated with GO terms before. The annotations for these proteins are presented in Appendix B.

## 3.4. Discussion

Several aspects of using domain fusion to identify functionally associated protein pairs have been explored. A new method, CODA, was developed and compared against existing implementations of gene/domain fusion using a benchmark based on the Gene Ontology (Harris et al., 2004).

CODA is a domain fusion rather than gene fusion method and several different protein domain representations were trialled in the development of the method. It was shown that Pfam domains give improved performance over CATH domains in identifying functional similarities, largely due to superior coverage of the genomes. Indeed, combining the two domain resources can increase domain coverage of the genomes and this may be used to improve performance in detecting functional relationships at some error rates.

Our approach considers all homologues of fusion proteins rather than focussing on orthologues alone. When large domain families are involved, many homologous pairs will not be involved in similar biological processes. Previous methods have either considered only orthologues (Snel et al., 2000), accepted high false positive rates (Enright and Ouzounis, 2001) or implemented a scoring system based on the frequencies of domain families in the whole target sequence database (Marcotte and Marcotte, 2002). Rather than using counts of domain frequency across all genomes as in previous methods, the CODA score uses domain counts within individual genomes. The CODA score was shown to cope well with the problem of promiscuous domains as well as large homologous domain families.

CODA was shown to outperform the gene fusion method from the STRING resource on the yeast genome at a range of error rates, finding up to four times as many functional associations. The gene fusion method implemented in the Prolinks resource (Prolinks-fusion) found ten times more hits than CODA at moderate error rates on the yeast genome, however many of the functional associations were between pairs of homologous proteins.

When these were removed, CODA and Prolinks-fusion perform similarly on yeast at moderate error rates. One advantage of genome context methods over traditional pair-wise sequence comparisons lies in the fact that they do not require homology between the functionally linked proteins. The domain fusion method of Truong & Ikura (2003) outperformed CODA at low error rates, however at moderate error rates CODA was able to find many more functional associations.

Gene/domain fusion methods in general have been thought to perform better in prokaryotes than eukaryotes as prokaryotes tend to have smaller families of homologous genes/domains (Marcotte and Marcotte, 2002). These methods have therefore rarely been benchmarked in more complex genomes. Here it was shown that CODA and STRING-fusion are both robust to the complexities of the human genome, achieving high accuracy and coverage, with CODA finding around seven times more results than STRING for a reasonable error rate. At very low error rates STRING outperformed CODA. Prolinks-fusion did not perform as well in human as in yeast, probably due to the increased problems of large homologous domain families and promiscuous domains. CODA could not be compared to the Truong & Ikura method on the human genome; however it was shown that their method was able to maintain accuracy on this dataset.

There are two niches that these methods seem to occupy. The methods which can achieve the highest accuracy but which provide a relatively small number of hits (STRING-fusion and Truong-fusion) are useful for identifying high quality sets of associations. However for any particular protein it is unlikely that they will find an association. Methods such as Prolinks-fusion and CODA can provide less certain associations for a greater number of hits and therefore would be more appropriate where the other methods cannot provide associations.

Interestingly there was little overlap between the methods in terms of the functional links they predicted and even the proteins included in the links. This suggests that the particular implementation greatly affects the

links obtained (e.g. using domains vs. whole proteins). Furthermore as different genome context methods have been combined to produce larger sets of confident predictions, using different implementations of the gene/domain fusion method could allow a greater number of predictions overall.

Finally it was shown that CODA was able to identify possible functional associations for uncharacterised proteins in humans. The associations found by CODA suggest that the uncharacterised protein Q6NZ37 (UniProt identifier) is involved in serotonin synthesis and potentially with neurological conditions such as depression. The uncharacterised protein Q96LW4 (UniProt identifier) was found to be associated with DNA ligases and indirectly with a DNA primase and it is therefore likely to have a role in DNA replication. These propositions of course remain to be shown directly by experiment. The other methods featured were not able to give clues about the function of these proteins.

Many previously unannotated human proteins were assigned high confidence GO terms using CODA suggesting that this approach will also be able to annotate previously undescribed proteins in many other genomes.

The methodology presented here allows accurate prediction of larger functional networks than previously determined by gene or domain fusion in higher eukaryotes. One future aim is to combine CODA with other functional association prediction methods. A new pipeline currently in development (Gene3D-BioMiner) will integrate predicted associations generated from methods including phylogenetic profiles (Phylo-Tuner; Ranea et al., 2007), gene expression, and inheritance of experimental protein-protein interactions. A project is currently underway to provide access to all these resources, including CODA, via webservices.

# Chapter 4 Comparative Evolutionary Analysis of Protein Complexes in *E. coli* & Yeast

## 4.1. Introduction

### 4.1.1. Protein Complexes

Most proteins in cells carry out their function as subunits of protein complexes (Alberts, 1998). These aggregations range in size from two to >70 individual peptide chains and can be complexed with other types of molecules such as RNA and DNA. Small complexes often comprise multiple copies of the same protein but large complexes such as the ribosome tend to contain many different proteins. Complexes can be stable as in the case of the proteasome or transient as in the case of a kinase interacting with its substrate. The role of these high order structures is to coordinate complex processes which require the colocation of separate functional elements.

Dezso et al. (2003) have shown that yeast protein complexes contain an essential, invariant core with irreplaceable biochemical function. The phenotype resulting from deletion of core proteins reflects the role of the complex as a whole. Furthermore recent work has suggested that complexes consist of cores, modules and attachments (Gavin *et al.*, 2006; Pang *et al.*, 2008). Gavin et al. (2006) repeatedly purified hundreds of yeast complexes using Tandem Affinity Purification (TAP) and clustered the components based on their frequency of occurrence. Complex members were then classified into three groups: cores, attachments and modules. Core proteins

were those which almost always appeared in a particular complex, attachments those which were less frequently observed. Modules were defined as groups of attachment proteins which always occurred together, often in different complexes. In functional terms, this suggests that attachment proteins are modifiers which are expressed at certain times to change aspects of complex function. A classic example of this is the variety of sigma factors available to bacterial RNA polymerase which alter its specificity for different promoter sequences (Ishihama, 2000).

It is currently unclear to what extent protein complexes are conserved between species. Given a particular complex in one species, many species have largely homologous complexes which are deficient in some of the subunits (Snel and Huynen, 2004). Additionally there is a very low overlap in Protein-Protein Interactions (PPIs) detected between species (Suthram et al., 2005) suggesting that PPIs may change rapidly during evolution (Mika and Rost, 2006), however this may also be due to a lack of experimental evidence. Recent work using combined PPI datasets suggests that pairs of complex members are well conserved between yeast and human (van Dam and Snel, 2008). Van Dam and Snel argue that PPIs between species rarely change within protein complexes but that complexes evolve through gain and loss of subunits. There is evidence that the Last Universal Common Ancestor (LUCA) contained protein complexes related to those of extant organisms (Ranea et al., 2006).

The evolutionary conservation of some complexes has been examined in detail. Comparisons of the eukaryotic SWI/SNF and RSC chromatin remodelling complexes have shown that they consist of an evolutionarily conserved core of subunits (Monahan et al., 2008). Across eukaryotes there are variations in accessory subunits involved in these complexes. Some subunits, present in multiple species, may be necessary for organismal viability in one case but not another.

Two contrasting modes of complex evolution are shown by the eukaryotic and prokaryotic NADH:Ubiquinone oxidoreductase, also termed

complex I. While the early prokaryotic complex is thought to have formed from the combination of small pre-existing complexes (Friedrick, 2001), it appears that the eukaryotic complex tripled in size by step-wise recruitment of new subunits (Gabaldon et al., 2005).

Many small complexes observed in structural data are homodimers and this arrangement confers several advantages. Firstly, homodimers can evolve stable interactions more parsimoniously than heterodimers (Levy et al., 2006). Secondly, producing larger complexes from a single component rather than multiple components allows for greater genetic efficiency, requiring only a single gene and regulatory mechanism.

It has been proposed that some homomeric complexes have diverged by duplication of the gene encoding the self-interacting protein (Pereira-Leal et al., 2007). The duplication of such a gene allows for divergence of one partner resulting in functional diversification and asymmetrical gain and/or loss of interactions in the complex. The F1 ATP synthase and the RecA recombinase homohexamer are examples of complexes which appear to have evolved in this manner, probably from the same homomeric ancestor (Yu and Egelman, 1997). There is evidence for between one tenth and a third of complexes in yeast having evolved in this way depending on the dataset considered (Pereira-Leal et al., 2007).

Duplication of complexes has been shown to be important in yeast (Pereira-Leal and Teichmann, 2005). It is thought that duplication results in complexes with similar general function but novel specificities. It appears that complexes rarely duplicate in their entirety, but more commonly in a partial, stepwise fashion.

### 4.1.2. Protein Complex Datasets

Protein complex datasets fall into four types. Those arguably most accurate are the relatively small curated datasets provided for yeast by the MIPS (Mewes et al., 2008) resource and for *E. coli* by EcoCyc (Karp et al., 2007). Complexes derived from structural data (e.g. Protein Quaternary

Structure database; Henrick and Thornton, 1998) are also thought to be very accurate, although relatively low in coverage and also biased towards stable interactions. Tandem Affinity Purification linked to Mass Spectrometry (TAP-MS) is a high-throughput experimental approach for identifying protein complexes. Large-scale datasets have been produced for yeast (Gavin et al., 2006; Krogan et al., 2006) and *E. coli* (Butland *et al.*, 2005; Arifuzzaman *et al.*, 2006) using this technique. Such datasets cover a greater proportion of interactomes than curated or structural data.

The fourth source of complex data comprises a range of approaches for computationally inferring complexes from pairwise protein-protein interaction data. Resources such as IntAct (Kerrien et al., 2007), MINT (Chatr-aryamontri et al., 2007) and BIND (Bader et al., 2003) provide datasets of protein-protein interactions in a range of species, derived from various low and high-throughput experiments including TAP-MS. Details of experiments found in IntAct and MINT are shown in Table 4.1 and Table 4.2 respectively. It has been shown that yeast protein complexes can be accurately inferred from pairwise PPI data using clustering techniques (Brohee and van Helden, 2006). Genetic interaction data (Bandyopadhyay et al., 2008) and predicted interactions such as those found in the STRING database (von Mering et al., 2007) have also been used (von Mering et al., 2003) for this purpose.

| Species | PPIs | Proteins | Genome coverage | Principal experiment types |
|---|---|---|---|---|
| Arabidopsis thaliana | 3256 | 928 | 3% | Two-hybrid 59% |
| | | | | Protein array 22% |
| Caenorhabditis elegans | 4902 | 2966 | 13% | Two-hybrid pooling 92% |
| Drosophila melanogaster | 26086 | 8271 | 52% | Two-hybrid 91% |
| Escherichia coli | 3280 | 2926 | 74% | Pull-down 98% |
| Homo sapiens | 23114 | 7398 | 21% | Anti-bait co-ip 34% |
| | | | | Two-hybrid pooling 27% |
| | | | | Two-hybrid 13% |
| Mus musculus | 3200 | 2353 | 7% | Two-hybrid 48% |
| | | | | Pull-down 9% |
| Plasmodium falciparum | 2744 | 1274 | 24% | Two-hybrid pooling 100% |
| Rattus norvegicus | 762 | 987 | 8% | Two-hybrid 18% |
| | | | | Pull-down 14% |
| Saccharomyces cerevisiae | 16035 | 5429 | 97% | Two-hybrid fragment pooling 29% |
| | | | | TAP 22% |
| | | | | Two-hybrid array 20% |
| | | | | Two-hybrid 15% |
| Schizosaccharomyces pombe | 578 | 314 | 6% | Pull-down 22% |
| | | | | Two-hybrid 21% |
| | | | | Anti-tag co-ip 21% |

Table 4.1 IntAct interaction datasets for genomes with more than 500 known interactions.

Genome coverage is the percentage of the genome which is captured in the interaction experiments. Only those experimental methods that make up more than 10% of the total number of experiments for an organisms are listed.

| Species | PPIs | Proteins | Genome coverage | Experiment types |
|---|---|---|---|---|
| Caenorhabditis elegans | 2798 | 1934 | 9% | Two-hybrid pooling 91% |
| Drosophila melanogaster | 19366 | 6734 | 42% | Two-hybrid pooling 97% |
| Escherichia coli | 2370 | 713 | 18% | Anti-tag co-ip 64% |
| Homo sapiens | 11476 | 3914 | 11% | Two-hybrid 27% |
| | | | | Pull-down 12% |
| | | | | Co-ip 11% |
| Mus musculus | 2573 | 1110 | 3% | Two-hybrid 14% |
| | | | | Anti-bait co-ip 12% |
| | | | | Pull-down 11% |
| Plasmodium falciparum | 604 | 574 | 11% | Two-hybrid fragment pooling 100% |
| Rattus norvegicus | 1459 | 503 | 4% | Anti-bait co-ip 13% |
| | | | | Pull-down 12% |
| Saccharomyces cerevisiae | 28057 | 3831 | 69% | TAP 46% |
| | | | | Two-hybrid pooling 34% |

Table 4.2 Genome-based interaction data from MINT.

See Table 4.1 legend for details.

### 4.1.3. Methodologies for Predicting Complexes

The *in silico* study of protein complexes has largely focussed on yeast where there is a greater quantity of data than for other organisms. Many of these studies have used structural and/or TAP-MS complexes (e.g. Pereira-Leal et al., 2007; Tamames et al., 2007). Several authors (Brohee and van Helden, 2006; Pereira-Leal et al., 2004; Bader and Hogue, 2003) have also explored complexes derived from Protein-Protein Interaction Networks (PINs) using clustering methods. This results in larger datasets of complexes, with greater coverage of genomes than are available from other sources. This is achievable because PINs have highly connected regions which have been shown to correlate with complexes (Bader and Hogue, 2003).

Several different clustering methods have been applied to the task of identifying complexes in PINs. The Markov CLustering algorithm (MCL - Enright et al., 2002) uses flow simulation in graphs to detect clusters and was used by Pereira-Leal et al. (2004) who showed that the clusters were functionally coherent in terms of regulatory and metabolic annotation, cellular localisation data and known complexes. MCODE (Bader and Hogue, 2003) uses local neighbourhood density to define clusters. Both Netcarto (Guimera and Nunes Amaral, 2005) and Restricted Neighbourhood Search Clustering (RNSC) (King et al., 2004) use a cost function and Monte Carlo methods to obtain a division of the graph. Netcarto was used by Tamames et al. (2007) to explore the relationship between reduction in genome size and network modularity. An analysis of several of these methods by Brohee & van Helden (2006) showed that MCL was the best overall method for determining known yeast complexes from PPI datasets.

### 4.1.4. Aims

The evolution of protein complexes is still poorly understood and differences between species have been difficult to study on a global scale. In this Chapter, protein complex datasets are created for a prokaryote (*Escherichia coli)* and a eukaryote (*Saccharomyces cerevisiae)* in order to probe

the differences in complex evolution between species. Combined PPI datasets are derived for each organism based on experimentally determined interactions and a clustering algorithm is used to identify protein complexes. These complexes are shown to be accurate representations of known complexes.

The clustered datasets are used examine the distribution of homologues amongst protein complexes to show how duplicates have been reused. Differences between *E. coli* and yeast are identified, suggesting that their complexes have evolved in different ways.

# 4.2. Methods

## 4.2.1. Summary

Figure 4.1 describes the motivation behind each part of this chapter and details where each type of dataset was used.

## 4.2.2. Experimental Protein-Protein Interaction Datasets

Protein-Protein Interaction (PPI) datasets for *E. coli* and yeast, from the MINT (Chatr-aryamontri et al., 2007) and IntAct (Kerrien et al., 2007) resources were extracted from Gene3D v5 (Yeats et al., 2008). Much of the data from these resources is from high-throughput experiments such as Two-Hybrid and Tandem Affinity Purification (TAP) but is also derived from small-scale pull-down and co-immunoprecipitation experiments. Although most of these interactions are pairwise, those derived from TAP-MS data are between one bait protein and multiple prey proteins. Pairwise PPIs can be extracted from this data using one of two models. The spoke model defines interactions between the bait protein and each of the prey. The matrix model however defines pairwise interactions between the bait and prey proteins and between each pair of prey proteins. TAP-MS data from MINT was already in the matrix form and bait-prey relationships could not be established. IntAct data could be converted into either. Ultimately, the spoke model was used to convert IntAct TAP-MS data into pairwise PPIs as it was shown to perform best in replicating known complexes (see 4.3.1.1).

For the majority of this chapter combined datasets, taking all interactions from both MINT and IntAct were used. For *E. coli* (NCBI taxon id: 562) there were 13941 interactions between 2865 proteins (~72% genome coverage) and for *S. cerevisiae* (NCBI taxon id: 4932) 38825 interactions covering 5735 proteins (~100% genome coverage).

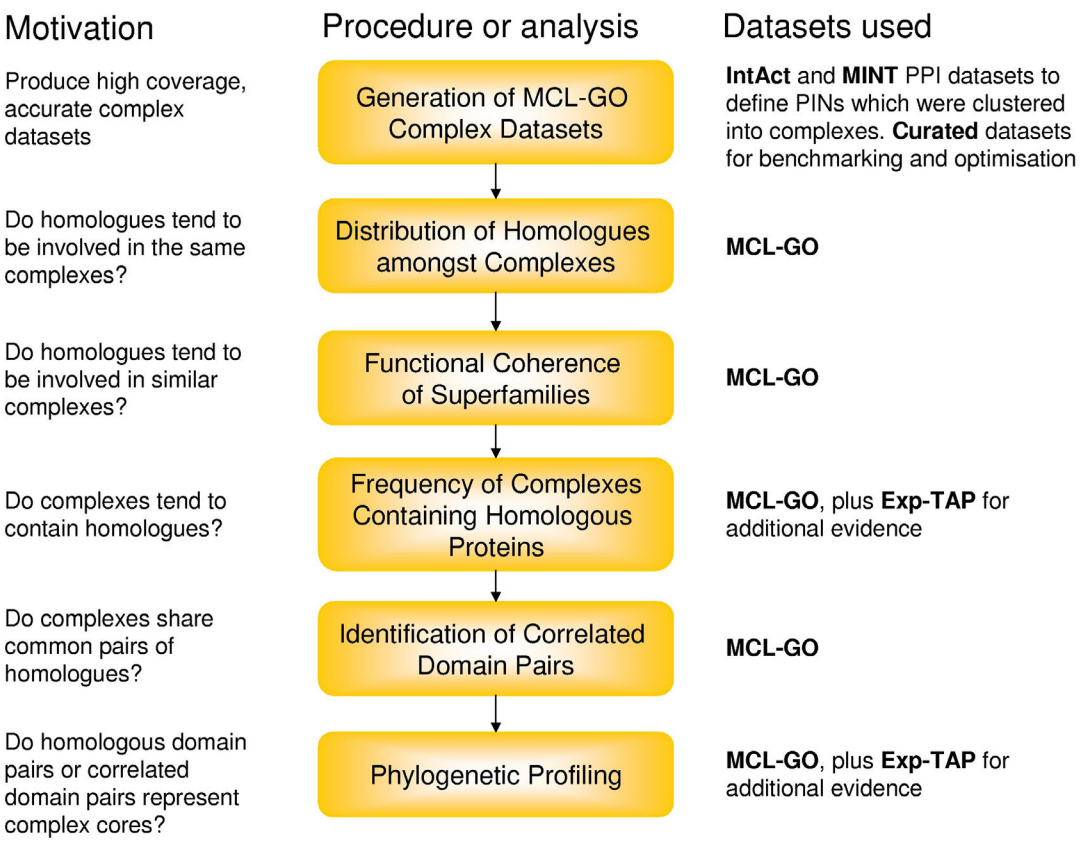



Figure 4.1 Summary of procedures and analyses presented in this chapter.

This figure summarises and highlights the motivations behind each part of this chapter and identifies the datasets used. MCL-GO is the automated approach for generating protein complex datasets used in this chapter and Exp-TAP is protein complex data derived purely from Tandem Affinity Purification Mass Spectrometry experiments.

### 4.2.3. Generating MCL-GO Complex Datasets from PPI Datasets

The *E. coli* and yeast combined PPI datasets described above were clustered into complex datasets using the MCL algorithm (Enright et al., 2002). It has been shown that enriching Protein Interaction Networks (PINs) with functional annotation improves detection of functional modules (Lubovac et al., 2006). Complex datasets were generated with and without weighting of the PINs. Each edge must have a positive weight in order to be considered; therefore unweighted edges were set to one. Weighted edges were set to one plus the Gene Ontology Semantic Similarity (GOSS) score. To generate these GOSS scores, proteins were annotated with GO biological process terms from Gene3D. The GO terms used were those described in 4.2.4. The terms were compared using the Resnik (1999) method described by Lord et al. (2003) to determine their functional similarity (discussed in detail in Chapter 1). Each edge in the network was weighted using the highest GOSS score between any pair of terms assigned to the relevant nodes. Complex datasets generated in this way are referred to as MCL-GO datasets.

The inflation parameter, which controls the granularity of the clusters produced, was optimised by comparing predicted complexes (clusters) with curated, *gold standard* complexes from MIPS in the case of yeast and Ecocyc in the case of *E. coli* (described further in 4.2.5.1). The comparison was performed in the same way as described by Brohee & van Helden (2006), using the same measures of sensitivity, Positive Predictive Value (PPV) and accuracy. When calculating sensitivity and PPV, only those clusters which had at least one member of a known complex were considered.

Sensitivity (Equation 4.1) is the weighted average over all complexes of the proportion of each gold standard complex *i* captured by the predicted cluster *j*, best reflecting that complex.

$$Sn = \frac{\sum_{i=1}^{n} N_i Sn_{co_i}}{\sum_{i=1}^{n} N_i}$$

Equation 4.1 Sensitivity.

In Equation 4.1, $N_i$ is the number of proteins in complex $i$ and $Sn_{co_i}$ is the complex-wise sensitivity defined in Equation 4.2.

$$Sn_{co_i} = \max_{j=1}^{m} Sn_{i,j}$$

Equation 4.2 Complex-wise sensitivity.

The complex-wise sensitivity is the maximum sensitivity $Sn_{i,j}$ for a particular complex $i$, taking the greatest value over all predicted clusters $j$.

$$Sn_{i,j} = \frac{T_{i,j}}{N_i}$$

Equation 4.3 Sensitivity for complex $i$ and cluster $j$.

In Equation 4.3, $T_{i,j}$ is the number of members of complex $i$ in cluster $j$.

Positive Predictive Value (PPV) is a measure of how pure the predicted clusters are, i.e. the maximum percentage of proteins from a known complex in each cluster.

$$PPV = \frac{\sum_{j=1}^{m} T_j PPV_{cl_j}}{\sum_{j=1}^{m} T_j}$$

Equation 4.4 Positive Predictive Value.

In Equation 4.4, $T_j$ is the number of members of cluster $j$ with membership of a known complex and $PPV_{cl_j}$ is the cluster-wise PPV described in Equation 1.1.

$$PPV_{cl_j} = \max_{j=1}^{m} PPV_{i,j}$$

Equation 4.5 Cluster-wise PPV.

The cluster-wise PPV takes the maximum value of $PPV_{i,j}$ for a particular cluster over all complexes. $PPV_{i,j}$ is described in Equation 4.6.

$$PPV_{i,j} = \frac{T_{i,j}}{T_j}$$

Equation 4.6 PPV for complex *i* and cluster *j*.

In Equation 4.6, $T_{i,j}$ is the number of members of cluster *j* in complex *i*.

The trade-off between sensitivity and PPV was captured by taking the geometric mean of the sensitivity and PPV, referred to as the accuracy (*Acc*; Equation 4.7).

$$Acc = \sqrt{Sn \cdot PPV}$$

Equation 4.7 Accuracy.

The accuracy achieved in recreating known complexes using the MCL-GO procedure was compared to that for randomly generated complexes to show that the procedure was useful, as was done by Brohee & van Helden (2006). This was achieved by clustering PPI datasets with MCL, then shuffling proteins between complexes while preserving complex size and benchmarking the resulting complexes. For each value of the MCL inflation parameter, randomisations were performed $10^5$ times.

## 4.2.4. Annotation of MCL-GO Complexes

CATH (Greene et al., 2007) protein domain superfamily annotation was extracted from Gene3D v5 (Yeats et al., 2008) to allow homologous

relationships between proteins to be identified. 2190 CATH domains were identified from 656 superfamilies in the 2210 proteins from the *E. coli* MCL-GO complexes, covering 1579 proteins (71%). The yeast MCL-GO complexes were annotated with 2666 CATH domains from 630 superfamilies over 2070 proteins (44% of protein in this dataset). Throughout this work, multiple members of the same superfamily are ignored within protein chains.

Functional data in the form of Gene Ontology (Gene Ontology Consortium, 2006) annotation was also extracted from Gene3D v5. For *E. coli*, coverage with GO terms derived from experimental annotation was very low and so Electronically Inferred Annotation (IEA) was included, only negative results (ND – No biological Data available) were excluded. This resulted in 3989 biological process terms over 1803 proteins (82% coverage). For yeast MCL-GO datasets, IEA terms were ignored. This resulted in 10622 terms over 3926 proteins for yeast (83% coverage).

FunCat (Ruepp et al., 2004) functional terms were extracted from Gene3D v5. Only the most general (level 1) terms were considered. These were used to annotate MCL-GO complexes as FunCat provides a suitable set of high level terms. There were 12257 terms covering 1573 *E. coli* proteins (71% coverage) and 12385 terms covering 3432 proteins in yeast (72% coverage).

## 4.2.5. Pre-defined Protein Complex Datasets

### *4.2.5.1. Curated Datasets Used to Validate Predicted Complexes*

Several pre-defined complex datasets were also used. As described, high-quality, curated datasets of known complexes were required in order to determine how accurately PPI datasets could be clustered into complexes. Such datasets were available from EcoCyc (Karp et al., 2007) for *E. coli* and from MIPS (Mewes et al., 2008) for yeast. The EcoCyc complexes comprised 232 unique, multi-subunit complexes containing a total of 586 distinct protein

sequences. The MIPS complexes comprised 192 non-redundant, multi-subunit complexes containing a total of 1036 distinct protein sequences.

#### *4.2.5.2. Experimental Datasets Used to Assess Trends*

Predicted MCL-GO complexes, derived by clustering PPIs from a variety of experimental approaches (see 4.2.3), were used throughout this work as they had higher coverage of the genomes of each organism than curated datasets or individual experimental approaches such as TAP. However, the MCL clustering method only allows each protein to exist in a single complex. In reality some proteins exist in multiple complexes and this discrepancy could bias inferences made based on the data. Therefore complexes based only on TAP data were also examined as these do allow individual proteins to appear in multiple complexes. TAP experiments identify relationships between one 'bait' protein and multiple 'prey', directly inferring complexes without the need for clustering. These are referred to collectively as Exp-TAP datasets. *E. coli* Exp-TAP complex datasets were derived from Butland et al. (2005) and Arifuzzaman et al. (2006) and downloaded from http://sunserver.cdfd.org.in:8080/protease/PPI/. Yeast Exp-TAP complexes derived from Gavin et al. (2006) and Krogan et al. (2006) were downloaded from BioGRID (Stark et al., 2006). These Exp-TAP datasets are referred to as Butland, Arifuzzaman, Gavin and Krogan, respectively.

Experimental datasets were annotated with GO terms and CATH domains using the same protocols as for the MCL-GO complexes.

### 4.2.6. Determining the Distribution of Homologues in Complexes

In order to examine the distribution of homologues in complexes, the distribution of each CATH domain superfamily was compared to that in randomised complexes. Only domain superfamilies with at least five members in different proteins were considered for this analysis to give the test sufficient statistical power. Of 656 superfamilies in *E. coli*, 101 had at least five members (62% of domains); of 630 in yeast 113 had at least five members

(68% of domains). For each superfamily, the number of distinct pairs of proteins containing that superfamily which were found in the same complex was determined. This was compared to the number of distinct pairs which were found together in $10^4$ randomised complex datasets. Complexes were randomised by shuffling members between complexes, retaining the complex size distribution. For each superfamily, p-values were calculated by determining the proportion of these $10^4$ randomised trials where the observed number of pairs was exceeded.

The False Discovery Rate (FDR) correction for multiple hypothesis testing, as introduced by Benjamini & Hochberg (1995), was applied. When testing a single hypothesis there is a one in 20 chance of a false positive if the p-value is 0.05. However, over 20 hypotheses one would expect one false positive if the p-values are 0.05 for each hypothesis. The FDR is thought to be a less conservative approach than the alternative Bonferroni correction. P-values for the superfamilies ($q_1..q_m$) were ordered such that $q_1 \leq q_2 \leq .. \leq q_m$. Superfamilies were considered non-randomly distributed where the p-value $q$ of that superfamily satisfied the inequality in Equation 4.8.

$$q \leq \frac{k\alpha}{m}$$

Equation 4.8 FDR correction

In Equation 4.8 $k$ is the rank of the ordered p-value, α is the accepted false discovery rate (0.01 in this case) and $m$ is the number of superfamilies.

## 4.2.7. Functional Coherence of Superfamilies

Whether two proteins occur in the same complex is one measure of functional similarity. Another measure of functional similarity, functional coherence, was used at three different levels to examine whether members of a superfamily tended to have a conserved role in the cell. A group of proteins is considered functionally coherent if the semantic similarity between their

GO terms is more similar than expected by chance. Functional coherence was firstly considered at the level of the superfamily, i.e. do proteins containing members of a particular superfamily perform more similar functions than random groups of proteins? Secondly, the functional similarity between those proteins which interact with members of a particular superfamily (interaction neighbourhood) was considered. In other words, do the interactors of one superfamily member perform similar functions to those of another superfamily member? Thirdly, the functional coherence of MCL-GO complexes containing members of a particular superfamily was considered.

At the superfamily level, functional coherence was calculated as the mean GOSS score between pairs of proteins containing that superfamily. GOSS scores between individual pairs were calculated using biological process GO terms as specified in 4.2.4.

At the neighbourhood level, mean GOSS scores were calculated between each of the direct interactors for one member of a superfamily and the interactors of another member of that superfamily. In other words, if protein A interacts with proteins B, C and D and protein A homologue A´ interacts with E, F and G, then each of B, C and D were compared to each of E, F and G. The mean GOSS score over these comparisons was then taken as the functional similarity of the neighbourhoods of the two homologues. For a superfamily, the functional similarity of the neighbourhoods was the mean over each pair of neighbourhood comparison.

At the complex level, the functional similarity between complexes containing a particular superfamily was determined in the same way as for neighbourhoods. Where members of a superfamily occurred in complexes A and B, each member of complex A was compared to each member of complex B and an average GOSS score taken. An average was then taken over each pair of complexes.

In each of the above analyses, the functional similarity of each superfamily was compared to random groups of proteins of the same size as the superfamily. Randomisations were performed $10^4$ times to derive p-

values. The FDR correction was used as described in section 4.2.6 with $\alpha$ = 0.01 (a standard value for this parameter). Superfamilies were considered if they had at least two members which were annotated with biological process GO terms. This criterion was met by 217 *E. coli* superfamilies and 302 yeast superfamilies.

### 4.2.8. Identification of Complexes Containing Homologous Pairs

In examining the proportion of complexes which contained multiple homologues, both domain and protein homologues were considered. Two proteins which shared a common CATH superfamily member were considered domain homologues. Two proteins which shared their entire CATH Multi-Domain Architecture (MDA) were considered protein homologues. MDA is defined as the series of domain annotations from N to C terminus, excluding multiple segments, gaps and tandem repeats. To determine whether complexes tended to contain pairs of homologues the number of complexes which contained at least one pair of homologous proteins (using either domain or protein homologues) was counted. To determine whether the number of observed complexes was significant, the observed count was compared against the distribution of counts derived from $10^4$ randomised complex datasets. P-values were calculated empirically. Complex datasets were randomised by shuffling complex membership while retaining the complex size distribution.

### 4.2.9. Identification of Correlated Domains

Correlated domains are pairs of domain superfamilies which occur together in a greater number of complexes than expected by chance. Instances of co-occurrence were only considered if the domains occur in separate proteins and these proteins do not share any common domains. Correlated pairs of Pfam domains have been identified previously by Betel et al. (2004). For each correlated domain pair occurring in at least two complexes, the frequency of occurrence was compared against frequencies found in $10^4$ randomised

complex datasets and an empirical p-value calculated by determining in what proportion of these datasets the frequency of co-occurrence of the pairs exceeded that observed in the MCL-GO complex dataset. Those pairs with a p-value >0.01 were excluded.

To determine whether proteins containing these correlated domain pairs tended to interact directly, the frequency with which they were observed to interact in MINT and IntAct data was compared to the frequencies of interaction of the same number of randomly chosen co-complex protein pairs. Sets of random co-complex pairs were created $10^4$ times to derive a p-value. To determine whether correlated pairs represented functional units within complexes, the average GOSS score between the proteins in each pair was compared with the average GOSS score between the same number of random co-complex protein pairs. Again this was performed $10^4$ times to derive a p-value.

### 4.2.10. Phylogenetic Profiling

To determine whether correlated domain pairs might represent protein complex cores, it was assumed that proteins in the core of complexes are older than other proteins. The analysis employed by Pereira-Leal et al. (2007) was used to determine the age of protein orthologues. In this approach the age of a protein was determined by the most ancient taxonomic group containing orthologues of the protein. Bidirectional best hit (BDBH) BLAST orthologues were determined for each *E. coli* and yeast protein amongst 32 species (listed in Appendix C). A pair of BDBH orthologues is defined as two proteins *i, j* from genomes *A* and *B* respectively such that when *i* is searched against genome *B*, *j* is the best match and when *j* is searched against genome A, *i* is the best match. This is an approximate approach but is sufficiently accurate for the analysis presented here (Pereira-Leal et al., 2007). Orthologues were defined as bi-directional best hits between two species with an E-value of ≤0.01. The point of origin of a particular protein was

defined by the age group in which an orthologue was found. Age groups were defined using the species tree of Baldauf (2003).

The age groups defined for *E. coli* in this analysis were '*E. coli* specific', 'Proteobacteria', 'Proteobacteria/Firmicutes', 'Bacteria', 'Eukaryota+Bacteria', 'Bacteria+Archaea' and 'Universal'. For yeast: '*Saccharomyces cerevisiae* specific', 'Fungi', 'Metazoa/Fungi', 'Eukaryota', 'Eukaryota+Archaea', 'Eukaryota+Bacteria' and 'Universal'.

The chi-square test was used to determine whether significant differences existed in the age distribution of different classes of proteins. For instance proteins containing correlated domains were compared with all other proteins from the complex dataset in which they were identified.

# 4.3. Results

## 4.3.1. Prediction and Functional Characterisation of Protein Complexes in *E. coli* and Yeast

### *4.3.1.1. Accurate Prediction of Protein Complexes by Clustering Protein Interaction Networks*

In order to study the evolution of protein complexes accurate datasets with high genome coverage were required. An approach similar to that employed by Brohee & van Helden (2006), Pereira-Leal et al. (2004) and Lubovac et al. (2006) was used. Protein-Protein Interactions (PPIs) were combined into Protein Interaction Networks (PINs) and clustered using the MCL algorithm (see 1.3.8.2). The MCL algorithm has been shown to be the best amongst several approaches available for clustering PINs into complexes (Brohee and van Helden, 2006). The MCL clustering algorithm requires a parameter to control the granularity of clusters known as the inflation parameter, *I*. This parameter was optimised on the yeast PIN by determining accuracy against the MIPS dataset of known yeast complexes as was done by Brohee & van Helden (2006), using the same measure of accuracy (see 4.2.3).

Two resources of PPI data were considered, IntAct (Kerrien et al., 2007) and MINT (Chatr-aryamontri et al., 2007). Some data from IntAct, derived from TAP-MS experiments did not directly specify pairwise PPIs. TAP-MS data identifies a complex between one bait protein and several prey and it was necessary to apply one of two models to generate pairwise interactions. The spoke model specifies an interaction between the bait and each of the prey, whereas the matrix model additionally specifies interactions between each pair of prey proteins. Figure 4.3 shows that, where a choice of models could be applied, the spoke model gave higher accuracy in identifying known yeast complexes from MIPS (Mewes et al., 2008). TAP-MS data from MINT had already been rendered using the matrix model. The spoke model was subsequently applied to all TAP-MS data from IntAct.

Edges in the PINs were then weighted using the semantic similarity of the biological process GO terms of the corresponding nodes (see 4.2.3). Figure 4.5 shows that combining MINT and IntAct and weighting edges with semantic similarity improved the performance over either method alone, with or without weighting. This optimised approach is referred to as MCL-GO and datasets derived from it as MCL-GO datasets.

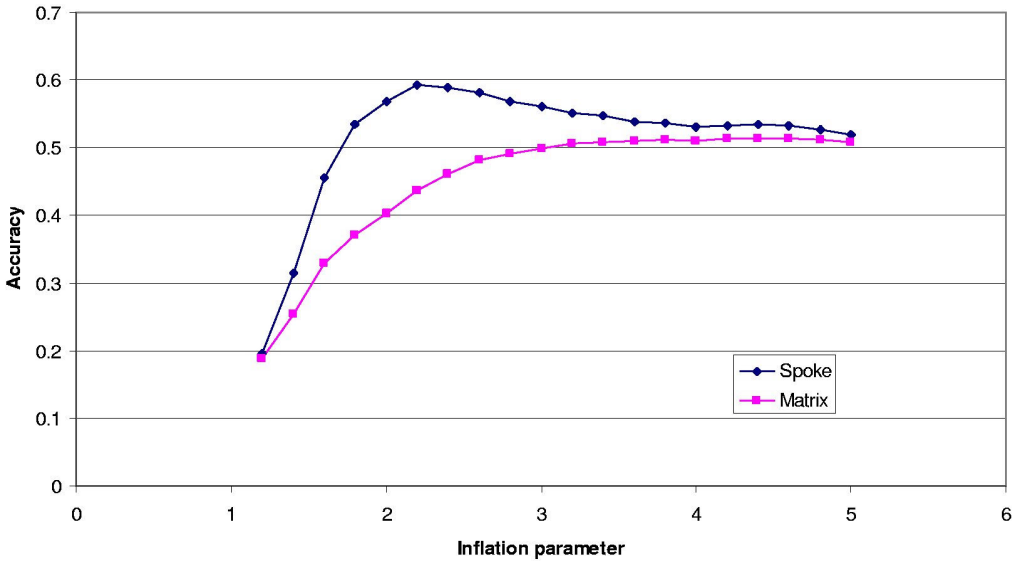



Figure 4.3 Difference in accuracy when clustering protein-protein interactions rendered in spoke and matrix models.

For yeast IntAct data, rendering TAP-MS data using the spoke model rather than the matrix model gave improved performance. All yeast IntAct data was included here, not just TAP-MS.

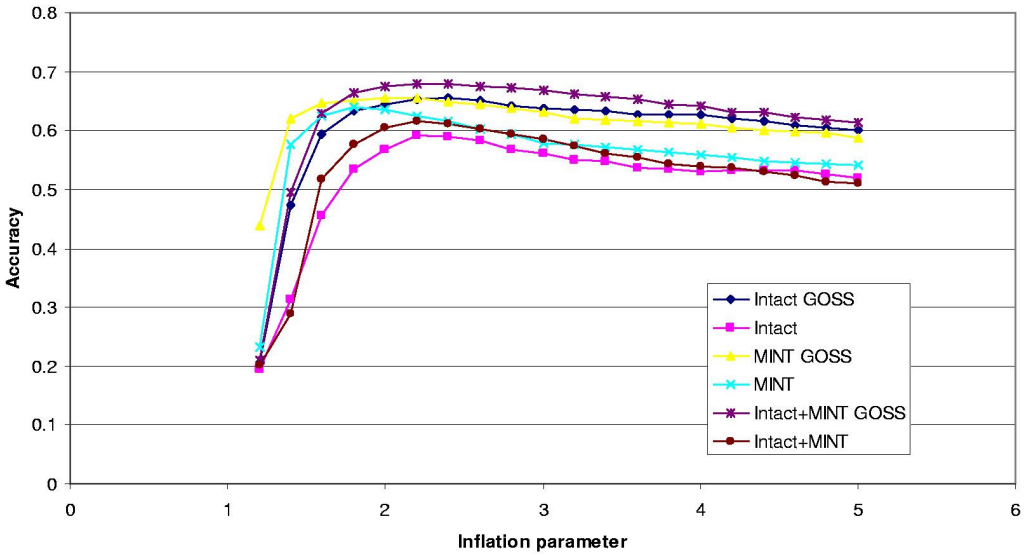



Figure 4.5 Combining IntAct and MINT datasets and weighting interactions with GOSS scores resulted in greater accuracy over either resource alone and without weighting.

Figure 4.7 shows that the maximal accuracy for reproducing yeast MIPS complexes was achieved with *I*=2.2, similar to the value of 1.8 found to be optimal by Brohee & van Helden (2006) on a different dataset. The accuracy achieved here (0.68) is comparable to that achieved in recent studies (Krogan *et al.*, 2006; Zheng *et al.*, 2008).

Figure 4.7 also shows the accuracy of *E. coli* MCL-GO complexes in reproducing the known *E. coli* complexes from EcoCyc. The optimal value of *I* was also 2.2. Although there is a slight increase in performance at higher inflation parameter values the separation from random is much greater at *I*=2.2. The accuracy for *E. coli* complexes is noticeably lower than for yeast, although still very much above random. This poorer performance may have been caused by lower coverage of the *E. coli* genome with PPIs compared to yeast.

The MCL-GO clusters for each species were filtered to remove clusters containing only one protein. This resulted in 574 predicted *E. coli* complexes containing a total of 2210 distinct protein sequences and 855 predicted yeast complexes containing a total of 4740 distinct protein sequences. These complex datasets thus cover roughly 56%, and 85% of *E. coli* and yeast genomes respectively based on genome sizes of 3952 and 5586 genes (genome sizes were taken from Integr8 (Kersey et al., 2005)). Figure 4.9 shows the size distribution of complexes. On average yeast complexes were larger than *E. coli* complexes.

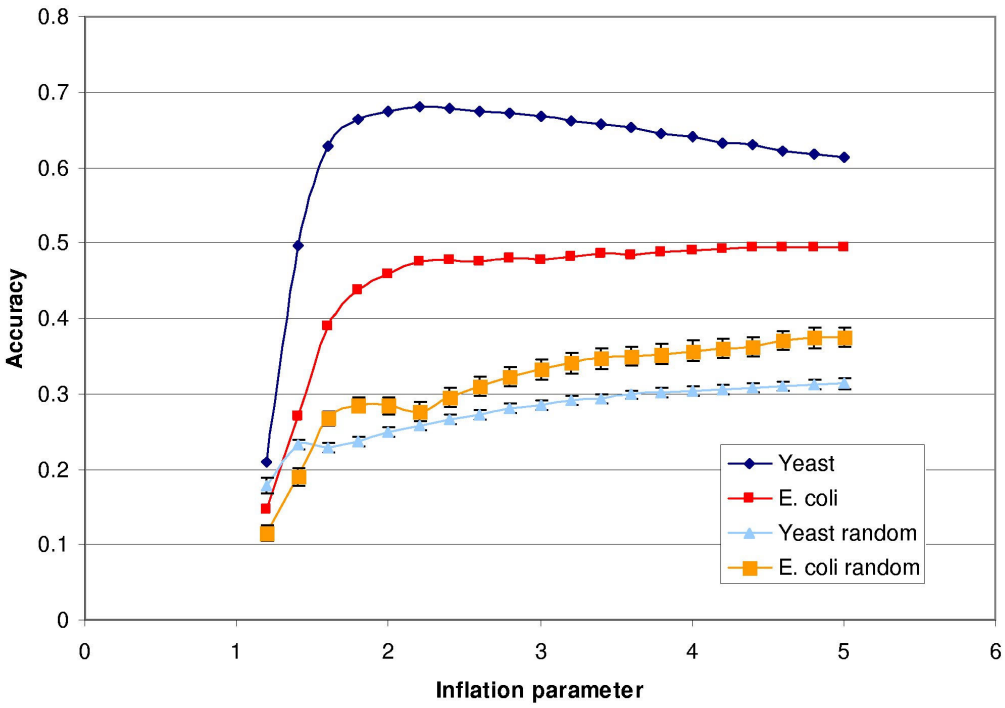



Figure 4.7 Accuracy of MCL-GO complexes (using MINT+IntAct and edge weighting) in capturing MIPS yeast complexes and EcoCyc *E. coli* complexes.

'Random' lines show mean accuracy achieved over $10^4$ sets of randomised clusters. Error bars show one standard deviation either side of the mean.

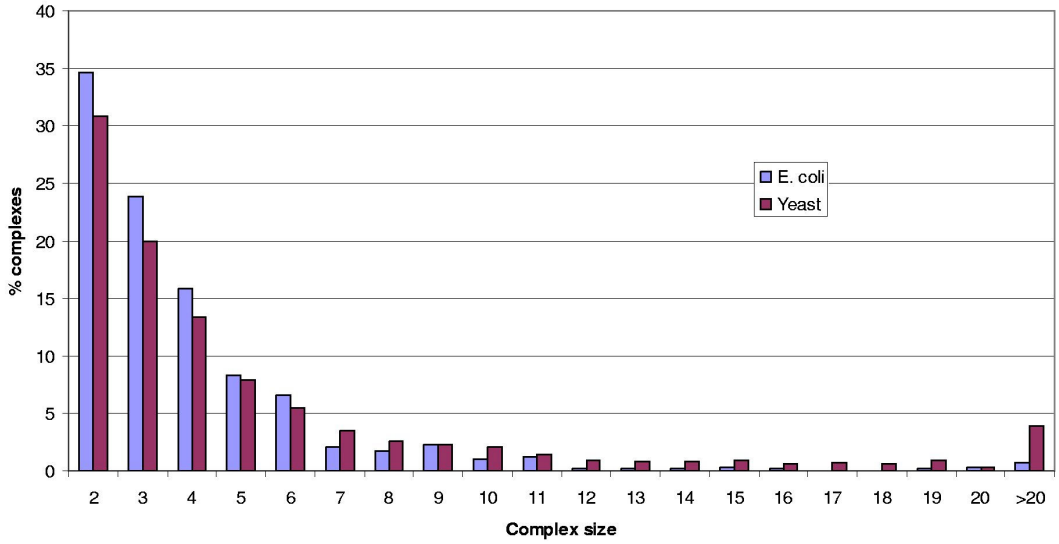



Figure 4.9 Size distribution of *E. coli* and yeast MCL-GO complexes.

Complex size is the number of proteins in the complex.

#### *4.3.1.2. Functional Classification of Predicted Protein Complexes*

To determine whether the MCL-GO complex datasets made biological sense, their functions were analysed using FunCat terms (Ruepp et al., 2004). The FunCat classification of protein function is described in detail in Chapter 1. Figure 4.11 shows the percentage of proteins in each complex which could be annotated with the most common level one FunCat term in that complex. Complexes with less than two terms were excluded leaving 453 *E. coli* complexes (79%) and 725 yeast complexes (85%). For both *E. coli* and yeast around one third of complexes were completely covered by only one term. The majority of proteins (>50%) could be described by a single functional term in ~75% of *E. coli* and yeast complexes. These results suggest that the MCL-GO complexes were generally functionally coherent, with the majority of proteins in the majority of complexes performing the same general function. Furthermore it suggests that, in both species, complexes can be reasonably well annotated using the most frequent term applied to their constituent proteins.

Each complex was then annotated using its most common FunCat term. Figure 4.13 shows the proportion of complexes in each species that were involved in different processes. *E. coli* had a larger proportion of complexes devoted to metabolism and energy than yeast whereas yeast had a greater proportion of complexes involved in the cell cycle, transcription and cellular transport. These results make sense as prokaryotes are known to focus much of their resources on metabolism, enabling utilisation of alternative energy sources for example. Their transcriptional machinery and cell cycle are also known to be less complicated than that of eukaryotes. Thus the MCL-GO complexes for *E. coli* and yeast appear to reflect the known biology of these species. This suggests that the complexes produced are functionally representative.

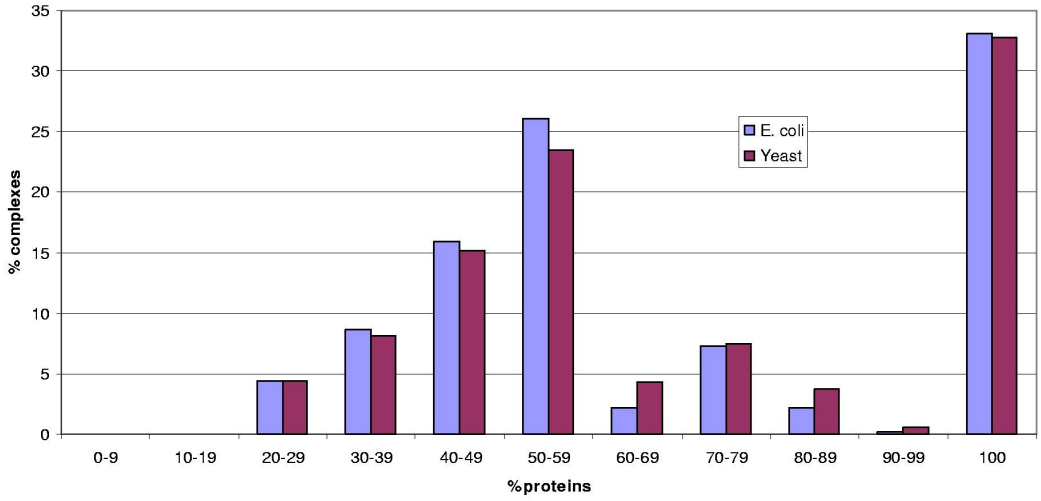



Figure 4.11 Percentage of proteins in complexes annotated with the most common term in each complex.

Complexes were classified using level one FunCat terms.

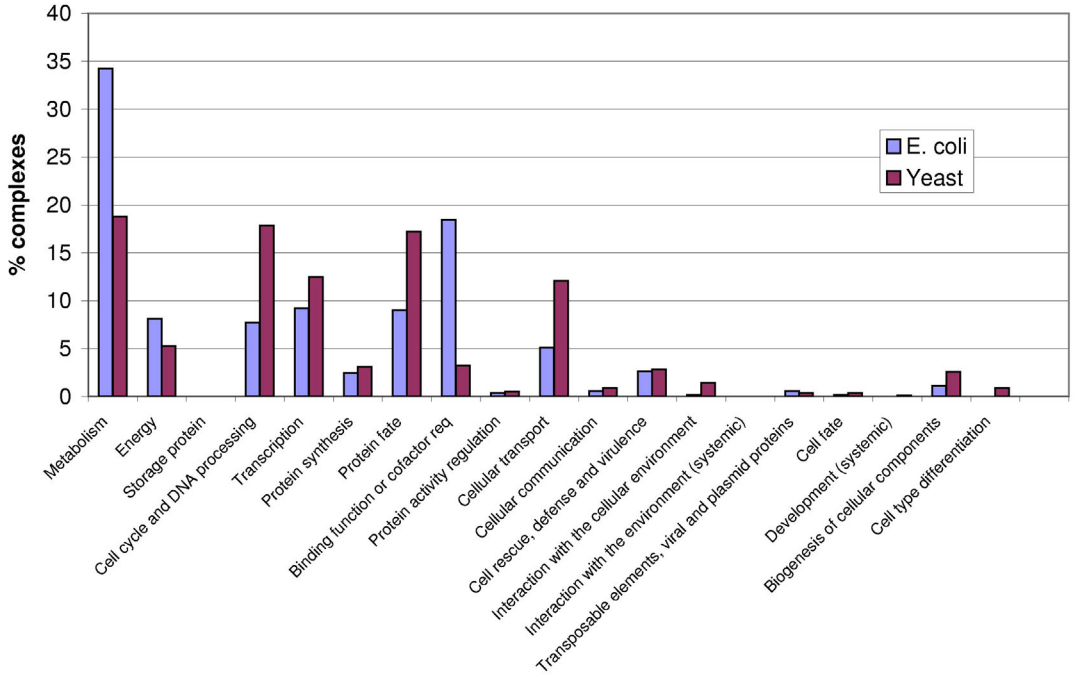



Figure 4.13 Principal functions of complexes in each species.

Complexes were classified using level one FunCat terms. Complexes with less than 2 annotated proteins were excluded.

### 4.3.2. Distribution of Protein Domain Superfamilies amongst Protein Complexes

There has been much debate about the fate of duplicated genes. It has been proposed that newly duplicated gene products which become fixed in a population initially retain common interactions which subsequently diverge (Wagner, 2001). There have been conflicting reports however regarding the extent to which paralogues within species tend to have common interactions and how fast they might lose them during evolution (Wagner, 2001; Baudot *et al.*, 2004). This part of the chapter examines how homologues are distributed in protein complexes and how this might relate to complex evolution. CATH domain superfamilies were used to define homologues as these allow distant evolutionary relationships to be established.

Figure 4.15 shows, for MCL-GO complexes, the number of superfamily members versus the number of different complexes in which these superfamilies are found. There was a strong positive correlation between superfamily size and the number of complexes in which that superfamily is found. For *E. coli* $r^2$ was 0.99 and for yeast 0.97. This suggests that after domains have duplicated they tend to change their interactions and move into new complexes.

Are there superfamilies which do not follow this trend and tend to conserve their complex membership? For each superfamily the frequency with which two proteins containing a member of that superfamily were found together in a complex was determined. This was compared to the number of co-complex pairs that would be expected if the proteins were distributed randomly amongst complexes (see 4.2.6). For most superfamilies, members did not co-occur in complexes more than would be expected by chance. 98% of *E. coli* superfamilies and 95% of yeast superfamilies were randomly distributed. The exceptional, non-randomly distributed superfamilies are discussed in the next section.

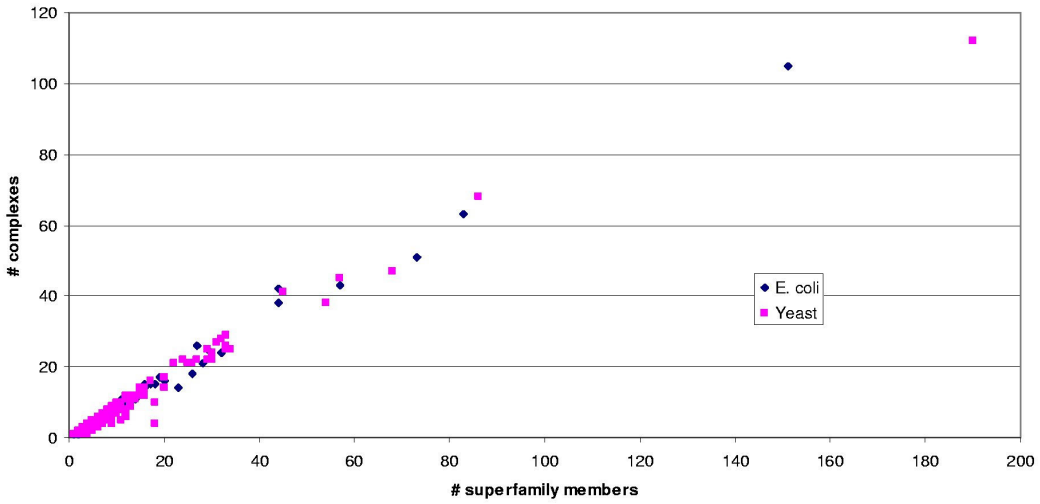



Figure 4.15 Number of CATH superfamily members versus number of complexes containing members of that superfamily for *E. coli* and yeast MCL-GO complexes.

Are different members of a superfamily involved in similar biological processes despite their random distribution amongst complexes? In other words, are they involved in complexes with related function? It was found, using GO terms, that 28% of superfamilies in *E. coli* and 22% in yeast had members which were involved in more similar biological processes than expected by chance ($p<0.01$). While homologous domains tend to become involved in different complexes after duplication, some superfamilies appear to be more conservative about changing their functional role.

When the functional similarity of the proteins with which each superfamily member was directly interacting was examined, there was less conservation. For example, if protein A interacts with proteins B, C and D and protein A homologue A´ interacts with E, F and G, then B, C and D were not functionally similar to E, F and G. Less than 1% of *E. coli* superfamilies had interactors with conserved function. 12% of yeast superfamilies had interactors with conserved function.

For each superfamily, the functional similarity of the complexes in which its members were found was also examined. Again, <1% of *E. coli* superfamilies were found in complexes with similar functions, whereas in yeast 6% were found in similar complexes.

These results suggest that those superfamilies which conserve their function to some extent tend to diversify into distinct aspects of similar processes in yeast. While 28% of superfamilies in yeast have conserved function, the functions of neighbours of around half of these superfamilies are not conserved and only 6% of superfamilies are in complexes with similar functions. In *E. coli*, while more than a quarter of superfamilies have a conserved function, almost no superfamilies have a conserved functional environment. The function of their interactors has changed.

### 4.3.3. Functional Analysis of Non-Randomly Distributed Superfamilies

A small number of superfamilies were found to be non-randomly distributed amongst MCL-GO complexes in the previous analysis; Table 4.3 shows details of these superfamilies. What is the functional significance of multiple homologues in complexes?

In *E. coli* there was only one non-randomly distributed superfamily identified, the NAD(P)-binding Rossmann-like Domain superfamily. This is a very large, universal (present in all three superkingdoms) domain superfamily which provides oxidoreductase activity in a wide variety of biological processes. Those complexes containing multiple members of this superfamily tended to be large, with diverse functional roles. It was therefore unclear as to the role of multiple members of this superfamily in individual complexes.

In yeast there were six non-randomly distributed superfamilies amongst MCL-GO complexes. These fell into three categories. The first was RNA processing, the second was the proteasome and the third was the signal transduction.

The RNA-binding superfamily was found in two complexes relating to the spliceosome. The spliceosome is a complex which removes introns from pre-mRNA and requires functions which include binding a variety of RNAs. Multiple members of the Quinoprotein Amine Dehydrogenase domain superfamily were found in complexes rich in annotation relating to the spliceosome in one case and rRNA processing in the other.

The ribosomal protein superfamily was found in a complex rich in annotation for rRNA processing. Ribosomal RNA processing is known to occur in the nucleolar complex which is involved in the production of ribosomes.

| P-value | Superfamily | Frequency | Function | Species distribution |
|---|---|---|---|---|
| E. coli | | | | |
| 0.0041 | NAD(P)-binding Rossmann-like Domain (3.40.50.720) | 73 | Oxidoreductase activity in a wide variety of processes | Universal |
| Yeast | | | | |
| 0.0001 | RNA binding (2.30.30.100) | 11 | RNA binding/ splicing | Universal |
| 0.0001 | Glutamine Phosphoribosylpyrophosphate, subunit 1, domain 1 (3.60.20.10) | 18 | Ubiquitin-mediated endopeptidase activity | Universal |
| 0.0007 | Quinoprotein amine dehydrogenase (2.130.10.10) | 68 | Wide range of activities including protein synthesis | Universal |
| 0.0016 | Protein tyrosine phosphatase superfamily (3.90.190.10) | 12 | Dephosphorylation in signalling pathways | Eukaryotic |
| 0.0019 | Ribosomal Protein (3.30.1370.10) | 4 | Binding activity in a variety of processes | Universal |
| 0.0021 | Ubiquitin-like superfamily (3.10.20.30) | 5 | TCA cycle | Universal |

Table 4.3 Superfamilies in *E. coli* and yeast MCL-GO complexes which were non-randomly distributed.

CATH codes are shown in brackets. Frequency is the number of proteins containing a member of that superfamily in that complex dataset. Functional descriptions are based on the most common GO terms from proteins containing the superfamily in that particular

organism, not for the specific complexes identified in the text. Superfamilies are considered to belong to a kingdom when they are found in at least 70% of completed genomes from that kingdom. Universal refers to eukaryotes, eubacteria and archaea.

There is a caveat to some of these results however. Associations related to rRNA processing may represent a bias in some of the experimental data used to generate the complexes. Some high-throughput complex identifications in yeast (Gavin *et al.*, 2002; Ho *et al.*, 2002) contain many complexes erroneously enriched in rRNA processing. This is thought to be the result of proteins connected by rRNA, rather than protein interactions (Betel et al., 2004). Results relating to rRNA processing should therefore be considered false positives. Independent evidence supports the relevance of the spliceosome however (Staley and Guthrie, 1998).

The second category is the proteasome. A complex was identified containing several copies of the Glutamine Phosphoribosylpyrophosphate superfamily which is involved in Ubiquitin-mediated endopeptidase activity via the proteasome complex and different members of the superfamily are required for different types of protease activity (Rubin and Finley, 1995).

The third category is signal transduction. Multiple copies of the protein tyrosine phosphatase superfamily were found in a complex involved in signal transduction via a MAP kinase pathway controlling pseudohyphal growth.

Multiple copies of homologous regulatory proteins may represent signalling/regulatory complexes with alternative regulatory subunits e.g. the Myc-Max and Mad-Max basic helix-loop-helix transcription factor complexes noted by Pereira-Leal et al. (2007). In MCL-GO complexes, complex variants with alternative regulatory subunits such as these are expected to be found as single complexes. Each protein can only occur in a single complex and therefore variant complexes which are largely composed of the same set of subunits cannot be resolved. If the alternative subunits of such variant complexes are homologous, they will be identified in this analysis, despite the fact that they would not be present together in a complex *in vivo*.

It appears that those members of superfamilies which clustered together tended to be involved in eukaryote-specific processes. They were almost exclusively universal superfamilies, suggesting that these eukaryotic

advancements have largely developed from duplication and divergence of pre-existing superfamilies.

### 4.3.4. Co-Occurrence of Homologues in Protein Complexes

It was shown in the previous analysis that homologous domains tend to be randomly distributed in protein complexes and that duplicates have therefore tended to diversify rather than remain involved in the same complex. An alternative analysis by Pereira-Leal et al. (2007) has shown that interacting, homologous pairs might be important for complex evolution in yeast. They found that 10-30% of complexes in this species contain homologous protein pairs. In the model of complex evolution they presented, the gene encoding a homodimer duplicates and diverges resulting in a paralogous, heterodimeric protein complex. Rather than examine the distribution of individual domain or protein families, they considered what proportion of complexes contained homologous pairs. Although the analysis described above suggested that superfamilies tend to be randomly distributed in complexes, this is a general trend and there might still be a significant number of cases of homologous pairs in complexes. In particular it was not possible to consider smaller superfamilies (less than 5 members) in the previous analysis due to statistical considerations. Therefore the extent of homologous pairs in complexes was re-examined from the perspective of complexes rather than superfamilies. This analysis builds on the previous work of Pereira-Leal et al. as the MCL-GO datasets are more extensive than those used previously and *E. coli* complexes can be examined as well as those of yeast.

For each predicted yeast complex it was determined whether there was at least one pair of proteins sharing, in the first case, a homologous domain or, in the second case, their entire multi-domain architecture (Figure 4.17). If there is a tendency for homologous proteins to occur together in complexes more than expected by chance, then this gives an upper bound for the

number involved in the model of complex evolution described by Pereira-Leal et al. (2007). Using individual domains allows more distant relationships to be identified which might otherwise be obscured by gain or loss of domains within homologous proteins.

It was found that the proportion of complexes containing homologues was greater than expected by chance in each species ($p<0.01$). In *E. coli* 7.5% of complexes contained homologues at the domain level; this is 1.5 times more complexes than expected by chance. There were 516 pairs of homologues co-occurring in *E. coli* complexes and these were found to interact more often than expected by chance ($p < 0.01$). For yeast the value was much higher: 18.4% of complexes contained homologues, 3.4 times more than expected. 720 pairs of co-complex homologues were identified in yeast and these tended to interact more than expected for random pairs of co-complex proteins ($p < 0.01$).

The result for yeast was within the bounds of 10-30% suggested by Pereira-Leal et al. (2007). *E. coli* had a much smaller proportion of complexes which could have evolved from interacting paralogues. 43 complexes were identified in *E. coli* compared to 157 in yeast.

The trends between species in terms of relative numbers of complexes involved and the difference between expected and observed counts were similar when considering homologues as proteins sharing at least one homologous domain (domain homologues) or as sharing entire multi-domain architectures (protein homologues). Using domain homologues was shown to be more powerful, detecting more cases of co-complex homologous pairs.

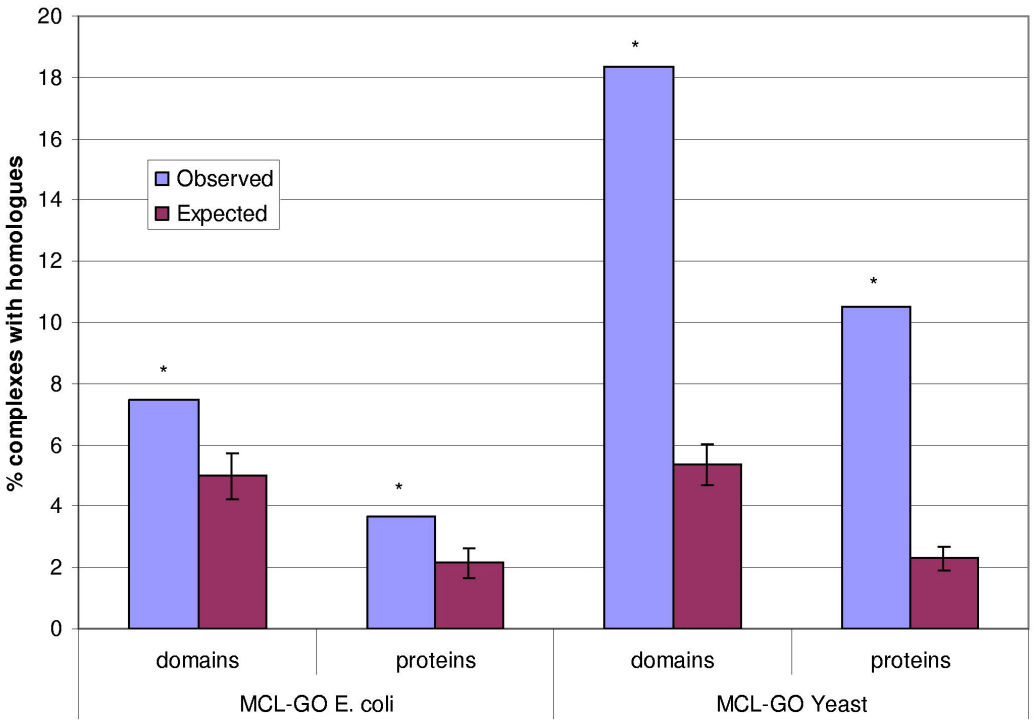



Figure 4.17 Percentage of complexes in each species in which at least one pair of homologues was observed.

Homologues were defined here as either proteins sharing a homologous domain (domains) or sharing a common domain architecture (proteins). All observed values were significantly larger than expected ($p < 0.01$). Asterisks highlight those observed values which were significantly greater than expected at $p = 0.01$.

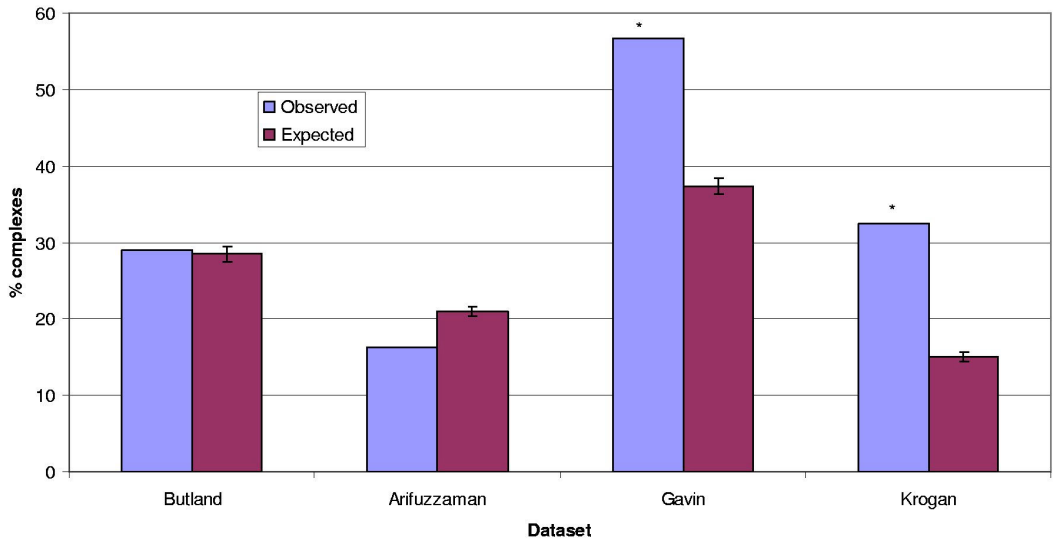



Figure 4.19 Percentage of TAP-MS complexes containing pairs of proteins with homologous domains.

Asterisks highlight those observed values which were significantly greater than expected at p = 0.01.

Exp-TAP complex datasets were also examined to determine whether they supported the above findings (Figure 4.19). Butland and Arifuzzaman Exp-TAP *E. coli* complexes showed no significant increase in the number of complexes containing homologous pairs relative to random complexes. However the Gavin and Krogan Exp-TAP yeast complexes showed significant proportions of complexes containing homologous pairs (57% for Gavin and 32% for Krogan). These results confirm the trends identified in MCL-GO complexes.

### 4.3.5. Identification of Correlated Domain Superfamily Pairs

In the previous analysis it was reaffirmed that homologous domain pairs are not present in the majority of yeast complexes. Furthermore, it was shown that homologous domain pairs are a less common feature of protein complexes in *E. coli* than in yeast. Another feature of protein complexes that has been identified in yeast is pairs of non-homologous domains which co-occur in multiple complexes (Betel et al., 2004). Might these represent an alternative route of complex evolution to that of homologous pairs?

Those pairs of superfamilies whose members co-occur in the same complex (in separate protein chains) and which are found in multiple complexes more often than expected by chance were determined. 189 pairs of correlated superfamilies were identified in *E. coli* MCL-GO complexes, involving 156 superfamilies. These pairs occurred in 68 separate complexes (~12%). This was a greater proportion of complexes than that containing paralogous pairs (~8%). In yeast MCL-GO complexes, 183 pairs were identified, involving 186 superfamilies and 83 complexes (~10%). Full details of the superfamily pairs identified are presented in Appendix D. Using IntAct and MINT PPI datasets it was determined whether these superfamily pairs tended to interact more often than expected by chance. In *E. coli* and yeast there was a significant tendency for interaction ($p < 0.001$). In both species the pairs were also significantly more functionally similar (using GOSS scores as described in 4.2.9) than expected by chance ($p < 0.001$). This

suggests that the correlated domain pairs have a tendency to interact and form functionally coherent parts of complexes in both species.

### 4.3.6. Do Co-Complex Homologues and Correlated Domain Pairs Correspond to Complex Cores?

Pereira-Leal et al. (2007) showed that homologous pairs represent cores of some yeast complexes. The analysis they used determined whether an arbitrary set of proteins tend to be older than other proteins. Specifically, the species distribution of the orthologues of proteins containing correlated domains was determined to ascertain whether they tended to emerge earlier in evolution than other proteins. Older proteins are more likely to represent evolutionary conserved complex cores, whereas more recently evolved proteins are likely to represent later modifications to complexes (van Dam and Snel, 2008; Pereira-Leal et al., 2007). The age of interacting, homologous domain pairs was examined to determine whether those that occur in *E. coli* represent complex cores, as they are thought to in yeast. Additionally, correlated domain pairs in both species were examined to determine whether they too represent cores.

Although there was a tendency for orthologues of interacting homologous pairs from *E. coli* to be present in more distantly related organisms than other proteins (Table 4.4) this trend was not found to be significant ($p = 0.09$). The same was true of proteins containing correlated domains ($p = 0.28$). This is further evidence that *E. coli* complexes have not evolved from interacting homologues, at least not to the extent seen in yeast. Furthermore it appears that correlated domain pairs tend not to be cores of *E. coli* complexes.

| | All proteins | Co-complex homologous domain pairs | Correlated domain pairs |
|---|---|---|---|
| E. coli | | | |
| *E. coli* K12 specific | 19.0% | 9.0% | 11.6% |
| Proteobacteria | 21.0% | 10.4% | 12.5% |
| Proteobacteria Firmicutes | 7.8% | 9.0% | 5.4% |
| Bacteria | 1.4% | 3.0% | 1.8% |
| Eukaryota+Bacteria | 25.1% | 29.9% | 37.5% |
| Bacteria+Archaea | 7.3% | 10.4% | 8.0% |
| Universal | 18.4% | 28.4% | 23.2% |
| Yeast | | | |
| S. cerevisiae-specific | 44.8% | 13.1% | 12.1% |
| Fungi | 11.1% | 9.3% | 12.1% |
| Fungi + Metazoa | 7.4% | 10.4% | 7.9% |
| Eukaryotes | 10.3% | 23.5% | 14.3% |
| Eukaryotes + Archaea | 4.2% | 9.7% | 10.0% |
| Eukaryotes + Bacteria | 13.2% | 18.3% | 26.4% |
| Universal | 9.0% | 15.7% | 17.1% |

Table 4.4 Relative age (emergence of orthologues) of all proteins, co-complex homologues and proteins which contain correlated domains for *E. coli* and yeast MCL-GO complexes.

Co-complex, homologous proteins in yeast were significantly older than proteins in general ($p < 0.01$). This reaffirmed the result of Pereira-Leal et al. (2007). It was also observed that, in yeast, proteins containing correlated domains were significantly older than proteins in general ($p < 0.01$). Most correlated proteins were found in all types of eukaryotes, whereas most yeast proteins were no older than the split between metazoa and fungi. This suggests that both co-complex homologues and correlated pairs are important as evolutionary cores of yeast protein complexes.

Table 4.5 shows that the *E. coli* Exp-TAP datasets supported the trends identified in the MCL-GO dataset. Neither co-complex homologues nor correlated domain pairs in the Arifuzzaman and Butland Exp-TAP *E. coli* datasets were significantly older than other proteins. The picture was less clear in the yeast Exp-TAP datasets. Although the Krogan dataset supported the finding that correlated domains are older than other proteins, the test for homologous pairs was not quite significant. In the Gavin Exp-TAP yeast dataset neither type was significant.

| Exp-TAP Dataset | Homologous Pairs (p-value) | Correlated Pairs (p-value) |
|---|---|---|
| *E. coli* (MCL-GO) | 0.095 | 0.281 |
| *E. coli* (Arifuzzaman) | 0.881 | 0.913 |
| *E. coli* (Butland) | 0.818 | 0.670 |
| Yeast (MCL-GO) | 9.55E-05* | 6.95E-05* |
| Yeast (Gavin) | 0.388 | 0.282 |
| Yeast (Krogan) | 0.053 | 0.006* |

Table 4.5 P-values indicating whether or not particular types of proteins are older than other proteins. Asterisks identify statistically significant results.

# 4.4. Discussion

In this chapter an analysis of the differences in evolution between the protein complexes of *E. coli* and yeast was presented. In order to achieve this, protein complex datasets representing high coverage of proteins in these organisms were generated and shown to accurately reproduce known complexes. CATH domain superfamilies were used to identify how duplicates are reused in complexes. This allowed distant relationships to be identified relative to other sequence comparison approaches.

It was found that homologous domains tended to be randomly distributed amongst complexes and therefore that duplicates tend to occupy distinct functional niches. Those exceptional domain superfamilies whose members were found together more than expected by chance tended to be involved in signalling/regulation or a limited number of eukaryote-specific complexes requiring colocation of similar functions. It has been shown that homologues are rarely found together in small molecule metabolic pathways of *E. coli* (Teichmann et al., 2001) and it was shown here that this appears to be the case for protein complexes as well.

Pereira-Leal et al. (2007) proposed that a proportion of yeast complexes have evolved from cores of homologous subunits. These subunits are proposed to originate from homodimers, encoded by single genes which then duplicated, resulting in dimers of paralogues. The results presented here suggest that this model of complex evolution is limited in prokaryotes. It was found that in *E. coli* there were a much smaller number of complexes which could have evolved in this way than in yeast. It is known that there is less gene copy redundancy in prokaryotes and that their gene families are smaller (Ranea et al., 2007), resulting from streamlined genomes (Ranea, 2006). Here it was shown that this may extend to fundamental differences in how complexes have evolved in *E. coli*. Furthermore, a functional analysis showed that those homologues which cluster in complexes tend to relate to eukaryotic functions. This process may therefore have been exploited

principally in developing the more complex processing and regulation required in the eukaryotic cell.

Pairs of correlated domains were identified which occur together in multiple complexes, as was done previously by Betel et al. (2004). It was shown that the proteins containing these domains tended to interact and be more functionally similar than other pairs of co-complex proteins. In yeast these protein pairs tended to be older than other pairs of proteins and might therefore represent complex cores; there was little evidence for this in *E. coli* however. Complexes are known to have duplicated in yeast and these correlated pairs are likely to include parts of duplicated complexes. The results imply that the cores of *E. coli* complexes tend not to be duplicated. This may be because one route through which complex duplication might occur is whole genome duplication, which is thought to have occurred in yeast (Wolfe and Shields, 1997), but is not known in *E. coli* (Ochman et al., 2005). It is possible that correlated domain pairs tend to be more recently evolved parts of complexes in *E. coli*.

In future studies it would be interesting to examine further the role of correlated pairs in *E. coli*, as it is unclear what role they play in complex evolution. Furthermore an analysis of higher eukaryotes would be an appropriate extension, to determine whether the processes of complex evolution discussed are more common than in yeast. *Drosophila melanogaster* was considered for analysis; however there was insufficient data to produce reliable complexes.

# Chapter 5 Discussion and Conclusions

## 5.1. Overview

The aim of this thesis was to explore the evolution and function of proteins using domain superfamilies. In Chapter 2 CATH structural domain superfamilies were used to benchmark methods for identifying homologous relationships between sequences. In Chapter 3 Pfam domain families were used to identify triplets of proteins related through gene fusion allowing the prediction of functional associations between proteins. In Chapter 4 CATH domain superfamilies were used to study differences in the evolution of protein complexes between prokaryotes and eukaryotes. This chapter introduces a wider perspective on the findings contained in the thesis.

## 5.2. Chapter 2

In Chapter 2 a thorough benchmark of current methods for remote homologue detection was developed. These methods can be used to identify domain family members in genomes and such data is used in Chapters 3 and 4 to examine the function and evolution of proteins. It was necessary to perform this benchmark as, at the time, there were no benchmarks encompassing the full range of methods available. The benchmarking resulted in the adoption of new methods into the CATH-Gene3D pipeline, resulting in improved datasets for the subsequent chapters.

The benchmarking assessed the ability of publicly available methods to detect homologues at different ranges, e.g. remote homologues in the twilight zone (<30% sequence identity) and very remote homologues in the midnight zone (<10% sequence identity). Furthermore the ability of these

methods to distinguish all homologues from all non-homologues was compared with their abilities in detecting their closest neighbour, which is a more relevant assessment of accuracy for genome annotation.

It has been recognized for some time that methods for remote homologue detection involving Hidden Markov Models (HMMs) are able to detect evolutionarily plausible relationships between proteins which are not classified as homologous based on their structures in CATH or SCOP (Gough et al., 2001). This appears to run counter to the commonly held assumption that structure is more conserved than sequence. This is not necessarily the case however; it seems in fact to result from certain assumptions about how structures evolve.

The β-propellers, for example, have long been classified into several different architectures in CATH based on their number of blades (units which fit together much like the blades of a propeller). It has been shown that the blades can duplicate within a protein and be inherited from other propeller proteins. This form of evolution alters the core of the structure, whereas most globular folds are assumed to maintain a conserved core and experience peripheral embellishments.

The FAD/NAD(P) binding domain fold (3.50.50) and the Rossmann fold (3.40.50) are also found in different architectures but were found to have many putative homologues between them. They both have a central β-sheet flanked on one side by α-helices and the other by either β (3.50.50) or α (3.40.50) structures. These have previously been proposed as potentially related folds (Harrison et al., 2002).

The most powerful methods of remote homologue detection which compare profiles of related sequences against each other (profile-profile) detect even more relationships between different folds and architectures than previous methods. Traditional benchmarking approaches for remote homologue detection rely on structural classifications to determine relationships between sequences and therefore novel benchmarking approaches are required to accurately measure performance of the most

powerful methods. To achieve this, structural comparison was combined with sequence comparison to exclude examples from the benchmark dataset which were scored highly by both sequence and structural methods. This was shown to accurately reproduce a manually curated set of putative homologues. The approach can be applied to any existing structure-based benchmark dataset such as those based on SCOP or FSSP. The approach presented was an extension of that developed by Soding for benchmarking his profile-profile method HHSearch (Soding, 2005). The improvements presented in Chapter 2 were the use of a leading structural comparison algorithm and optimisation of the approach's accuracy using manually classified examples.

After the publication of the work in Chapter 2 (Reid et al., 2007), a paper was published by Qi et al. (Qi et al., 2007) describing an alternative approach to this problem. They used a Support Vector Machine (SVM) trained on both sequence and structural similarity scores between SCOP domains from different classes and those from the same superfamily to classify previously ambiguous relationships between domains. Rather than remove ambiguous relationships from the dataset as was the aim in this work, their aim was include as many ambiguous relationships as possible by explicitly classifying them as homologous or non-homologous. The benefit of this approach relative to that presented in this thesis is that the benchmarking dataset is enlarged rather than reduced. Chapter 2 describes a manually validated approach which although resulting in smaller a dataset (the dataset is still very large) should produce fewer incorrectly classified examples.

Employing this novel benchmarking strategy it was shown that different methods were optimal for different tasks, the method which was best for identifying families of homologues was not the best for annotating genomes. Furthermore it was shown that profile-profile methods were able to detect up to 10 times more very remote homologues (<10% sequence identity) than BLAST at low error rates. The profile-profile method PRC

performed best in distinguishing homologues from non-homologues whereas a different profile-profile method, COMPASS, performed best at annotating genomes (i.e. finding just the closest homologue).

The ability to identify relationships between more distant homologues allows more ancient evolutionary events to be examined. For instance details of the biology of the Last Universal Common Ancestor (Ranea et al., 2006) have been inferred using CATH domain superfamilies to identify ancient homologous relationships between proteins. Although very distant domain relatives tend to have divergent functions, function is more conserved within families than between unrelated proteins. The organisation of structural protein domains into families and their enrichment with sequence relatives also aids the study of functional evolution (Todd et al., 2001).

The realisation from this work that profile-profile methods, in particular PRC, are able to detect very remote homologues classified in different CATH folds has led to its incorporation into the CATH update pipeline. In this context it is used to identify incorrectly classified domains and to better resolve the superfamilies.

In the final part of Chapter 2 it was shown that combining methods of remote homologue detection could improve coverage at very low error rates, particularly in annotating genomes. This approach could therefore be applied where highly accurate genome annotation is needed in projects such as that organised by the ENCODE Project Consortium (2004) to analyse 1% of the human genome in great detail.

## 5.3. Chapter 3

In Chapter 3 an improved method was introduced for predicting functional associations between proteins by identifying instances of domain fusion. Domain fusion occurs when two genes, whose products interact or are otherwise involved in a common process, fuse so that their products are expressed together in a single protein chain. This method was developed

with the aim of improving prediction of functional associations in the human genome. There had been no comparison of different methods in the literature and little examination of how well methods performed in the human genome (with the exception of (Truong and Ikura, 2003)). The human genome includes larger protein domain families than bacteria and lower eukaryotes which make it more difficult to accurately detect functionally related proteins using domain fusion.

A novel scoring approach was introduced which takes into account the frequency of domain homologues in query and target genomes when identifying and scoring putative relationships between proteins. The resulting approach, named CODA, for Co-Occurrence of Domains Analysis, was shown to give improved performance over several comparable approaches.

Perhaps the most similar method to CODA is that of Truong & Ikura (2003). Their method represented the only available instance of domain fusion. However, this did not include a scoring method and to cope with large, promiscuous domain families, they simply excluded all fusions involving such families. CODA was able to more subtly downweight the effect of such families. Although Truong's method was able to find 189 functionally related pairs with high accuracy, CODA was able to find many more functionally linked pairs (~1000) with moderate accuracy. Additionally CODA was shown to accurately find more hits than gene fusion methods in both yeast and human genomes.

Each alternative method examined tended to find functional relationships between distinct sets of proteins compared to CODA. This suggests that different implementations of the fusion method find quite different sets of functional relationships, perhaps due to alternative sequence representations (genes vs. domains) and alternative methods of dealing with large domain families/promiscuous domains (exclusion vs. scoring). This implies that these alternative implementations could be combined to improve coverage.

CODA has been integrated with other methods of function prediction to produce a suite of tools called BioMiner (*in preparation*) which is being used in several collaborations to identify networks of functionally related proteins. One project in particular, part of ENFIN (Kahlem and Birney, 2007), has shown that BioMiner, in combination with other approaches, successfully improved the detection of proteins involved in the human mitotic spindle (*in preparation*). Such approaches allow experimental characterisation of proteins to be used more efficiently, reducing the amount of time and money applied to interpreting genomes and characterising biological processes.

## 5.4. Chapter 4

In chapter 4 the evolutionary mechanisms which generate protein complexes in *E. coli* and yeast were shown to differ. This analysis was facilitated by the generation of accurate, high coverage datasets of complexes for these species. The datasets were generated by clustering large, combined protein-protein interaction networks, an approach which had been shown previously to generate accurate complexes for yeast. In Chapter 4, the same methodology was applied to *E. coli* and although the accuracy and coverage of the predicted *E. coli* complexes was lower than for yeast, it was significantly better than for randomly generated complexes.

Using these datasets it was shown that members of the vast majority of protein domain superfamilies are randomly spread amongst complexes. Those which were not were essentially limited to yeast and were found to be involved in eukaryote-specific complexes such as the spliceosome and proteasome, as well as one example involved in signal transduction. It is known that domain families tend to be smaller in prokaryotes than eukaryotes with lower gene copy redundancy (Ranea et al., 2007) and these results suggest further that protein complexes tend not to contain homologous pairs in *E. coli*. Conservation of complex membership between

homologues is not the rule in either species and thus duplicate genes are generally not reused in the same complexes.

Pereira-Leal et al (2007) examined the occurrence of homologues in complexes from the perspective of complexes rather than superfamilies. They proposed that between 10 and 30% of protein complexes in yeast have evolved from an evolutionary core of interacting homologues. The Pereira-Leal model was re-examined, with the inclusion of *E. coli*. The model was upheld and ~18% of yeast complexes were found to contain pairs of homologous proteins, well within the bounds previously identified. These pairs were subsequently shown to have properties expected of complex cores as had been done by the authors of the original work. In *E. coli* however, there were fewer complexes (~8%) containing homologous pairs and those which were identified were not found to be significantly associated with the properties expected of complex cores.

An alternative model of complex evolution was examined, namely the role of correlated domains: pairs of non-homologous domains which co-occur in multiple complexes. These were found in ~12% of E. coli complexes and ~8% of yeast complexes and shown to represent interacting pairs with highly similar functions. These were found to represent complex cores in yeast, but not in *E. coli*. Complexes are known to have duplicated in yeast and at least some of these correlated pairs are likely to relate to duplicated complexes. The results imply that the cores of *E. coli* complexes tend not to be duplicated. This may be because one route through which complex duplication can occur is whole genome duplication, which is thought to have occurred in yeast (Wolfe and Shields, 1997), but is not known in *E. coli* (Snel et al., 2002). It is possible that correlated pairs tend to be more recently evolved parts of complexes in this organism.

The field of protein complex analysis is relatively young, since much of the appropriate data has only recently been collected. For species other than yeast there is still a noticeable paucity of data, however on the positive side there is much interest in protein complexes and protein-protein interactions

in general. This is stimulating further data gathering and calls for projects to systematically identify complexes (Bravo and Aloy, 2006). Further barriers include the current lack of understanding regarding the accuracy of current datasets (Jensen and Bork, 2008) and uncertainty over the extent to which interactions can be inherited to other proteins (Mika and Rost, 2006). These are active areas of research.

Part of the reason for the current interest in protein-protein interactions is that in the post-genomic era it has become clear that the parts list of an organism, its genes and proteins, is not sufficient to explain its complexity (Hahn and Wray, 2002). Humans and nematodes, for example, have similar numbers of genes, but humans appear much more complex, having many more different cell types (Vogel and Chothia, 2006). Current thinking suggests that the origin of this complexity can be understood through the interactions between proteins and the ways in which they are regulated. This fundamental problem in molecular biology underscores the importance of examining differences in protein complexes between prokaryotes and eukaryotes.

## 5.5. Future Work

There are many interesting possibilities for examining the evolution of protein interactions and complexes. It would be useful, for instance, to determine the role of changes in multi-domain architecture. Given a particular CATH superfamily, how do the partner domains of its members affect the interactions it is involved in? Some domain superfamilies are involved directly in inter-chain protein-protein interactions and therefore are likely to have an effect on the interactions of the proteins in which they are found. Other superfamilies are not directly involved in such interactions but will inherit the interactions of the domains to which they are covalently linked. It would be most interesting to investigate how this affects the distribution of the different superfamilies and what it might mean for the

prediction of protein-protein interactions based on domain architecture. Additionally it may be possible to discern examples of how domain combinations directly alter the interactions of particular superfamily members. Several resources of domain-domain interaction data have been established (Stein *et al.*, 2005; Jefferson *et al.*, 2007; Finn *et al.*, 2005), however, there is currently a relatively small amount of experimental data (Schuster-Bockler and Bateman, 2007). Furthermore it has proven difficult to accurately predict domain-domain interactions based on domain architecture (Nye et al., 2005).

There is little experimental protein-protein interaction data for the vast majority of species and it is not trivial to inherit protein-protein interactions between species. It has been possible however, to study the evolution of complexes across the eukaryotes. This was achieved for a single complex by identifying whether orthologues of a well characterised yeast complex were present in other eukaryotes (Gabaldon et al., 2005). If one assumes that orthologues perform the same function in different species, then it may be possible to map a core set of complexes across eukaryotes or prokaryotes, for example. Given a set of complexes in one species, e.g. yeast, orthologues could be identified in all other eukaryotes with complete genomes. Those complexes whose constituent proteins have orthologues in all these genomes could be considered as core eukaryotic complexes. This approach is likely to underestimate the true number of core complexes due to an incomplete list of complexes in yeast and missed orthologues. On the other hand, some proteins identified as orthologues may have changed their function and may no longer be involved in a particular complex. Despite these caveats, it is possible that important aspects of protein complex biology might be identified.

# Bibliography

(2004) The ENCODE (ENCyclopedia Of DNA Elements) Project. *Science* **306,** 636-640.

(2008) The Universal Protein Resource (UniProt) 2009. *Nucleic Acids Res.*

Alberts, B. (1998) The cell as a collection of protein machines: preparing the next generation of molecular biologists. *Cell* **92,** 291-294.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990) Basic local alignment search tool. *J Mol Biol* **215,** 403-410.

Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* **25,** 3389-3402.

Andreeva, A., Howorth, D., Brenner, S. E., Hubbard, T. J., Chothia, C., and Murzin, A. G. (2004) SCOP database in 2004: refinements integrate structure and sequence family data. *Nucleic Acids Res* **32,** D226-D229.

Andreeva, A., Howorth, D., Chandonia, J. M., Brenner, S. E., Hubbard, T. J., Chothia, C., and Murzin, A. G. (2008) Data growth and its impact on the SCOP database: new developments. *Nucleic Acids Res* **36,** D419-D425.

Apic, G., Gough, J., and Teichmann, S. A. (2001b) Domain combinations in archaeal, eubacterial and eukaryotic proteomes. *J Mol Biol* **310,** 311-325.

Apic, G., Gough, J., and Teichmann, S. A. (2001a) An insight into domain combinations. *Bioinformatics* **17 Suppl 1,** S83-S89.

Arifuzzaman, M., Maeda, M., Itoh, A., Nishikata, K., Takita, C., Saito, R., Ara, T., Nakahigashi, K., Huang, H. C., Hirai, A., Tsuzuki, K., Nakamura, S., Altaf-Ul-Amin, M., Oshima, T., Baba, T., Yamamoto, N., Kawamura, T., Ioka-Nakamichi, T., Kitagawa, M., Tomita, M., Kanaya, S., Wada, C., and Mori, H. (2006) Large-scale identification of protein-protein interaction of Escherichia coli K-12. *Genome Res* **16,** 686-691.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet* **25,** 25-29.

Attwood, T. K., Bradley, P., Flower, D. R., Gaulton, A., Maudling, N., Mitchell, A. L., Moulton, G., Nordle, A., Paine, K., Taylor, P., Uddin, A., and Zygouri, C. (2003) PRINTS and its automatic supplement, prePRINTS. *Nucleic Acids Res* **31,** 400-402.

Bader, G. D., Betel, D., and Hogue, C. W. (2003) BIND: the Biomolecular Interaction Network Database. *Nucleic Acids Res* **31,** 248-250.

Bader, G. D. and Hogue, C. W. (2003) An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics* **4,** 2.

Baldauf, S. L. (2003) The deep roots of eukaryotes. *Science* **300,** 1703-1706.

Bandyopadhyay, S., Kelley, R., Krogan, N. J., and Ideker, T. (2008) Functional maps of protein complexes from quantitative genetic interaction data. *PLoS Comput Biol* **4,** e1000065.

Barnes, D. E., Tomkinson, A. E., Lehmann, A. R., Webster, A. D., and Lindahl, T. (1992) Mutations in the DNA ligase I gene of an individual with immunodeficiencies and cellular hypersensitivity to DNA-damaging agents. *Cell* **69,** 495-503.

Basu, M. K., Carmel, L., Rogozin, I. B., and Koonin, E. V. (2008) Evolution of protein domain promiscuity in eukaryotes. *Genome Res* **18,** 449-461.

Bateman, A. and Finn, R. D. (2007) SCOOP: A simple method for identification of novel protein superfamily relationships. *Bioinformatics*.

Baudot, A., Jacq, B., and Brun, C. (2004) A scale of functional divergence for yeast duplicated genes revealed from analysis of the protein-protein interaction network. *Genome Biol* **5,** R76.

Bellivier, F., Chaste, P., and Malafosse, A. (2004) Association between the TPH gene A218C polymorphism and suicidal behavior: a meta-analysis. *Am J Med Genet B Neuropsychiatr Genet* **124B,** 87-91.

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J R Stat Soc ser B* **57,** 289-300.

Berman, H., Henrick, K., Nakamura, H., and Markley, J. L. (2007) The worldwide Protein Data Bank (wwPDB): ensuring a single, uniform archive of PDB data. *Nucleic Acids Res* **35,** D301-D303.

Betel, D., Isserlin, R., and Hogue, C. W. (2004) Analysis of domain correlations in yeast protein complexes. *Bioinformatics* **20 Suppl 1,** i55-i62.

Birney, E. and Durbin, R. (2000) Using GeneWise in the Drosophila annotation experiment. *Genome Res* **10,** 547-548.

Bowers, P. M., Pellegrini, M., Thompson, M. J., Fierro, J., Yeates, T. O., and Eisenberg, D. (2004) Prolinks: a database of protein functional linkages derived from coevolution. *Genome Biol* **5,** R35.

Bravo, J. and Aloy, P. (2006) Target selection for complex structural genomics. *Curr Opin Struct Biol* **16,** 385-392.

Brenner, S. E., Chothia, C., and Hubbard, T. J. (1998) Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc Natl Acad Sci U S A* **95,** 6073-6078.

Brohee, S. and van Helden, J. (2006) Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics* **7,** 488.

Brown, K. R. and Jurisica, I. (2005) Online predicted human interaction database. *Bioinformatics* **21,** 2076-2082.

Bru, C., Courcelle, E., Carrere, S., Beausse, Y., Dalmar, S., and Kahn, D. (2005) The ProDom database of protein domain families: more emphasis on 3D. *Nucleic Acids Res* **33,** D212-D215.

Buchan, D. W., Shepherd, A. J., Lee, D., Pearl, F. M., Rison, S. C., Thornton, J. M., and Orengo, C. A. (2002) Gene3D: structural assignment for whole genes and genomes using the CATH domain structure database. *Genome Res* **12,** 503-514.

Bult, C. J., Eppig, J. T., Kadin, J. A., Richardson, J. E., and Blake, J. A. (2008) The Mouse Genome Database (MGD): mouse biology and model systems. *Nucleic Acids Res* **36,** D724-D728.

Butland, G., Peregrin-Alvarez, J. M., Li, J., Yang, W., Yang, X., Canadien, V., Starostine, A., Richards, D., Beattie, B., Krogan, N., Davey, M., Parkinson, J., Greenblatt, J., and Emili, A. (2005) Interaction network containing conserved and essential protein complexes in Escherichia coli. *Nature* **433,** 531-537.

Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, D., Harte, N., Lopez, R., and Apweiler, R. (2004) The Gene Ontology Annotation (GOA) Database: sharing knowledge in Uniprot with Gene Ontology. *Nucleic Acids Res* **32,** D262-D266.

Casbon, J. A. and Saqi, M. A. (2006) On single and multiple models of protein families for the detection of remote sequence relationships. *BMC Bioinformatics* **7,** 48.

Chatr-aryamontri, A., Ceol, A., Palazzi, L. M., Nardelli, G., Schneider, M. V., Castagnoli, L., and Cesareni, G. (2007) MINT: the Molecular INTeraction database. *Nucleic Acids Res* **35,** D572-D574.

Cherkasov, A. and Jones, S. J. (2004) Structural characterization of genomes by large scale sequence-structure threading. *BMC Bioinformatics* **5,** 37.

Chothia, C. and Lesk, A. M. (1986) The relation between the divergence of sequence and structure in proteins. *EMBO J* **5,** 823-826.

CRICK, F. H., BARNETT, L., BRENNER, S., and WATTS-TOBIN, R. J. (2009) General nature of the genetic code for proteins. *Nature* **192,** 1232.

Dandekar, T., Snel, B., Huynen, M., and Bork, P. (1998) Conservation of gene order: a fingerprint of proteins that physically interact. *Trends Biochem Sci* **23,** 324-328.

de Silva, E., Thorne, T., Ingram, P., Agrafioti, I., Swire, J., Wiuf, C., and Stumpf, M. P. (2006) The effects of incomplete protein interaction data on structural and evolutionary inferences. *BMC Biol* **4,** 39.

Devos, D. and Valencia, A. (2000) Practical limits of function prediction. *Proteins* **41,** 98-107.

Dezso, Z., Oltvai, Z. N., and Barabasi, A. L. (2003) Bioinformatics analysis of experimentally determined protein complexes in the yeast Saccharomyces cerevisiae. *Genome Res* **13,** 2450-2454.

Diaz-Mejia, J. J., Perez-Rueda, E., and Segovia, L. (2007) A network perspective on the evolution of metabolism by gene duplication. *Genome Biol* **8,** R26.

Durbin, R., Eddy, S. R., Krogh, A., and Mitchison, G. (1998) *Biological Sequence Analysis*. Cambridge University Press: Cambridge.

Eddy, S. R. (1998) Profile hidden Markov models. *Bioinformatics* **14,** 755-763.

Eddy, S. R. (1996) Hidden Markov models. *Curr Opin Struct Biol* **6,** 361-365.

Enright, A. J., Iliopoulos, I., Kyrpides, N. C., and Ouzounis, C. A. (1999) Protein interaction maps for complete genomes based on gene fusion events. *Nature* **402,** 86-90.

Enright, A. J., Kunin, V., and Ouzounis, C. A. (2003) Protein families and TRIBES in genome sequence space. *Nucleic Acids Res* **31,** 4632-4638.

Enright, A. J. and Ouzounis, C. A. (2001) Functional associations of proteins in entire genomes by means of exhaustive detection of gene fusions. *Genome Biol* **2,** RESEARCH0034.

Enright, A. J., Van Dongen, S., and Ouzounis, C. A. (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res* **30,** 1575-1584.

Euler, L. (1741) Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae* **8,** 128-140.

Fields, S. and Song, O. (1989) A novel genetic system to detect protein-protein interactions. *Nature* **340,** 245-246.

Finn, R. D., Marshall, M., and Bateman, A. (2005) iPfam: visualization of protein-protein interactions in PDB at domain and amino acid resolutions. *Bioinformatics* **21,** 410-412.

Finn, R. D., Mistry, J., Schuster-Bockler, B., Griffiths-Jones, S., Hollich, V., Lassmann, T., Moxon, S., Marshall, M., Khanna, A., Durbin, R., Eddy, S. R., Sonnhammer, E. L., and Bateman, A. (2006) Pfam: clans, web tools and services. *Nucleic Acids Res* **34,** D247-D251.

Finn, R. D., Tate, J., Mistry, J., Coggill, P. C., Sammut, S. J., Hotz, H. R., Ceric, G., Forslund, K., Eddy, S. R., Sonnhammer, E. L., and Bateman, A. (2008) The Pfam protein families database. *Nucleic Acids Res* **36,** D281-D288.

Fitch, W. M. (1970) Distinguishing homologous from analogous proteins. *Syst Zool* **19,** 99-113.

Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., Kirkness, E. F., Kerlavage, A. R., Bult, C. J., Tomb, J. F., Dougherty, B. A., Merrick, J. M., and . (1995) Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science* **269,** 496-512.

Fleming, K., Muller, A., MacCallum, R. M., and Sternberg, M. J. (2004) 3D-GENOMICS: a database to compare structural and functional annotations of proteins between sequenced genomes. *Nucleic Acids Res* **32,** D245-D250.

Friedrick, T. (2001) Complex I: a chimaera of a redox and conformation-driven proton pump? *Journal of Bioenergetics and Biomembranes* **33,** 169-177.

Gabaldon, T., Rainey, D., and Huynen, M. A. (2005) Tracing the evolution of a large protein complex in the eukaryotes, NADH:ubiquinone oxidoreductase (Complex I). *J Mol Biol* **348,** 857-870.

Gavin, A. C., Aloy, P., Grandi, P., Krause, R., Boesche, M., Marzioch, M., Rau, C., Jensen, L. J., Bastuck, S., Dumpelfeld, B., Edelmann, A., Heurtier, M. A., Hoffman, V., Hoefert, C., Klein, K., Hudak, M., Michon, A. M., Schelder, M., Schirle, M., Remor, M., Rudi, T., Hooper, S., Bauer, A., Bouwmeester, T., Casari, G., Drewes, G., Neubauer, G., Rick, J. M., Kuster, B., Bork, P., Russell, R. B., and Superti-Furga, G. (2006) Proteome survey reveals modularity of the yeast cell machinery. *Nature* **440,** 631-636.

Gavin, A. C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J. M., Michon, A. M., Cruciat, C. M., Remor, M., Hofert, C., Schelder, M., Brajenovic, M., Ruffner, H., Merino, A., Klein, K., Hudak, M., Dickson, D., Rudi, T., Gnau, V., Bauch, A., Bastuck, S., Huhse, B., Leutwein, C., Heurtier, M. A., Copley, R. R., Edelmann, A., Querfurth, E., Rybin, V., Drewes, G., Raida, M., Bouwmeester, T., Bork, P., Seraphin, B., Kuster, B., Neubauer, G., and Superti-Furga, G. (2002) Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* **415,** 141-147.

Gene Ontology Consortium (2006) The Gene Ontology (GO) project in 2006. *Nucleic Acids Res* **34,** D322-D326.

Gotoh, O. (1982) An improved algorithm for matching biological sequences. *J Mol Biol* **162,** 705-708.

Gough, J., Karplus, K., Hughey, R., and Chothia, C. (2001) Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure. *J Mol Biol* **313,** 903-919.

Greene, L. H., Lewis, T. E., Addou, S., Cuff, A., Dallman, T., Dibley, M., Redfern, O., Pearl, F., Nambudiry, R., Reid, A., Sillitoe, I., Yeats, C., Thornton, J. M., and Orengo, C. A. (2007) The CATH domain structure database: new protocols and classification levels give a more comprehensive resource for exploring evolution. *Nucleic Acids Res* **35,** D291-D297.

Grigoriev, A. (2001) A relationship between gene expression and protein interactions on the proteome scale: analysis of the bacteriophage T7 and the yeast Saccharomyces cerevisiae. *Nucleic Acids Res* **29,** 3513-3519.

Guimera, R. and Nunes Amaral, L. A. (2005) Functional cartography of complex metabolic networks. *Nature* **433,** 895-900.

Hahn, M. W. and Wray, G. A. (2002) The g-value paradox. *Evol Dev* **4,** 73-75.

Harris, M. A., Clark, J., Ireland, A., Lomax, J., Ashburner, M., Foulger, R., Eilbeck, K., Lewis, S., Marshall, B., Mungall, C., Richter, J., Rubin, G. M., Blake, J. A., Bult, C., Dolan, M., Drabkin, H., Eppig, J. T., Hill, D. P., Ni, L., Ringwald, M., Balakrishnan, R., Cherry, J. M., Christie, K. R., Costanzo, M. C., Dwight, S. S., Engel, S., Fisk, D. G., Hirschman, J. E., Hong, E. L., Nash, R. S., Sethuraman, A., Theesfeld, C. L., Botstein, D., Dolinski, K., Feierbach, B., Berardini, T., Mundodi, S., Rhee, S. Y., Apweiler, R., Barrell, D., Camon, E., Dimmer, E., Lee, V., Chisholm, R., Gaudet, P., Kibbe, W., Kishore, R., Schwarz, E. M., Sternberg, P., Gwinn, M., Hannick, L., Wortman, J., Berriman, M., Wood, V., de la, C. N., Tonellato, P., Jaiswal, P., Seigfried, T., and White, R. (2004) The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res* **32,** D258-D261.

Harrison, A., Pearl, F., Mott, R., Thornton, J., and Orengo, C. (2002) Quantifying the similarities within fold space. *J Mol Biol* **323,** 909-926.

Hartwell, L. H., Hopfield, J. J., Leibler, S., and Murray, A. W. (1999) From molecular to modular cell biology. *Nature* **402,** C47-C52.

Hawkins, T., Chitale, M., Luban, S., and Kihara, D. (2008) PFP: Automated prediction of gene ontology functional annotations with confidence scores using protein sequence data. *Proteins*.

Heger, A., Wilton, C. A., Sivakumar, A., and Holm, L. (2005) ADDA: a domain database with global coverage of the protein universe. *Nucleic Acids Res* **33,** D188-D191.

Henikoff, S. and Henikoff, J. G. (1992) Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A* **89,** 10915-10919.

Henikoff, S. and Henikoff, J. G. (1993) Performance evaluation of amino acid substitution matrices. *Proteins* **17,** 49-61.

Henrick, K. and Thornton, J. M. (1998) PQS: a protein quaternary structure file server. *Trends Biochem Sci* **23,** 358-361.

Hirschman, J. E., Balakrishnan, R., Christie, K. R., Costanzo, M. C., Dwight, S. S., Engel, S. R., Fisk, D. G., Hong, E. L., Livstone, M. S., Nash, R., Park, J., Oughtred, R., Skrzypek, M., Starr, B., Theesfeld, C. L., Williams, J., Andrada, R., Binkley, G., Dong, Q., Lane, C., Miyasato, S., Sethuraman, A., Schroeder, M., Thanawala, M. K., Weng, S., Dolinski, K., Botstein, D., and Cherry, J. M. (2006) Genome Snapshot: a new resource at the Saccharomyces Genome Database (SGD) presenting an overview of the Saccharomyces cerevisiae genome. *Nucleic Acids Res* **34,** D442-D445.

Ho, Y., Gruhler, A., Heilbut, A., Bader, G. D., Moore, L., Adams, S. L., Millar, A., Taylor, P., Bennett, K., Boutilier, K., Yang, L., Wolting, C., Donaldson, I., Schandorff, S., Shewnarane, J., Vo, M., Taggart, J., Goudreault, M., Muskat, B., Alfarano, C., Dewar, D., Lin, Z., Michalickova, K., Willems, A. R., Sassi, H., Nielsen, P. A., Rasmussen, K. J., Andersen, J. R., Johansen, L. E., Hansen, L. H., Jespersen, H., Podtelejnikov, A., Nielsen, E., Crawford, J., Poulsen, V., Sorensen, B. D., Matthiesen, J., Hendrickson, R. C., Gleeson, F., Pawson, T., Moran, M. F., Durocher, D., Mann, M., Hogue, C. W., Figeys, D., and Tyers, M. (2002) Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry. *Nature* **415,** 180-183.

Hoffman, S. and Edelman, G. M. (1983) Kinetics of homophilic binding by embryonic and adult forms of the neural cell adhesion molecule. *Proc Natl Acad Sci U S A* **80,** 5762-5766.

Holm, L. and Sander, C. (1993) Protein structure comparison by alignment of distance matrices. *J Mol Biol* **233,** 123-138.

Holm, L. and Sander, C. (1994) The FSSP database of structurally aligned protein fold families. *Nucleic Acids Res* **22,** 3600-3609.

Holt, R. A., Subramanian, G. M., Halpern, A., Sutton, G. G., Charlab, R., Nusskern, D. R., Wincker, P., Clark, A. G., Ribeiro, J. M., Wides, R., Salzberg, S. L., Loftus, B., Yandell, M., Majoros, W. H., Rusch, D. B., Lai, Z., Kraft, C. L., Abril, J. F., Anthouard, V., Arensburger, P., Atkinson, P. W., Baden, H., de, B., V, Baldwin, D., Benes, V., Biedler, J., Blass, C., Bolanos, R., Boscus, D., Barnstead, M., Cai, S., Center, A., Chaturverdi, K., Christophides, G. K., Chrystal, M. A., Clamp, M., Cravchik, A., Curwen, V., Dana, A., Delcher, A., Dew, I., Evans, C. A., Flanigan, M., Grundschober-Freimoser, A., Friedli, L., Gu, Z., Guan, P., Guigo, R., Hillenmeyer, M. E., Hladun, S. L., Hogan, J. R., Hong, Y. S., Hoover, J., Jaillon, O., Ke, Z., Kodira, C., Kokoza, E., Koutsos, A., Letunic, I., Levitsky, A., Liang, Y., Lin, J. J., Lobo, N. F., Lopez, J. R., Malek, J. A., McIntosh, T. C., Meister, S., Miller, J., Mobarry, C., Mongin, E., Murphy, S. D., O'Brochta, D. A., Pfannkoch, C., Qi, R., Regier, M. A., Remington, K., Shao, H., Sharakhova, M. V., Sitter, C. D., Shetty, J., Smith, T. J., Strong, R., Sun, J., Thomasova, D., Ton, L. Q., Topalis, P., Tu, Z., Unger, M. F., Walenz, B., Wang, A., Wang, J., Wang, M., Wang, X., Woodford, K. J., Wortman, J. R., Wu, M., Yao, A., Zdobnov, E. M., Zhang, H., Zhao, Q., Zhao, S., Zhu, S. C., Zhimulev, I., Coluzzi, M., della, T. A., Roth, C. W., Louis, C., Kalush, F., Mural, R. J., Myers, E. W., Adams, M. D., Smith, H. O., Broder, S., Gardner, M. J., Fraser, C. M., Birney, E., Bork, P., Brey, P. T., Venter, J. C., Weissenbach, J., Kafatos, F. C., Collins, F. H., and Hoffman, S. L. (2002) The genome sequence of the malaria mosquito Anopheles gambiae. *Science* **298,** 129-149.

Hughey, R. and Krogh, A. (1996) Hidden Markov models for sequence analysis: extension and analysis of the basic method. *Comput Appl Biosci* **12,** 95-107.

Hulo, N., Bairoch, A., Bulliard, V., Cerutti, L., De Castro, E., Langendijk-Genevaux, P. S., Pagni, M., and Sigrist, C. J. (2006) The PROSITE database. *Nucleic Acids Res* **34,** D227-D230.

Huynen, M., Snel, B., Lathe, W., III, and Bork, P. (2000) Predicting protein function by genomic context: quantitative evaluation and qualitative inferences. *Genome Res* **10,** 1204-1210.

Ishihama, A. (2000) Functional modulation of Escherichia coli RNA polymerase. *Annu Rev Microbiol* **54,** 499-518.

Jansen, R., Greenbaum, D., and Gerstein, M. (2002) Relating whole-genome expression data with protein-protein interactions. *Genome Res* **12,** 37-46.

Jawad, Z. and Paoli, M. (2002) Novel sequences propel familiar folds. *Structure* **10,** 447-454.

Jefferson, E. R., Walsh, T. P., Roberts, T. J., and Barton, G. J. (2007) SNAPPI-DB: a database and API of Structures, iNterfaces and Alignments for Protein-Protein Interactions. *Nucleic Acids Res* **35,** D580-D589.

Jensen, L. J. and Bork, P. (2008) Biochemistry. Not comparable, but complementary. *Science* **322,** 56-57.

Jones, D. T., Taylor, W. R., and Thornton, J. M. (1992) A new approach to protein fold recognition. *Nature* **358,** 86-89.

Kahlem, P. and Birney, E. (2007) ENFIN a network to enhance integrative systems biology. *Ann N Y Acad Sci* **1115,** 23-31.

Kamburov, A., Goldovsky, L., Freilich, S., Kapazoglou, A., Kunin, V., Enright, A. J., Tsaftaris, A., and Ouzounis, C. A. (2007) Denoising inferred functional association networks obtained by gene fusion analysis. *BMC Genomics* **8,** 460.

Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K. F., Itoh, M., Kawashima, S., Katayama, T., Araki, M., and Hirakawa, M. (2006) From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res* **34,** D354-D357.

Kaplan, N., Sasson, O., Inbar, U., Friedlich, M., Fromer, M., Fleischer, H., Portugaly, E., Linial, N., and Linial, M. (2005) ProtoNet 4.0: a hierarchical classification of one million protein sequences. *Nucleic Acids Res* **33,** D216-D218.

Karchin, R. and Hughey, R. (1998) Weighting hidden Markov models for maximum discrimination. *Bioinformatics* **14,** 772-782.

Karp, P. D., Keseler, I. M., Shearer, A., Latendresse, M., Krummenacker, M., Paley, S. M., Paulsen, I., Collado-Vides, J., Gama-Castro, S., Peralta-Gil, M., Santos-Zavaleta, A., Penaloza-Spinola, M. I., Bonavides-Martinez, C., and Ingraham, J. (2007) Multidimensional annotation of the Escherichia coli K-12 genome. *Nucleic Acids Res* **35,** 7577-7590.

Karplus, K., Barrett, C., and Hughey, R. (1998) Hidden Markov models for detecting remote protein homologies. *Bioinformatics* **14,** 846-856.

Karplus, K., Karchin, R., Shackelford, G., and Hughey, R. (2005) Calibrating E-values for hidden Markov models using reverse-sequence null models. *Bioinformatics* **21,** 4107-4115.

Kelley, L. A., MacCallum, R. M., and Sternberg, M. J. (2000) Enhanced genome annotation using structural profiles in the program 3D-PSSM. *J Mol Biol* **299,** 499-520.

Kerrien, S., Alam-Faruque, Y., Aranda, B., Bancarz, I., Bridge, A., Derow, C., Dimmer, E., Feuermann, M., Friedrichsen, A., Huntley, R., Kohler, C., Khadake, J., Leroy, C., Liban, A., Lieftink, C., Montecchi-Palazzi, L., Orchard, S., Risse, J., Robbe, K., Roechert, B., Thorneycroft, D., Zhang, Y., Apweiler, R., and Hermjakob, H. (2007) IntAct--open source resource for molecular interaction data. *Nucleic Acids Res* **35,** D561-D565.

Kersey, P., Bower, L., Morris, L., Horne, A., Petryszak, R., Kanz, C., Kanapin, A., Das, U., Michoud, K., Phan, I., Gattiker, A., Kulikova, T., Faruque, N., Duggan, K., Mclaren, P., Reimholz, B., Duret, L., Penel, S., Reuter, I., and Apweiler, R. (2005) Integr8 and Genome Reviews: integrated views of complete genomes and proteomes. *Nucleic Acids Res* **33,** D297-D302.

King, A. D., Przulj, N., and Jurisica, I. (2004) Protein complex prediction via cost-based clustering. *Bioinformatics* **20,** 3013-3020.

Kleywegt, G. J. (1996) Use of non-crystallographic symmetry in protein structure refinement. *Acta Crystallogr D Biol Crystallogr* **52,** 842-857.

Kolodny, R., Koehl, P., and Levitt, M. (2005) Comprehensive evaluation of protein structure alignment methods: scoring by geometric measures. *J Mol Biol* **346,** 1173-1188.

Krishna, S. S. and Grishin, N. V. (2004) Structurally analogous proteins do exist! *Structure* **12,** 1125-1127.

Krissinel, E. and Henrick, K. (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. *Acta Crystallogr D Biol Crystallogr* **60,** 2256-2268.

Krogan, N. J., Cagney, G., Yu, H., Zhong, G., Guo, X., Ignatchenko, A., Li, J., Pu, S., Datta, N., Tikuisis, A. P., Punna, T., Peregrin-Alvarez, J. M., Shales, M., Zhang, X., Davey, M., Robinson, M. D., Paccanaro, A., Bray, J. E., Sheung, A., Beattie, B., Richards, D. P., Canadien, V., Lalev, A., Mena, F., Wong, P., Starostine, A., Canete, M. M., Vlasblom, J., Wu, S., Orsi, C., Collins, S. R., Chandran, S., Haw, R., Rilstone, J. J., Gandi, K., Thompson, N. J., Musso, G., St Onge, P., Ghanny, S., Lam, M. H., Butland, G., Altaf-Ul, A. M., Kanaya, S., Shilatifard, A., O'Shea, E., Weissman, J. S., Ingles, C. J., Hughes, T. R., Parkinson, J., Gerstein, M., Wodak, S. J., Emili, A., and Greenblatt, J. F. (2006) Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature* **440,** 637-643.

Krogh, A., Brown, M., Mian, I. S., Sjolander, K., and Haussler, D. (1994) Hidden Markov models in computational biology. Applications to protein modeling. *J Mol Biol* **235,** 1501-1531.

Krogh, A., Larsson, B., von Heijne, G., and Sonnhammer, E. L. (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol* **305,** 567-580.

Kulikova, T., Akhtar, R., Aldebert, P., Althorpe, N., Andersson, M., Baldwin, A., Bates, K., Bhattacharyya, S., Bower, L., Browne, P., Castro, M., Cochrane, G., Duggan, K., Eberhardt, R., Faruque, N., Hoad, G., Kanz, C., Lee, C., Leinonen, R., Lin, Q., Lombard, V., Lopez, R., Lorenc, D., McWilliam, H., Mukherjee, G., Nardone, F., Pastor, M. P., Plaister, S., Sobhany, S., Stoehr, P., Vaughan, R., Wu, D., Zhu, W., and Apweiler, R. (2007) EMBL Nucleotide Sequence Database in 2006. *Nucleic Acids Res* **35,** D16-D20.

Kummerfeld, S. K. and Teichmann, S. A. (2005) Relative rates of gene fusion and fission in multi-domain proteins. *Trends Genet* **21,** 25-30.

Lee, D., Grant, A., Marsden, R. L., and Orengo, C. (2005) Identification and distribution of protein families in 120 completed genomes using Gene3D. *Proteins* **59,** 603-615.

Lee, D., Redfern, O., and Orengo, C. (2007) Predicting protein function from sequence and structure. *Nat Rev Mol Cell Biol* **8,** 995-1005.

Letunic, I., Copley, R. R., Pils, B., Pinkert, S., Schultz, J., and Bork, P. (2006) SMART 5: domains in the context of genomes and networks. *Nucleic Acids Res* **34,** D257-D260.

Levy, E. D., Pereira-Leal, J. B., Chothia, C., and Teichmann, S. A. (2006) 3D complex: a structural classification of protein complexes. *PLoS Comput Biol* **2,** e155.

Liu, J. and Rost, B. (2004) CHOP proteins into structural domain-like fragments. *Proteins* **55,** 678-688.

Lord, P. W., Stevens, R. D., Brass, A., and Goble, C. A. (2003) Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation. *Bioinformatics* **19,** 1275-1283.

Lubovac, Z., Gamalielsson, J., and Olsson, B. (2006) Combining functional and topological properties to identify core modules in protein interaction networks. *Proteins* **64,** 948-959.

M.O.Dayhoff, Schwartz, R. M., and Orcutt, B. C. (1978) A model of evolutionary change in proteins. In: *Atlas of Protein Sequence and Structure*, pp. 345-352. Ed M.O.Dayhoff. Natl. Biomed. Res. Found.: Washington, DC.

Madera, M. (2008) Profile Comparer: a program for scoring and aligning profile hidden Markov models. *Bioinformatics* **24,** 2630-2631.

Madera, M. PRC - The Profile Comparer. 2006.
Ref Type: Unpublished Work

Madera, M. and Gough, J. (2002) A comparison of profile hidden Markov model procedures for remote homology detection. *Nucleic Acids Res* **30,** 4321-4328.

Marcotte, C. J. and Marcotte, E. M. (2002) Predicting functional linkages from gene fusions with confidence. *Appl Bioinformatics* **1,** 93-100.

Marcotte, E. M., Pellegrini, M., Ng, H. L., Rice, D. W., Yeates, T. O., and Eisenberg, D. (1999) Detecting protein function and protein-protein interactions from genome sequences. *Science* **285,** 751-753.

Marsden, R. L., Lee, D., Maibaum, M., Yeats, C., and Orengo, C. A. (2006) Comprehensive genome analysis of 203 genomes provides structural genomics with new insights into protein family space. *Nucleic Acids Res* **34,** 1066-1080.

Marsden, R. L., Lewis, T. A., and Orengo, C. A. (2007) Towards a comprehensive structural coverage of completed genomes: a structural genomics viewpoint. *BMC Bioinformatics* **8,** 86.

Marsden, R. L. and Orengo, C. A. (2008) The classification of protein domains. *Methods Mol Biol* **453,** 123-146.

Martin, D. M., Berriman, M., and Barton, G. J. (2004) GOtcha: a new method for prediction of protein function assessed by the annotation of seven genomes. *BMC Bioinformatics* **5,** 178.

May, A. C. (1999) A cautionary note on interpretation of hierarchical classifications of protein folds. *Structure* **7,** R213.

McDowall, M. D., Scott, M. S., and Barton, G. J. (2008) PIPs: human protein-protein interaction prediction database. *Nucleic Acids Res.*

McGuffin, L. J. and Jones, D. T. (2003) Improvement of the GenTHREADER method for genomic fold recognition. *Bioinformatics* **19,** 874-881.

McGuffin, L. J., Street, S. A., Bryson, K., Sorensen, S. A., and Jones, D. T. (2004) The Genomic Threading Database: a comprehensive resource for structural annotations of the genomes from key organisms. *Nucleic Acids Res* **32,** D196-D199.

McKusick, V. A. (1998) *Mendelian Inheritance in Man. A Catalog of Human Genes and Genetic Disorders.* Johns Hopkins University Press: Baltimore.

Meinel, T., Krause, A., Luz, H., Vingron, M., and Staub, E. (2005) The SYSTERS Protein Family Database in 2005. *Nucleic Acids Res* **33,** D226-D229.

Mellor, J. C., Yanai, I., Clodfelter, K. H., Mintseris, J., and DeLisi, C. (2002) Predictome: a database of putative functional links between proteins. *Nucleic Acids Res* **30,** 306-309.

Mewes, H. W., Dietmann, S., Frishman, D., Gregory, R., Mannhaupt, G., Mayer, K. F., Munsterkotter, M., Ruepp, A., Spannagl, M., Stumpflen, V., and Rattei, T. (2008) MIPS: analysis and annotation of genome information in 2007. *Nucleic Acids Res* **36,** D196-D201.

Mika, S. and Rost, B. (2006) Protein-protein interactions more conserved within species than across species. *PLoS Comput Biol* **2,** e79.

Monahan, B. J., Villen, J., Marguerat, S., Bahler, J., Gygi, S. P., and Winston, F. (2008) Fission yeast SWI/SNF and RSC complexes show compositional and functional differences from budding yeast. *Nat Struct Mol Biol* **15,** 873-880.

Mulder, N. J., Apweiler, R., Attwood, T. K., Bairoch, A., Bateman, A., Binns, D., Bork, P., Buillard, V., Cerutti, L., Copley, R., Courcelle, E., Das, U., Daugherty, L., Dibley, M., Finn, R., Fleischmann, W., Gough, J., Haft, D., Hulo, N., Hunter, S., Kahn, D., Kanapin, A., Kejariwal, A., Labarga, A., Langendijk-Genevaux, P. S., Lonsdale, D., Lopez, R., Letunic, I., Madera, M., Maslen, J., McAnulla, C., McDowall, J., Mistry, J., Mitchell, A., Nikolskaya, A. N., Orchard, S., Orengo, C., Petryszak, R., Selengut, J. D., Sigrist, C. J., Thomas, P. D., Valentin, F., Wilson, D., Wu, C. H., and Yeats, C. (2007) New developments in the InterPro database. *Nucleic Acids Res* **35,** D224-D228.

Muller, A., MacCallum, R. M., and Sternberg, M. J. (1999) Benchmarking PSI-BLAST in genome annotation. *J Mol Biol* **293,** 1257-1271.

Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C. (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol* **247,** 536-540.

Najmanovich, R. J., Torrance, J. W., and Thornton, J. M. (2005) Prediction of protein function from structure: insights from methods for the detection of local structural similarities. *Biotechniques* **38,** 847, 849, 851.

Needleman, S. B. and Wunsch, C. D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol* **48,** 443-453.

Nye, T. M., Berzuini, C., Gilks, W. R., Babu, M. M., and Teichmann, S. A. (2005) Statistical analysis of domains in interacting protein pairs. *Bioinformatics* **21,** 993-1001.

Ochman, H., Daubin, V., and Lerat, E. (2005) A bunch of fun-guys: the whole-genome view of yeast evolution. *Trends Genet* **21,** 1-3.

Ohlson, T., Wallner, B., and Elofsson, A. (2004) Profile-profile methods provide improved fold-recognition: a study of different profile-profile alignment methods. *Proteins* **57,** 188-197.

Ohno, S. (1970) *Evolution by Gene Duplication*. Springer: New York.

Orengo, C. A. and Taylor, W. R. (1996) SSAP: sequential structure alignment program for protein structure comparison. *Methods Enzymol* **266,** 617-635.

Orengo, C. A. and Thornton, J. M. (2005) Protein families and their evolution-a structural perspective. *Annu Rev Biochem* **74,** 867-900.

Page, G. P., Zakharkin, S. O., Kim, K., Mehta, T., Chen, L., and Zhang, K. (2007) Microarray analysis. *Methods Mol Biol* **404,** 409-430.

Pang, C. N., Krycer, J. R., Lek, A., and Wilkins, M. R. (2008) Are protein complexes made of cores, modules and attachments? *Proteomics* **8,** 425-434.

Park, J., Karplus, K., Barrett, C., Hughey, R., Haussler, D., Hubbard, T., and Chothia, C. (1998) Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. *J Mol Biol* **284,** 1201-1210.

Pearson, W. R. and Lipman, D. J. (1988) Improved tools for biological sequence comparison. *Proc Natl Acad Sci U S A* **85,** 2444-2448.

Pellegrini, M., Marcotte, E. M., Thompson, M. J., Eisenberg, D., and Yeates, T. O. (1999) Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proc Natl Acad Sci U S A* **96,** 4285-4288.

Pereira-Leal, J. B., Enright, A. J., and Ouzounis, C. A. (2004) Detection of functional modules from protein interaction networks. *Proteins* **54,** 49-57.

Pereira-Leal, J. B., Levy, E. D., Kamp, C., and Teichmann, S. A. (2007) Evolution of protein complexes by duplication of homomeric interactions. *Genome Biol* **8,** R51.

Pereira-Leal, J. B. and Teichmann, S. A. (2005) Novel specificities emerge by stepwise duplication of functional modules. *Genome Res* **15,** 552-559.

Petryszak, R., Kretschmann, E., Wieser, D., and Apweiler, R. (2005) The predictive power of the CluSTr database. *Bioinformatics* **21,** 3604-3609.

Pietrokovski, S. (1996) Searching databases of conserved sequence regions by aligning protein multiple-alignments. *Nucleic Acids Res* **24,** 3836-3845.

Porter, C. T., Bartlett, G. J., and Thornton, J. M. (2004) The Catalytic Site Atlas: a resource of catalytic sites and residues identified in enzymes using structural data. *Nucleic Acids Res* **32,** D129-D133.

Portugaly, E., Linial, N., and Linial, M. (2007) EVEREST: a collection of evolutionary conserved protein domains. *Nucleic Acids Res* **35,** D241-D246.

Pruitt, K. D., Tatusova, T., and Maglott, D. R. (2005) NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res* **33,** D501-D504.

Puig, O., Caspary, F., Rigaut, G., Rutz, B., Bouveret, E., Bragado-Nilsson, E., Wilm, M., and Seraphin, B. (2001) The tandem affinity purification (TAP) method: a general procedure of protein complex purification. *Methods* **24,** 218-229.

Qi, Y., Sadreyev, R. I., Wang, Y., Kim, B. H., and Grishin, N. V. (2007) A comprehensive system for evaluation of remote sequence similarity detection. *BMC Bioinformatics* **8,** 314.

Ranea, J. A. (2006) Genome evolution: micro(be)-economics. *Heredity* **96,** 337-338.

Ranea, J. A., Sillero, A., Thornton, J. M., and Orengo, C. A. (2006) Protein superfamily evolution and the last universal common ancestor (LUCA). *J Mol Evol* **63,** 513-525.

Ranea, J. A., Yeats, C., Grant, A., and Orengo, C. A. (2007) Predicting protein function with hierarchical phylogenetic profiles: the Gene3D Phylo-Tuner method applied to eukaryotic genomes. *PLoS Comput Biol* **3,** e237.

Redfern, O. C., Harrison, A., Dallman, T., Pearl, F. M., and Orengo, C. A. (2007) CATHEDRAL: a fast and effective algorithm to predict folds and domain boundaries from multidomain protein structures. *PLoS Comput Biol* **3,** e232.

Reeves, G. A., Dallman, T. J., Redfern, O. C., Akpor, A., and Orengo, C. A. (2006) Structural diversity of domain superfamilies in the CATH database. *J Mol Biol* **360,** 725-741.

Reid, A. J., Yeats, C., and Orengo, C. A. (2007) Methods of remote homology detection can be combined to increase coverage by 10% in the midnight zone. *Bioinformatics*.

Resnik, P. (1999) Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research* **11,** 95-130.

Rison, S. C., Teichmann, S. A., and Thornton, J. M. (2002) Homology, pathway distance and chromosomal localization of the small molecule metabolism enzymes in Escherichia coli. *J Mol Biol* **318,** 911-932.

Rost, B. (2002) Enzyme function less conserved than anticipated. *J Mol Biol* **318,** 595-608.

Rubin, D. M. and Finley, D. (1995) Proteolysis. The proteasome: a protein-degrading organelle? *Curr Biol* **5,** 854-858.

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., Tetko, I., Guldener, U., Mannhaupt, G., Munsterkotter, M., and Mewes, H. W. (2004) The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Res* **32,** 5539-5545.

Sadreyev, R. and Grishin, N. (2003) COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance. *J Mol Biol* **326,** 317-336.

Sadreyev, R. I., Tang, M., Kim, B. H., and Grishin, N. V. (2007) COMPASS server for remote homology inference. *Nucleic Acids Res*.

Salwinski, L., Miller, C. S., Smith, A. J., Pettit, F. K., Bowie, J. U., and Eisenberg, D. (2004) The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res* **32,** D449-D451.

Sanger, F., Nicklen, S., and Coulson, A. R. (1977) DNA sequencing with chain-terminating inhibitors. *Proc Natl Acad Sci U S A* **74,** 5463-5467.

Schuster-Bockler, B. and Bateman, A. (2007) Reuse of structural domain-domain interactions in protein networks. *BMC Bioinformatics* **8,** 259.

Schwikowski, B., Uetz, P., and Fields, S. (2000) A network of protein-protein interactions in yeast. *Nat Biotechnol* **18,** 1257-1261.

Shindyalov, I. N. and Bourne, P. E. (1998) Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng* **11,** 739-747.

Siew, N., Elofsson, A., Rychlewski, L., and Fischer, D. (2000) MaxSub: an automated measure for the assessment of protein structure prediction quality. *Bioinformatics* **16,** 776-785.

Sillitoe, I., Dibley, M., Bray, J., Addou, S., and Orengo, C. (2005) Assessing strategies for improved superfamily recognition. *Protein Sci* **14,** 1800-1810.

Sjolander, K., Karplus, K., Brown, M., Hughey, R., Krogh, A., Mian, I. S., and Haussler, D. (1996) Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology. *Comput Appl Biosci* **12,** 327-345.

Smith, T. F. and Waterman, M. S. (1981) Identification of common molecular subsequences. *J Mol Biol* **147,** 195-197.

Snel, B., Bork, P., and Huynen, M. (2000) Genome evolution. Gene fusion versus gene fission. *Trends Genet* **16,** 9-11.

Snel, B., Bork, P., and Huynen, M. A. (2002) Genomes in flux: the evolution of archaeal and proteobacterial gene content. *Genome Res* **12,** 17-25.

Snel, B. and Huynen, M. A. (2004) Quantifying modularity in the evolution of biomolecular systems. *Genome Res* **14,** 391-397.

Soding, J. (2005) Protein homology detection by HMM-HMM comparison. *Bioinformatics* **21,** 951-960.

Staley, J. P. and Guthrie, C. (1998) Mechanical devices of the spliceosome: motors, clocks, springs, and things. *Cell* **92,** 315-326.

Stark, C., Breitkreutz, B. J., Reguly, T., Boucher, L., Breitkreutz, A., and Tyers, M. (2006) BioGRID: a general repository for interaction datasets. *Nucleic Acids Res* **34,** D535-D539.

Stein, A., Russell, R. B., and Aloy, P. (2005) 3did: interacting protein domains of known three-dimensional structure. *Nucleic Acids Res* **33,** D413-D417.

Stumpf, M. P., Thorne, T., de Silva, E., Stewart, R., An, H. J., Lappe, M., and Wiuf, C. (2008) Estimating the size of the human interactome. *Proc Natl Acad Sci U S A* **105,** 6959-6964.

Subbiah, S., Laurents, D. V., and Levitt, M. (1993) Structural similarity of DNA-binding domains of bacteriophage repressors and the globin core. *Curr Biol* **3,** 141-148.

Suhre, K. and Claverie, J. M. (2004) FusionDB: a database for in-depth analysis of prokaryotic gene fusion events. *Nucleic Acids Res* **32,** D273-D276.

Suthram, S., Sittler, T., and Ideker, T. (2005) The Plasmodium protein network diverges from those of other eukaryotes. *Nature* **438,** 108-112.

Szathmary, E., Jordan, F., and Pal, C. (2001) Molecular biology and evolution. Can genes explain biological complexity? *Science* **292,** 1315-1316.

Tamames, J., Moya, A., and Valencia, A. (2007) Modular organization in the reductive evolution of protein-protein interaction networks. *Genome Biol* **8,** R94.

Tatusov, R. L., Fedorova, N. D., Jackson, J. D., Jacobs, A. R., Kiryutin, B., Koonin, E. V., Krylov, D. M., Mazumder, R., Mekhedov, S. L., Nikolskaya, A. N., Rao, B. S., Smirnov, S., Sverdlov, A. V., Vasudevan, S., Wolf, Y. I., Yin, J. J., and Natale, D. A. (2003) The COG database: an updated version includes eukaryotes. *BMC Bioinformatics* **4,** 41.

Taylor, W. R. (1986) The classification of amino acid conservation. *J Theor Biol* **119,** 205-218.

Taylor, W. R. and Orengo, C. A. (1989) Protein structure alignment. *J Mol Biol* **208,** 1-22.

Teichmann, S. A. and Babu, M. M. (2002) Conservation of gene co-regulation in prokaryotes and eukaryotes. *Trends Biotechnol* **20,** 407-410.

Teichmann, S. A., Rison, S. C., Thornton, J. M., Riley, M., Gough, J., and Chothia, C. (2001) The evolution and structural anatomy of the small molecule metabolic pathways in Escherichia coli. *J Mol Biol* **311,** 693-708.

Tian, W. and Skolnick, J. (2003) How well is enzyme function conserved as a function of pairwise sequence identity? *J Mol Biol* **333,** 863-882.

Todd, A. E., Orengo, C. A., and Thornton, J. M. (2001) Evolution of function in protein superfamilies, from a structural perspective. *J Mol Biol* **307,** 1113-1143.

Todd, A. E., Orengo, C. A., and Thornton, J. M. (2002) Sequence and structural differences between enzyme and nonenzyme homologs. *Structure* **10,** 1435-1451.

Tress, M. L., Martelli, P. L., Frankish, A., Reeves, G. A., Wesselink, J. J., Yeats, C., Olason, P. I., Albrecht, M., Hegyi, H., Giorgetti, A., Raimondo, D., Lagarde, J., Laskowski, R. A., Lopez, G., Sadowski, M. I., Watson, J. D., Fariselli, P., Rossi, I., Nagy, A., Kai, W., Storling, Z., Orsini, M., Assenov, Y., Blankenburg, H., Huthmacher, C., Ramirez, F., Schlicker, A., Denoeud, F., Jones, P., Kerrien, S., Orchard, S., Antonarakis, S. E., Reymond, A., Birney, E., Brunak, S., Casadio, R., Guigo, R., Harrow, J., Hermjakob, H., Jones, D. T., Lengauer, T., Orengo, C. A., Patthy, L., Thornton, J. M., Tramontano, A., and Valencia, A. (2007) The implications of alternative splicing in the ENCODE protein complement. *Proc Natl Acad Sci U S A* **104,** 5495-5500.

Truong, K. and Ikura, M. (2003) Domain fusion analysis by applying relational algebra to protein sequence and domain databases. *BMC Bioinformatics* **4,** 16.

Tweedie, S., Ashburner, M., Falls, K., Leyland, P., McQuilton, P., Marygold, S., Millburn, G., Osumi-Sutherland, D., Schroeder, A., Seal, R., and Zhang, H. (2008) FlyBase: enhancing Drosophila Gene Ontology annotations. *Nucleic Acids Res*.

van Dam, T. J. and Snel, B. (2008) Protein complex evolution does not involve extensive network rewiring. *PLoS Comput Biol* **4,** e1000132.

Van Dongen, S. Graph Clustering by Flow Simulation. 2000. University of Utrecht.
Ref Type: Thesis/Dissertation

Vogel, C., Bashton, M., Kerrison, N. D., Chothia, C., and Teichmann, S. A. (2004) Structure, function and evolution of multidomain proteins. *Curr Opin Struct Biol* **14,** 208-216.

Vogel, C. and Chothia, C. (2006) Protein family expansions and biological complexity. *PLoS Comput Biol* **2,** e48.

Vogel, C., Teichmann, S. A., and Pereira-Leal, J. (2005) The relationship between domain duplication and recombination. *J Mol Biol* **346,** 355-365.

von Mering, C., Jensen, L. J., Kuhn, M., Chaffron, S., Doerks, T., Kruger, B., Snel, B., and Bork, P. (2007) STRING 7--recent developments in the integration and prediction of protein interactions. *Nucleic Acids Res* **35,** D358-D362.

von Mering, C., Zdobnov, E. M., Tsoka, S., Ciccarelli, F. D., Pereira-Leal, J. B., Ouzounis, C. A., and Bork, P. (2003) Genome evolution reveals biochemical networks and functional modules. *Proc Natl Acad Sci U S A* **100,** 15428-15433.

Wagner, A. (2001) The yeast protein interaction network evolves rapidly and contains few redundant duplicate genes. *Mol Biol Evol* **18,** 1283-1292.

Walhout, A. J. and Vidal, M. (2001) High-throughput yeast two-hybrid assays for large-scale protein interaction mapping. *Methods* **24,** 297-306.

Wang, Z. and Zhang, J. (2007) In search of the biological significance of modular structures in protein networks. *PLoS Comput Biol* **3,** e107.

Wass, M. N. and Sternberg, M. J. (2008) ConFunc--functional annotation in the twilight zone. *Bioinformatics* **24,** 798-806.

Webb, E. C. (1992) Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology. In: *Enzyme Nomenclature*, Academic Press: New York.

Webber, C. and Barton, G. J. (2003) Increased coverage obtained by combination of methods for protein sequence database searching. *Bioinformatics* **19,** 1397-1403.

Wilson, D., Madera, M., Vogel, C., Chothia, C., and Gough, J. (2007) The SUPERFAMILY database in 2007: families and functions. *Nucleic Acids Res* **35,** D308-D313.

Wolfe, K. H. and Shields, D. C. (1997) Molecular evidence for an ancient duplication of the entire yeast genome. *Nature* **387,** 708-713.

Yanai, I., Derti, A., and DeLisi, C. (2001) Genes linked by fusion events are generally of the same functional category: a systematic analysis of 30 microbial genomes. *Proc Natl Acad Sci U S A* **98,** 7940-7945.

Yeats, C., Lees, J., Reid, A., Kellam, P., Martin, N., Liu, X., and Orengo, C. (2008) Gene3D: comprehensive structural and functional annotation of genomes. *Nucleic Acids Res* **36,** D414-D418.

Yona, G. and Levitt, M. (2002) Within the twilight zone: a sensitive profile-profile comparison tool based on information theory. *J Mol Biol* **315,** 1257-1275.

Yona, G., Linial, N., and Linial, M. (2000) ProtoMap: automatic classification of protein sequences and hierarchy of protein families. *Nucleic Acids Res* **28,** 49-55.

Yu, H., Luscombe, N. M., Lu, H. X., Zhu, X., Xia, Y., Han, J. D., Bertin, N., Chung, S., Vidal, M., and Gerstein, M. (2004) Annotation transfer between genomes: protein-protein interologs and protein-DNA regulogs. *Genome Res* **14,** 1107-1118.

Yu, X. and Egelman, E. H. (1997) The RecA hexamer is a structural homologue of ring helicases. *Nat Struct Biol* **4,** 101-104.

Zheng, H., Wang, H., and Glass, D. H. (2008) Integration of genomic data for inferring protein complexes from global protein-protein interaction networks. *IEEE Trans Syst Man Cybern B Cybern* **38,** 5-16.

# Appendix A

Probabilities of significant GOSS scores in several yeast and human sequence datasets

| Dataset | Yeast | Human |
|---|---|---|
| Gene3D v5 | 0.01671 | 0.00082 |
| STRING | 0.01418 | 0.00148 |
| Prolinks | 0.01783 | 0.00137 |
| Truong | 0.03709 | 0.00623 |

# Appendix B

Gene Ontology biological process annotations produced by CODA for 107 human proteins.

| UniProt identifier | GO terms |
|---|---|
| Q8NI37 | GO:0007165\|TAS GO:0006816\|IDA<br>GO:0009187\|NAS GO:0007165\|NAS<br>GO:0007601\|TAS |
| Q5JTI7 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q5JTI9 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q5VWA5 | GO:0006672\|TAS GO:0007165\|TAS |
| Q5QGT2;Q6NUK4;Q6PJY4;Q5JQR5;Q6PEW8 | GO:0006357\|TAS GO:0006357\|TAS |
| Q7L9B9 | GO:0006281\|NAS GO:0006260\|NAS<br>GO:0000279\|IEP GO:0006281\|NAS<br>GO:0006260\|NAS GO:0000279\|IEP |
| Q5M7Z8 | GO:0007585\|TAS GO:0008535\|TAS<br>GO:0007585\|TAS GO:0008535\|TAS |
| Q5T0R4 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q7Z7A3;Q96GZ7 | GO:0006520\|TAS GO:0009113\|TAS<br>GO:0006564\|NAS GO:0008615\|NAS<br>GO:0006461\|TAS GO:0000096\|TAS<br>GO:0009113\|TAS |
| Q5T0R7 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |

| | |
|---|---|
| Q9H8W0 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q6ZUX2 | GO:0006401\|TAS GO:0009615\|TAS<br>GO:0006401\|TAS GO:0009615\|TAS |
| Q5T8V1 | GO:0006412\|NAS GO:0006412\|NAS |
| Q7Z327;Q8IY39;Q7Z6V5 | GO:0007585\|TAS GO:0008535\|TAS |
| Q5JTI5 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q8TCB7;Q96LU4 | GO:0008152\|IDA GO:0006633\|IDA GO:0008152\|IDA<br>GO:0006633\|IDA |
| Q96HR9;Q96LM0 | GO:0006357\|TAS GO:0006357\|TAS |
| Q5VVM0 | GO:0045333\|NAS GO:0006118\|NAS |
| Q0VGD4 | GO:0009113\|TAS GO:0009113\|TAS |
| Q7Z5U5 | GO:0009113\|TAS GO:0009113\|TAS |
| Q5T0R9 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q86U90 | GO:0006796\|TAS GO:0006796\|TAS<br>GO:0006796\|TAS |
| Q86YL1 | GO:0006401\|TAS GO:0009615\|TAS<br>GO:0006928\|TAS GO:0006928\|TAS<br>GO:0006401\|TAS GO:0009615\|TAS<br>GO:0006928\|TAS GO:0006928\|TAS |
| Q5JUX3 | GO:0006944\|TAS GO:0006944\|TAS |
| Q96EI3;Q96IX1;Q6FI88;Q9Y6B4;Q6XYB0;Q9H0W9 | GO:0008544\|TAS GO:0006582\|NAS |
| Q5T6J8 | GO:0006118\|IDA GO:0009051\|IDA |
| Q5JTI8 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q86XN3 | GO:0006545\|NAS GO:0009165\|NAS<br>GO:0006545\|NAS GO:0009165\|NAS |
| O43341 | GO:0006401\|TAS GO:0009615\|TAS<br>GO:0006401\|TAS GO:0009615\|TAS |
| Q9P1A0 | GO:0006401\|TAS GO:0009615\|TAS<br>GO:0006401\|TAS GO:0009615\|TAS<br>GO:0006954\|TAS GO:0006800\|TAS |

| | |
|---|---|
| Q5T0S2 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q9H825 | GO:0008152\|IDA GO:0006633\|IDA GO:0008152\|IDA<br>GO:0006633\|IDA |
| Q86WR0;Q9NV98;Q96SI2 | GO:0009113\|TAS GO:0009113\|TAS |
| Q6ZW71;Q9BVQ3;Q9NRG7 | GO:0007159\|NAS GO:0042351\|TAS<br>GO:0005975\|TAS GO:0007159\|NAS<br>GO:0042351\|TAS GO:0009225\|TAS<br>GO:0006703\|TAS GO:0006702\|TAS<br>GO:0006629\|TAS GO:0007159\|NAS<br>GO:0042351\|TAS GO:0005975\|TAS<br>GO:0006629\|TAS |
| Q6ZRJ8 | GO:0006556\|IDA GO:0006556\|IDA |
| Q8NA58 | GO:0007292\|TAS GO:0009451\|TAS<br>GO:0007292\|TAS GO:0009451\|TAS |
| Q8N7X5;Q5TAP7 | GO:0007292\|TAS GO:0009451\|TAS |
| Q6DKI4;Q96LA8;Q9NVR8 | GO:0008152\|IDA GO:0006633\|IDA GO:0008152\|IDA<br>GO:0006633\|IDA |
| Q5TAW9;Q9BWV3 | GO:0007585\|TAS GO:0008535\|TAS |
| Q86SK8 | GO:0006464\|TAS GO:0006464\|TAS |
| Q9H6I5;Q9H6H4;Q86VL1;Q9HBP4 | GO:0006357\|TAS GO:0006357\|TAS |
| Q9HAU7;Q8IUT9;Q9HAT2;Q9NT71 | GO:0007399\|TAS GO:0006629\|TAS<br>GO:0006954\|TAS |
| Q8N467 | GO:0007585\|TAS GO:0007585\|TAS |
| Q9BZH2 | GO:0007585\|TAS GO:0008535\|TAS<br>GO:0007585\|TAS GO:0008535\|TAS |
| Q6IT77 | GO:0006298\|TAS GO:0008630\|TAS<br>GO:0006298\|TAS GO:0008630\|TAS |
| Q86SK7 | GO:0006464\|TAS |
| Q0VG05 | GO:0006366\|TAS |
| Q8TAR0;Q8NBX0;Q9Y363 | GO:0006595\|TAS GO:0006555\|TAS<br>GO:0015992\|TAS GO:0006099\|TAS<br>GO:0006118\|TAS GO:0006595\|TAS<br>GO:0006555\|TAS GO:0015992\|TAS<br>GO:0006099\|TAS GO:0006118\|TAS |
| Q8IUQ5 | GO:0006508\|TAS GO:0006508\|TAS |

| | |
|---|---|
| Q8N140 | GO:0006412\|TAS |
| Q6VNZ8 | GO:0009399\|TAS GO:0016226\|TAS<br>GO:0009399\|TAS GO:0016226\|TAS |
| Q9NWU2;Q8N5M5 | GO:0007049\|TAS GO:0007067\|TAS<br>GO:0006364\|TAS |
| Q96FC6;Q9H993;Q9UFY5 | GO:0006355\|NAS |
| A4D2M5 | GO:0018279\|IDA GO:0018279\|IDA GO:0018279\|IDA<br>GO:0018279\|IDA |
| Q5T0R6 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q5T0R5 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q0VAC6 | GO:0006464\|NAS GO:0006464\|NAS |
| Q3B7J1 | GO:0008152\|IDA GO:0006633\|IDA GO:0008152\|IDA<br>GO:0006633\|IDA |
| Q5T0R2 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| O94903;Q6FI94 | GO:0006139\|TAS GO:0006139\|TAS |
| Q5FWF4;Q9H0E8 | GO:0006298\|IMP GO:0043570\|IMP GO:0006298\|IDA<br>GO:0006284\|IDA GO:0007131\|TAS<br>GO:0006298\|IMP GO:0043570\|IMP GO:0006298\|IDA<br>GO:0006284\|IDA |
| Q8N7C5;Q8N4J0;Q7Z383 | GO:0006412\|NAS |
| Q05BX1 | GO:0006397\|TAS GO:0008380\|TAS<br>GO:0006917\|TAS GO:0006397\|TAS<br>GO:0008380\|TAS GO:0006917\|TAS |
| Q5T0R1 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q86TP1 | GO:0006419\|TAS GO:0008033\|TAS<br>GO:0006412\|NAS GO:0006412\|NAS<br>GO:0006419\|TAS GO:0008033\|TAS |
| Q0P663 | GO:0009113\|TAS GO:0009113\|TAS |

| | |
|---|---|
| Q5T017 | GO:0009103\|NAS GO:0009103\|NAS |
| Q5T0S3 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q5JUX6 | GO:0006944\|TAS GO:0006944\|TAS |
| Q9NV41;Q96HH6;Q53FY3 | GO:0008654\|NAS GO:0008654\|NAS<br>GO:0007601\|TAS GO:0007165\|TAS<br>GO:0006629\|TAS |
| Q9BYW9 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q8WY66;Q9NXX6;Q5SQQ5;Q6P673;Q9BS90 | GO:0006412\|TAS |
| Q96IQ6 | GO:0007292\|TAS GO:0009451\|TAS |
| Q9UBU6 | GO:0015942\|NAS GO:0015942\|NAS |
| Q8WZ99 | GO:0007186\|NAS GO:0007399\|TAS<br>GO:0009887\|TAS |
| Q96EY9 | GO:0007585\|TAS GO:0008535\|TAS |
| Q4LE72 | GO:0006944\|TAS GO:0006944\|TAS |
| A4FTY4 | GO:0006401\|TAS GO:0009615\|TAS |
| Q8TBR4 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q5T0R8 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q5VVM3 | GO:0045333\|NAS GO:0006118\|NAS |
| Q5T014 | GO:0009103\|NAS GO:0009103\|NAS |
| Q8N1G4;Q9ULN5 | GO:0006418\|TAS GO:0006935\|TAS<br>GO:0007165\|NAS GO:0006954\|TAS<br>GO:0006418\|TAS GO:0006935\|TAS<br>GO:0007165\|NAS GO:0006954\|TAS |
| A2A397 | GO:0030423\|IEP |
| Q9Y6N5;Q9UQM8 | GO:0006401\|TAS GO:0009615\|TAS |
| Q7Z5B1 | GO:0009399\|TAS GO:0016226\|TAS<br>GO:0009399\|TAS GO:0016226\|TAS |

| | |
|---|---|
| Q8NA83;Q8N3H2;Q8NHP6 | GO:0006366\|TAS |
| Q4G104 | GO:0006508\|TAS |
| Q5JPJ8 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q5T0R3 | GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS<br>GO:0007049\|NAS GO:0000070\|TAS<br>GO:0048015\|NAS GO:0007051\|NAS |
| Q5VTK4 | GO:0006595\|TAS GO:0006555\|TAS<br>GO:0015992\|TAS GO:0006099\|TAS<br>GO:0006118\|TAS GO:0006595\|TAS<br>GO:0006555\|TAS GO:0015992\|TAS<br>GO:0006099\|TAS GO:0006118\|TAS |
| O75423;O75424;O75425 | GO:0006366\|TAS |
| Q5H9C5;Q5H9C7;Q9UJG1 | GO:0006366\|TAS |
| Q9H7H0 | GO:0007585\|TAS GO:0008535\|TAS |
| Q96IZ6;Q9H9G9;Q9P0B5;Q9NUI8 | GO:0008152\|IDA GO:0006633\|IDA GO:0008152\|IDA<br>GO:0006633\|IDA |
| Q5JUX4 | GO:0006944\|TAS GO:0006944\|TAS |
| Q5T015 | GO:0009103\|NAS GO:0009103\|NAS |
| Q5TFJ4 | GO:0001561\|IDA GO:0001561\|IDA<br>GO:0008285\|TAS GO:0006436\|TAS<br>GO:0008285\|TAS GO:0006436\|TAS |
| Q5T9J8 | GO:0006139\|TAS GO:0006378\|NAS<br>GO:0006139\|TAS |
| Q9NW94;Q8N3B7;Q8IZV6;Q9BRR8 | GO:0000389\|TAS GO:0006376\|TAS<br>GO:0006397\|TAS GO:0000389\|TAS<br>GO:0006376\|TAS GO:0006397\|TAS |
| Q96EH3 | GO:0006656\|TAS GO:0006656\|TAS<br>GO:0008654\|TAS |
| A4FTW1;Q15493;Q53FC9;Q5JRR5 | GO:0006801\|TAS GO:0015680\|TAS |
| Q5TCW7 | GO:0006370\|IMP |
| Q6P275 | GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA<br>GO:0006367\|IDA GO:0030521\|IDA GO:0045944\|IDA |
| Q7L8W6;Q96HJ6 | GO:0006449\|TAS GO:0006449\|TAS |

| | |
|---|---|
| Q5TGK5;Q6PDA1;Q8NEV7;Q8IWS9;Q8IWT0;Q8NEV6 | GO:0006955\|NAS |
| A2A3L6 | GO:0007165\|TAS GO:0006468\|TAS<br>GO:0007165\|TAS GO:0006468\|TAS |
| Q8NI37 | GO:0007165\|TAS GO:0006816\|IDA<br>GO:0009187\|NAS GO:0007165\|NAS<br>GO:0007601\|TAS |

# Appendix C

## Species used in identifying orthologous groups (Chapter 4)

| Species | Classification | NCBI taxon Id |
|---|---|---|
| Oryza sativa | Eukaryota; Viridiplantae; Streptophyta | 39947 |
| Arabidopsis thaliana | Eukaryota; Viridiplantae; Streptophyta | 3702 |
| Dictyostelium discoideum | Eukaryota; Mycetozoa; Dictyosteliida | 352472 |
| Caenorhabditis elegans | Eukaryota; Metazoa; Nematoda | 6239 |
| Mus musculus | Eukaryota; Metazoa; Chordata | 10090 |
| Homo sapiens | Eukaryota; Metazoa; Chordata | 9606 |
| Danio rerio | Eukaryota; Metazoa; Chordata | 7955 |
| Anopheles gambiae | Eukaryota; Metazoa; Arthropoda | 180454 |
| Drosophila melanogaster | Eukaryota; Metazoa; Arthropoda | 7227 |
| Ustilago maydis | Eukaryota; Fungi; Basidiomycota; Ustilaginomycetes | 5270 |
| Saccharomyces cerevisiae | Eukaryota; Fungi; Ascomycota; Saccharomycotina | 4932 |
| Schizosaccharomyces pombe | Eukaryota; Fungi; Ascomycota; Schizosaccharomycetes | 4896 |
| Aspergillus fumigatus | Eukaryota; Fungi; Ascomycota; Pezizomycotina | 5085 |

| | | |
|---|---|---|
| Plasmodium falciparum 3D7 | Eukaryota; Alveolata; Apicomplexa | 36329 |
| Vibrio cholerae | Bacteria; Proteobacteria; Gammaproteobacteria | 666 |
| Pseudomonas putida KT2440 | Bacteria; Proteobacteria; Gammaproteobacteria | 160488 |
| Haemophilus influenzae | Bacteria; Proteobacteria; Gammaproteobacteria | 727 |
| Yersinia pestis | Bacteria; Proteobacteria; Gammaproteobacteria | 632 |
| Escherichia coli K12 | Bacteria; Proteobacteria; Gammaproteobacteria | 562 |
| Buchnera aphidicola (Bp) | Bacteria; Proteobacteria; Gammaproteobacteria | 135842 |
| Mycoplasma genitalium | Bacteria; Firmicutes; Mollicutes | 2097 |
| Clostridium acetobutylicum | Bacteria; Firmicutes; Clostridia | 1488 |
| Clostridium tetani | Bacteria; Firmicutes; Clostridia | 1513 |
| Bacillus subtilis | Bacteria; Firmicutes; Bacillales | 1423 |
| Thermus thermophilus HB27 | Bacteria; Deinococcus-Thermus; Deinococci | 262724 |
| Synechococcus elongatus | Bacteria; Cyanobacteria; Chroococcales | 32046 |
| Mycobacterium tuberculosis | Bacteria; Actinobacteria; Actinobacteridae | 1773 |
| Nanoarchaeum equitans | Archaea; Nanoarchaeota; Nanoarchaeum | 160232 |
| Thermoplasma acidophilum | Archaea; Euryarchaeota; Thermoplasmatasma | 2303 |
| Pyrococcus furiosus | Archaea; Euryarchaeota; Ther mococci | 2261 |
| Methanocaldococcus jannaschii | Archaea; Euryarchaeota; Methanococci | 2190 |
| Aeropyrum pernix | Archaea; Crenarchaeota; Thermoprotei | 56636 |

# Appendix D

## Correlated domain pairs identified in *E. coli* and yeast.

The tables presented here show the superfamily pairs identified as correlated in *E. coli* and yeast and the number of complexes in which those superfamilies were found together.

## D1. E. coli

| Superfamily A | Superfamily B | Number of complexes |
|---|---|---|
| 1.10.10.10 | 3.30.450.40 | 6 |
| 2.40.40.20 | 3.30.70.20 | 4 |
| 3.10.20.30 | 3.40.190.10 | 4 |
| 3.40.228.10 | 3.30.70.20 | 4 |
| 3.40.50.150 | 3.40.50.1820 | 4 |
| 3.50.50.60 | 2.60.120.10 | 4 |
| 3.90.55.10 | 3.30.70.20 | 4 |
| 1.10.1040.10 | 3.40.605.10 | 3 |
| 1.10.443.10 | 3.40.720.10 | 3 |
| 1.20.1090.10 | 3.40.50.150 | 3 |
| 1.20.1090.10 | 3.40.50.2300 | 3 |
| 1.25.40.10 | 2.40.50.100 | 3 |
| 2.40.50.140 | 1.10.730.10 | 3 |
| 2.60.120.10 | 1.25.40.10 | 3 |

| | | |
|---|---|---|
| 3.10.50.40 | 3.40.50.2300 | 3 |
| 3.30.450.40 | 3.40.930.10 | 3 |
| 3.30.930.10 | 2.160.10.10 | 3 |
| 3.40.1280.10 | 3.40.50.2300 | 3 |
| 3.40.50.10490 | 3.40.720.10 | 3 |
| 3.40.50.1220 | 3.40.50.150 | 3 |
| 3.40.50.1970 | 3.40.50.150 | 3 |
| 3.40.50.1970 | 3.40.50.2300 | 3 |
| 3.40.50.2300 | 3.30.870.10 | 3 |
| 3.40.50.300 | 3.40.50.1580 | 3 |
| 3.40.50.620 | 1.20.1090.10 | 3 |
| 3.40.50.620 | 3.20.20.150 | 3 |
| 3.40.50.620 | 3.40.50.1970 | 3 |
| 3.40.50.720 | 3.30.1490.20 | 3 |
| 3.40.50.970 | 3.30.420.40 | 3 |
| 3.40.630.30 | 1.20.1090.10 | 3 |
| 3.40.630.30 | 3.40.50.1970 | 3 |
| 3.90.226.10 | 3.20.20.120 | 3 |
| 3.90.226.10 | 3.90.1150.10 | 3 |
| 3.90.226.10 | 3.90.550.10 | 3 |
| 1.10.10.60 | 3.90.1200.10 | 2 |
| 1.10.1060.10 | 3.10.50.40 | 2 |
| 1.10.1660.10 | 3.60.10.10 | 2 |
| 1.10.260.40 | 3.40.50.1580 | 2 |
| 1.10.260.40 | 3.90.1530.10 | 2 |
| 1.10.443.10 | 3.40.50.1580 | 2 |
| 1.10.443.10 | 3.60.21.10 | 2 |
| 1.20.1090.10 | 1.10.1680.10 | 2 |
| 1.20.1090.10 | 3.10.290.10 | 2 |

| | | |
|---|---|---|
| 1.20.1090.10 | 3.20.20.150 | 2 |
| 1.20.1090.10 | 3.30.870.10 | 2 |
| 1.20.1090.10 | 3.40.1090.10 | 2 |
| 1.20.1090.10 | 3.90.110.10 | 2 |
| 1.20.1090.10 | 3.90.1200.10 | 2 |
| 1.20.58.100 | 1.10.8.60 | 2 |
| 1.20.58.100 | 3.30.450.20 | 2 |
| 2.130.10.10 | 3.40.50.2300 | 2 |
| 2.160.10.10 | 3.90.1200.10 | 2 |
| 2.40.160.10 | 3.40.605.10 | 2 |
| 2.40.160.10 | 3.40.930.10 | 2 |
| 2.40.240.10 | 1.10.1040.10 | 2 |
| 2.40.240.10 | 1.20.1090.10 | 2 |
| 2.40.240.10 | 2.40.50.140 | 2 |
| 2.40.240.10 | 3.10.129.10 | 2 |
| 2.40.240.10 | 3.10.290.10 | 2 |
| 2.40.240.10 | 3.30.870.10 | 2 |
| 2.40.240.10 | 3.40.50.150 | 2 |
| 2.40.240.10 | 3.40.50.1970 | 2 |
| 2.40.240.10 | 3.40.50.2300 | 2 |
| 2.40.240.10 | 3.40.50.620 | 2 |
| 2.40.240.10 | 3.40.605.10 | 2 |
| 2.40.240.10 | 3.40.630.30 | 2 |
| 2.40.240.10 | 3.90.110.10 | 2 |
| 2.40.240.10 | 3.90.79.10 | 2 |
| 2.40.50.140 | 3.30.160.100 | 2 |
| 2.60.120.10 | 3.20.20.10 | 2 |
| 2.60.120.10 | 3.40.1090.10 | 2 |
| 2.60.120.10 | 3.90.1200.10 | 2 |

| | | |
|---|---|---|
| 2.60.120.260 | 1.10.150.130 | 2 |
| 2.60.120.260 | 1.10.443.10 | 2 |
| 2.60.40.1090 | 2.60.40.1070 | 2 |
| 2.60.40.320 | 1.10.150.130 | 2 |
| 2.60.40.320 | 1.10.443.10 | 2 |
| 2.70.98.10 | 1.10.150.130 | 2 |
| 2.70.98.10 | 1.10.443.10 | 2 |
| 2.70.98.10 | 3.10.290.10 | 2 |
| 3.10.129.10 | 1.10.940.10 | 2 |
| 3.10.129.10 | 1.20.1090.10 | 2 |
| 3.10.129.10 | 3.30.870.10 | 2 |
| 3.10.129.10 | 3.40.50.1970 | 2 |
| 3.10.129.10 | 3.90.110.10 | 2 |
| 3.10.20.30 | 3.20.20.120 | 2 |
| 3.10.20.30 | 3.90.1530.10 | 2 |
| 3.10.50.40 | 3.40.50.9600 | 2 |
| 3.20.20.140 | 3.30.160.100 | 2 |
| 3.20.20.140 | 3.40.50.1580 | 2 |
| 3.20.20.150 | 1.10.1680.10 | 2 |
| 3.20.20.80 | 1.10.150.130 | 2 |
| 3.20.70.20 | 2.40.50.100 | 2 |
| 3.30.110.40 | 3.40.640.10 | 2 |
| 3.30.110.40 | 3.90.1150.10 | 2 |
| 3.30.230.10 | 3.40.50.2000 | 2 |
| 3.30.300.30 | 2.60.40.420 | 2 |
| 3.30.390.30 | 3.30.160.100 | 2 |
| 3.30.450.40 | 3.90.230.10 | 2 |
| 3.30.470.20 | 3.40.50.10540 | 2 |
| 3.30.70.920 | 3.90.1150.10 | 2 |

| | | |
|---|---|---|
| 3.30.870.10 | 3.10.290.10 | 2 |
| 3.30.870.10 | 3.40.605.10 | 2 |
| 3.30.870.10 | 3.90.110.10 | 2 |
| 3.30.930.10 | 3.40.1090.10 | 2 |
| 3.30.930.10 | 3.90.1200.10 | 2 |
| 3.40.1090.10 | 1.10.10.60 | 2 |
| 3.40.1090.10 | 1.25.40.10 | 2 |
| 3.40.1090.10 | 2.160.10.10 | 2 |
| 3.40.1090.10 | 3.90.1200.10 | 2 |
| 3.40.1190.20 | 3.30.1490.20 | 2 |
| 3.40.1280.10 | 2.60.40.1070 | 2 |
| 3.40.1280.10 | 2.60.40.360 | 2 |
| 3.40.1280.10 | 3.30.870.10 | 2 |
| 3.40.190.10 | 3.40.1090.10 | 2 |
| 3.40.192.10 | 3.30.1490.20 | 2 |
| 3.40.30.10 | 3.40.1090.10 | 2 |
| 3.40.30.10 | 3.90.1200.10 | 2 |
| 3.40.50.1000 | 3.40.1090.10 | 2 |
| 3.40.50.1000 | 3.90.1200.10 | 2 |
| 3.40.50.10540 | 3.30.1490.20 | 2 |
| 3.40.50.10540 | 3.40.1190.20 | 2 |
| 3.40.50.1100 | 1.10.1680.10 | 2 |
| 3.40.50.1240 | 3.30.1490.20 | 2 |
| 3.40.50.1240 | 3.40.50.10540 | 2 |
| 3.40.50.150 | 3.40.1090.10 | 2 |
| 3.40.50.150 | 3.90.1200.10 | 2 |
| 3.40.50.1580 | 3.20.20.100 | 2 |
| 3.40.50.1580 | 3.30.70.20 | 2 |
| 3.40.50.1580 | 3.60.21.10 | 2 |

| | | |
|---|---|---|
| 3.40.50.1820 | 3.40.1090.10 | 2 |
| 3.40.50.1820 | 3.90.1200.10 | 2 |
| 3.40.50.1970 | 1.10.1680.10 | 2 |
| 3.40.50.1970 | 3.10.290.10 | 2 |
| 3.40.50.1970 | 3.20.20.150 | 2 |
| 3.40.50.1970 | 3.30.870.10 | 2 |
| 3.40.50.1970 | 3.40.1090.10 | 2 |
| 3.40.50.1970 | 3.90.110.10 | 2 |
| 3.40.50.1970 | 3.90.1200.10 | 2 |
| 3.40.50.20 | 3.40.192.10 | 2 |
| 3.40.50.2300 | 3.30.1330.10 | 2 |
| 3.40.50.2300 | 3.40.1090.10 | 2 |
| 3.40.50.2300 | 3.40.50.1360 | 2 |
| 3.40.50.2300 | 3.90.1200.10 | 2 |
| 3.40.50.2300 | 3.90.650.10 | 2 |
| 3.40.50.261 | 1.10.1680.10 | 2 |
| 3.40.50.620 | 2.30.38.10 | 2 |
| 3.40.50.620 | 3.40.1090.10 | 2 |
| 3.40.50.620 | 3.90.1200.10 | 2 |
| 3.40.50.980 | 2.60.40.420 | 2 |
| 3.40.630.30 | 3.20.20.150 | 2 |
| 3.40.630.30 | 3.30.870.10 | 2 |
| 3.40.630.30 | 3.40.1090.10 | 2 |
| 3.40.630.30 | 3.90.110.10 | 2 |
| 3.40.630.30 | 3.90.1200.10 | 2 |
| 3.40.640.10 | 3.40.1090.10 | 2 |
| 3.40.640.10 | 3.40.50.9600 | 2 |
| 3.40.640.10 | 3.90.1200.10 | 2 |
| 3.40.720.10 | 3.20.20.30 | 2 |

| | | |
|---|---|---|
| 3.40.930.10 | 1.20.1090.10 | 2 |
| 3.40.930.10 | 3.20.20.150 | 2 |
| 3.40.930.10 | 3.40.50.170 | 2 |
| 3.40.930.10 | 3.40.50.1970 | 2 |
| 3.40.980.10 | 3.60.120.10 | 2 |
| 3.50.50.60 | 2.60.300.12 | 2 |
| 3.50.50.60 | 3.30.160.100 | 2 |
| 3.50.50.60 | 3.40.1090.10 | 2 |
| 3.50.50.60 | 3.40.220.10 | 2 |
| 3.50.50.60 | 3.90.1200.10 | 2 |
| 3.90.110.10 | 3.10.290.10 | 2 |
| 3.90.110.10 | 3.40.605.10 | 2 |
| 3.90.1150.10 | 3.40.1090.10 | 2 |
| 3.90.1150.10 | 3.90.1200.10 | 2 |
| 3.90.1200.10 | 1.25.40.10 | 2 |
| 3.90.226.10 | 1.20.1090.10 | 2 |
| 3.90.226.10 | 3.30.390.10 | 2 |
| 3.90.226.10 | 3.40.1090.10 | 2 |
| 3.90.226.10 | 3.40.50.1970 | 2 |
| 3.90.226.10 | 3.90.1200.10 | 2 |
| 3.90.55.10 | 3.40.1160.10 | 2 |
| 3.90.550.10 | 1.25.40.20 | 2 |
| 3.90.550.10 | 3.40.1090.10 | 2 |
| 3.90.550.10 | 3.90.1200.10 | 2 |
| 3.90.700.10 | 1.10.8.60 | 2 |
| 3.90.700.10 | 3.30.450.20 | 2 |
| 3.90.79.10 | 3.30.870.10 | 2 |
| 3.90.79.10 | 3.90.110.10 | 2 |
| 4.10.520.10 | 1.20.1090.10 | 2 |

| | | |
|---|---|---|
| 4.10.520.10 | 3.40.50.1970 | 2 |

# D2. Yeast

| Superfamily A | Superfamily B | Number of complexes |
|---|---|---|
| 1.10.10.60 | 4.10.240.10 | 6 |
| 1.10.10.60 | 1.10.10.10 | 5 |
| 1.25.10.10 | 3.30.450.60 | 5 |
| 3.10.110.10 | 3.30.40.10 | 5 |
| 3.10.20.90 | 1.10.10.60 | 5 |
| 1.10.10.10 | 3.40.630.10 | 4 |
| 1.25.10.10 | 2.60.40.1170 | 4 |
| 3.40.50.1240 | 3.40.30.10 | 4 |
| 1.10.10.60 | 1.20.920.10 | 3 |
| 2.60.120.200 | 1.10.238.10 | 3 |
| 3.10.129.10 | 3.20.20.80 | 3 |
| 3.10.129.10 | 3.30.420.40 | 3 |
| 3.10.129.10 | 3.40.30.10 | 3 |
| 3.20.20.70 | 3.30.428.10 | 3 |
| 3.20.20.80 | 2.60.260.20 | 3 |
| 3.20.20.80 | 3.40.630.10 | 3 |
| 3.30.420.40 | 1.10.245.10 | 3 |
| 3.30.450.60 | 2.60.40.1170 | 3 |
| 3.30.70.240 | 3.30.40.10 | 3 |
| 3.30.70.870 | 3.30.40.10 | 3 |
| 3.40.50.720 | 3.10.129.10 | 3 |
| 3.40.50.720 | 3.40.1190.20 | 3 |
| 3.50.50.60 | 3.90.180.10 | 3 |
| 1.10.10.10 | 2.170.120.12 | 2 |

| | | |
|---|---|---|
| 1.10.10.10 | 3.10.120.10 | 2 |
| 1.10.10.10 | 3.30.1490.120 | 2 |
| 1.10.10.10 | 3.40.720.10 | 2 |
| 1.10.10.60 | 1.20.1070.10 | 2 |
| 1.10.10.60 | 2.30.30.70 | 2 |
| 1.10.10.60 | 3.40.800.20 | 2 |
| 1.10.1000.11 | 3.60.40.10 | 2 |
| 1.10.1370.10 | 3.30.50.10 | 2 |
| 1.10.220.20 | 3.30.420.40 | 2 |
| 1.10.220.20 | 3.60.40.10 | 2 |
| 1.10.287.600 | 3.30.1370.50 | 2 |
| 1.10.287.600 | 3.40.30.10 | 2 |
| 1.10.555.10 | 3.40.1190.20 | 2 |
| 1.10.600.10 | 3.30.780.10 | 2 |
| 1.10.730.10 | 3.10.50.40 | 2 |
| 1.20.1050.10 | 1.10.600.10 | 2 |
| 1.20.1050.40 | 3.30.50.10 | 2 |
| 1.20.58.90 | 3.30.1520.10 | 2 |
| 1.20.910.10 | 3.40.50.720 | 2 |
| 1.25.10.10 | 2.30.130.10 | 2 |
| 1.25.10.10 | 3.40.50.10480 | 2 |
| 1.25.40.10 | 1.20.58.90 | 2 |
| 1.25.40.10 | 3.40.1180.10 | 2 |
| 1.25.40.20 | 3.40.50.2300 | 2 |
| 1.50.10.20 | 1.25.40.120 | 2 |
| 2.130.10.10 | 3.40.50.10480 | 2 |
| 2.170.120.12 | 1.10.10.60 | 2 |
| 2.30.130.10 | 2.130.10.10 | 2 |
| 2.30.130.10 | 3.30.70.330 | 2 |

| | | |
|---|---|---|
| 2.30.250.10 | 1.10.45.10 | 2 |
| 2.30.250.10 | 3.30.43.10 | 2 |
| 2.30.38.10 | 3.90.470.20 | 2 |
| 2.40.50.140 | 2.170.120.12 | 2 |
| 2.40.50.140 | 3.30.1360.10 | 2 |
| 2.40.50.150 | 1.10.10.10 | 2 |
| 2.40.50.150 | 1.10.10.60 | 2 |
| 2.40.50.150 | 2.170.120.12 | 2 |
| 2.40.50.150 | 2.40.50.140 | 2 |
| 2.40.50.150 | 3.30.1360.10 | 2 |
| 2.40.50.150 | 3.30.1490.120 | 2 |
| 2.60.260.20 | 3.80.10.10 | 2 |
| 2.60.40.1170 | 2.60.40.1230 | 2 |
| 2.60.40.1180 | 3.40.190.10 | 2 |
| 3.10.110.10 | 1.10.1040.10 | 2 |
| 3.10.110.10 | 1.10.245.10 | 2 |
| 3.10.120.10 | 1.10.245.10 | 2 |
| 3.10.120.10 | 2.10.230.10 | 2 |
| 3.10.120.10 | 2.60.260.20 | 2 |
| 3.10.120.10 | 3.10.110.10 | 2 |
| 3.10.120.10 | 3.30.420.40 | 2 |
| 3.10.120.10 | 3.40.630.10 | 2 |
| 3.10.129.10 | 1.10.245.10 | 2 |
| 3.10.129.10 | 2.10.230.10 | 2 |
| 3.10.129.10 | 2.60.260.20 | 2 |
| 3.10.129.10 | 3.10.110.10 | 2 |
| 3.10.129.10 | 3.10.120.10 | 2 |
| 3.10.129.10 | 3.30.360.10 | 2 |
| 3.10.129.10 | 3.40.630.10 | 2 |

| | | |
|---|---|---|
| 3.10.20.30 | 1.10.245.10 | 2 |
| 3.10.20.30 | 3.10.110.10 | 2 |
| 3.10.20.30 | 3.10.120.10 | 2 |
| 3.10.20.30 | 3.10.129.10 | 2 |
| 3.10.20.30 | 3.40.630.10 | 2 |
| 3.10.20.30 | 3.50.50.60 | 2 |
| 3.10.20.90 | 3.30.70.100 | 2 |
| 3.10.260.10 | 2.60.200.20 | 2 |
| 3.20.20.140 | 2.60.40.1180 | 2 |
| 3.20.20.140 | 3.30.1550.10 | 2 |
| 3.20.20.140 | 3.40.50.790 | 2 |
| 3.20.20.70 | 3.40.50.1170 | 2 |
| 3.20.20.80 | 1.10.245.10 | 2 |
| 3.20.20.80 | 1.10.840.10 | 2 |
| 3.20.20.80 | 1.20.58.90 | 2 |
| 3.20.20.80 | 2.10.230.10 | 2 |
| 3.20.20.80 | 3.10.120.10 | 2 |
| 3.20.20.80 | 3.30.1520.10 | 2 |
| 3.20.20.80 | 3.30.360.10 | 2 |
| 3.20.20.80 | 3.40.1180.10 | 2 |
| 3.20.20.80 | 3.40.20.10 | 2 |
| 3.20.20.80 | 3.40.720.10 | 2 |
| 3.30.1330.20 | 3.30.1370.50 | 2 |
| 3.30.1330.20 | 3.40.30.10 | 2 |
| 3.30.1360.10 | 2.170.120.12 | 2 |
| 3.30.1360.10 | 3.30.1490.120 | 2 |
| 3.30.1360.70 | 3.10.50.40 | 2 |
| 3.30.1490.120 | 1.10.10.60 | 2 |
| 3.30.1490.120 | 2.170.120.12 | 2 |

| | | |
|---|---|---|
| 3.30.1490.40 | 3.30.70.330 | 2 |
| 3.30.160.60 | 2.30.29.30 | 2 |
| 3.30.310.30 | 3.30.450.60 | 2 |
| 3.30.360.10 | 2.10.230.10 | 2 |
| 3.30.360.10 | 2.60.260.20 | 2 |
| 3.30.360.10 | 3.30.420.40 | 2 |
| 3.30.360.10 | 3.40.640.10 | 2 |
| 3.30.420.40 | 3.10.28.10 | 2 |
| 3.30.450.60 | 2.60.40.1230 | 2 |
| 3.30.460.10 | 3.40.50.1000 | 2 |
| 3.30.470.20 | 2.40.50.100 | 2 |
| 3.30.50.10 | 3.10.260.10 | 2 |
| 3.30.60.20 | 1.10.10.10 | 2 |
| 3.30.60.20 | 1.10.10.60 | 2 |
| 3.30.70.240 | 3.10.110.10 | 2 |
| 3.30.70.240 | 3.30.930.10 | 2 |
| 3.30.70.330 | 2.30.170.20 | 2 |
| 3.30.70.330 | 3.40.50.2300 | 2 |
| 3.30.70.870 | 3.10.110.10 | 2 |
| 3.30.70.870 | 3.30.930.10 | 2 |
| 3.30.930.10 | 1.10.600.10 | 2 |
| 3.30.930.10 | 2.60.120.260 | 2 |
| 3.30.930.10 | 3.30.780.10 | 2 |
| 3.40.1180.10 | 2.60.260.20 | 2 |
| 3.40.1180.10 | 4.10.240.10 | 2 |
| 3.40.1190.20 | 3.20.20.100 | 2 |
| 3.40.1190.20 | 3.40.190.10 | 2 |
| 3.40.190.10 | 3.20.20.100 | 2 |
| 3.40.250.10 | 1.10.150.50 | 2 |

| | | |
|---|---|---|
| 3.40.30.10 | 1.10.600.10 | 2 |
| 3.40.30.10 | 3.10.120.10 | 2 |
| 3.40.30.10 | 3.30.1370.50 | 2 |
| 3.40.30.10 | 3.40.1180.10 | 2 |
| 3.40.50.1000 | 3.40.720.10 | 2 |
| 3.40.50.1170 | 3.50.50.60 | 2 |
| 3.40.50.1380 | 3.40.50.1000 | 2 |
| 3.40.50.1440 | 3.30.1370.50 | 2 |
| 3.40.50.1440 | 3.40.30.10 | 2 |
| 3.40.50.150 | 3.40.50.10480 | 2 |
| 3.40.50.20 | 2.40.50.100 | 2 |
| 3.40.50.20 | 3.30.470.20 | 2 |
| 3.40.50.410 | 2.130.10.10 | 2 |
| 3.40.50.620 | 3.10.50.40 | 2 |
| 3.40.50.720 | 2.40.180.10 | 2 |
| 3.40.50.720 | 3.30.160.20 | 2 |
| 3.40.50.720 | 3.90.470.20 | 2 |
| 3.40.50.790 | 3.30.1550.10 | 2 |
| 3.40.50.800 | 1.10.600.10 | 2 |
| 3.40.50.800 | 3.30.780.10 | 2 |
| 3.40.50.980 | 3.90.470.20 | 2 |
| 3.40.630.10 | 1.10.245.10 | 2 |
| 3.40.630.10 | 1.10.45.10 | 2 |
| 3.40.630.10 | 3.30.43.10 | 2 |
| 3.40.630.10 | 3.40.720.10 | 2 |
| 3.50.50.60 | 3.30.1610.10 | 2 |
| 3.60.10.10 | 1.10.555.10 | 2 |
| 3.60.10.10 | 1.20.1070.10 | 2 |
| 3.60.110.10 | 3.50.50.60 | 2 |

| | | |
|---|---|---|
| 3.60.15.10 | 3.80.10.10 | 2 |
| 3.60.15.10 | 4.10.240.10 | 2 |
| 3.80.10.10 | 1.10.580.10 | 2 |
| 3.80.10.10 | 3.40.605.10 | 2 |
| 3.90.1100.10 | 1.10.10.10 | 2 |
| 3.90.1100.10 | 1.10.10.60 | 2 |
| 3.90.1100.10 | 2.170.120.12 | 2 |
| 3.90.1100.10 | 2.40.50.140 | 2 |
| 3.90.1100.10 | 3.30.1360.10 | 2 |
| 3.90.1100.10 | 3.30.1490.120 | 2 |
| 3.90.230.10 | 3.40.50.10190 | 2 |
| 3.90.550.10 | 1.10.245.10 | 2 |
| 3.90.550.10 | 3.10.120.10 | 2 |
| 3.90.79.10 | 3.10.110.10 | 2 |