Automated methods for the determination of homologous relationships and functional similarities between protein domains

Oliver Charles Redfern

Department of Biochemistry and Molecular Biology
University College London

A thesis submitted to the University of London in the Faculty of Science for the degree of Doctor of Philosophy

June 2007

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Abstract

CATH is a protein database of structural domains which are assigned to superfamilies through evidence of a common evolutionary ancestor. These superfamilies are further grouped by overall structural similarity into folds. This thesis explores several automated methods for recognising homologous relationships between these domains using the structural data from the Protein Data Bank (PDB). The aim of this work was to aid the manual classification of domains into the database and provide putative functional assignments to structures solved by the structural genomics initiatives.

A fast and novel algorithm, CATHEDRAL, was developed to make fold assignments to regions of polypeptide chains. By combining a fast secondary-structure method (GRATH) and a slower residue-based method (SSAP), the algorithm was able to accurately assign boundaries for distant relatives, undetectable by sequence methods.

Sequence and structural conservation patterns were combined in a novel algorithm, FLORA, to develop structural templates specific to catalytic function. FLORA was able to predict the correct functional site in 80% of cases and combined with global structure comparison, it was able to assign domains to enzyme families within diverse superfamilies.

Techniques in structure comparison were also applied to *ab initio* models of protein domains, in order to assign them to fold groups within the CATH database. A novel scoring method was developed to pre-select models that were more likely to have adopted the correct fold. A selected sample of models for each target structure was then compared against representatives from the CATH database using the MAMMOTH and SSAP algorithms. Data from these alignments were combined using a Support Vector Machine to assign the target to a fold group within CATH.

This work was generously supported by the Engineering and Physical Sciences Research Council.

Acknowledgements

When I arrived at UCL as a fresh-faced twenty-one year old, I never imagined how much I would learn and how many different people I would meet over the course of my PhD. I am at pains to forget to mention anyone who has tolerated my incessant questioning and pleas for help, so I will do my utmost to cover them all and apologise profusely to those I inadvertently omit.

First and foremost, I cannot thank Christine Orengo enough for her first-rate supervision, unwavering support and consistent encouragement over the years. As a consequence of her immense hard work and passion for science, I have benefited from an excellent lab and been surrounded by fantastic colleagues. I will be forever grateful for the opportunity she has given me.

I also wish to pay a special tribute to Russell Marsden who has devoted so much time to reading this thesis and has helped me more than he realises to get to a stage where I feel I can submit it.

Since I began in the lab knowing very little about the field of bioinformatics, a huge number of past and current members of the Thornton and Orengo groups have helped along the way. Gabby Reeves, Daniel Buchan, Stuart Rison and James Bray gave me support in the early years and made the lab such a fun and welcoming place to work. Ian Sillitoe was always at hand with his trademark words of wisdom and gave advice on Chapter 4. Tony Lewis deserves a lot of thanks for his programming tips and particularly for helping to set up the web site for CATHEDRAL. Tim Dallman has been a joy to work with and ever tolerant of my rapidly evolving programs. Thanks also to everyone else in the CATH group: Alison Cuff, Adam Reid, David Lee, Corin Yeats, Jonathan Lees, Benoit Dessailly.

I thank Roman Laskowski for kindly running his SiteSeer program on my data set.

Special thanks go to those people who have patiently calmed me in hours of need and given me confidence to carry on when things were looking bleak: Ellen Hardy, Henrietta Gordon, Douglas Corrigall and Joana Rzepa.

Finally, I wish to thank my family for caring so much and realising how important my PhD was to me, without having a clue what it was about. To Mum, Nanny, Auntie Dot and Uncle John: I hope you feel it was worth the wait.

List of Abbreviations

Abbreviation	Details
BLAST	Basic Local Alignment Search Tool
BLOSUM	Blocks Substitution Matrices
CATH	Class, Architecture, Toplogy and Homologous superfamily
CATHEDRAL	CATH's Existing Recognition Algorithm
CE	Combinatorial Extension algorithm
CORA	Consensus Of Residue Attributes
EBI	European Bioinformatics Inistitute
FLORA	Functional Listing Of Residue Attributes
FSSP	Fold classification based on Structure-Structure alignment of Proteins
HMM	Hidden Markov Model
NCBI	National Center of Biotechnology Information
NMR	Nuclear Magnetic Resonance
NRDB	Non-Redundant DataBase
PDB	Protein Data Bank
PSI-BLAST	Position Specific Iterated-BLAST
PSSM	Position Specific Score Matrices
RMSD	Root Mean Squared Deviation
SCOP	Structural Classification Of Proteins
SSAP	Sequential Structural Alignment Program
SVM	Support Vector Machine

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 What are proteins?

Proteins comprise approximately 15% of our body mass and are fundamental to the majority of biological processes. Through the polymerisation of just 20 amino acids, these macromolecules perform a vast array of functions, from reaction catalysis to providing mechanical support within the cell. In addition, they are capable of forming complex interaction networks that govern both inter and intra-cellular signalling pathways and gene transcription. Key to the huge diversity of protein function is the subtly different ways in which polypeptide chains of a given sequence can fold into a unique three-dimensional structure. To fully understand how protein functions are achieved at the molecular level is one of the major goals of modern biology, as it would provide an unparalleled insight into the underlying mechanisms of development and disease. Furthermore, it could bring about a revolution in drug development through the rational design of molecules able to affect known disease-associated targets with a high degree of specificity.

1.1.1 Primary structure

The primary structure of a protein describes the sequence of amino acids along the polymer chain. All amino acids have a central C-α carbon attached to an amine group (NH_2), carboxyl group (COOH) and a hydrogen; but the distinguishing feature of each is the 'sidechain' group. Sidechains vary considerably in their physicochemical properties, but can be broadly grouped into three main classes: mainly hydrophobic, charged and polar (Figure 1.1). Glycine is the exception as its sidechain is simply a hydrogen atom, although it is sometimes classified as a hydrophobic residue. Polypeptide chains are synthesised on the ribosome in a condensation reaction between the carboxyl and amino termini to form the amide/peptide bond.

1.1.2 Secondary structure

Water-soluble, globular proteins are energetically driven to fold into their three-dimensional structure by the packing of mainly hydrophobic amino acids into the interior, leaving a surface of hydrophilic sidechains. The main chain polar N-H and C=O groups, which are similarly buried with the hydrophobic sidechains, are neutralised by the formation of hydrogen bonds. These often give rise to regular hydrogen bonding patterns to create secondary structure elements. The configuration of the amino acids units relative to one another in these elements can be described by the angles between the C-α, carbonyl carbon and amide nitrogen. Two angles, phi and psi denote the angles around the N—Cα bond and the Cα—carbonyl carbon bonds respectively. The two main types of secondary structure are the alpha-helix and the beta-sheet, although there are a number of less stable hydrogen bonded motifs observed in nature.

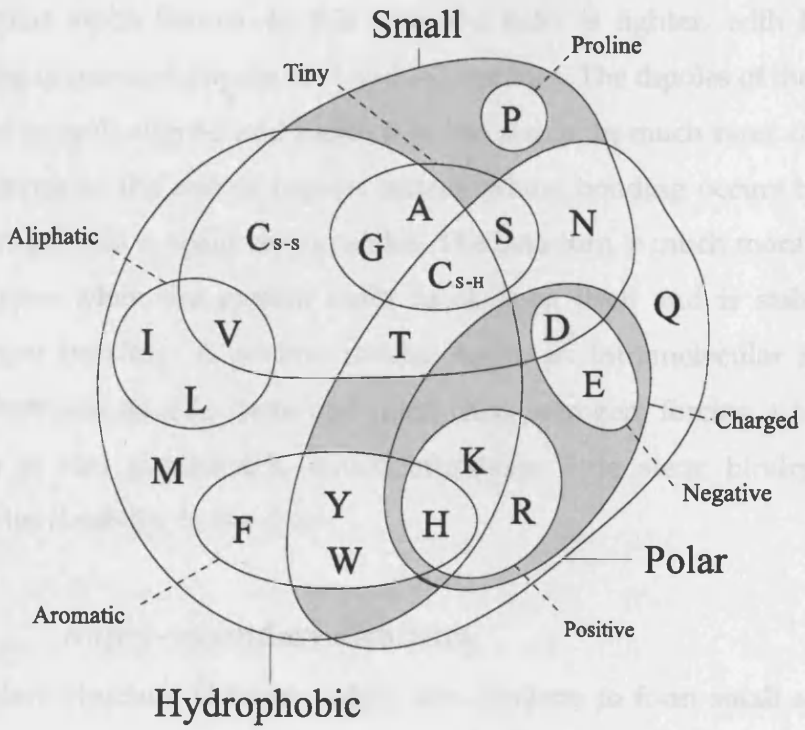


Figure 1.1 A Venn diagram describing the chemical and physical properties of amino acids (Taylor, 1986). The single letter code is translated in the abbreviations section.

In an alpha helix, the C=O group of residue i forms a hydrogen bond to the N—H of residue i+4, causing the chain to adopt a cylindrical helix structure with approximately 3.6 residues per turn. The helix is right-handed with psi and phi angles of -60 degrees and -50 respectively.

Beta sheets are made up of two or more continuous regions of beta-strand. Hydrogen bonds form in such a way as to allow the C=O and N-H groups of adjacent residues to bond to one another. Fully-formed beta sheets can be described as parallel, anti-parallel or mixed, depending on the bonding patterns between individual strands. Parallel sheets have average phi/psi angles of -119 and 133 respectively; anti-parallel, an average of -139 and 135.

In addition, other less stable and thus rarer secondary structure elements exist. The 3_{10} helices are invariably short and frequently found at the termini of regular alpha helices. In this case, the helix is tighter, with hydrogen bonding occurring between the i and i+3 residues. The dipoles of the 3_{10} helix are not so well aligned and hence it is less stable. In much rarer cases, a π-helix forms at the end of regular helices, where bonding occurs between i and i+5 but this is again very unstable. The beta-turn is much more common and arises when the protein chain turns upon itself and is stabilised by hydrogen bonding. A proline residue forms an intramolecular hydrogen bond between its side chain and main chain nitrogen, forcing a bend. The region is also glycine-rich, which introduces little steric hindrance and promotes flexibility in the chain.

1.1.3 Super-secondary structure

Secondary structure elements might also combine to form small secondary motifs or super-secondary structures. Some are associated with particular functions, such as DNA binding, whereas others are merely components of larger structural and functional assemblies.

β-hairpins consist of two adjacent anti-parallel strands, joined by looped region. They can occur either in isolation or as part of a more complex beta-sheet. Sibanda and Thornton (Sibanda and Thornton, 1991) showed that 70% of beta-hairpins are less than 7 residues in length, with the 'two-residue turns' being the most distinctive. Concatenation of several anti-parallel beta strands connected by beta-hairpins form a motif known as a beta meander.

The helix-turn-helix motif (EF hand) frequently has a specific functional role in binding calcium ions and was first discovered in parvalbumin where two of the three calciums interact in this way. The positive charge of the calcium is neutralised by the negative sidechain carboxyl groups and main chain carbonyl. Similarly, the helix-loop-helix motif is associated with DNA binding and found in proteins that control transcription, such as the Cro repressor in bacteriophage λ.

1.1.4 Tertiary structure

Secondary structure elements and larger motifs pack together to form the overall three-dimensional conformation or tertiary structure of a protein. A combination of electrostatic, Van de Waals forces, and covalent disulfide bonds act to stabilise the globular fold. This unit is often described as a domain.

1.1.5 Protein domains

Richardson (Richardson, 1981) described the protein domain as a semi-independent globular folding unit that formed the building blocks for larger, multi-domain chains. Structural domains are often defined by the observation that residue contacts are greater within the domain, than between other folding units. In addition, secondary structure elements (particularly beta-strands) are rarely shared between domains (Taylor 1999). As a consequence, connecting loop regions between domains can be sites for residue insertions, as they do not disrupt the overall fold of the protein.

1.1.6 Quaternary structure

Two or more protein chains can associate via electrostatic and covalent bonds to form oligomeric complexes, conferring a *quaternary* structure. These multimeric complexes further increase the functional repertoire of proteins and can also facilitate regulation, as these associations are often temporary or transient. This allows mechanisms such as allosteric control, where co-factors can modulate the shape of enzymatic sites and hence affect the reaction rate. In addition, new active sites can form at interfaces between chains, which allow a convenient way to build signalling networks and molecular machines (Liu and Eisenberg, 2002).

1.2 Evolution of protein domains

It is a widely accepted tenet of modern biology that organisms have evolved through a process of mutation and natural selection to produce the huge diversity of species we see in nature today. At the molecular level, it is the recombination and mutation of DNA that results in the myriad of proteins observed in the cell.

Proteins with similar structures and evidence of a common evolutionary ancestor are termed *homologues*. Despite often retaining the same function, they may differ significantly in their primary sequence as they have mutated independently from their parent ancestral gene. Identifying homologous relationships is often possible through comparative sequence analysis; however, protein structure is generally more conserved than protein sequence (Chothia and Lesk, 1986) and therefore structural similarities can be more informative when these data are available. Proteins that perform the same function in different organisms are termed *orthologues* (Figure 1.2).

When a gene duplicates, the new copy is not subject to the same evolutionary pressures as its parent gene and is potentially free to evolve a different

function. Genes related by this mechanism are termed *paralogues* (Figure 1.2). New functions can evolve through an extensive modification of functionally active regions of the protein structure, or through amino acid substitutions of key catalytic or substrate binding residues. As with homologue detection, paralogues can be identified through sequence similarities, although they tend to be more diverse and hence often require structural information.

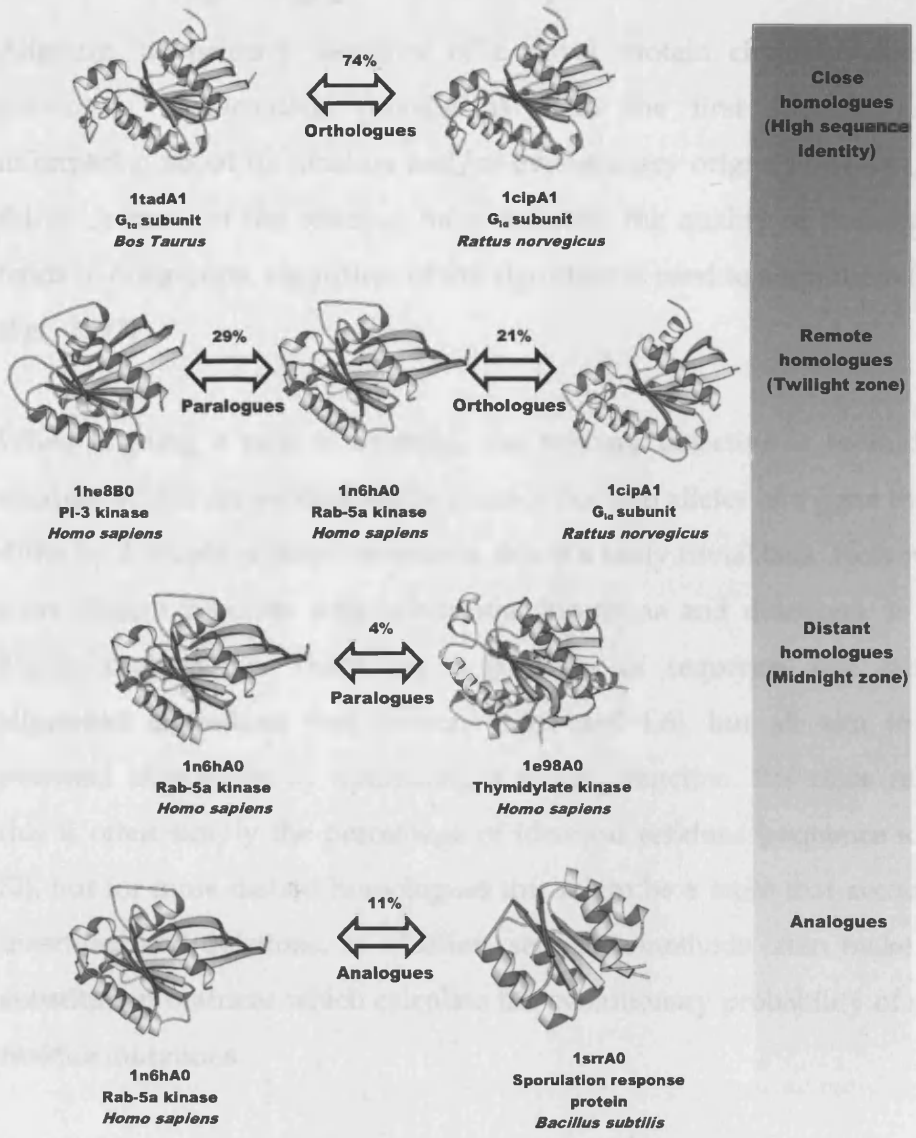


Figure 1.2 Schematic representation of the progression from close homologues, through more remote (twilight zone) (Doolittle, 1986) and very remote (midnight zone) (Rost, 1997) homologues and finally analogous/homologous structural relatives.

1.3 Aligning protein sequences

Aligning the primary sequence of a novel protein chain or domain to previously characterised proteins is often the first step in deriving information about its function and/or evolutionary origin. However, if two thirds or more of the residues have mutated, the quality of the alignment tends to deteriorate, regardless of the algorithm is used to align them (Martin *et al.*, 1997).

When aligning a pair of proteins, the primary objective is to find those residues which are evolutionarily related. For two alleles of a gene that only differ by a couple of point mutations, this is a fairly trivial task. However, for more distant relatives with substantial insertions and deletions, it can be highly problematic. There are a plethora of sequence and structural alignment algorithms (see Sections 1.3.3 and 1.6), but all aim to assess potential alignments by optimising a scoring function. For close relatives, this is often simply the percentage of identical residues (sequence identity, SI), but for more distant homologues this might be a score that accounts for insertions and deletions. In addition, sequence methods often make use of substitution matrices which calculate the evolutionary probability of specific residue mutations.

Alignment methods can usually be split into two types: local and global. The latter optimise equivalences across the entire length of two protein chains. This is useful when aligning two known homologues. However, for multi-domain chains that share only one common domain, a method which is biased towards local similarities (local alignment method) is more appropriate.

1.3.1 Substitution Matrices

1.3.1.1 Using physicochemical properties

As discussed in Section 1.1.1, amino acid residues can be grouped according

to shared chemical or physical properties. It is a reasonable assumption that the substitution of one residue for another is more likely to be tolerated in evolution if they possess similar characteristics. A mutation of leucine to valine, for example, is likely to have a minimal effect on the stability and function of a protein structure, as they share comparable hydrophobicity and molecular size.

1.3.1.2 Dayhoff or Point Accepted Mutation (PAM) Matrices

This approach of property comparison can be extended by calculating amino acid similarity based on an empirically-derived evolutionary method. By examining a large number of alignments of known relatives, substitutions probabilities between all 20 amino acids can be calculated and used to fill a mutation data matrix (MDM).

In the late 1970s, Margaret Dayhoff and co-workers used the sequences in their database of protein families to generate alignments of close evolutionary relatives (>85% sequence identity) (Dayhoff, 1978). The alignments were subject to a so-called global optimisation, where sequence identity was optimised to give maximum sequence identity across the whole protein sequence. The frequencies of residue substitutions were calculated and normalised so that each probability represented a residue substitution in an evolutionary period of one mutation every 100 residues.

1.3.1.3 The BLOcks SUbstitution Matrices (BLOSUM)

In a similar way, BLOSUM matrices are generated from regions of locally aligned sequences from the BLOCKS database (Henikoff and Henikoff, 1991). Proteins with a sequence identity greater than a given threshold are clustered together. Substitution values are calculated and used to populate a matrix, representing different evolutionary distances (e.g. BLOSUM50 clusters sequences at 50% identity). These matrices have been shown to be more effective in searching for homologous relationships than PAM matrices (Henikoff and Henikoff, 1993)

1.3.2 Methods for identifying conserved residues positions

Global sequence comparison methods seek to identify proteins showing significant sequence similarity and a high probability of being evolutionarily related and thus possessing similar functions. Many groups have developed more general algorithms to detect amino acid conservation across families in an effort to predict functional sites. Valdar and Thornton (Valdar and Thornton, 2001) developed the *ScoreCons* program to analyse and predict protein-protein interfaces. They calculated the diversity, or entropy, of amino acids at each position in a multiple alignment, quantified by using mutational matrices of evolutionary distance. This was then used to predict conserved residues that may be important for binding. A review of other methods can be found in Valdar (2002).

1.3.3 Protein sequence alignment methods

Proteins do not only evolve simply through substitutions: DNA recombination and the presence of transposable elements can also cause a gene sequence to expand or contract (insertions and deletions, indels). In the translated protein structure, these indels often occur in the loop regions connecting secondary structure elements as this is less likely to disrupt the overall stability of the fold; however, they are still able to modulate the ligand binding capabilities and hence, the function. When comparing the sequences of more diverse proteins, an alignment algorithm must be able to account for indels of varying lengths. An optimal alignment ought to consider every possible combination of residues, including potential indels. Nevertheless, this is computationally expensive and can become impractical when searching large databases.

1.3.3.1 Global Alignment

Needleman and Wunsch (Needleman and Wunsch, 1970) were the first to apply the dynamic programming algorithm to the field of bioinformatics —

it is still widely used today. The method begins by populating a matrix containing scores that reflect the similarity of all residues in protein A with those in protein B. The algorithm then starts at the bottom right hand corner to populate an accumulation matrix, as depicted in Figure 1.3. Each cell in this matrix takes the value of the scoring function S(*i,j*), which is determined by the values of previous cells to the below and to the right. It should be noted that if the value diagonally below S(i+1,j+1) is not selected, a gap penalty is invoked to penalise the introduction of a gap in the alignment. The final stage is to traceback through the matrix to determine the highest scoring path and hence the optimal alignment.

$$S(i,j) = S(i,j) + \max \begin{cases} S(i+1, j+1) \\ S(i+1, j+2..J) + G \\ S(i+2..I, j+1) + G \end{cases}$$

Where I = length of the row, J = length of column and G = gap penalty

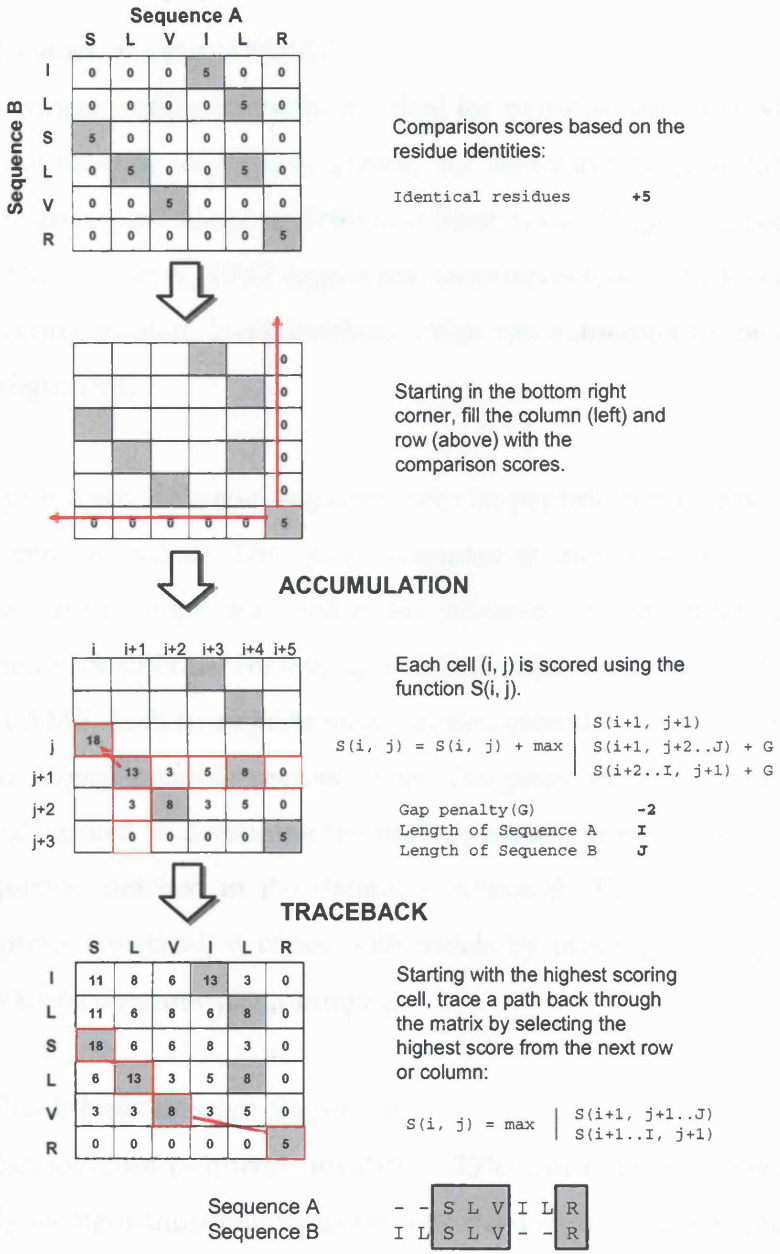


Figure 1.3 The Needleman and Wunsch dynamic programming algorithm. Each residue in sequence A and B is scored for similarity and these scores are used to populate a matrix. The accumulation step populates another matrix using the function S(i,j), where gaps are penalised. The final traceback step looks for the highest scoring path.

A modification of this algorithm was introduced by Smith and Waterman (1981) that focused on providing local, rather than global, alignments. When tracing back through the matrix, the paths can start anywhere and are terminated when the score falls below zero.

1.3.3.2 Local alignment and BLAST

Dynamic programming methods are ideal for pairwise sequence alignment, but computationally expensive. Hence, for searching large databases the FASTA (Pearson and Lipman, 1988) and Basic Local Alignment Search Tool (BLAST) (Altschul *et al.*, 1990) algorithms were developed, which concentrate on discovering smaller, local matches, which can subsequently be extended to a full alignment.

BLAST splits each database sequence into tri-peptide fragments (although this size can be varied). The query sequence is then searched against all fragments, with scope for mutations allowed by invoking BLOSUM substitution probabilities. For example, ACE is allowed to match ACE, GCE, GME and AME. Each tri-peptide match is then extended in both directions to create the largest possible segment pair. The pairs are scored, assigned E-values and ranked to determine the highest scoring segment pair (HSP) for each sequence matched in the database. Although BLAST is essentially a local alignment method, it copes with indels by refining the alignment of good hits using dynamic programming.

1.3.3.3 Profile-based sequence comparison

Remote homologues (sequence identity < 35%) can often be detected more effectively by algorithms that focus on conserved regions or sequence motifs. Protein motifs represent small, highly conserved stretches of contiguous sequence, which may be associated with a particular evolutionary family or biological function. Searching for these recurring 'fingerprints' is frequently successful in the twilight zone (Doolittle, 1986), where global pairwise sequence similarity becomes unreliable. In a more sophisticated way,

sequence profile methods, such as Hidden Markov Models (HMMs) (Eddy, 1996) and PSI-BLAST (Altschul *et al.*, 1997), have made it possible to capture the probability of certain residue mutations and insertions/deletions occurring across families of related proteins. These methods effectively measure the likelihood of finding a given amino acid or gap at each position in the alignment.

Not all residues in a protein are of equal evolutionary importance. Those that are critical for molecular interactions, catalysis or the stability of the fold are subject to far greater evolutionary conservation than the average residue. Hence, the actual probability of residue substitution is dependent both on the type of amino acid exchange and location in the three-dimensional (3D) structure.

By aligning a large family of protein sequences, one can observe specific amino acids which remain invariant despite substantial sequence diversity across the whole population. It follows that these residues are likely to have some functional or structural importance for the protein. By combining positional information with residue exchange probabilities, a Position Specific Score Matrix (PSSM) can be generated, which acts as a profile or sequence 'fingerprint' for the family.

PSI-BLAST (Altschul *et al.*, 1997) is an extension of the popular BLAST algorithm, which uses an iterative approach to refine a profile of the original query sequence. An initial BLAST database search is used to find close relatives, from which a multiple alignment can be built. A PSSM is then generated based on the residue propensities at each position in the multiple alignment. This is invoked to detect more remote homologues in subsequent searches of the database. The multiple alignment is then rebuilt and the PSSM refined. PSI-BLAST iterates through this process until no more relatives can be found below a given E-value cut-off.

HMMs (Karplus *et al.*, 2005; Eddy, 1996) have been shown to outperform PSI-BLAST (Park *et al.*, 1998) and are widely used by protein family databases, such as Pfam. HMMs implement a statistical framework which is based on state-transition probabilities in a multiple sequence alignment. A probability is calculated for each position for one of three states: match, delete or insert. The match state is further quantified by the distribution of residues at that position. Transition probabilities are calculated between all states and positions in the alignment. By traversing this probabilistic network, a distribution of residues is 'emitted' at each position to create the model (Figure 1.4). Each new sequence can be scored against the model and an E-value calculated. The most commonly used methods are SAM-T and HMMER.

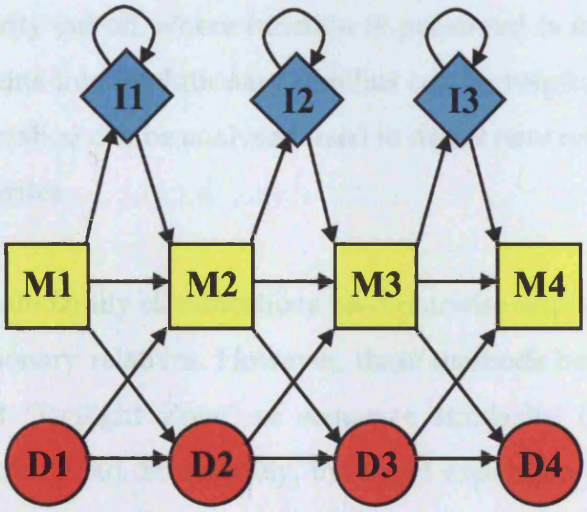


Figure 1.4 Overview of Hidden Markov Model (HMM), showing transition probabilities between match (M), delete (D) and insert (I) states.

1.4 Sequence Based Protein Family Classifications

Since the advent of genome sequence projects, the sequence repositories have always been several orders of magnitude larger than the structure databases.

There has, in fact, been an exponential increase in the sizes of both types of data since the early 1970s but the largest sequence database, GenBank (Benson *et al.*, 2006) still contains nearly three million non-redundant sequences (October 2006), compared to ~35000 protein entries in the Protein Data Bank (PDB) (Berman *et al.*, 2000) (see Section 1.5).

These data can be exploited to investigate the mechanisms of evolution and annotate novel genes with a putative function, based on their similarity to experimentally characterised proteins. The two areas of research are intimately linked as more effective annotation can be achieved through an understanding of how differences between genes affect their function. Small mutations can inactivate an enzyme's ability to catalyse a reaction; yet a given enzyme can exhibit large sequence diversity across different organisms and still retain its primary role in the cell. Hence, finding a universal sequence similarity cut-off where function is preserved is impossible. Thus classifying proteins into evolutionary families can be helpful, as patterns of sequence conservation can be analysed used to detect new relatives and infer functional properties.

The earliest protein family classifications used pairwise sequence comparison to detect evolutionary relatives. However, these methods become unreliable in the so-called 'Twilight Zone' of sequence similarity (<30% sequence identity) (Doolittle, 1990). Fortunately, the rapid expansion of the sequence databases over that past ten years has increased the populations of the protein families, enabling the derivation of family-based sequence profiles and motifs.

Despite the success of the new profile methods (e.g. PSI-BLAST, HMMs), very distant homologues can still be undetectable at low error rates. However, members that share significant sequence similarity may possess similar or identical biological functions. Many resources choose to cluster whole protein chains. However, databases such as Pfam (Bateman *et al.*,

2002) now identify separate domains within genes (often defined using protein structure data) and group them accordingly. Thus, one gene may comprise several domains that are members of different protein families. In reviewing the databases below, the distinction between those which simply cluster whole protein chains and those which focus on the domain level is highlighted.

Table 1.1 summarises the current populations of the major sequence family databases and the methodologies used to create them. An important recent development has been the establishment of the Integrated Resource of Protein Families, Domains and Sites (InterPro) Database (Apweiler *et al.*, 2001) at the EBI. This resource integrates all the major protein family classifications and provides regular mappings from these family resources onto primary sequences in the UniProt database (Apweiler *et al.*, 2004) which contains over 3 million sequences as of July 2005. InterPro is a collaboration that aims to provide an integrated interface of protein signature databases. Databases in the collaboration include UniProt, PROSITE (Hulo *et al.*, 2004), PRINTS (Attwood *et al.*, 2003), Pfam (Bateman *et al.*, 2004), ProDom (Corpet *et al.*, 1998) , SMART (Ponting *et al.*, 1999), TIGRFAMs, PIR SuperFamily, SUPERFAMILY (Gough, 2002) and Gene3D (Buchan *et al.*, 2002).

Resource	Group	Source(s)	No. Families	Method	URL
PRINTS	Zygouri	SWISSPROT, TrEMBL	1800 entries, 10,931 motifs	Iterative motif searches	http://www.bioinf.man.ac.uk/dbbrowser/PRINTS/
Pfam	Eddy	SWISSPROT, TrEMBL	7459 families	HMM	www.sanger.ac.uk/Software/Pfam
SMART	Bork	Selected proteins	667 domains	HMM	http://www.smart.embl-heidelberg.de
ProDom	Kahn	SWISSPROT, TrEMBL	501,917 families, (186,303 non-singleton)	PSI-BLAST	http://www.protein.toulouse.inra.fr/prodom/current/html/home.php
InterPro	Zbobnov	UniProt, PROSITE, PRINTS, Pfam, ProDom, SMART, TIGRFAMs, PIR SuperFamily, SUPERFAMILY	11,007 entries (including 2573 domains, 8166 families)	Multiple methods (HMM, PSI-BLAST, Regular Expression)	http://www.ebi.ac.uk/interpro
TIGRFAMs	White	SWISSPROT, TrEMBL	1976 families	HMM	http://www.tigr.org/TIGRFAMs/index.shtml
ADDA	Holm	SWISSPROT, TrEMBL, PIR, PDB, WORMPEP, ENSEMBL	34,000 families (plus 60,000 singleton)		http://ekhidna.biocenter.helsinki.fi:8080/examples/servlets/adda/index.html
CHOP	Rost	62 complete genomes	63,300 clusters (plus 118,108 singleton clusters)	PSI-BLAST	http://cubic.bioc.columbia.edu/services/CHOP

Resource	Group	Source(s)	No. Families	Method	URL
TRIBES	Ouzounis	83 Complete Genomes	60,934 or 82,692 depending on granularity	TribeMCL	http://maine.ebi.ac.uk:8000/services/tribes
ProtoNet	Linial	SWISSPROT, TrEMBL	User-defined	BLAST	http://www.protonet.huji.ac.il
SYSTERS	Vingron	SWISSPROT, TrEMBL, ENSEMBL (complete genomes), the Arabidopsis Information Resource, SGD and GeneDB	158,153 disjoint clusters	BLAST	http://systers.molgen.mpg.de
SWISSPROT	Schneider	Primary database	153,871 proteins	N/A	http://us.expasy.org/sprot
COG/KOG	Natale	66 unicelluar and 7 eukaryotic complete genomes	4873 COG, 4852 KOG	Bidirectional best hit	http://www.ncbi.nlm.nih.gov/COG

Table 1.1 Protein family resources (compiled July 2005)

1.4.1 Families of sequence domains

Pfam (Bateman *et al.*, 2004) is a highly comprehensive resource providing an optimised set of Hidden Markov Model profiles for protein domain families. Families are defined using multiple sequence alignments and HMMs which cover many common protein domains and families. Pfam consists of two parts, the first is the curated part of Pfam (Pfam-A), the second is an automatically generated supplement called Pfam-B.

Similarly, the Simple Modular Architecture Research Tool (Ponting *et al.*,

1999) (SMART) domain families are selected with a particular emphasis on mobile eukaryotic domains and as such are widely found among nuclear, signalling and extracellular proteins. SMART domain families are annotated with function, sub-cellular localization, phylogenetic distribution and tertiary structure.

COG and KOG (Tatusov *et al.*, 2003) are databases of clusters of orthologous groups of proteins, defined by groups of three or more proteins in complete genomes. KOG contains 7 eukaryotic genomes whilst COG contains 66 complete unicellular genomes.

1.4.2 Families of whole protein chain sequences

TIGRFAMs protein families are built in a similar fashion to Pfam but also contain whole protein chains. ProtoNet developed by Linial and co-workers (Sasson *et al.*, 2003), uses three clustering methods (harmonic, geometric and arithmetic) to group sequences in the UniProt database on the basis of their similarity. Likewise, the SYSTERS (Krause *et al.*, 2000) and TRIBES (Enright *et al.*, 2003) methods make use of graph-based methods and Markov clustering respectively to generate protein families of varying granularity.

The PRINTS database (Attwood *et al.*, 2003) is a collection of protein 'fingerprints' — conserved sequence motifs used to characterise a protein family. These motifs are generated via multiple protein sequence alignments by identifying regions of local sequence conservation. They can subsequently be used to scan a larger sequence set (e.g. UniProt (Apweiler *et al.*, 2004)) to recruit new family members. The majority of families are defined by multiple motifs and all must be present for a relative to be added to the group.

A number of other resources exist that automatically cluster sequences from the completed genomes or from the large sequence repositories (e.g. GenBank or UniProt) into putative domain families. The ProDom resource (Corpet *et al.*, 1998) contains protein sequence families derived from

sequences in UniProt. These protein sequences are chopped into protein domains using an iterative PSI-BLAST domain boundary prediction algorithm and have been used to seed the majority of Pfam families.

Holm and co-workers developed the ADDA algorithm to cluster sequences into domain families (Heger *et al.*, 2005), which takes alignments from all-against-all sequence comparison to define domains within protein sequences and cluster them into families. Recently, almost 800,000 non-redundant sequences were condensed into 100,000 domain families (33% of the families containing more than one member) covering all of the currently available sequence space. A related algorithm, CHOP (Liu and Rost, 2004) designed by Rost and co-workers, assigns domain boundaries by BLAST sequence comparison and then clusters the subsequent domain-like fragments into sequence families using the CLUP clustering method. 62 completed genomes were chopped and clustered into 118,108 single and 63,300 multi-member clusters. Gene3D (Yeats *et al.*, 2006) clusters families at a range of sequence identities and now contains over 2000 domain superfamilies.

There are an ever-increasing number of web-accessible classifications of protein sequence families (see Table 1.1). The number of families identified by those performing automated clustering of large sequence repositories varies from 65,000 to 186,000 depending on the algorithm used. Ouzounis and co-workers revealed that each newly sequenced genome leads to an increase in the total number of protein families characterised (Janssen *et al.*, 2003). That is, currently a certain proportion of genome sequences (between 10 and 25%) in every genome are singletons, or belong to families not present in other sequenced genomes. This may reflect limitations in the current sequence-based homologue detection algorithms; or alternatively these may be genuinely novel families that have arisen following speciation. The organism-specific families may be important for expanding the functional repertoire and phenotype of the organism, perhaps by providing unique biological processes or changes in gene regulation.

1.5 The Protein Data Bank (PDB) and Macromolecular Structure Database (MSD)

The Protein Data Bank (PDB) (Berman *et al.*, 2000) was established in 1971 as a worldwide repository for the three-dimensional structures of proteins and nucleic acids. It contains structures solved using experimental techniques, such as X-ray crystallography, nuclear magnetic resonance (NMR) and cryoelectron microscopy. PDB files hold standardised coordinate data for atoms in the structures. However, other data is non-standardised, with many of the fields accepting free text of the author's choice.

The Macromolecular Structure Database (MSD) (Velankar et al., 2005) is curated at the European Bioinformatics Institute (EBI) in Cambridge and is also a database of macromolecular structures. However, unlike the PDB, which was designed as a data bank, the MSD focuses on providing a more rigid framework for data and optimising automatic searching. Manual intervention has been employed to correct errors such as spelling mistakes and the consistent nomenclature of amino acids residues and other chemical groups.

1.6 Aligning protein structures

As two proteins diverge from a common ancestor, their sequences can change beyond recognition. However, their three-dimensional structures usually remain similar. This was originally demonstrated in 1986 by Chothia and Lesk who plotted sequence similarity against structural similarity for homologues in the PDB (Chothia and Lesk, 1986). A more recent analysis of several hundred well populated superfamilies in the CATH database, containing three or more sequence families, confirmed that even in very remote relatives (<20% sequence identity) at least 50% of the structure

remains conserved (Reeves *et al.*, 2006). The most highly conserved positions usually correspond to residues in secondary structures in the buried core of the protein.

Computational structure comparison methods were introduced in the 1970s, shortly after the advent of the PDB. Although they can be used to align entire multi-domain chains, it is often useful to separate proteins into their constituent domains, as the connectivity and orientation of domains can vary widely and this can have negative effects on the quality of the structural alignment.

There are well over 50 different structure comparison algorithms cited to date but most are variations on a number of techniques. In general, the alignment is determined in two stages: a measure of similarity of residues and/or secondary structure features between both proteins is calculated and then an optimisation strategy is employed to find an alignment that maximises the score of aligned positions. The majority of methods use the geometric properties of C_α or C_β atoms and/or secondary structure information, such as distances or intramolecular vectors. Physicochemical properties, such as hydrophobicity, hydrogen bonding and solvent accessibility are also sometimes used to identify equivalent residues (Orengo and Taylor, 1993).

1.6.1 Calculating Structural Similarity

Irrespective of the method used to align two protein structures, a transformation matrix can be calculated to superpose them in the same co-ordinate space. If a quantitative measure of similarity is required, the most widely used is the Root Mean Square Deviation (RMSD). This is simply the square root of the average squared distance between equivalent atoms ((Equation 1.1).

$$RMSD = \sqrt{\frac{\sum_{i=1}^{N} d_i^2}{N}}$$ (Equation 1.1 Root Mean Square Deviation (RMSD))

Similar protein folds tend to give an RMSD below 4.0; however, this can be higher for the folds of distant relatives with more than 400 residues. Moreover, superposing the same protein with and without its bound ligands, can also result in a large RMSD if there is a sizeable conformational change during binding (Grindley *et al.*, 1993). This makes it highly sensitive to hinge movements between two domains and this highlights the main problem with using RMSD as a measure of similarity: namely, that it is dependent on the number of aligned positions. It is therefore important to consider both the RMSD and the number of equivalent residue pairs when assessing the significance of the similarity. Despite its limitations, RMSD remains a widely used and valuable measure.

1.6.2 Rigid body superposition methods

It is possible to treat two protein structures as rigid objects and simply find the best way of minimising the distance between them when superposing one on top of the other. It should be noted that this is distinct from structural alignment, which maps equivalent residues between two proteins. Rigid body superposition was the rationale of the methods pioneered by Rossman and Argos in the 1970s and can be thought of in three stages:

1. Moving both structures to a common position in the co-ordinate frame, usually by translating their centre of mass to the origin.
2. Finding putative equivalent positions to start the optimisation.
3. Rotating one protein, relative to the other, around to three major axes to look for the "best fit" (i.e. giving the lowest RMSD).

The major difficulty with this method lies in identifying putative equivalent positions to begin the optimisation and reduce the search space. For close relatives (>35% sequence identity), standard sequence alignment methods can be used. However, for more distantly related proteins, this is unreliable and the algorithm often requires manual input to define known equivalent residues, such as catalytic residues in the active site.

Therefore, rigid body superposition is generally only used to compare closely related proteins, or to superpose structures once alternative algorithms with the ability to handle extensive insertions and deletions have determined equivalent positions.

1.6.3 Secondary Structure Based Methods

One approach to handling insertions and deletions (indels) in distant homologues is simply to compare the secondary structures, as a large proportion of indels occur in the loops connecting secondary structures. Graph theoretical methods (Grindley *et al.*, 1993; Artymiuk *et al.*, 1994; Harrison *et al.*, 2003) tend to dominate this approach to structure comparison, as they are both fast and effective. The majority concentrate on the distances and angles between secondary structures in both proteins, which are then compared to find equivalent pairs.

1.6.3.1 GRATH, SSM

Graph theory is a comprehensive branch of mathematics that has been applied to many different areas of biology and computer science. A graph consists of points, *nodes*, in two-dimensional space connected by lines, *edges*, which describe the relationship between them. A protein structure can be reduced to a graph where the nodes are secondary structures and the edges describe the geometric relationships between them (e.g. distances, angles). Grindley and co-workers (Grindley *et al.*, 1993), were the first to use these techniques in 1993, although Harrison *et al.* (2002; Harrison *et al.*, 2003) have applied them more recently to detect fold similarities as part of the

classification procedure in the CATH database (Orengo *et al.*, 1997) (GRATH).

In the GRATH method, linear vectors are used to represent the secondary structures and the edges are then labelled with distances between the midpoints and angles, describing the tilt and rotation between the vectors (Figure 1.5). The resulting two protein graphs are then evaluated to detect common secondary structure 'cliques or complete sub-graphs', by identifying equivalent edges that are labelled with similar distances and angles (Harrison *et al.*, 2003). This forms the basis of the correspondence graph, where each node represents two secondary structures (one from each protein) and edges are constructed where their angles and distances are within prescribed cut-offs. The Bron-Kerbosch method (Bron and Kerbosch, 1973) is then used to detect the common secondary clique. The algorithm operates in a recursive fashion by gradually eliminating nodes that do not have sufficient edges, until the clique is found.

Krissinel and co-workers (Krissinel and Henrick, 2004) have optimised a sub-graph matching algorithm, on which they base their SSM method. Much like GRATH, it labels edges with distances and angles to determine equivalent relationships (Figure 1.5). However, a greater emphasis is placed on the similarity between the sizes of secondary structures, a feature which was explicitly found to be unhelpful by Harrison et al. (Harrison *et al.*, 2003). The major difference is that SSM does not search for fully-connected cliques. This is compensated for by also examining equivalent connectivity, i.e. matched secondary structures must be in the same order along the protein chain.

Methods based on secondary structure matching are extremely fast at searching databases of protein folds (particularly for proteins that contain < 20 secondary structures elements) and very effective at identifying distant fold similarities. They are often used to find putative structural relatives,

which can then be aligned more accurately to the query structure using residue-based methods.

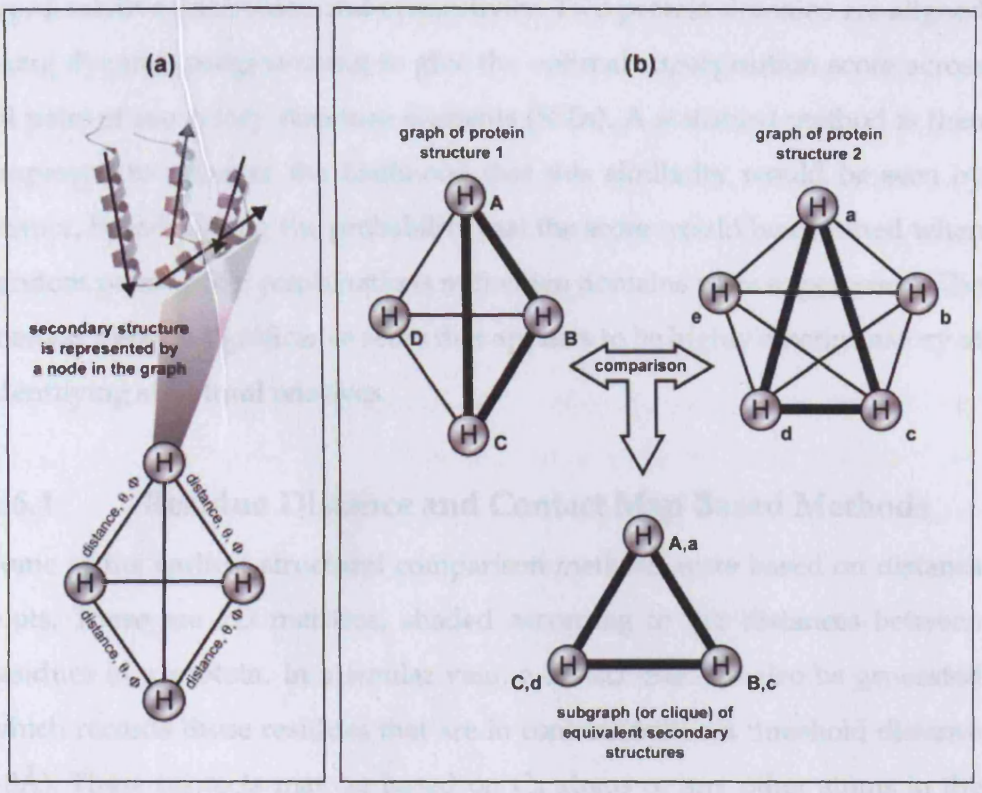


Figure 1.5 Illustration of graph theory-based structure comparison algorithms. a) Linear vectors are calculated through each secondary structure and used to represent each node in a graph. The relationships between these vectors (e.g. angles and midpoint distances) then annotate the edges between them. b) Two protein graphs are compared by looking for equivalent edges (highlighted in bold). Whereas SSM looks only for common sub-graphs, GRATH looks for fully-connected cliques. The resulting secondary structure graphs can represent a common topology shared by the two protein domains.

1.6.3.2 VAST

Entrez at the NCBI provides a web resource of structural alignments and superpositions of around 10,000 domain substructures within the PDB using the VAST (Vector Alignment Search Tool) algorithm (Madej *et al.*, 1995). In a similar way to graph theory methods, VAST focuses on the relationship between secondary structures. The authors define "units" of similar tertiary

structure as pairs of secondary structure elements that share equivalent types, relative orientation and connectivity. Two protein domains are aligned using dynamic programming to give the optimal superposition score across all pairs of secondary structure elements (SSEs). A statistical method is then employed to measure the likelihood that this similarity would be seen by chance, by calculating the probability that the score would be obtained when random pairs of SSE combinations in the two domains were superposed. The method yields a significance score that appears to be highly discriminatory at identifying structural relatives.

1.6.4 Residue Distance and Contact Map Based Methods

Some of the earliest structural comparison methods were based on distance plots. These are 2D matrices, shaded according to the distances between residues in a protein. In a similar vein, a *contact map* can also be generated which records those residues that are in contact (within a threshold distance ~8Å). These contacts may be based on C_α atoms or any other atoms in the residue side chains. The patterns arising in the resulting matrix are often characteristic of a particular protein fold. For example, dense stretches of contacts indicate closely packed secondary structures. Protein structures can be aligned by overlaying their contact maps. However, as with rigid body methods, it is difficult to overlay the maps of distant homologues; although some strategies have been developed to cope with indels, which are described below.

1.6.4.1 DALI and CE

One approach to aligning distant structural relatives is to divide each protein into fragments. The Combinatorial Extension (CE) algorithm (Shindyalov and Bourne, 1998b) and DALI (Holm and Sander, 1993) are popular examples of methods that discover equivalent fragments and subsequently combining them to calculate a global alignment, using some manner of optimisation strategy.

Holm and Sander developed the DALI algorithm (Holm and Sander, 1993), which fragments protein structures into hexapeptides and compares their contact maps (Figure 1.6). Potentially equivalent fragments are identified by looking for similar patterns of distances between residues, within a specific threshold. These pairs are then concatenated to extend the alignment using a Monte Carlo optimisation. An RMSD value is calculated to assess the quality of the extension as the concatenation progresses.

In a similar way, CE fragments the polypeptide chain into octapeptides and aligns residues based on the characteristics of their local geometry (as defined by vectors between C_α positions). Matching fragments are termed Aligned Fragment Pairs (AFPs). Heuristics are used to define a set of optimal paths joining AFPs, with gaps inserted as required. The pairs with the best RMSD are subjected to dynamic programming to achieve an optimal alignment. For specific families of diverse proteins, additional characteristics are used to weight the alignment.

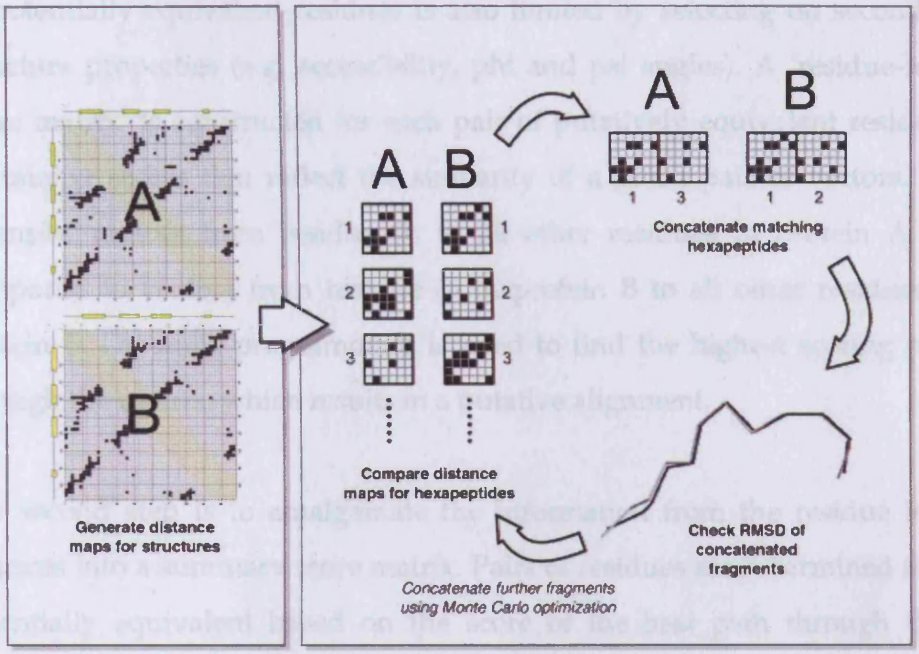


Figure 1.6 The DALI method of Holm and Sander(1993). Proteins are fragmented into hexapeptides and their contact maps compared to find equivalent fragments. Fragments are concatenated and their RMSD checked to find valid extensions. Monte Carlo optimisation is used to guide the extension process to a full alignment.

1.6.4.2 SSAP

Another approach to comparing distances between residues was developed by Taylor and Orengo (Taylor and Orengo, 1989). They sought to deal with the structural embellishments observed between distant relatives by applying the dynamic programming techniques used in sequence alignment methods. In the SSAP algorithm, dynamic programming is in fact utilized twice; firstly to compare residue environments and secondly to determine the optimal global alignment (Figure 1.7).

At the heart of the comparison lies the concept of 'residues views'. These are vectors calculated between a specific C_β (side chain carbon) atom and all C_β atoms within a structure. The vectors are compared between the two proteins

by a score based on the magnitude of the vector between them. The number of potentially equivalent residues is also limited by selecting on secondary structure properties (e.g. accessibility, phi and psi angles). A 'residue-level score matrix' is constructed for each pair of putatively equivalent residues, containing scores that reflect the similarity of a given pair of vectors. For example, vectors from residue (i) to all other residues in protein A are compared to vectors from residue (j) in protein B to all other residues in protein B. Dynamic programming is used to find the highest scoring path through the matrix, which results in a putative alignment.

The second step is to amalgamate the information from the residue level matrices into a summary score matrix. Pairs of residues are determined to be potentially equivalent based on the score of the best path through their residue level matrix. All optimal paths returning scores above a given threshold are collated in the summary matrix and an overall optimal path calculated using dynamic programming.

The SSAP algorithm has historically been used to classify domains in the CATH database. In keeping with the idea of vector comparison, SSAP bases its primary scoring scheme on an average of the vector environment similarity of equivalent residues.

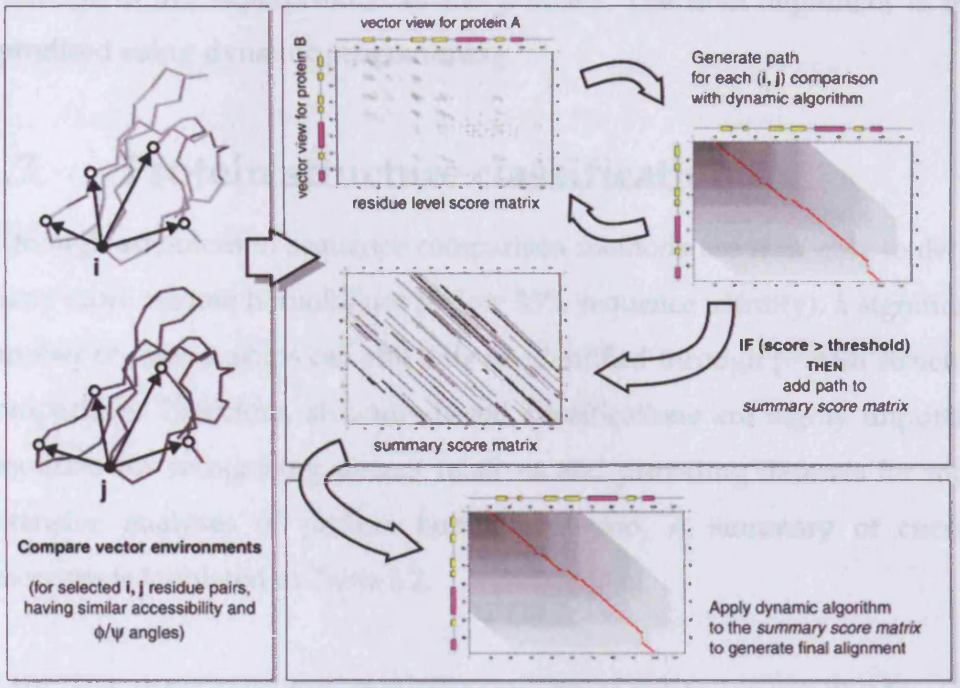


Figure 1.7 Flowchart of the SSAP algorithm. Vector environments are compared between pairs of potentially equivalent residues in each protein. A residue level score matrix is constructed for each pair and optimal paths (putative alignments) are calculated by dynamic programming. High scoring paths are then added to the summary score matrix. Dynamic programming is then applied to the summary matrix to generate the final optimal alignment of the two structures.

1.6.4.3 COMPARER

COMPARER (Sali and Blundell, 1990) uses intermolecular superposition and then subsequently assesses relationships between residues within each structure. Residue properties, such as secondary structure type, side-chain orientations and torsional angles are then compared between proteins and used to populate a 2D matrix. These are combined with intramolecular information (C_α distances, hydrogen bonding patterns, distances to the protein's centre of mass) to find equivalent residues. Putative equivalences are optimised by rigid body superposition followed by a technique known as simulated annealing. This applies a probabilistic Boltzmann energy function, which calculates drops in energy as temperature decreases to find optimal

solutions to the superposition of the proteins. The final alignment is then optimised using dynamic programming.

1.7 Protein structure classification

Although advances in sequence comparison methods are now able to detect many more remote homologues (below 35% sequence identity), a significant number of relationships can still only be identified through protein structure comparison. Therefore, structure-based classifications are highly important resources for recognising distant relatives and providing datasets for more extensive analyses of protein family evolution. A summary of current resources is tabulated in Table 1.2.

Since 1994, there have been two major structural databases, SCOP (Murzin *et al.*, 1995) and CATH (Orengo *et al.*, 1997), which group protein domains into evolutionary superfamilies. Domains are further classified under a hierarchy, the top level of which corresponds to the protein class – the proportion of residues adopting α-helical or β-strand conformations. This gives rise to three major classes, mainly-α, mainly-β and α-β, although SCOP divides the alpha-beta class into alternating α/β and α+β, depending on the segregation of α-helices and β-strands along the polypeptide chain.

1.7.1 SCOP

The Structural Classification of Proteins (SCOP) database was established in 1995 by Murzin and co-workers (1995) and uses almost entirely manual validation for recognising structural similarities between proteins to generate evolutionary superfamilies. Although time consuming, this has resulted in a very high quality resource where domain boundaries are also manually assigned. Domains are further clustered at the family level if they share greater than 30% sequence identity, or have a close structural or functional relative.

1.7.2 CATH

CATH is an acronym of its hierarchy of: Class, Architecture, Topology, Homologous superfamily Figure 1.8. It uses a combination of manual and automated approaches. Robust structure comparison methods (SSAP, CORA, GRATH) have been developed to recognise structural relatives; although evolutionary relationships are only assigned following manual assessment of all available data. Several automatic methods are used for domain boundary recognition but, again, assignments are all manually validated.

Within each of the three protein classes defined in CATH, structures are grouped by architecture, which describes the overall arrangement of secondary structures. For example, the ubiquitous three-layer α-β-α sandwich which is defined by a core β-sheet surrounded by two alpha helical regions. The topology (or fold) level further delineates domains by the different connectivities of their secondary structure elements. Significant structure similarity, often identified through structural comparison, must be in evidence for domains to share the same fold level.

Finally, proteins are only grouped at the superfamily (H) level where there is additional evidence of an evolutionary relationship (e.g. high structural/sequence similarity or comparable functions). Two of the following criteria must be met:

1. Similar structures (SSAP score > 80) with at least 60%overlapping residues.
2. Similar sequence (> 35% identity or significant HMM E-value).
3. Functional similarity (e.g. sharing of first 3 E.C. numbers).

Version 2.6 of the CATH database contained 67, 054 domains in 1572 superfamilies, 907 folds and 39 unique architectures. Within each superfamily, proteins are further sub-clustered by sequence identity into families of close relatives (e.g. > 35%) (Figure 1.8). The vast majority of

structures in the same sequence family (S35 group) share very high structural and functional similarity. As such, datasets of domains can be reduced in size by only taking one representative from each S35 cluster – this is termed the SRep.

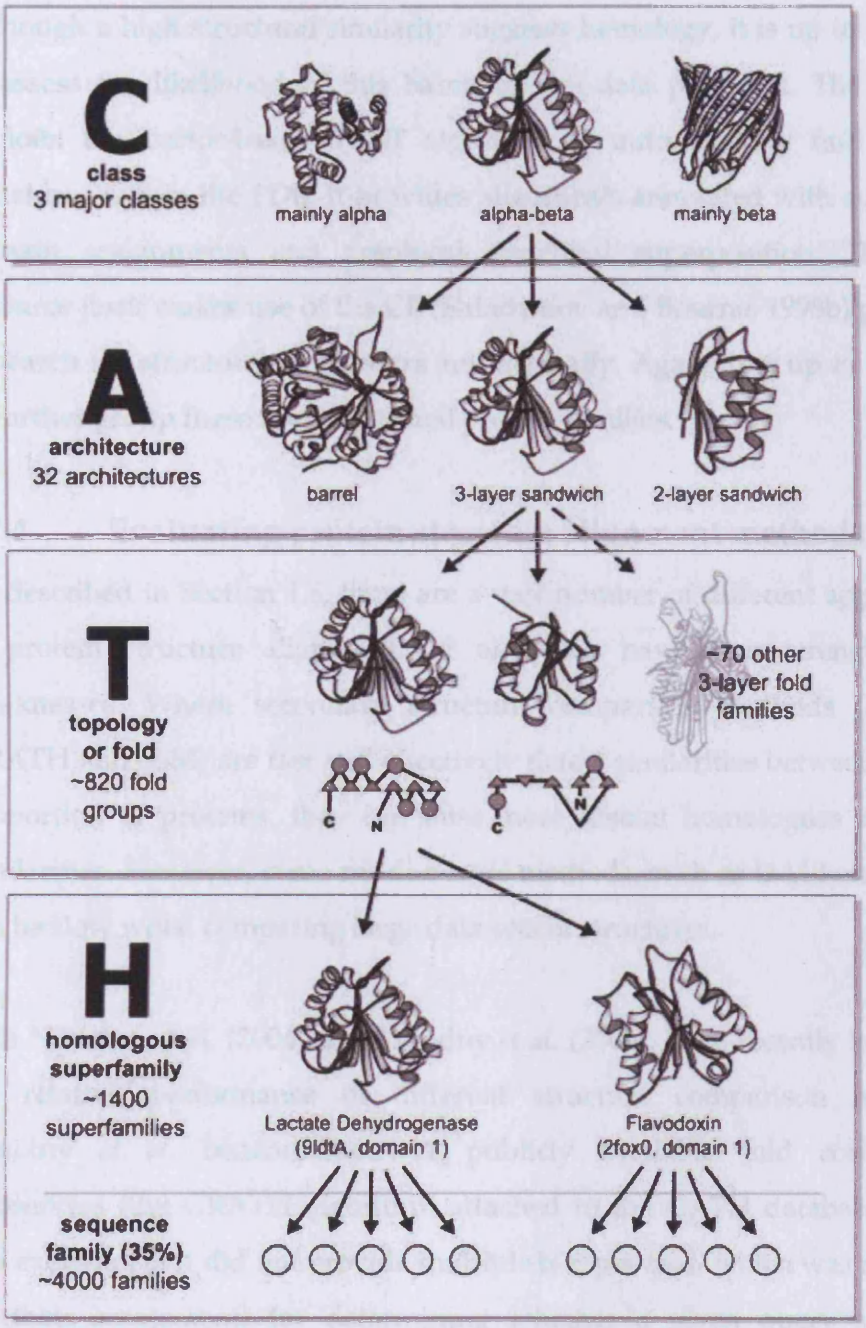


Figure 1.8 Diagram of the CATH hierarchy

1.7.3 Other structural resources

In addition to hierarchical classifications, there are several online resources (e.g. FSSP (Holm and Sander, 1997), MMDB (Marchler-Bauer *et al.*, 1999)) that provide lists of structural neighbours for a given query. FSSP provides a search tool that exploits the DALI algorithm to find structural relatives. Although a high structural similarity suggests homology, it is up to the user to assess the likelihood of this based on the data provided. The MMDB exploits the vector-based VAST algorithm to automatically find similar structures within the PDB. It provides alignments annotated with automatic domain assignments and graphical structural superposition. The PDB resource itself makes use of the CE (Shindyalov and Bourne, 1998b) program to search for structural neighbours automatically. Again, it is up to the user to further group these into individual protein families.

1.7.4 Evaluating protein structure alignment methods

As described in Section 1.6, there are a vast number of different approaches to protein structure alignment, all of which have their strengths and weaknesses. Where secondary structure comparison methods (such as GRATH and SSM) are fast and effectively detect similarities between a large proportion of proteins, they can miss more distant homologues and fold similarities. However, some residue-level methods, such as DALI and SSAP, can be slow when comparing large data sets of structures.

Both Novotny *et al.* (2004) and Kolodny *et al.* (2005), have recently looked at the relative performance of different structure comparison methods. Novotny *et al.* benchmarked 11 publicly available fold comparison webservers (the GRATH algorithm attached to the CATH database server was excluded at it did not provide multiple hits per fold, which was required for their assessment) for determining whether a given query structure represented a novel fold, according to the CATH classification. The authors concluded that CE, DALI and VAST performed well for detecting similar

folds, although recommended using a combination of algorithms to be able to confidently assume that a given query structure had a novel fold.

Kolodny *et al.* (2005) took a subset of structure comparison methods to align representative domains from the CATH database with 6 different structure comparison methods, including SSAP. The authors compared the native structural similarity scoring schemes with their own geometric scores based on the RMSD upon superposition of aligned residues. Interestingly, they concluded that SSAP, among other methods, performed better in Receiver-Operator Curve (ROC) analysis of all domain pairs related at the fold level when using their geometric scores, rather than the native scoring of a given method. However, it was also suggested that ROC curve analysis with respect to CATH could unfairly penalise methods that detect structural similarities between domains in different folds.

To further evaluate the comparative performance of each method, Kolodny *et al.* examined the fraction of all same-fold domain pairs that were aligned with a transformed RMSD score (SAS) below a given cut-off. By this analysis, SSAP is judged to perform poorly compared with the other structural alignment methods. However, the authors note that when greater emphasis is given to the number of aligned residues, SSAP is the best performing algorithms, despite fairing worse than STRUCTAL in their other benchmarks.

It could be argued that one of the problems with the authors' conclusions is that a correct structural alignment cannot purely be judged on the geometric superposition score. Although it is important for a given structure comparison method to provide a score that performs well for identifying fold similarities and homologous relationships, it is also designed to produce a biologically meaningful alignment. As the SSAP algorithm is used extensively in this thesis, its relative performance for generating structural alignments and scoring structural similarity will be assessed and revised in

Summary table of protein structure databases and structural comparison algorithms

DATABASE	LOCATION AND AUTHOR	COVERAGE	STRUCTURE COMPARISON METHOD	TYPE	DESCRIPTION
CAMPASS	Cambridge University, UK *Sowdhamini*	7580 domains in 1409 superfamilies	COMPARER (Sali and Blundell 1990), SEA (Rufino and Blundell, 1994)	Structure-based sequence alignments of SCOP superfamilies.	CAMbridge database of Protein Alignment organised as Structural Superfamilies. Provides sequence alignments of structural domains within a superfamily.
CATH Gene3D	UCL, London, UK *Orengo*	58,000 domains in 1459 superfamilies	SSAP (Taylor and Orengo 1989), GRATH (Harrison *et al.*, 2002)	Automatic structural and sequence comparison methods are combined with manual validation of superfamily alignments and domain boundaries.	CATH is a hierarchical classification of protein domains structures, clustered by Class, Architecture, Topology and Homologous Superfamily.
CE	SDSC, La Jolla, CA, USA *Bourne*	All chains in PDB	CE (Shindyalov and Bourne 1998)	Fully automatic. Nearest neighbours.	Combinatorial Extension of the optimal path. A database of structural alignments and similarities between all

					structures in the PDB.
DHS	UCL, London UK	1459 superfamilies in CATH	SSAP (Taylor and Orengo, 1989) CORA (Orengo 1999)	Fully automatic multiple structure alignments of close relatives in CATH superfamilies.	Dictionary of Homologous Superfamilies. Multiple structure alignments of homologous domains as defined by superfamilies in the CATH database. These are further annotated with functional information from UniProt, ENZYME, GO, KEGG.
ENTREZ/MMDB	NCBI, Bethesda, MD, USA *Bryant*	All in PDB	VAST (Madej *et al.*, 1995)	Fully automatic. Nearest Neighbours	MMDB contains pre-calculated pairwise structural comparisons and alignment between all structures in the PDB.
HOMSTRAD	Cambridge University, UK *Blundell*	7500 domains in over 1400 superfamilies	COMPARER (Sali and Blundell, 1990)	Manual classification of close protein homologues	HOMologous STRucture Alignment Database, Database of annotated structural alignments for homologous protein families, utilising SCOP, Pfam and

					SMART to identify relatives.
SCOP SUPERFAMILY	LMB-MRC, Cambridge, UK *Murzin*	54745 domains in 1294 superfamilies	Manual	Manual classification	Structural Classification Of Proteins. Hierarchical classification by Class, Fold, Superfamily, Family.

Table 1.2 Protein Structure databases (July 2005)

1.7.5 Structural Genomics Initiatives

Although solving the structure of proteins at the atomic level is a non-trivial task, it can provide important insights into the mechanics of protein function. Such efforts can enable us to rationalise why certain proteins interact and elucidate the unique molecular apparatus afforded by enzymes to catalyse chemical reactions under physiological conditions. Nevertheless, given the large number of proteins in nature, it is unrealistic to hope to solve all structures with current techniques. The 'Holy Grail' of structural bioinformatics is therefore to be able to accurately predict structure from sequence.

Proteins are molecules, albeit very large ones, which obey the laws of chemistry and quantum physics, and therefore many believe it should be feasible to go from sequence to structure using *ab initio* methods (for a review, see Hardin *et al.* (2002). However, the process of protein folding is far from well understood and the best performing structure prediction techniques have been those that utilise empirical data on known sequence-structure relationships. These include 'threading' sequences through a library of structural templates (Jones *et al.*, 1992) and modelling from

homologous structures (homology modelling). Working on this principal structural genomics is aiming to put each protein sequence within the reach of these reliable homology modelling methods.

1.8 Predicting protein function

As discussed in the previous sections, annotating novel proteins with function can be achieved by finding close relatives through global sequence or structure comparison. Many studies have shown that enzyme domains which share at least 40% sequence identity are highly likely to share a common function (Todd et al. 2001), although this figure must be raised to 60% for multi-domain proteins. Indeed, algorithms such as BLAST and FASTA are widely used to rapidly scan large databases of genome sequences in order to detect close relatives with experimentally characterised functions.

1.8.1 Defining Protein Function

'Protein Function' is a term frequently used in the literature, but should always be carefully defined. If we take glycogen synthetase as an example, we can say that its physiological function is to store excess blood glucose as glycogen in muscle and liver tissue. At a cellular level, it interacts with other metabolic enzymes to interconvert carbohydrates. On the molecular level, it catalyses the polymerisation of glucose-6-phosphate to glycogen. These three very different descriptions illustrate that when designing methods to predict protein function, it is important to decide on which level of function one is trying to focus.

Analysis of primary sequence and structure are most likely to give us information on a protein's biochemistry and molecular interactions. We can hope to identify motifs associated with a particularly binding property, such as ATP-binding, or enzymatic function and even cellular localisation. However, an important caveat in genome annotation is that although we may be able to predict the binding partners and reaction chemistry, these

descriptions can be of limited use to experimentalists. For example, protein kinases are so ubiquitous in cell signalling pathways that simply identifying a protein's ability to catalyse phosphorylation says nothing about its role in a cellular context. It would be helpful to go further to predict its substrates and interaction partners.

1.8.2 Whole protein function vs. domain function

As reviewed above, polypeptide chains can fold to form a number of distinct structural domains. Furthermore, several chains can interact via electrostatic and hypdrophobic interactions to form protein complexes. Functional sites and enzymatic cavities can span more than one chain or arise in the gap between two globular domains. Even when the catalytic site is entirely located on a specific chain or domain, it may only be active in the full-complexed state. Hence, it is often only valid to ascribe a function to a whole protein, rather than a chain or domain. This is confirmed by the large range of functions observed across superfamilies of domains in the CATH database (Pearl *et al.*, 2005).

1.8.3 Structured descriptors of protein function

As was highlighted in Section 1.8.1, protein function can be described on a number of levels. To further complicate matters, many enzymes and substrates have a number of synonyms. Therefore, several efforts have been made to more formally and consistently describe the huge diversity of functions observed in nature. One of the earliest in the field, was the Enzyme Classification (E.C.) (Bairoch, 2000) which groups enzymes into six major classes based on the chemistry of the reactions they catalyse. Each E.C. number consists of 4 digits (e.g. 2.7.7.1) where the first three describe the catalytic action of the enzyme and the forth usually denotes its substrate specificity. Rison and co-workers (Rison and Thornton, 2002) have shown that proteins are recruited into metabolic pathways based on their reaction chemistry and allowed to evolve the required substrate specificity. Hence,

many different enzymes in the same superfamily frequently share the same reaction chemistry (i.e. first 3 E.C. numbers).

The Gene Ontology (GO) (Ashburner *et al.*, 2000) was set-up to provide consistent descriptors of proteins in every species. The consortium developed three controlled vocabularies (ontologies) to describe a protein's molecular function, its role in biological processes and its association with other cellular components. Unlike the hierarchical E.C. classification, each ontology is constructed as a directional graph, where each term may have multiple parents. For example, an ATP-dependent DNA helicase is a child of 'DNA binding', 'DNA helicase' and 'ATP-binding'. One of the major goals of GO is to facilitate automatic annotation of newly sequenced genomes by comparison to well-characterised genes in experimentally tractable organisms. For example, Cdc9p in yeast is able to perform DNA ligation during replication, repair and recombination. It is not known whether this is true for the equivalent enzyme in higher organisms, but the ontology captures these three functions independently and therefore gives the experimentalist the opportunity to test each individually.

1.8.4 Predicting functional residues through incorporating sequence and structural information

Even profile-based sequence methods can result in sub-optimal alignments of distant relatives. Assigning function from remote homologues frequently requires structural data and many groups have sought to combine this with sequence information.

Lichtarge and co-workers (1996) pioneered a method known as the 'Evolutionary Trace' to identify sequence motifs associated with specific functions, such as ligand binding specificity. A phylogenetic tree is built from protein families and the conservation at each alignment position is calculated at different levels of global sequence similarity across the tree.

Conserved residues specific to certain clades of the tree were mapped onto a representative structure to locate the functional site and identify binding residues, as they often clustered together in three dimensions. Landgraf *et al.* (2001) extended this to automatically select a representative structure from a cluster of functionally related homologues and identify conserved residue clusters that characterized protein surfaces, such as SH2 domains. A global conservation score was calculated for each position in the multiple alignment and a second score measured the local conservation in a 10Å radius around each position. After statistical analysis, they generated a regional conservation score, C(x), and a similarity deviation score, S(x), for each residue in the alignment. C(x) defined the conservation of the local environment relative to the whole protein and was particularly effective at identifying poorly conserved transient interfaces in the MAPK transcription factor, ERK2. Whereas S(x) detected highly variable residue clusters that were shown to confer the various binding specificities to members of a family of aldolases.

1.8.5 Using electrostatics to predict functional sites

Molecular interactions in the cell — either between protein surfaces or proteins and their ligands — rely on electrostatic contacts between charged or polar residues. Many groups have examined ways of analysing and classifying proteins by the physico-chemical properties of their surfaces. Pawlowski and Godzik (Pawlowski and Godzik, 2001) took a molecular cartography approach to reduce protein surfaces to a spherical map. Focussing on charged and hydrophobic residues, they were able to calculate the similarity between two protein maps. They showed that this simple measure was capable of identifying functional subgroups within protein families, such as distinguishing between monomeric and tetrameric haemoglobin subunits. This method has been made available as a webserver (Sasin *et al.*, 2007). A similar resource (the electrostatic-surface of functional site (ef-Site) database (Kinoshita and Nakamura, 2004)) provides information

about electrostatic potential surfaces that can be used to identify similar patterns of charge in binding and interaction sites.

1.8.6 Using knowledge-based catalytic structural templates

To retain function through evolution, the structure of two proteins may stay the same, despite significant divergence of their sequences. This can be due to the constraints of maintaining the overall fold, but structure is also particularly conserved in the environment of functional sites.

In 1997, Wallace and co-workers (1997) built a database of catalytic sites (PROCAT) that were characterised by hand. This has now been superseded by the Catalytic Site Atlas (Porter *et al.*, 2004) and contains over 14,000 structures, with each catalytic residue (up to 6 per protein) annotated with information from the literature. The 3D conformation of these functional residues is often conserved over evolution to preserve function, even when other regions of the structure may vary. A fast search algorithm (JESS) is used to compare small catalytic templates to structures of unknown function to assign a putative E.C number (Barker and Thornton, 2003). In spite of this, there are two main problems with the approach.

Firstly, catalytic residues can frequently move relative to one another when the substrate binds, causing their geometry to vary considerably between structures with and without bound ligands. Secondly, the probability of these small templates matching regions in functionally-unrelated proteins is high, making it difficult to distinguish between true and false matches simply by RMSD. The SiteSeer algorithm (Laskowski *et al.*, 2005) attempts to address this problem by also comparing the local environments of the known catalytic residues and the corresponding residues in the matched protein. They exploit the idea that the environment around the active site often exhibits higher sequence similarity than suggested by a global alignment of the query and match structures. A statistical scoring function improves

matters by producing a more biologically-meaningful ranking for each search of a given query protein.

Other methods (DRESPAT (Wangikar *et al.*, 2003), PINTS (Stark and Russell, 2003)) look for structural motifs that are common to both the annotated and 'hypothetical proteins'. They have the advantage of not requiring a user definition of functionally relevant residues; however, there is no guarantee that structural similarities are not a product of stabilising the protein fold, rather than true functional conservation.

1.8.7 Using surface cleft analysis to identify binding pockets

One of the key reasons enzymes can catalyse reactions so effectively is that they are able to isolate their substrates in binding pockets or clefts, creating a unique chemical environment. Indeed, the active site is usually found in one of the two largest surface clefts (Laskowski *et al.*, 1996). In a similar fashion to the template searching discussed in the previous section, binding sites in unannotated proteins can be compared against a library of known sites, such as those implemented in the pvSOAR/CASTp server (Liu *et al.*, 2007). SiteEngine (Shulman-Peleg *et al.*, 2005) goes further than similar geometric matching by also examining the physico-chemical properties of the amino acids in the site. The conservation of charge and hydrophilicity often provides an important addition to pick out genuine functional homologues.

Although these methods can be used to effectively assign function, they are again constrained by the fact that similar binding sites can exhibit different geometries depending on the presence, absence or identity of the bound ligand.

1.9 Support Vector Machines (SVMs)

Support Vector Machines (SVMs) are a class of learning machines that aim to distinguish between two classes of data based on different values of common features. For example, given ion and enzyme levels in samples of human blood, an SVM could conceivably distinguish between diseased and non-diseased patients. SVMs aim to maximise a particular mathematical function with respect to a given data set and have been used extensively in bioinformatics over recent years in protein-fold recognition (Rangwala and Karypis, 2005; Rangwala and Karypis, 2006; Miller *et al.*, 1996), structural class prediction, secondary structure prediction and subcellular localisation. SVMs are underpinned by statistical learning theory, which provides a theoretical framework from which to fit a function to separate two classes of data by placing a division (or hyperplane) between them (Vladimir N.Vapnik, 1995).

1.9.1 Calculating a separating hyperplane

To construct a classifying function to separate two classes of data, the SVM calculates a hyperplane. Figure 1.9a shows an example where, with respect to two variables, the two classes can be fairly easily delineated. However, there are obviously many different hyperplanes that could separate these data. A statistical learning theorem shows that the most probable hyperplane classifier between two classes of data is the one that adopts the maximal distance (soft margin) from the nearest data points, i.e. in the middle of the two sets. Clearly, real data sets rarely separate this cleanly and for the SVM to come to a solution, it has to be able to deal with imperfect solutions. This is achieved by introducing a user-defined parameter (C, or soft margin), which essentially determines how many of data points are allowed to be misclassified when training the SVM without affecting the chosen hyperplane.

1.9.2 Choosing a SVM kernel

Some data sets can be easily separated by a linear hyperplane, whereas in other cases such a solution is not possible. However, SVMs can be extended to use more complex kernels. Figure 1.9b shows two classes described by two features, one of which does not vary. However, by squaring the variable feature (effectively placing the solution in higher dimensions), it is possible to separate the classes using a linear hyperplane. This approach is referred to as the 'kernel trick'. Figure 1.9d shows a more realistic example where the data points have been transformed into 4 dimensions by the kernel function, producing a non-linear solution in 2 dimensions.

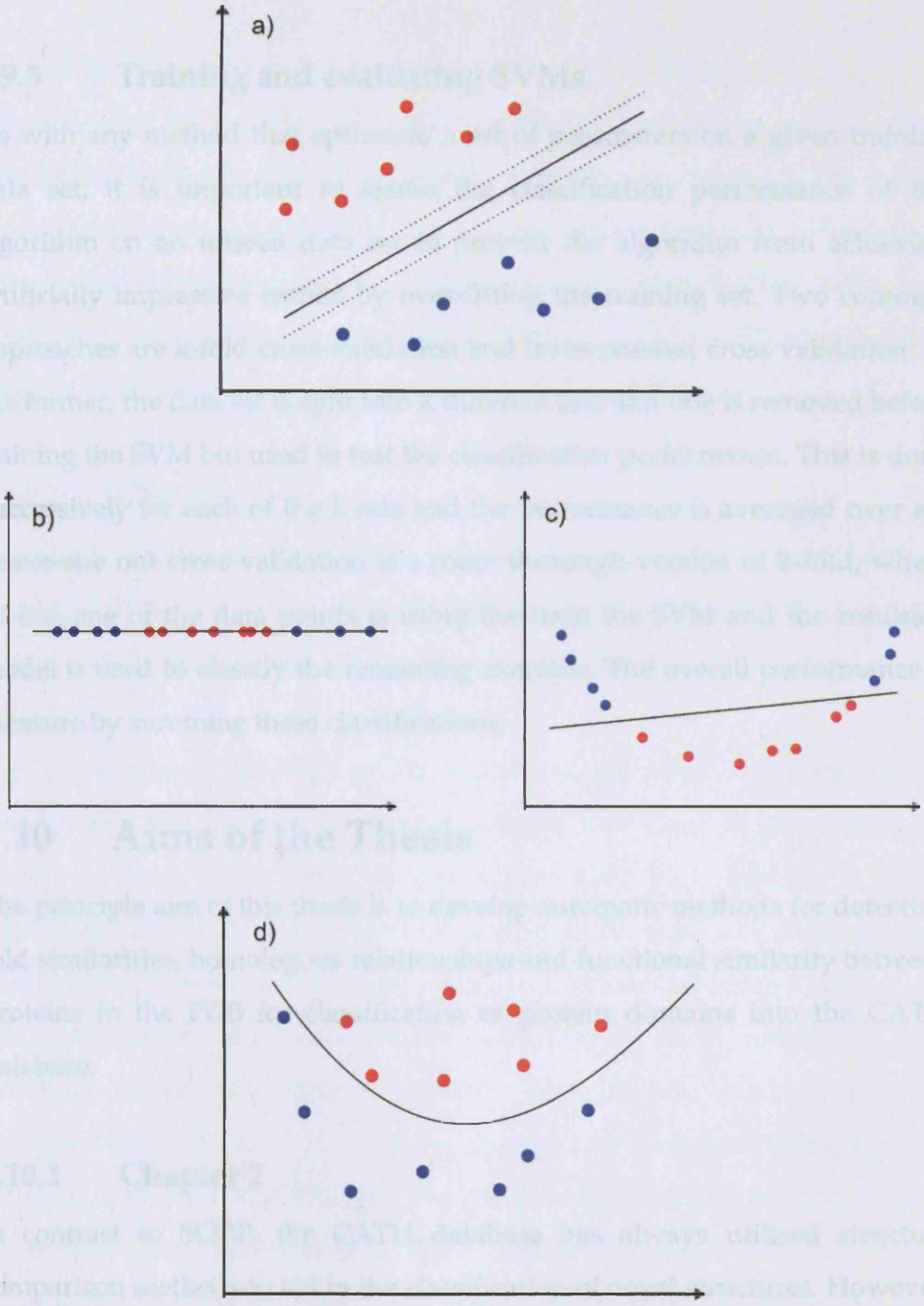


Figure 1.9 a) Separating two classes of data using a linear hyperplane. The soft margin (C parameter) is shown by the dotted lines. b) Two classes of data that cannot be separated in two dimensions using a line. c) By squaring the x feature in b) using the 'kernel trick', a linear solution can be found. d) A line separating two classes of data, which is linear in 4 dimensions, but not in 2.

1.9.3 Training and evaluating SVMs

As with any method that optimises a set of parameters on a given training data set, it is important to assess the classification performance of the algorithm on an unseen data set to prevent the algorithm from achieving artificially impressive results by over-fitting the training set. Two common approaches are k-fold cross-validation and leave-one-out cross validation. In the former, the data set is split into k different sets and one is removed before training the SVM but used to test the classification performance. This is done successively for each of the k sets and the performance is averaged over all. Leave-one out cross-validation is a more thorough version of k-fold, where all but one of the data points is using the train the SVM and the resulting model is used to classify the remaining example. The overall performance is measure by summing these classifications.

1.10 Aims of the Thesis

The principle aim of this thesis is to develop automatic methods for detecting fold similarities, homologous relationships and functional similarity between proteins in the PDB for classification of protein domains into the CATH database.

1.10.1 Chapter 2

In contrast to SCOP, the CATH database has always utilised structure comparison methods to aid in the classification of novel structures. However, assigning domain boundaries to multi-domain chains is still one of major bottlenecks in the curation process. However, upto 90% of new structures contain previously observed folds. Chapter 2 details the development of CATHEDRAL: a new algorithm to automatically assign domain folds and boundaries. It compares a query multi-domain protein chain against a library of previously-classified folds in CATH by modifying and combining features from the GRATH and SSAP algorithms.

Around 50 structural comparison algorithms have been published in the literature over the last 30 years, the vast majority of which are not in regular use by the bioinformatics or structural biology community. Those which have gained popularity tend to have a web-based interface for users to submit their own structures or structures from the PDB. CATHEDRAL was designed to be implemented as a crucial stage in the CATH classification protocol and to be made available to the scientific community.

1.10.2 Chapter 3

Chapter 3 concerns another new algorithm, FLORA, which exploits multiple structure alignments of functionally similar domains to discover structural motifs, which can then be used to assign function to new domains.

The central goal of this work was to exploit sequence and structural data to detect conserved patterns in protein families that recur in enzymes with similar catalytic mechanisms, as defined by their E.C. number. A novel algorithm, FLORA, was developed to analyse multiple structural alignments of domains in these families and discover a conserved motif. Patterns of sequence conservation and residue accessibility were combined with structural conservation data to identify these motifs, which were then encoded into templates and compared against new structures using a graph matching program, FLORAScan. The primary focus of the method was to discriminate between domains with different functions, yet a common evolutionary origin (i.e. from the same CATH superfamily) in a more effective way than global structure comparison.

1.10.3 Chapter 4

Chapter 4 takes structure comparison methods a step further through combination with SVM technology to predict the fold of *ab initio* models. The aim of the work presented here was to further the efforts of De La Cruz *et al.*

(2002) and Simons et al. (Simons *et al.*, 2001) in utilising structural comparison methods to compare *ab initio* predictions (models) for a given target sequence to a library of known domains in CATH in order to assign it to a fold group. Once a fold prediction is made, the structural alignment between a model and library structure can be combined with homology modelling methods to further refine the structure prediction.

Chapter 2 CATHEDRAL: Detecting homologues and assigning domain boundaries

2.1 Background and Aims

Over 7000 new proteins structures were deposited in the Protein Data Bank (PDB) (Berman *et al.*, 2000) in 2005, many of which contain multiple polypeptide chains. Furthermore, as observed in structural classification databases, a significant fraction of protein chains comprise two or more domains (known as multi-domain structures). Indeed, nearly 50% of polypeptide chains classified in version 2.6 (May 2005) of the CATH database (Orengo *et al.*, 1997) are multi-domain and the proportion of this type of structure in the PDB in likely to increase with improvements in techniques for experimental structure determination. Figure 2.1 shows that the majority of multi-domain chains in v2.6 of CATH comprise two domains, although some larger structures have been solved with 3, 4 and even over 5 domains. Moreover, recent analyses of completed genomes have suggested that the proportion of multi-domain structures in some organisms, particularly eukaryotes, may be as high as 80% (Apic *et al.*, 2001).

To classify such structures into the CATH domain database, it is necessary to delineate their domain boundaries and subsequently assign each component domain to a homologous superfamily, with both processes requiring significant manual intervention. However, the majority of newly solved structures contain previously observed domain folds and accordingly it is feasible to exploit structural comparison methods to recognise these folds in their multi-domain context. Even a recent analysis of domains solved by the Structural Genonomics Initiatives (SGIs) (Todd *et al.*, 2005) — which aim to

target novel folds — showed that approximately 90% adopt structures similar to those already observed in the PDB.

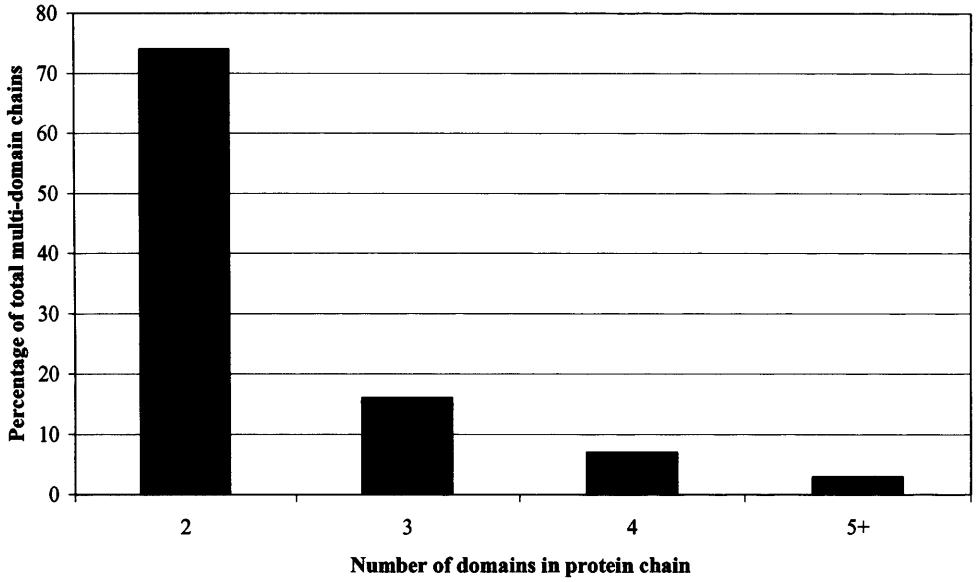


Figure 2.1 Percentage of multi-domain chains with a given number of component domains.

Proteins are comprised of individual folding units known as domains. In general, each domain takes the form of a specific topology and it is estimated that there are up to several thousand such folds in nature (Chothia, 1992; Orengo *et al.*, 1994; Grant *et al.*, 2004). Domains are thought to be important evolutionarily conserved units, and structural classification databases, such as SCOP (Murzin *et al.*, 1995) and CATH (Orengo *et al.*, 1997), aim to classify their structures into fold groups and superfamilies. Although members of domain superfamilies can exhibit sequence similarity of < 30%, relatives generally maintain comparable topologies in the core of their structures (Orengo *et al.*, 1997; Reeves *et al.*, 2006).

Various structural methods have been developed to detect domain boundaries through *a priori* knowledge of domain structure, folding and interactions. The method of Taylor (1999) uses a technique similar to an

Isling model in which the structural elements of the model changes state according to a function of the state of their neighbours. Each residue in the protein chain is assigned a numerical label and if a residue is surrounded by neighbours that possess (on average) a higher label, its label increases. The DOMAK algorithm of Siddiqui and Barton (1995) assumes that a domain makes more internal contacts (intra-domain) than external contacts (contact with residues in the remainder of the structure). A "split value" is calculated from the number of contacts measured when a protein is divided into two parts at different points, which is highest when the two parts of the split structure are distinct domains. By contrast, the DETECTIVE algorithm (Swindells, 1995) aims to determine the hydrophobic core in each domain unit. The Parser for protein Unfolding Units (PUU) algorithm by Holm and Sander (1994) uses a harmonic model to describe inter-domain dynamics and this is then used to define domains for the FSSP database (Holm and Sander, 1998).

The original CATH classification protocol, (Jones *et al.*, 1998), attempted to use a consensus of the results from the three independent methods: PUU, DOMAK, DETECTIVE. However, although many of these individual methods reported between 70-80% accuracy in benchmarking tests, this does not seem to have been born out on a practical level when updating CATH and manual validation becomes the only secure way to resolve conflicting predictions. An additional complication is that approximately 30% of domains are discontiguous in sequence — i.e. the structure of the individual domains is formed from disconnected regions of the polypeptide chain – and assigning these types of domains remains a problem for most automated methods (Jones *et al.*, 1998).

Another problem with *ab initio* domain prediction is that it provides no indication of whether each domain is similar to other folds in CATH. Hence, even after manually validating the domain boundaries, it is still necessary to compare each domain against a database of known structures if one is to

classify the fold. As has been suggested, there are a limited number of folds in nature and newly solved multi-domain structures are likely to contain at least one previously observed fold. Therefore, exploiting the concept of domain recurrence appears a sensible strategy to classify the majority of new polypeptide chains. The idea of automatically detecting recurring domains is not new and has been successfully exploited by the DALI Domain Database (Holm and Sander, 1998), which uses a combination of structural comparison and automated domain detection to classify new structures.

Several powerful structural comparison algorithms exists — GRATH (Harrison *et al.*, 2003), SSAP (Taylor and Orengo, 1989), CE (Shindyalov and Bourne, 1998a), DALI(Holm and Sander, 1993), SSM (Krissinel and Henrick, 2004), STRUCTAL (Kolodny *et al.*, 2005) and VAST (Madej *et al.*, 1995) — each of which have been reviewed in more detail in Section 1 of this Thesis. The performance of any alignment method should be measured on its ability to generate biologically-meaningful alignments and its capacity to accurately detect similar folds and structural homologues by means of a robust scoring scheme. As a recent analysis by Kolodny and co-workers (2005) highlighted, the accuracy of the latter feature is vital when comparing novel structures with a database such as CATH. They found that the RMSD of a given alignment, normalised by the number of aligned residues (termed the SAS score), was the best score for discriminating domains with genuine fold similarities. However, the original RMSD is still dependent on the number of equivalent residues in the alignment — although the SAS score provides a more uniform measure across proteins of different sizes, it may still score small motif matches disproportionately highly. Hence, a good SAS score does not necessarily indicate that a globally optimal alignment with the maximum number of equivalent residues has been found. For the purpose of assigning accurate domain boundaries from structural comparison, it is particularly important to align as many residues as possible, as this alignment will be used to allocate the domain region. Simply superposing

the conserved core of two domains with the same fold will often leave the equivalences in the rest of the structure to guesswork.

This chapter concerns the development of the CATHEDRAL algorithm, a novel domain identifier that exploits the fold-recurrence philosophy. CATHEDRAL is an acronym for CATH's Existing-Domain Recognition ALgorithm. It compares a query multi-domain protein chain against a library of previously-classified folds in CATH by modifying and combining features from the GRATH and SSAP algorithms.

SSAP is a residue-based method that uses double dynamic programming to produce accurate alignments, even for distant structural relatives. However, it attempts to solve a highly computationally intensive problem and is slow for large-scale analysis and protein structure database scanning. Conversely, GRATH is extremely fast and seeks the largest common 'clique' of equivalent secondary structures between two structures. It is able to identify equivalent folds with high sensitivity and selectivity, but unlike SSAP does not provide a detailed or globally optimised alignment.

To improve the speed yet maintain the fidelity of detecting domains with similar folds, this work explores using GRATH as a filter for pre-selecting similar structures in the CATH database, which can then be aligned more accurately with SSAP. Initially, this was optimised by comparing domains which had already been classified in CATH. By using GRATH to pre-filter putative structural relatives before generating more accurate SSAP alignments, a 100 fold increase in speed is achieved, depending on the size of the query structure, at no cost to the quality of the domain alignments. This approach was shown to perform well against several other publicly available structure comparison methods at the domain level.

Assigning individual domains to multi-domain chains poses a more challenging problem for structural comparison, not least because in some

cases the definition of a 'domain region' can be highly subjective. Scanning newly solved chains against the CATH library using GRATH often effectively identifies constituent domain folds; however, it can also erroneously match highly recurrent structural motifs that are present across many areas of fold space. Several measures (e.g. SAS score, RMSD, number of aligned residues, number of matched secondary structures) should be taken into consideration when gauging whether a valid fold assignment can be made to the query chain. When developing an algorithm, many workers choose to experiment manually with different scoring schemes and then optimise the parameters on their chosen test set. However, machine learning methodologies, such as Support Vector Machines (SVMs), can also be used in order to enable this optimisation to be performed automatically, rendering the process easier and potentially more powerful.

Around 50 structural comparison algorithms have been published in the literature over the last 30 years, the vast majority of which are not in regular use by the bioinformatics or structural biology community. Those which have gained popularity tend to have a web-based interface for users to submit their own structures or structures from the PDB. CATHEDRAL has been implemented as a crucial stage in the CATH classification protocol and another aim of this chapter was to make these tools available to the scientific community. Hence, a new webserver was created to allow users to make their own domain assignments.

2.2 Methods

2.2.1 Overview of Methods

This section briefly details some of the more technical methods and optimisations used in the development of the CATHEDRAL algorithm. The main steps of the algorithm are outline in Section 2.3.2. SVM technology was used to predict fold assignments and this optimisation is also included.

2.2.2 Data sets

2.2.2.1 *Selecting domain library and CathScop data sets for benchmarking GRATH and SSAP*

Representative domains were taken from 6003 sequence families (SReps) in CATH v2.6.0 to form a data set where potential evolutionary relationships could not be identified reliably by sequence methods with all domains sharing less than 35% sequence identity. These formed the *domain library* with representatives from all 907 fold groups.

GRATH and SSAP were benchmarked against four other structural comparison methods: STRUCTAL, DALI, LSQMAN and CE. An all-against-all structural comparison was performed between all domains in the domain library, for each of the different structural comparison methods, giving over 18 million individual comparisons. To minimise any bias towards CATH classifications, a second data set that was subset of CATH v2.6.0 and SCOP v1.65 was constructed. Each of 6003 CATH (SRep) domains was checked to see if it had an equivalent SCOP domain with at least 80% residue overlap and was in the same SCOP family sharing 80% of the members. This created the *CathScop* data set with 1779 SReps encompassing 406 folds.

2.2.2.2 *Selecting a dataset of proteins chains for optimising CATHEDRAL*

A set of 1071 non-redundant (at 35% sequence identity) representatives (SReps) from multi-domain sequence families were selected. From this set, those chains containing domains from folds with less than 2 SReps were removed. The remaining set contained 680 chains with 1593 constituent domains.

2.2.3 Benchmarking SSAP against other publicly available structure comparison methods

There are several publicly available methods that have been endorsed by widespread community use and/or validation by comparative

benchmarking against established methods. The programs selected here were based on those previously benchmarked by Kolodny and co-workers (2005) for performance in fold recognition and alignment accuracy. These were: CE, LSQMAN, DALI and STRUCTAL.

Structure alignment methods were compared using Receiver-Operator Curves (ROC). These plot true positive rate (sensitivity) against the false positive rate (1 - specificity) for different similarity scores returned by the individual methods. A binary classifier was defined by the CATH hierarchy whereby a positive match is one where both domains share the same fold or superfamily classification whilst negative match does not. The matches for each method were ordered by the structural similarity score of their alignment, and the number of true positives and true positives calculated at varying thresholds.

Kolodny and co-workers tested several measures for assessing the accuracy of structural alignments. They found the most useful to be the SAS score (see (Equation 2.1), which normalises Root Mean Squared Deviation (RMSD) by the number of aligned residues and penalises alignments where less than 100 residues have been aligned.

$$SAS = 100 \times \frac{rmsd}{nAl}$$ (Equation 2.1 SAS score, nAl = aligned residues)

In addition to this geometric measure, alignment accuracy was also assessed by comparison against a set of manually curated alignments. BAliBASE (Thompson *et al.*, 1999) is a database of manually-refined multiple structure alignments specifically designed for the evaluation and comparison of multiple sequence alignment programs. The alignments in BAliBASE are selected from the FSSP (Holm and Sander, 1997) or HOMSTRAD (Mizuguchi *et al.*, 1998) structural databases, or from manually constructed structural alignments taken from the literature. Functional sites are identified using the

PDBsum database (Laskowski *et al.*, 1997) and the alignments are manually verified and adjusted, in order to ensure that conserved residues and secondary structure elements are well aligned.

Fourteen BaliBase multiple alignments were selected comprising 108 pairwise structural comparisons. All the alignments represented single protein domain chains that shared less than 25% sequence identity, making alignment non-trivial. All protein classes were represented and the quality of the alignments generated by the different structure comparison methods was measured by the score, fm, which is the number of amino acids correctly aligned in the structural alignment divided by the total number of aligned residues in the BaliBase alignment. CE was not appropriate for this analysis as the alignments it outputs only show the largest continuous motif.

2.2.4 Guiding residue alignments using SSAP

Although secondary structure matching using GRATH is both fast and effective at finding fold similarities, it tends only to identify highly conserved regions of secondary structure. A large amount of structural variation around this common core is observed across some superfamilies (Reeves *et al.*, 2006), even more so within a fold group. Conversely, the SSAP algorithm has been optimised to find as many equivalent residues and hence the optimal global alignment between two domains.

The first step in SSAP is to find putative equivalent pairs of residues, by selecting those that share comparable torsional angles and solvent accesbility. Each pair is then aligned using dynamic programming to compare their residue environments. For two large domains, the numbers of putative residues pairs can run into several hundred, each of which must be subjected to the same computational expensive algorithm. The paths from these matrices are added to a summary matrix, provided their total score exceeds a threshold. The top 20 highest scoring residue pairs are then compared again using dynamic programming. The summary matrix is then

reset and these 20 paths are added. From this, the final alignment is calculated using dynamic programming. Previous versions of SSAP have sought to increase the speed of the process by performing an initial secondary structure alignment (Orengo *et al.*, 1992). It was proposed that the conserved clique identified by GRATH alignment could similarly be used to reduce the search space in SSAP in this way.

When a clique of secondary structures is matched, it aligns equivalent secondary structures in a pair of domains. This was used to guide a more comprehensive residue-level alignment, by modifying SSAP to use the clique to guide the initial selection of residue pairs. This is achieved by populating a binary matrix, which dictates which residue pairs are selected, based on the equivalent secondary structures identified by GRATH.

In the first step, residues in equivalent secondary structures are simply paired with one another. As equivalent strands and helices can vary in length (e.g. a helix with 11 residues could be aligned to one with 8), it must be an All-vs-All pairing (represented by a square of '1' values in the matrix). Similarly, residues on the end of aligned secondary structures could potentially be paired with residues in the loop regions, so the boundary is extended by 10 residues on either side.

Secondly, although the alignment for residues outside the clique is unknown, it is possible to exclude certain pairings. The clique effectively orientates the alignment and dictates that if helix 1 in protein A is equivalent to helix 2 in protein B, it cannot simultaneously be equivalent to helix 3 in protein B. Moreover, it sets the overall direction of the alignment and allows the regions between the clique secondary structures to be linked together.

Finally, the alignment of the beginning and end of the domains, outside embellishments to the core secondary structures in the clique is unspecified. However, it is known that these cannot be aligned to any of the core residue

pairs. Hence, the starts and ends of the domains are paired up for SSAP to decide where the equivalences lie.

As is standard in the SSAP algorithm, the torsional angles and accessibility of the potentially equivalent residue pairs are still assessed to determine which to select in the first phase of dynamic programming, which helps to reduce the search space further.

2.2.5 Training an SVM to recognise domain folds in multi-domain chains

Both GRATH and SSAP SAS scores give a good measure of the structural similarity of two domains (see Section 2.3.1). Nevertheless, their ability to discriminate between genuine fold similarities and simply matching smaller structural motifs also relies on accounting for the alignment overlap in relation to the largest domain. Indeed, recognising domain folds within a multi-domain context poses a more difficult problem if the domain boundaries are unknown, as it is not possible to accurately determine the overlap with the largest domain. In this case, all factors (such as the number of aligned residues, domain size and structural similarity) should be considered. In order to develop a robust scoring scheme for CATHEDRAL, an SVM was used to combine a series of scores from GRATH and SSAP and other indicators of alignment quality for the data set of protein chains described in Section 2.2.2.2. The primary aim was to generate a combined score that could be easily calculated to rank potential folds matches to a query chain.

The SVMLight package (Joachims 1999) was used in this instance to train a classifier. It provides a choice of 4 kernels: linear, polynomial, radial-basis function (RBF) and sigmoid. In addition, the user can define a tailored knowledge-based kernel. Initial investigations showed that the RBF kernel did not perform any better than using a linear kernel. Therefore, since the

SVMLight outputs linear weightings for each of the inputs in the latter case, and this could be directly implemented in the source code of CATHEDRAL, the linear kernel was chosen.

2.2.5.1 Creating a dataset as input to the SVM

Machine learning with neural networks or SVMs is usually undertaken using a training set where there are equal numbers of positive and negative examples. Unbalanced sets can bias the optimisation function to predict the majority class exclusively. However, SVMLight allows to user to modify the relative weighting of positive and negative inputs when training the kernel (using the –j parameter). This feature lends itself to CATHEDRAL as the negative examples in a database scan can outweigh the genuine matches by as much as 4 times. It also allows the SVM to train on all available examples, which is not the case when artificially balancing the data sets by randomly sampling negative examples. Therefore, we used this feature to weight genuine hits according to the ratio by which they were overrepresented by unrelated domains.

In order to ensure fair testing in machine learning applications, it is vital to assess the performance of the model on a separate dataset to the one on which it has been trained. An extension of this is five-fold cross-validation, which was used here. In this procedure, the dataset is split into 5 sets and each one is successively taken as the test set, while the model is trained on the other 4 sets. The performance is then calculated as an average over the 5 test sets. This guarantees that evaluation of the classifier is not biased by the any random fluctuations in the composition of the training or test sets.

2.2.5.2 Feature selection

As inputs to the SVM, measures of structural similarity and other alignment features from GRATH and SSAP were calculated. The features used are listed below:

1. GRATH score
2. GRATH clique size
3. SSAP score
4. Residue overlap (as calculated by SSAP)
5. RMSD
6. Number of aligned residues (as calculated by SSAP)
7. SAS score from SSAP alignment

To improve the performance and speed of the classifier, all features were normalised between values 0 and 1.

2.2.5.3 Optimising and assessing the performance of the SVM

The performance of machine learning methods can be measured in several ways: Error rate, percentage of correct assignments, Matthews Correlation Coefficient, ROC curve analysis. The latter is simply a plot of the true positive versus false positive rates over a range of score cut-offs. It is a useful measure of a score's ability to discriminate between correct and incorrect classifications. In this work, it was used to evaluate different scoring mechanisms for domain assignment.

When using a linear SVM, it is still advantageous to optimise the penalty variable, C. This determines how much each wrongly classified example is penalised when evaluating different hyperplanes. Depending on how the SVM is going to be used and the size of the data set, different values of C can result in classifiers with better or worse performance. In this instance, C was optimised by exploring a range of values and assessing performance based on the average area under a ROC curve. The results in Figure 2.2 show that a value of C of 10 or above gives the optimum performance. However, the overall increase is very small (< 0.02) and hence varying the C parameter has little effect (< 0.12 increase in ROC Area) on the dataset.

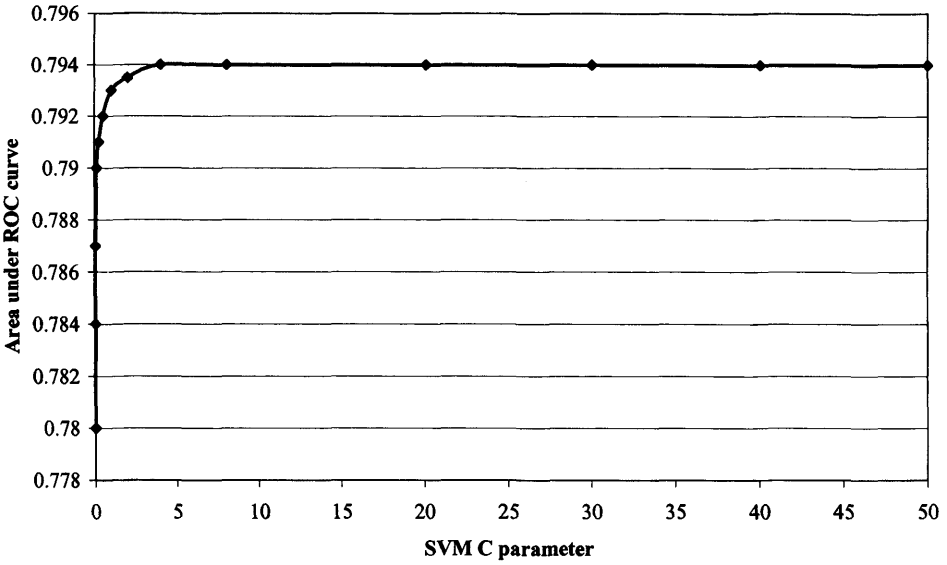


Figure 2.2 Optimisation of SVMLight cost parameter on chain CATHEDRAL dataset

2.3 Algorithm Development and Results

The ultimate aim of this work was to assign domains to multi-domain chains using the GRATH and SSAP structural comparison algorithms, using the CATHEDRAL algorithm. However, to ensure that the resultant method was going to be accurate, it was desirable to first benchmark the performance of these two component approaches for recognising structural similarity at the single domain level.

2.3.1 Assessing performance of GRATH and SSAP

GRATH and SSAP were compared to assess their ability to correctly identify fold and superfamily relatives from a data set of domains from CATH (*CathScop* data set, see Section 2.2.2.1). GRATH is several orders of magnitude faster than SSAP, however, it is limited by solely comparing secondary structure elements and the scoring scheme is based on the number of shared equivalent secondary structures. Although this can be effective at detecting the conserved core, it does not give a measure of the overall similarity between two domains. Conversely, SSAP calculates a SAS score (

(Equation 2.1) based on the RMSD of a residue superpostion, which gives a more accurate quantification of protein fold similarity.

2.3.1.1 *Ranking fold matches with GRATH and SSAP database scans*

The coverage and accuracy of GRATH and SSAP was assessed by scanning the dataset against a library of CATH domains and the results ranked by GRATH E-value, SSAP SAS score. The rank of the first appearance of the correct fold was noted and a frequency distribution calculated. It can be seen that SSAP finds the correct fold as the top hit over 94% of the time compared to nearly 84% for the GRATH E-value. Nevertheless, the correct fold is within the top 10 hits nearly 94% of the time for GRATH. SSAP appears to be a superior discriminator of fold similarities, yet the performance of GRATH is impressive given its superior speed and the simplicity of its approach.

2.3.1.2 *Ranking superfamily matches with GRATH and SSAP database scans*

For ranking homologous superfamily matches, SSAP maintains the same high performance with the correct hit at the top of the list 94% of the time (Table 2.1). However, GRATH drops to 76% as it unable to distinguish between homology and fold similarity. Interestingly, SSAP is better at discriminating superfamily matches than fold matches. This could be due to the fact that the correct hits are closer structural matches to the search domain. Moreover, fold groups are a more arbitrary grouping within CATH and different levels of structural variability are tolerated in different areas of fold space.

	FOLD		SUPERFAMILY	
Rank	GRATH E-value	SSAP SAS	GRATH E-value	SSAP SAS
1	83.8	94.6	76.2	93.7
2	4.2	1.5	5.8	1.9
3	1.6	0.5	2.9	0.7
4	1.3	0.3	2.1	0.4
5	0.6	0.3	1.0	0.3
6	0.3	0.2	0.6	0.1
7	0.5	0.1	1.1	0.1
8	0.5	0.2	0.6	0.3
9	0.3	0.1	0.5	0.1
10	0.3	0.1	0.4	0.2
>10	6.6	2.1	8.9	2.2

Table 2.1 A dataset of domains was scanned against the CATH library using GRATH and SSAP and the ranked by GRATH and SSAP scores respectively. The percentage of domains with the correct fold or superfamily at each ranking is tabulated.

2.3.1.3 ROC analysis

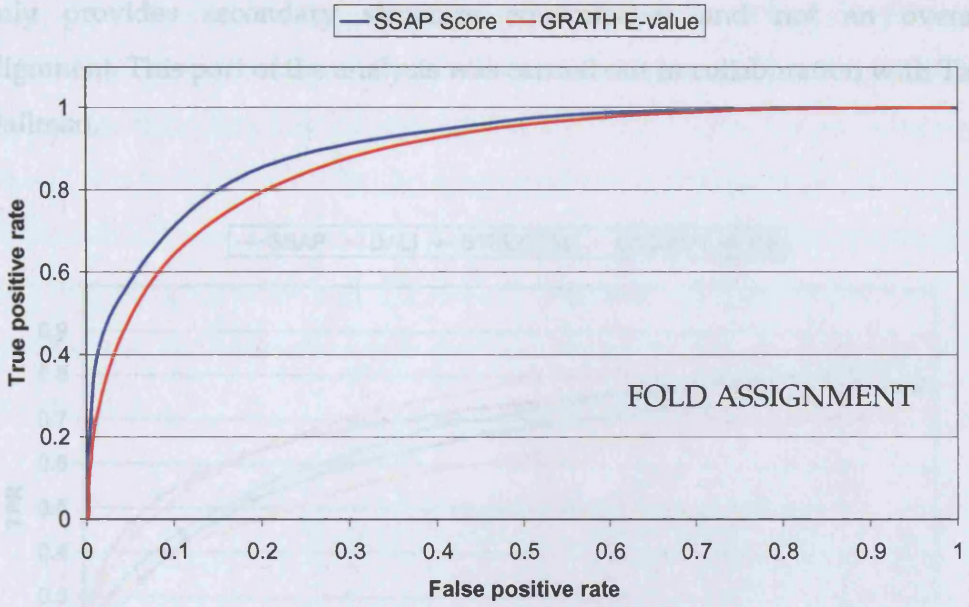
To further compare the ability of GRATH and SSAP to discriminate between different folds and superfamilies, Receiver-Operator Curves (ROC) were plotted for each method, Figure 2.3. These help to assess the ability of a score to differentiate between two classes, in this case (same fold/superfamily). A perfect ROC curve would form a mirror image of the x and y axes, with a true positive rate of 1 with 0 false positives. A random predictor would result in a linear graph of unit gradient, with true positive and false positives rates of 0.5. Integrating the area under the curve gives a measure of the overall performance of the score.

For fold prediction (Figure 2.3a), SSAP and GRATH perform fairly similarly, with SSAP performing slightly better. The area under the SSAP and GRATH

E-value curves are 0.91, 0.88 respectively, indicating that both methods perform well. Again, the main reason for the methods' lower than optimal performance is probably the inconsistency of fold clustering. It has been suggested that fold space is in fact a continuum (Harrison *et al.*, 2002); however, CATH and SCOP show a generally good correlation which suggests there is at least a common consensus for many areas of fold space (Hadley and Jones, 1999).

For predicting homologous relationships (Figure 2.3b), SSAP performs very well, significantly better than GRATH. The area under the SSAP and GRATH curves are 0.97, 0.90 respectively. Nevertheless, both methods appear to be better at identifying genuine homologues than fold matches. This again may be due to the aforementioned fact that folds are less well defined than superfamilies. However, it could also be that homologues are generally more structurally similar and the more distant fold matches are harder to assess by a simple geometric score.

a)



b)

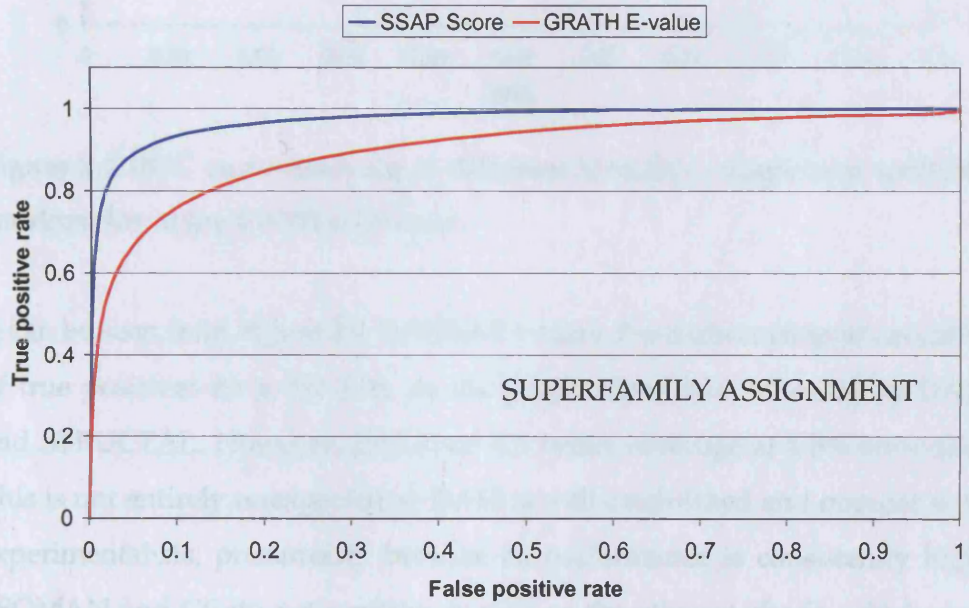


Figure 2.3 ROC curve analysis of GRATH and SSAP scores for a) fold assignment, b) superfamily assignment.

2.3.1.4 Comparing SSAP to other publicly available methods

SSAP was also compared against the performance of several other publicly available methods (DALI, STRUCTAL, LSQMAN, CE) using the alternate

CathScop data set (see Section 2.2.2.1). GRATH was not benchmarked as it only provides secondary structure equivalences and not an overall alignment. This part of the analysis was carried out in collaboration with Tim Dallman.

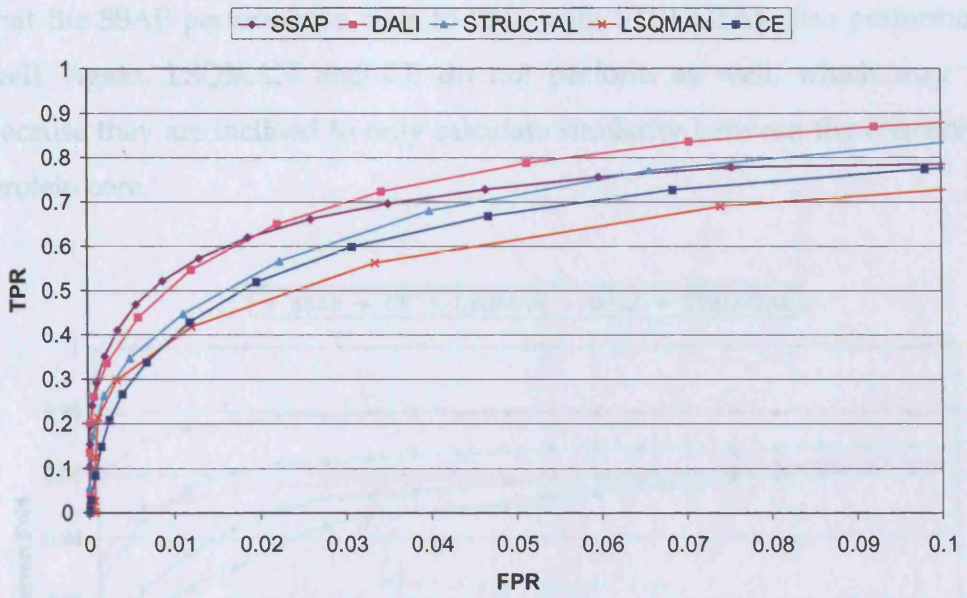


Figure 2.4 ROC curve analysis of different structure comparison methods for domains at the CATH fold level.

It can be seen from Figure 2.4 that SSAP returns the highest proportion (53%) of true positives for a 1% (0.01 on the graph) error rate, followed by DALI and STRUCTAL. However, DALI has 4% better coverage at a 5% error rate. This is not entirely unexpected as DALI is well-established and popular with experimentalists, presumably because its performance is consistently high. LSQMAN and CE do not perform as well as the other methods, which may be because they tend to score only residues that superpose well. This might suggest that maximising alignment length and calculating global similarity is most informative for detecting fold/superfamily relationships.

As well as the ability of the SAS score to discriminate between true and false fold matches, for the purpose of developing a domain boundary recognition

algorithm it is important to identify the closest relative within a fold group in order to obtain the best alignment. Therefore, the correct fold should rank highly in the list of matches. It can be seen from Figure 2.5 that SSAP assigns the correct fold as its top hit over 96% of the time. When the percentage of correct fold matches with the top ten matches are considered, it can be seen that the SSAP performance rises to 98%, with STRUCTAL also performing well. Again, LSQMAN and CE do not perform as well, which may be because they are inclined to only calculate similarity between the conserved protein core.

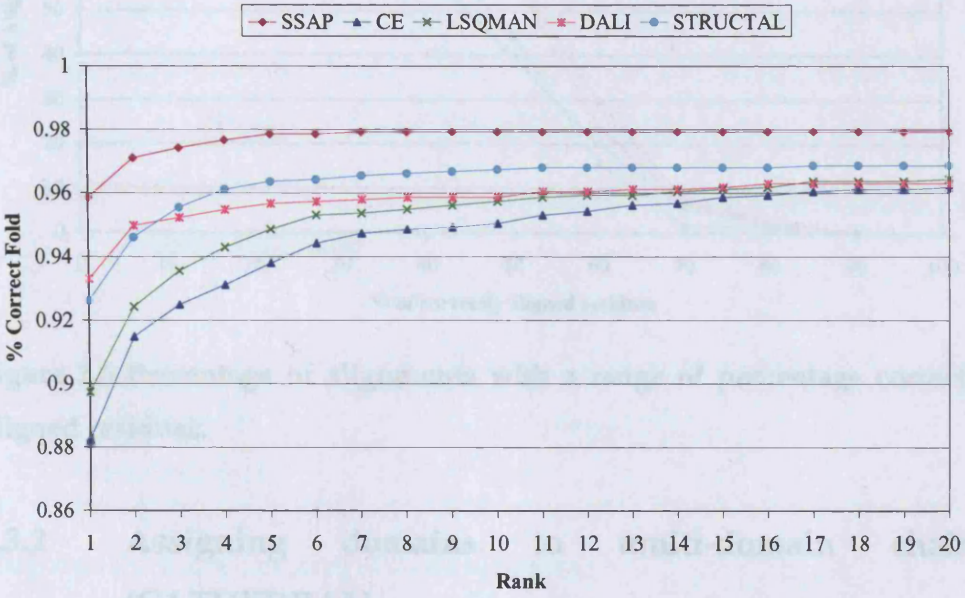


Figure 2.5 Plot of percentage of correct folds matched against the ranked native score for the CATH-SCOP data set.

2.3.1.5 Comparison to manually-curated alignments

The only true way to assess automatic structure alignments is to compare them to a manually validated dataset. We therefore compared all methods (excluding CE, see Section 2.2.3) against curated alignments in the BAliBASE. Figure 2.6 shows that DALI and SSAP produce alignments closer to the BaliBase alignments with nearly 60% of DALI and SSAP alignments having at least 50% residues correctly aligned, compared to 45% for LSQMAN and

40% for STRUCTAL. For LSQMAN, this is most likely due to the fact that it restricts its alignments to the 100 most conserved positions. However, it is interesting to note that although STRUCTAL does not align as many residues as SSAP and DALI, it still performs well in discriminating similarities at the fold level.

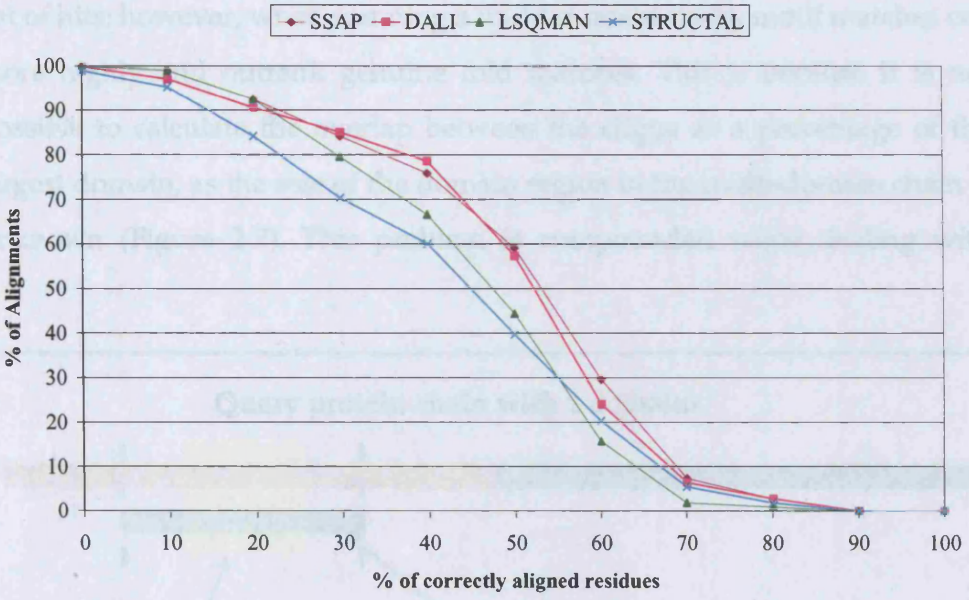


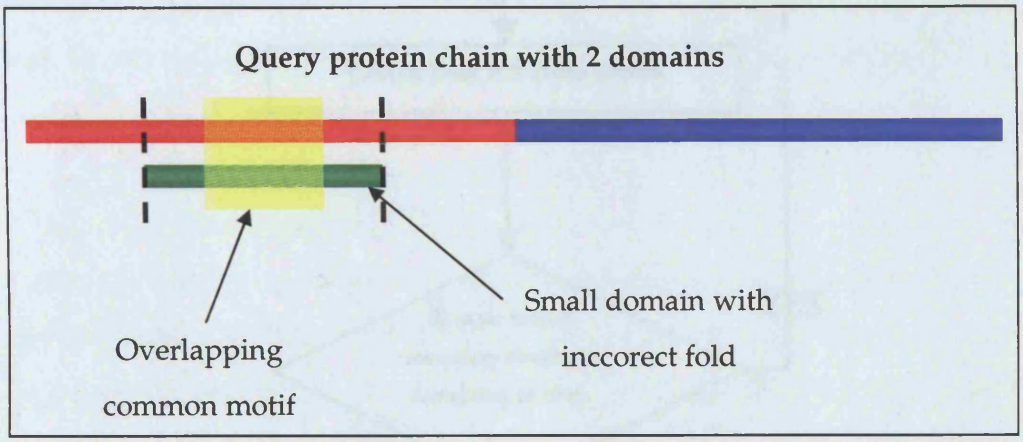
Figure 2.6 Percentage of alignments with a range of percentage correctly aligned residues.

2.3.2 Assigning domains to multi-domain chains (CATHEDRAL)

Assigning domain folds to multi-domain chains using structural comparison methods can initially appear as straightforward as scanning the chain against a domain library and allocating the highest ranking hit for each region. However, there are number of important caveats to this solution.

GRATH is very good at identifying common secondary structure motifs in two domains. These overlaps can often be large and indicative of a particular fold (Harrison *et al.*, 2002). However, they may also represent motifs that are observed in unrelated folds across the whole of the protein universe. A small domain containing a beta-alpha-beta motif, for example, may match a region

of the query chain which is in fact part of a much larger domain. This hit would score highly if only the overlap to the small domain was considered. Indeed, as well as hitting genuine relatives, folds such as the Rossmann will match many such motifs in small domains (Figure 2.7). Clearly, when scanning with a single domain, the largest match will still be at the top of the list of hits; however, when scanning a multi-domain chain, motif matches can score highly and outrank genuine fold matches. This is because it is not possible to calculate the overlap between the clique as a percentage of the largest domain, as the size of the domain region in the multi-domain chain is unknown (Figure 2.7). This problem is compounded when dealing with



domains which are discontiguous in sequence.

Figure 2.7 The problem of matching common structural motifs in small domains when scanning protein chains against the domain library, which leads to false domain boundaries despite a high local structural similarity.

The solution proposed here was to develop an iterative algorithm (CATHEDRAL), where domains are allocated in a stepwise fashion and the remainder of the chain re-scanned against the library for each assignment. This permits larger domains to be assigned first before comparing the remainder of the chain to smaller domains. The algorithm is summarised in Figure 2.8.

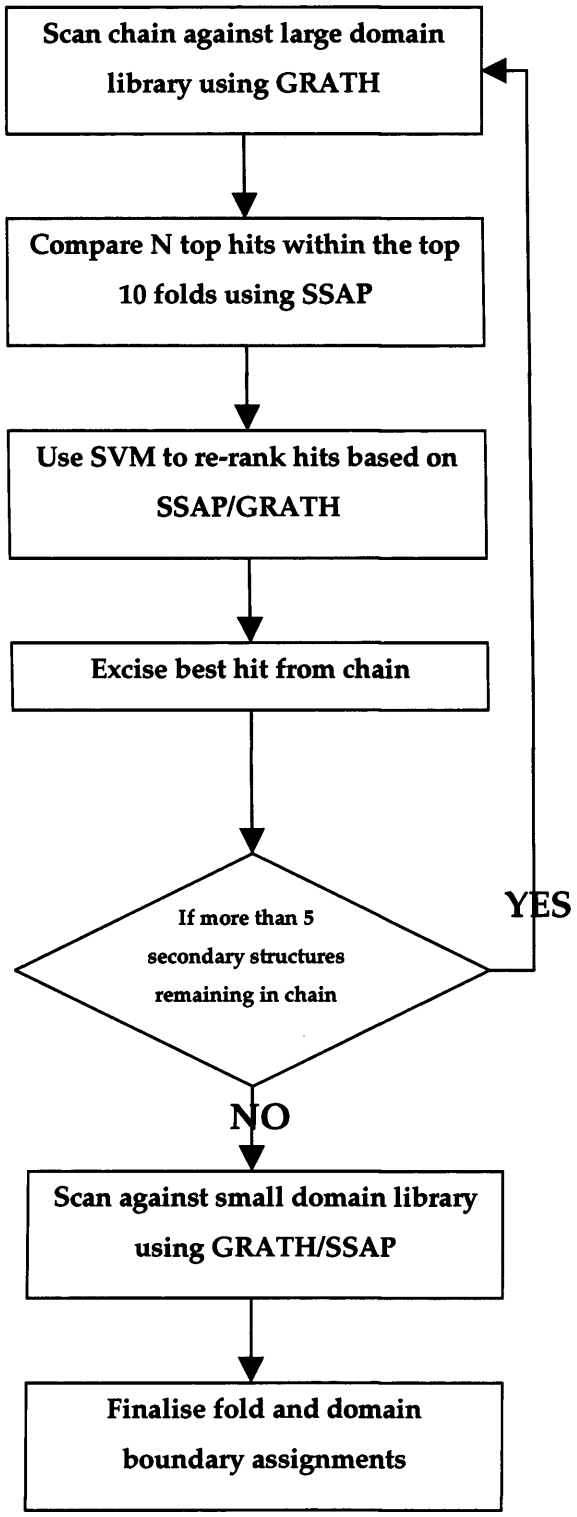


Figure 2.8 Flow chart of CATHEDRAL algorithm for assigning folds and domain boundaries to protein chains.

2.3.2.1 Scanning a chain against a library of domains using GRATH

The first step in the algorithm is to compare the query protein chain against the CATH domain library of folds using GRATH. To lessen the impact of motif matching to small domains, the library is split into domains with 5 or more secondary structures (*large library*) and those domains with less than 5 (*small library*). All small domain assignments are made at the end of the protocol. Chains are scanned against the large library and the hits ranked by the GRATH E-value.

N representatives from each of the top 10 folds identified by GRATH are taken forward for further analysis using SSAP. In Section 2.3.1.1, it was shown that this should cover 94% of correct folds. The motivation for this was to increase the chance of finding the closest structural match, which should result in the best domain boundaries.

2.3.2.2 Compare top hits using SSAP

Although GRATH is effective at matching common secondary structures, residue-based methods, such as SSAP, produce alignment scores that better represent the similarity between the two domain regions. The top 10 fold representatives from GRATH were therefore subjected to a SSAP alignment, guided by the secondary structure clique as described in Section 2.2.4. The scoring scheme used in SSAP for domain-vs-domain alignments involves normalisation over the size of the largest protein. When aligning a chain against a domain, the length of the putative domain region in the chain is unknown. Hence, it was decided to take the length of the region of the chain that had been assigned to the matched domain as a substitute of the largest domain size.

2.3.2.3 Excising the top hit and re-scanning

The hits are then re-ranked by an SVM (see Section 2.2.5) score and the top hit is excised from the chain. The assigned regions are determined by the SSAP alignment, which provides a list of residues in the chain which are to

be excluded for future searches of the library. A new secondary structure graph for the chain is created, where any secondary structures within the assigned region are excluded. If the chain contains more than 3 secondary structures it is then re-scanned against the library using GRATH.

Potentially valid fold representatives are selected as before and passed to SSAP. After the first iteration, SSAP was modified to exclude any residues assigned to the previous domain. The reasons for this are two-fold. Firstly, and most importantly, it avoids SSAP attempting to align residues that have already been assigned. Although the SSAP score penalises gaps as it assesses potential alignments, it also looks for the best global alignment along the length of the whole chain. Excluding these residues therefore increases the chance that SSAP will find the best alignment to the hit domain. This is especially pertinent when aligning discontiguous domains, as the gaps between segments are not penalised. Secondly, SSAP can be very slow to align large chains, so excising previously assigned regions can reduce the search space and hence enhance the speed. CATHEDRAL continues for up to 10 iterations or until there are less than 5 secondary structures left to be assigned.

2.3.2.4 Scanning the small library and collating results

After cycling through the iterative scans against the large library of domains, the remaining stretches of the protein chain are compared against the small library. The top 10 folds are selected as before for SSAP alignment. The results of all the GRATH and SSAP comparisons so far are then collated and written out as a list of hits, ranked by their SVM score.

2.3.2.5 Analysis of SVM score

The CATHEDRAL algorithm was used to generate a list of potential domain matches to all chains in the dataset. The parameters described in Section 2.2.5 were used to train the SVM and it was optimised using five-fold cross-validation. A ROC curve analysis was used to assess the performance of

different scoring schemes (Grath E-value, SAS, RMSD) at low error rates, as this would be crucial in determining the correct fold match for each domain in the chain. As can be seen in Figure 2.9, the SVM score outperforms all other measures of structural similarity, with RMSD performing the worst. At a false positive rate of 0.02, the SVM has coverage of 0.70 compared to less than 0.60 for the SAS score. Interestingly, the GRATH E-value curve has a greater area than the SAS and RMSD, despite only looking at secondary structure similarities. Overall, these data appear to confirm the hypothesis that combining alignment scoring features in the SVM is more effective at separating true and false fold matches.

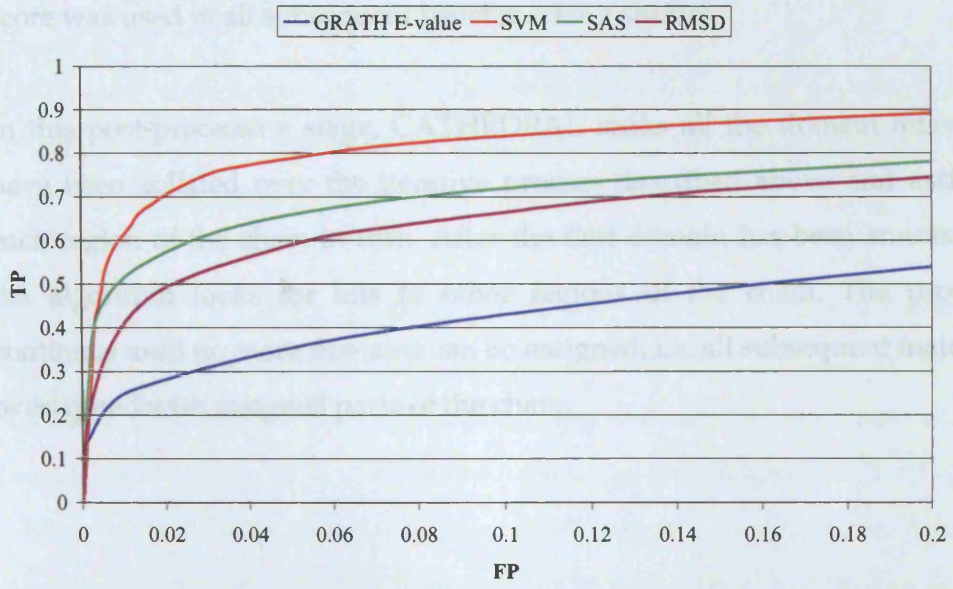


Figure 2.9 Comparison of GRATH, SSAP and RMSD scores with the SVM score for assigning domains to multi-domain chains.

2.3.2.6 Testing the algorithm

A non-redundant set of multidomain chains (see Section 2.2.2.2) were scanned against the domain database using the CATHEDRAL algorithm to assign domains. Any matches to domains with >35% sequence identity were discarded as 'trivial hits' that could be picked up by sequence methods (such as BLAST or HMMs), so assignments were only made for distant relatives. These may be genuine homologues or domains with similar folds.

2.3.2.7 Assigning folds and domain boundaries

Although folds are assigned to the chain during the iterative scanning algorithm, the final domain assignment is reserved until all the results have been collated. This is for two reasons. Firstly, although the top hits for each cycle were selected previously, they are not necessarily valid matches. CATHEDRAL does not use any empirical cut-offs in the first structure comparison stage, so there may simply be no valid fold match in the CATH library to a domain region. Secondly, the assignment of small domains (less than 5 secondary structures) is still required, in addition to domains with less than 3 secondary structures which cannot be identified by GRATH. The SVM score was used in all subsequent benchmarking studies.

In this post-processing stage, CATHEDRAL ranks all the domain hits that have been collated over the iterative process described above and assigns each region of the chain in turn. After the first domain has been annotated, the algorithm looks for hits to other regions of the chain. The process continues until no more domains can be assigned, i.e. all subsequent matches overlapped with assigned parts of the chain.

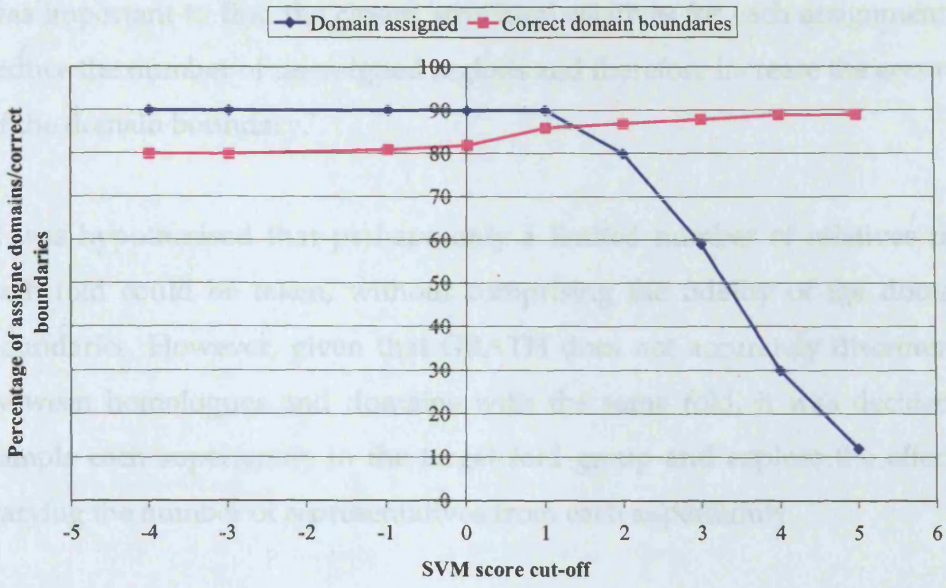


Figure 2.10 Percentage of domain assigned (blue) and percentage of domain boundaries within 10 residues of verified boundaries (pink) at a range of SVM score cutoffs.

Figure 2.10 shows that CATHEDRAL was able to assign 90% of domains in the query data set to the correct fold group, with 86% of these within 15 residues of the actual boundary. Although the data set only contained multi-domain chains where all component domains were represented in the CATH library, this is not always the case in classifying novel structures. Indeed, assigning erroneous folds to chains could adversely affect the quality of the domain boundaries. However, no improvement in domain boundary assignment performance could be achieved by increasing the SVM score cut-off above 1.5, suggesting that this is an appropriate threshold to use in CATHEDRAL.

2.3.2.8 Optimising number of fold representatives aligned in each iteration

The major speed increase in CATHEDRAL is due to the fact that GRATH pre-selects representatives for SSAP to align to the query chain. By default, it takes all relatives (SReps) in each fold group, even if this produces thousands of comparisons, as it does with large folds such as the Rossmann. This can

result in much longer running times for some query chains. Nevertheless, it was important to find the closest structural relatives for each assignment, to reduce the number of unassigned regions and therefore increase the accuracy of the domain boundary.

It was hypothesised that perhaps only a limited number of relatives from each fold could be taken, without comprising the fidelity of the domains boundaries. However, given that GRATH does not accurately discriminate between homologues and domains with the same fold, it was decided to sample each superfamily in the target fold group and explore the effect of varying the number of representatives from each superfamily.

CATHEDRAL was run as described above (by targeting the top 10 fold groups at each iteration) but the number of representatives (*fr*) taken from each superfamily to be aligned by SSAP was varied. Figure 2.11 shows the number of correctly assigned domain boundaries (within 15 residues of manually validated boundary) at each of these levels. It appears that taking any more than 7 reps does not increase the number of good assignments and hence was an appropriate level to set the *fr* parameter.

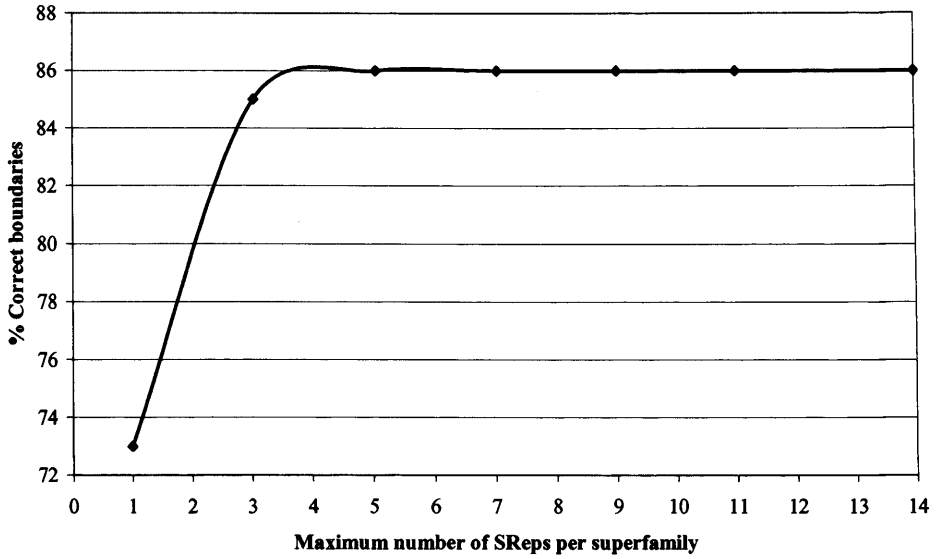


Figure 2.11 Percentage of domains with correct domain boundaries (within 15 residues) when varying the number of representatives taken from each superfamily in the targeted fold groups.

2.3.2.9 Correcting domain boundaries

When CATHEDRAL determines which fold to assign to a region of the protein chain, it is also making judgement of where the domain boundaries lie. The fidelity of this latter process is arguably dependent on the structural similarity between the domain region in the chain and the domain it has matched in the library. A number of methods were employed to increase the accuracy of the boundaries.

Firstly, domains were allowed to overlap by a maximum of 30% of their length with other assigned domains. This conflict was resolved by assuming that the highest scoring domain is most likely to have the correct boundaries. The boundaries of the other domain were cropped to exclude the shared region.

Secondly, some chains may contain small regions at the start and end that are unassigned. This was often less than 20 residues and unlikely to contain

another domain, or comprise an additional segment of a discontiguous domain. In these instances, CATHEDRAL assigns the extra residues at the beginning and end of the chain to the first and last domains respectively. Similarly, some chains contained small regions between assigned segments belonging to different domains. In these cases the algorithm, splits the unassigned residues equally between the two neighbouring segments.

Dealing with discontiguous domains has been found to be problematic with other domain boundary prediction algorithms (Jones *et al.*, 1998). For CATHEDRAL, one of the reasons for this is that even domains with the same fold can vary considerably in size (Reeves *et al.*, 2006). Therefore, it is difficult to determine whether an insertion in the alignment between a given matched domain to the query chain is genuine, or indicates that the gap is part of another fold in the chain. The algorithm deals with this by re-examining the chain for unassigned regions after all domains have been allocated. For a gap of less than 40 residues, it looks to see whether other assigned domains have residues that have aligned to residues in that gap and extends these to create two discontiguous domains. If not, it assumes there is an insertion and extends the size of the initial domain accordingly.

2.3.2.10 Domain assignment vs sequence identity

Figure 2.12 shows the relationship between the accuracy of the domain boundary and the sequence identity between the assigned domain region and best structural match used to assign the boundary. As sequence identity increases above 10%, there is an increase in the number of correct domain boundaries. It might be expected that the closer the relative from which the assignment is made, the greater chance of it being correct. However, it is encouraging to note that 60% of assignments with sequence identities between 5% and 10% show very little deviation from the manually verified boundaries.

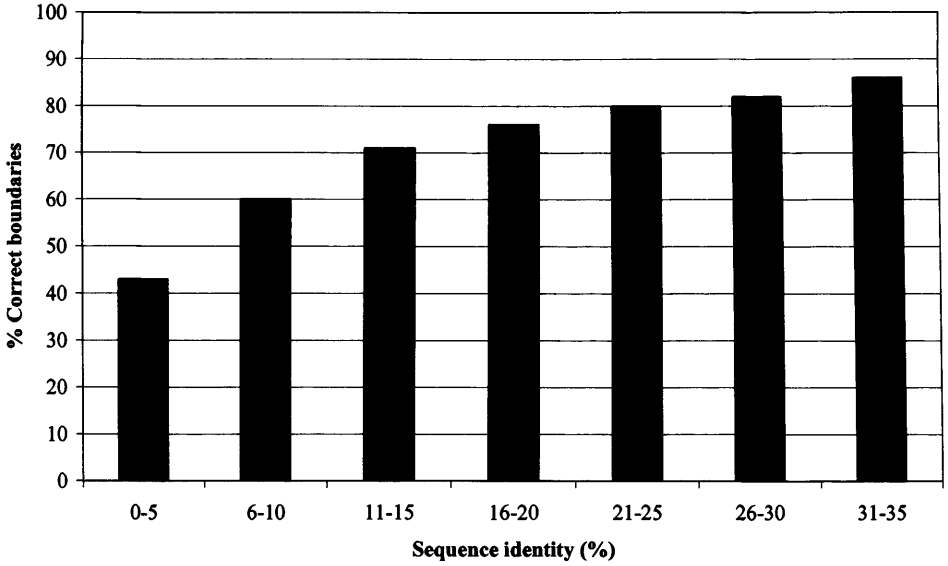
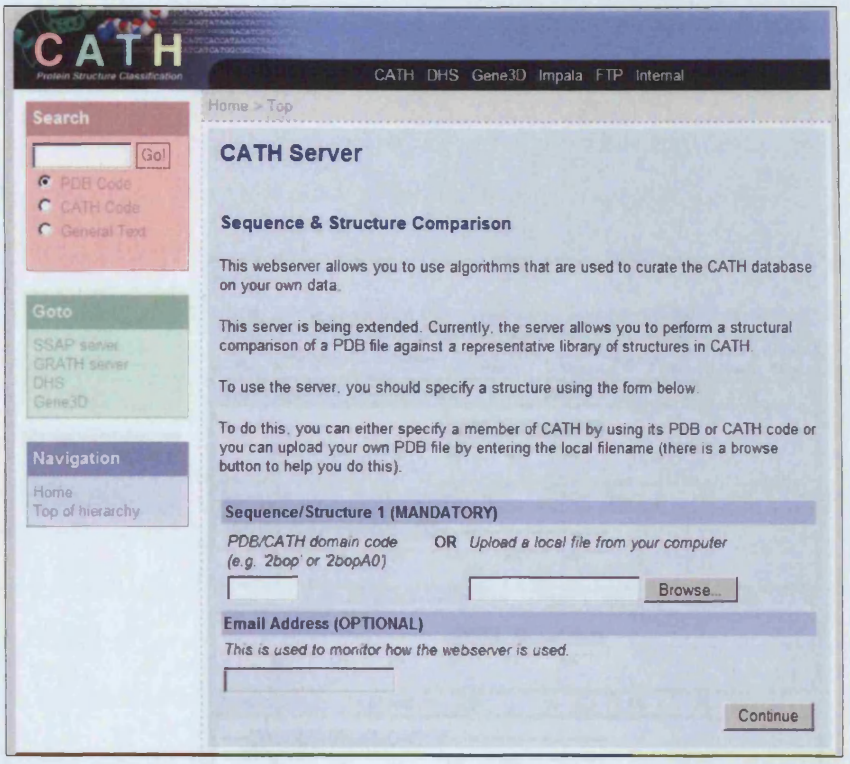


Figure 2.12 A plot of the percentage of correct (within 15 residues) domain boundaries against the sequence identity between the assigned region and the matched domain

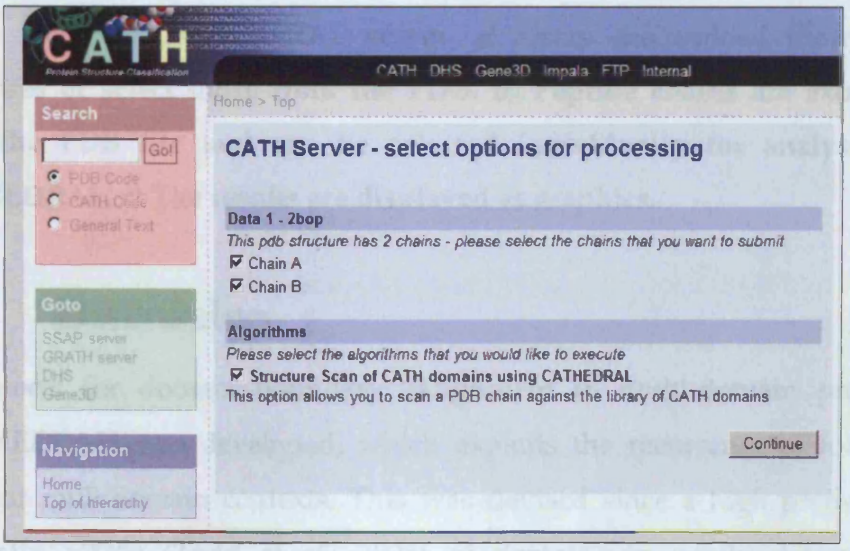
2.3.3 The CATHEDRAL Server

The structural comparison and domain assignment methods in this chapter were implemented as a server on the World Wide Web for access to the bioinformatics and structural biology community. Users can upload their own structures in PDB file format or use a PDB code to access the structures files stored at University College London (UCL) on a mirror of the PDB (Figure 2.13a). Once submitted, the PDB file is analysed and only peptide chains are selected. The user can then choose which chains they wish to submit for analysis (Figure 2.13b). Domain hits to each chain are displayed graphically in the context of the query chain (Figure 2.13c).

a)


CATH
Protein Structure Classification
CATH DHS Gene3D Impala FTP Internal
Search
Go!
PDB Code
CATH Code
General Text
Goto
SSAP server
GRATH server
DHS
Gene3D
Navigation
Home
Top of hierarchy
Home > Top
CATH Server
Sequence & Structure Comparison
This webserver allows you to use algorithms that are used to curate the CATH database on your own data.
This server is being extended. Currently, the server allows you to perform a structural comparison of a PDB file against a representative library of structures in CATH.
To use the server, you should specify a structure using the form below.
To do this, you can either specify a member of CATH by using its PDB or CATH code or you can upload your own PDB file by entering the local filename (there is a browse button to help you do this).
Sequence/Structure 1 (MANDATORY)
PDB/CATH domain code (e.g. '2bop' or '2bopA0')
OR Upload a local file from your computer
Browse...
Email Address (OPTIONAL)
This is used to monitor how the webserver is used.
Continue

b)


CATH
Protein Structure Classification
CATH DHS Gene3D Impala FTP Internal
Search
Go!
PDB Code
CATH Code
General Text
Goto
SSAP server
GRATH server
DHS
Gene3D
Navigation
Home
Top of hierarchy
Home > Top
CATH Server - select options for processing
Data 1 - 2bop
This pdb structure has 2 chains - please select the chains that you want to submit
Chain A
Chain B
Algorithms
Please select the algorithms that you would like to execute
Structure Scan of CATH domains using CATHEDRAL
This option allows you to scan a PDB chain against the library of CATH domains
Continue

c)

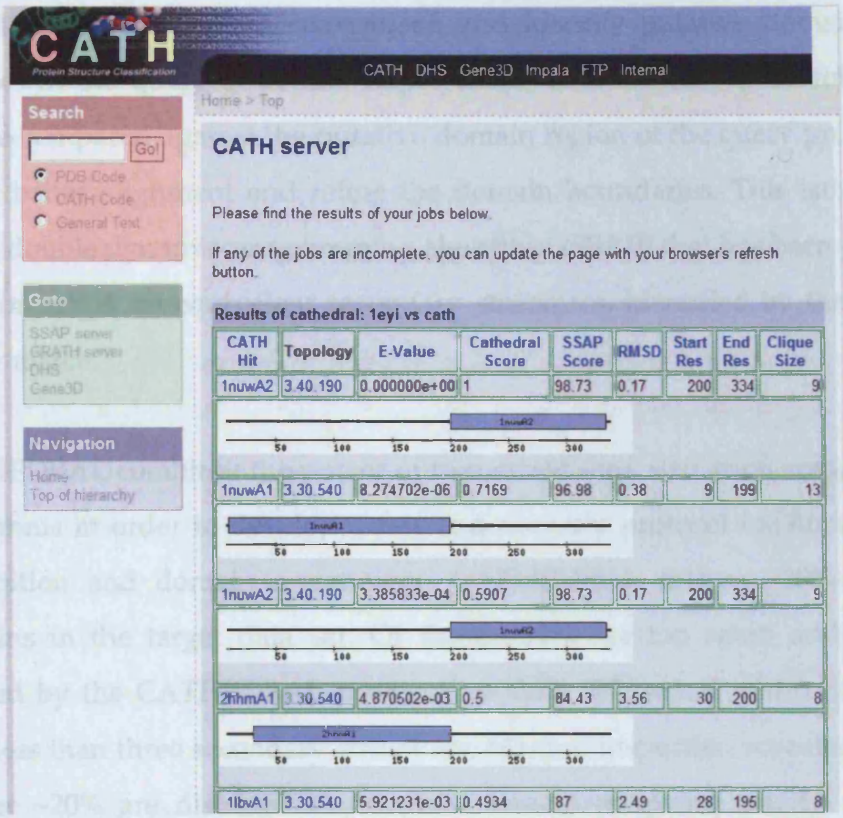


Figure 2.13 The CATHEDRAL server. a) Users can upload their own structures or select those from the PDB. b) Peptide chains are extracted from the PDB file and can be selected individually for analysis by CATHEDRAL. c) The results are displayed as graphics.

2.4 Discussion

A protocol for domain boundary assignment in multi-domain proteins (CATHEDRAL) was developed, which exploits the recurrence of folds in different multidomain contexts. This was devised since a high proportion (currently >90%) (Todd *et al.*, 2005) of domains in newly determined structures comprise folds which have been previously classified in CATH.

CATHEDRAL scans a query structure against a library of folds from the CATH databases. The algorithm first exploits graph theory to perform a secondary structure-based comparison and identify putative domain fold matches in the query structure. Representatives from the top 10 folds are then re-compared against the putative domain region of the query protein to obtain better alignment and refine the domain boundaries. This latter step uses a double dynamic programming algorithm (SSAP) that has been guided by information on equivalent secondary structures, identified by the graph theory match.

CATHEDRAL combines the power of two established structural comparison algorithms in order to develop a fast and accurate protocol for homologue recognition and domain assignment. CATHEDRAL misses ~10% of the domains in the target data set. Of these ~30% are too small and so are ignored by the CATHEDRAL protocol, as GRATH cannot match domains with less than three secondary structures. Manual inspection revealed that a further ~20% are distorted or irregular structures giving poorly defined graphs. The remaining ~50% are missed because they do not pass the score similarity cut-off, as the relatives are too distant and related structural motifs in neighbouring fold groups are better matched. This percentage should reduce as new structures are solved and CATH becomes more highly populated.

The CATH classification of protein folds gives a discrete description of fold space (Orengo *et al.*, 1997). However, there are difficulties in identifying distinct folds in some populated regions of fold space where the structural universe can more reasonably be represented as a continuum (Orengo *et al.*, 1994). In many cases, as the size of the protein increases, the repertoire of folds appears to consist of extensions to existing motifs. It has been shown by Koppensteiner et al. (2000) that it is possible to "walk" from one α/β sandwich fold to another, through the extension of α/β motifs. Furthermore, certain motifs, described as "attractors", occur as the core of a protein's

structure more frequently than others (Holm and Sander, 1996). Recent analyses of the overlaps between fold groups has shown that for some protein architectures (αβ sandwiches and mainly-β sandwiches) extensive overlaps between fold groups are observed due to large common structural motifs (Harrison *et al.*, 2002).

For 86% of the multi-chain data set, all domain boundaries within the multi-domain were correctly assigned within 15 residues of the true boundaries. This is a considerable improvement over a previous consensus protocol (DBS, (Jones *et al.*, 1998)), described above, for which on average only 10-20% of domains could be identified as having reliable boundary assignments from agreement between 3 independent methods. Especially since domain folds recognised by CATHEDRAL can be simultaneously classified in the CATH database, without the need for further structure comparison as in previous classification protocols (Orengo *et al.*, 1997). Furthermore, the data set used excluded hits with > 35% sequence identity, which would be non-trivial for a sequence-comparison method to identify.

Since CATH aims to maintain high quality domain boundary assignments (Veretnik *et al.*, 2004), results returned by the CATHEDRAL algorithm will be manually assessed. However, the high accuracy of the approach will considerably facilitate this process. Since the proportion of domain folds classified within CATH is likely to increase significantly over the next decade, due to the progress of the structural genomics initiatives, the CATHEDRAL algorithm will considerably enhance the speed of classification of new multi-domain structures and their constituent folds within CATH.

2.5 Future Work

As discussed above, CATHEDRAL generally fails to assign domains boundaries correctly when GRATH misses the correct fold in the list of hits

passed to SSAP. Smaller alpha domains cause the most difficulty and work is in progress to separate these into a library that can be compared purely using SSAP. As these contain fewer residues, it should not increase the overall runtime significantly.

Although a difference of 15 residues between the CATHEDRAL result and manually validated boundaries for 86% of query chains is fairly small, it could certainly be improved. The post-processing of CATHEDRAL results to assign domain boundaries presented here is fairly basic. For example, gaps between assigned domains are resolved simply by placing the domain boundary in the centre of the gap. Although this may still be useful if the data are to be subsequently adjusted manually, it does not lend itself to full automation. Work is now in progress to implement a more sophisticated decision algorithm that takes each residue in unassigned regions and calculates its proximity in three-dimensions to assigned domain regions in the chain. It also takes into account secondary structure e.g. preferring not to place domain boundary within a beta sheet or alpha helix.

Another problem that arises is that of unassigned regions or fragments in the CATH domain definitions file. Removing regions of coil at the termini of protein structures and domain linker regions is often desirable before assigning domain boundaries, as it produces neater definitions for sequence profile comparisons. However, this can be confusing for CATHEDRAL, as it aims to assign as much of the chain as possible. Currently, post-processing techniques to detect domain linkers and disordered termini is being explored by seeking sections of solvent accessible residues.

Chapter 3 FLORA: Using structural data to build functional templates

3.1 Background and Aims

One of the major goals of molecular biology is to understand the functions of all genes in nature, both biochemically and in the context of the cell. Bioinformatics techniques, such as sequence and structure comparison, can aid the functional annotation of novel genes by finding homologous relationships with experimentally characterised proteins. However, no methods are currently able to achieve 100% accuracy, as the level of global similarity required to transfer function varies considerably between protein families. The inherent problem with relying on overall sequence or structural homology is that even small mutations can inactivate a catalytic site or change the binding partners of a protein; hence, modifying its function. To further complicate matters, training a functional prediction algorithm relies on the assumption that proteins with the "same function" can be grouped together in the first place. Although this concept may be fairly straightforward when looking at related enzymes which perform equivalent functions in two similar organisms (orthologues), it becomes more complex when attempting to transfer function between more distant evolutionary relatives where many aspects of their role in the cell might have been modified. Therefore, any prediction method must seek to clearly define the level(s) of functional similarity it is trying to detect (e.g. catalysing the same chemical reaction). In order to enhance both the scope and fidelity of *in silico* predictions, it is vital to develop a better understanding of the sequence-structure-function paradigm and how it relates to different levels of functional conservation.

The success of large-scale genome sequencing projects has provided a flood of genomic data; however, our knowledge of the three-dimensional structure of the proteins they encode is far more limited. This is primarily due to the substantial experimental overheads involved in crystallising proteins compared to sequencing DNA. Hence, over the last six years the Structural Genomics Initiatives (SGIs) (Todd *et al.*, 2005) have sought to redress the balance, by targeting protein families where little or no structural data was present in the Protein Data bank (PDB) (Berman *et al.*, 2000) – more specifically, those families whose genes are more likely to adopt novel folds. Advances in high-throughput robotic techniques then allow multiple experimental parameters to be explored simultaneously, drastically reducing the time taken to grow viable crystals. This approach is in sharp contrast to that taken by crystallographers over the last 50 years, where structures were determined to complement experimental data for well-characterised genes. As a result, an increasing number of structures being deposited in the PDB come with little or no functional annotation (frequently denoted as 'hypothetical proteins'). This compounds the practical problems associated with assigning new domains to superfamilies in the CATH database (Orengo *et al.*, 1997).

Pair-wise sequence comparison algorithms, such as BLAST, are still commonly used to assign function by identifying close relatives which perform the same biological function. However, several groups (Todd *et al.*, 2002a; Rost, 2002; Tian and Skolnick, 2003) have highlighted the need to apply simple pair-wise identity cut-offs with caution. Where a close homologue cannot not be found for a given query protein, sequence profile methods (HMMs (Eddy, 1996), PSI-BLAST (Altschul *et al.*, 1997)) can be used to detect more distant evolutionary relationships and identifying proteins that may perform the same function. The power behind these methods is due to the ability of profiles to detect patterns of amino acid conservation that are specific to a given protein family, rather than applying universal rules across

the whole of sequence space. This permits the construction of resources, such as Pfam (Bateman *et al.*, 2004), where sequence domains are grouped according to a common evolutionary source, which often correlates with function. However, to maintain the accuracy of this approach requires extensive manual adjustment of multiple alignments and HMM cut-off values for an individual family, as it remains problematic to construct universal rules about the sequence-function relationship.

As structure is more conserved across protein families than sequence (Chothia and Lesk, 1986), structure comparison methods are able to detect far more distant relationships than the most powerful profile methods. However, even domains in the same superfamily can exhibit large amounts of structural variation (Reeves *et al.*, 2006). This may be due to different protein or domain interactions, or requirements to attach to distinct cellular environments, or might simply be due to random evolutionary drift. Consequently, these structural deviations can mean that even an accurate alignment of two structures can produce a global similarity score that falls below reliable thresholds for transferring a specific function.

In a similar vein to the way PRINTS (Attwood *et al.*, 2003) and PROSITE (Hulo *et al.*, 2006) focus on smaller conserved sequence patterns, there are several approaches to identifying local structure motifs that are associated with specific functions. For example, the Catalytic Site Atlas (Porter *et al.*, 2004) concentrates on building 3D motifs of residues that are directly involved in ligand binding or the catalytic mechanism in an enzyme. As *ab initio* prediction of functional residues is a complex problem in itself, the Thornton group at the European Bioinformatic Institute (EBI) have focussed on mining the primary literature to obtain the information on which to build templates. Torrance et al. (2005) analysed the performance of this approach for enzymes with more than 2 catalytic residues. They were able to discriminate related proteins from random with 85% accuracy and found that it was important to focus on C-alpha/C-beta residues as their position is

better conserved than side chain atoms. However, even by capturing the correct functionally active residues — for example, the catalytic triad in the serine proteases — the flexibility of active sites significantly impacts on the ability of these templates to detect these mobile residues in X-ray crystal structures with different bound ligands.

In contrast to exploiting information on known functional residues, the DRESPAT method (Wangikar *et al.*, 2003) uses graph theory to extract recurring structural patterns across superfamilies in the SCOP database (Murzin *et al.*, 1995). DRESPAT makes no assumptions about the location or nature of the motif positions, except by excluding hydrophobic residues. A statistical model is built to assess the significance of each recurring pattern and the authors were able to identify different metal binding sites in distantly related proteins. However, as with many methods which seek small structural motifs, distinguishing between genuine similarities and background is hampered by high false positive rates.

The PINTS methods (Stark and Russell, 2003) also shows promise for automatically detecting structural motifs in protein families, although is not able to annotate novel proteins with high accuracy. Again recurring side chain patterns are identified through a pair-wise comparison of diverse members within a protein family. These motifs can then be used to scan against a novel structure.

Instead of detecting 3D templates based on their structural conservation across an enzyme family, Polacco and Babbitt (Polacco and Babbitt, 2006) used a genetic algorithm (GASP) to generate a functional template from a given structure based on its ability to identify members of the same enzyme superfamily against a background of unrelated proteins in the SCOP database. An initial PSI-BLAST step builds a multiple sequence alignment for each enzyme structure that is used to create a set of conserved residues, from which a small number (~10) are selected at random to build a template. The

performance of each template is then evaluated by using a geometric matching algorithm, SPASM, to score matches to the functional relatives and the SCOP library. Interestingly, the best template generally contains known functional amino acids, although there are also a few additional residues with no known functional role. This method is a promising development, although each template takes up to 18 hours to generate and the performance was only evaluated for five superfamilies.

The central goal of this work was to exploit sequence and structural data to detect conserved patterns in protein families that recur in enzymes with similar catalytic mechanisms, as defined by their E.C. number. A novel algorithm, FLORA, was developed to analyse multiple structural alignments of domains in these families and discover a conserved motif. Patterns of sequence conservation and residue accessibility were combined with structural conservation data to identify these motifs, which were then encoded into templates and compared against new structures using a graph matching program, FLORAScan. The primary focus of the method was to discriminate between domains from different enzyme families, yet having a common evolutionary origin (i.e. from the same CATH superfamily).

3.2 Methods

3.2.1 Overview of methods

This section will outline the creation of a data set of enzyme families from diverse superfamilies and the more technical details of the FLORA algorithm. All the optimisation studies and a full outline of the algorithm are presented in section 3.3.

3.2.2 Multiple structure alignment using CORA

The CORA multiple alignment program (Orengo, 1999) is an integral part of the FLORA algorithm. Based on the double-dynamic programming (DDP) approach used in SSAP (described in Chapter 2), CORA uses an iterative

alignment approach to build a multiple structure alignment of protein domains, which can be used to analyse structural conservation within a superfamily or fold group. For example, Reeves *et al.* (2006) previously showed how a CORA alignment can be used to analyse structural changes across functionally variable superfamilies. The addition of secondary structure embellishments, for example, can modulate the active site and facilitate functional divergence.

CORA begins by calculating a SSAP pairwise comparison between all unique protein domain pairs to be aligned. Starting with the closest pair of structural relatives, vectors between C_β atoms are compared to score the similarity of the structural environments of residues pairs. The first stage of dynamic programming is then used to find putative alignment paths through the matrix of scores for each residue pair. High-scoring paths above a certain score threshold are added to a summary matrix. The top 20 highest scorings residue pairs are then recalculated and used to populate a final summary matrix, which is then subjected to a second stage of dynamic programming to discover the optimal global alignment of the two domains. From this alignment, equivalent residue pairs are selected and used to build a 'consensus structure' based on the average vectors between aligned residues. The next domain in the list is then selected and aligned to the consensus, using the same double dynamic algorithm. This iterative protocol is applied until all domains are aligned and a full multiple structure alignment has been calculated.

3.2.3 Data set: selecting enzyme families from diverse superfamilies

Domains in v2.6 of CATH were annotated with a 4 digit Enzyme Classification (E.C.) number using PDBSProtEC (Martin, 2004). Protein domains frequently do not have a clearly delineated enzymatic function of their own, hence the E.C. number (Bairoch, 2000) assignment merely

designates them as a component of the enzymatic function of their protein chain. Furthermore, the quaternary structure of a protein can result in a complex that catalyses more than one chemical reaction and may possess multiple E.C. numbers. For simplicity, domains in this category were removed from the dataset.

The first 3 E.C. numbers describe the overall catalytic mechanism performed by the enzyme, whereas the 4th generally denotes the substrate specificity. Preliminary analysis revealed a number of superfamilies that contained E.C. annotations which only deviated in their 4th digit. It was hypothesised that structural templates could capture this conserved catalytic framework. A group of domains which share their first 3 E.C. numbers will be subsequently referred to as an *enzyme family*. A data set of CATH superfamilies predicted to contain more than one of these enzyme families was compiled for testing the FLORA algorithm.

386 highly populated superfamilies (> 3 SReps) in CATH were analysed and reduced initially to 71 superfamilies, containing at least one enzyme family with three or more SReps (redundant at 35% sequence identity) and complete functional annotation. Of these, only 12 superfamilies contained more than one different enzyme family, resulting in a total of 21 enzyme families. The domains in these 21 families were selected to comprise a *dataset* for testing FLORA. For all families in this dataset, a representative was removed to construct a test data set. The remaining SRep relatives in the dataset were used to build templates for the corresponding enzyme family. This was done using a jack-knifing approach whereby all domains were used as the test domain at some stage – this produced 125 test domains with 125 different templates.

3.2.4 CoraXPlode

After using CORA to produce multiple structure alignments of each enzyme family, a modified version of the CoraXPlode protocol (Sillitoe *et al.*, 2005)

was applied to search for related sequences in the UniProt90 NRDB (Apweiler *et al.*, 2004) that could be used for sequence conservation analysis in the FLORA algorithm.

The first step in the original version of CoraXPlode is to take the sequence of each domain in the enzyme family and build a HMM profile using the SAM-T99 program. Each profile is then used to search for related sequences in UniProt90 NRDB (Wu *et al.*, 2006). However, in this case, a more conservative profile was desired that would be biased towards closer relatives of the query enzyme sequence where function was conserved, i.e. mainly orthologous sequences. SAM-T99 was replaced by PSI-BLAST using a 1×10^{-40} E-value cut-off with 10 iterations, in accordance with the work of Bartlett *et al.* (2002) that examined conservation patterns of catalytic residues in known enzyme structures.

After CoraXPlode has identified close sequence relatives for each structure in the enzyme family, these sequences need to be integrated into the multiple structure alignment. Instead of realigning the new relatives with the original structures, these sequences are simply inserted into the CORA alignment according to their alignment to the query sequence given by PSI-BLAST (see Figure 3.1). Any extra residues in the UniProt sequences that are not present in the query structures are then discarded.

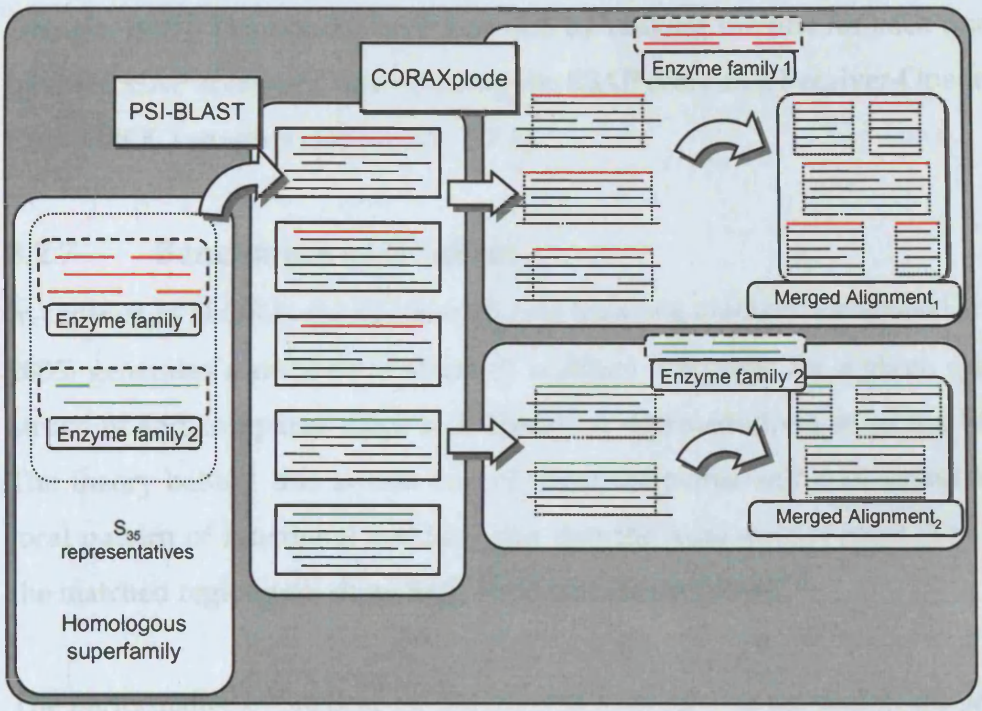


Figure 3.1 Flowchart showing main steps in the CoraXPlode protocol

3.2.5 Benchmark of PSI-BLAST

In order to place the performance of FLORA in context, it was compared against PSI-BLAST. PSI-BLAST (Altschul *et al.*, 1997) was chosen as an established standard method for assigning function and the performance was measured by taking all domains in the enzyme data set as query sequences. These sequences were also embedded in the Uniref90 database (Apweiler *et al.*, 2004) to allow for PSI-BLAST to build a sufficient profile. An E-value cut-off of 3×10^{-3} was used for acceptance into the profile at each iteration, with an overall E-value cut-off of 1×10^{-3} for hits over 5 iterations. These parameters are identical to those used by George *et al.* (2006) for searching for enzyme homologues.

3.2.6 Benchmark of SSAP

PSI-BLAST does not make use of structural data, so it could be argued that it is at a disadvantage compared to FLORA. Therefore, to assess the performance of FLORA with respect to global structure comparison, all domains in the data set were aligned and scored using SSAP (Taylor and

Orengo, 1989). The results were assessed by ranking the hits for each query by their SSAP score and also by using the SSAP score in a Receiver-Operator Curve (ROC) analysis (see Section 3.3.6).

3.2.7 Benchmark of SiteSeer

In contrast to FLORA, the SiteSeer reverse template method (Laskowski *et al.*, 2005) generates a number of small (3 residue) templates for a given query structure and compares these to a library of representatives from the PDB. The theory behind this is that one of these templates will correspond to a local pattern of functional residues and that the local environment (10Å) of the matched region will show high sequence conservation.

The performance of SiteSeer on the diverse data set was applied in the same way as SSAP, by using each domain succesively as a query and comparing this to the remainder of the data set. The program was kindly run and results were provided by Roman Laskowski at the European Bioinformatic Institute in Cambridge, UK.

3.3 Algorithm Development and Results

This section outlines the main steps of the FLORA algorithm to produce templates for enzyme families within CATH superfamilies and the optimisations that were undertaken. A second program, FLORAScan, was developed to compare these templates against the enzyme test set from CATH v2.6. In addition, the performance of FLORA templates was compared to PSI-BLAST, SSAP, CORA and the SiteSeer template method.

3.3.1 FLORA – designing structural templates specific for catalytic function

The main steps of the FLORA algorithm are outlined in Figure 3.2.

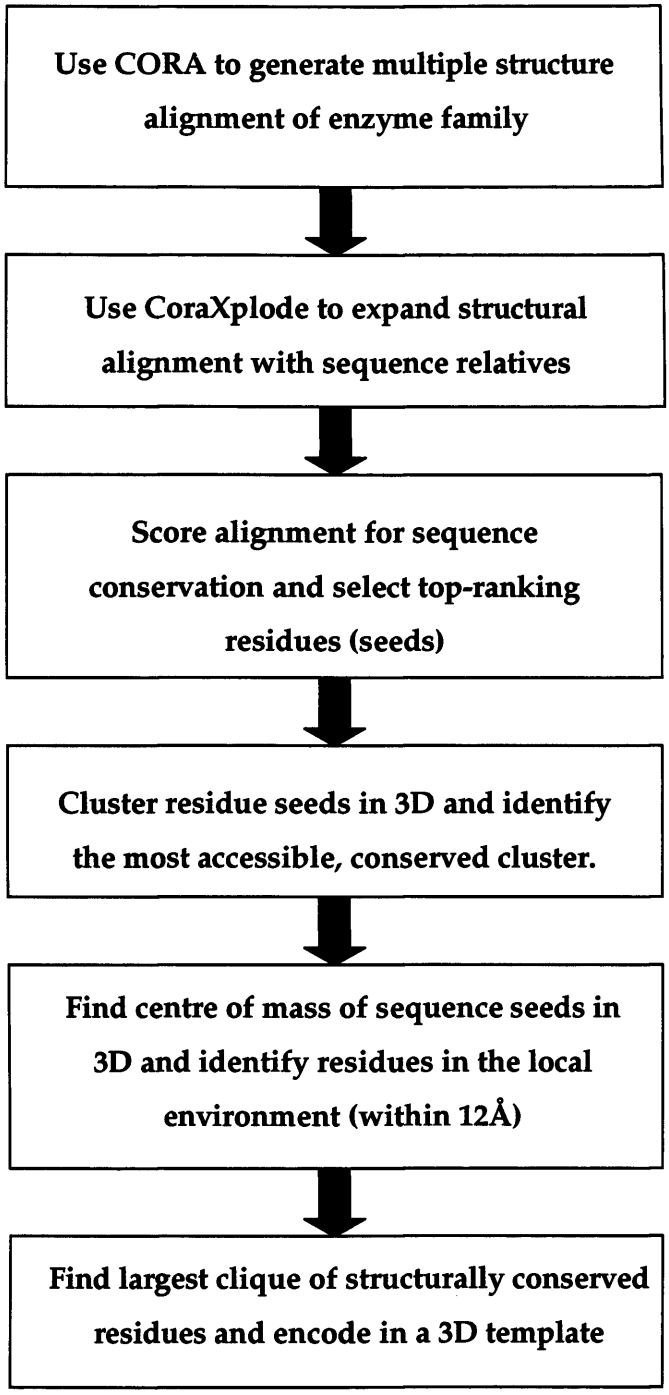


Figure 3.2 Flow diagram of main steps in FLORA algorithm used to generate a 3D template for enzyme families.

3.3.1.1 Generating multiple alignments using CORA

All the SRep domains in the data set (described in Section 3.2.3) for a given enzyme family were aligned using the multiple structural alignment algorithm CORA. For those domains with annotations in the Catalytic Site Atlas (CSA), a sample of the alignments was inspected manually to confirm that they had been aligned correctly with respect to known catalytic residues.

3.3.1.2 Expanding alignments with sequence relatives

In order for FLORA to predict a putative functional site, it first requires an analysis of sequence conservation at each position in the alignment. Since many of the enzyme families contained as few as three structural domains, extra sequence relatives were required to more accurately calculate the sequence conservation at each alignment position. The CoraXPlode protocol (see Section 3.2.4) was utilised to expand the CORA alignment with sequence homologues for each domain in the alignment by generating a relatively conservative PSI-BLAST profile (as described in Section 3.2.4).

3.3.1.3 Calculating sequence conservation using ScoreCons

An optimised version of ScoreCons (Valdar and Thornton, 2001) (re-implemented in the C programming language) was used to calculate sequence conservation at each position in the multiple alignments of each enzyme family. A Diversity of Positions (DOPs) score was calculated as described in Valdar and Thornton (2001) to measure the evolutionary variation in the multiple alignment. Although ScoreCons accounts for sequence redundancy across the alignment, conservation scores at each position are only considered accurate if there is sufficient overall sequence diversity across the multiple alignment. This is reflected in a DOPs score of greater than 0.9. All alignments were found to meet these criteria, which may be expected as the original structural domains shared less than 35% pair-wise sequence identity.

3.3.1.4 *Identifying and clustering sequence-conserved alignment positions in 3D to locate the functional site*

Many function prediction methods, such as the evolutionary trace (Yao *et al.*, 2006; Lichtarge *et al.*, 1996), rely on the premise that residues that are highly conserved across protein families are important for function and can therefore be used to locate the functional site. However, work on protein folding has also shown that hydrophobic residues in the core are often also well conserved. These are thought to act by promoting stability through the formation of intermediates in the folding pathway (Mirny and Shakhnovich, 2001). Others (Bartlett *et al.*, 2002; Wangikar *et al.*, 2003) have reported that catalytic residues are far more likely to be polar residues. Accurately predicting residues that may be involved in substrate binding or catalysis based on sequence conservation is a more challenging problem than defining the general area of the functional site. Consequently, a straight-forward, yet effective, approach was taken with the FLORA algorithm.

All positions in the multiple alignments were ranked by the sequence conservation calculated by ScoreCons. This set was reduced to only those positions where residues were present in all sequences (i.e. non-gapped positions in the alignment). The top 20 residues conserved by sequence were then selected (*sequence seeds*).

Manual inspection of a selection of enzyme families revealed that the sequence seeds tended to be present both within the active site and in the hydrophobic core of the protein, as expected. To bias the selection towards putative functional residues, all seeds where 80% or more of the residues were hydrophobic were eliminated. To generate a list of putative active sites, the remaining hydrophilic seed residues were clustered together using a complete linkage approach and a cut-off of 7Å. We initially used a cut-off of 5Å, in keeping with the Drespat method (Wangikar *et al.*, 2003) but manual inspection revealed this produced too many singleton clusters. For each

enzyme family, the collection of clustered residue positions will be referred to as the *seed clusters*.

3.3.1.5 Predicting a putative active site

Although there maybe several functionally-relevant regions of the domain, the aim of FLORA was to capture one active site associated with the catalytic action of each enzyme family. Hence, it was necessary to select one seed cluster from the previous stage that was most likely to co-locate with the active site. Across all enzyme families in the data set, it was found that clusters varied considerably in size and average sequence conservation. As active sites tend to possess conserved residues near the protein surface (Bartlett *et al.*, 2002), the solvent accessibility of each residue across the enzyme family was also calculated using the NAccess program (Hubbard and Thornton, 1993).

A small manual analysis was performed on 16 enzyme families where the catalytic residues had been annotated from the literature in the CSA. This revealed that the largest seed cluster with the highest accessibility and sequence conservation tended to co-locate with the catalytic residues. Moreover, only one seed cluster containing residues with these properties in each enzyme family overlapped with the known catalytic residues. It was observed that choosing the seed cluster (ignoring singletons) with the greatest sequence conservation and an average surface accessibility greater than zero produced the correct functional cluster for 80% of enzyme families. This is referred to as the *top seed cluster*.

3.3.1.6 Expanding the sequence seeds by selecting residues in the local environment of the predicted functional site

The goal of FLORA was to build a static template of structurally conserved residues important for function. However, catalytic residues often move during enzyme catalysis and hence might change their relative positions depending on whether a ligand is bound or not in the structure. This does

not only apply to functional residues and substantial change can be observed within the same protein, depending on the bound ligand or absence thereof. This compounds the problem of detecting regions of structural conservation between relatives that perform similar functions. To address this problem in FLORA, although the *top seed cluster* was chosen to predict the functional site on the domains in each enzyme family, it was hypothesised that other residues in the vicinity could provide a more static, structurally conserved framework. To identify these residues, the centre of mass (CoM) of the top seed cluster was calculated for each domain in the enzyme family. Any residues that fell within a 12Å of the CoM in each domain were used to generate a set of *site positions* (the top seed cluster residues being a subset of the site positions). As with the identification of the sequence seeds, alignment positions without a residue present in all domains (i.e. gapped positions) were excluded. Initially, a cut-off of 10Å was chosen as this is consistent with other function prediction methods (e.g. SiteSeer (Laskowski *et al.*, 2005)), but this did not identify a sufficient number of residues for the template sizes explored later in the protocol, hence the radius was expanded to 12Å. The process of identifying other residues in the locality of the active site is shown in the first 2 steps of Figure 3.3.

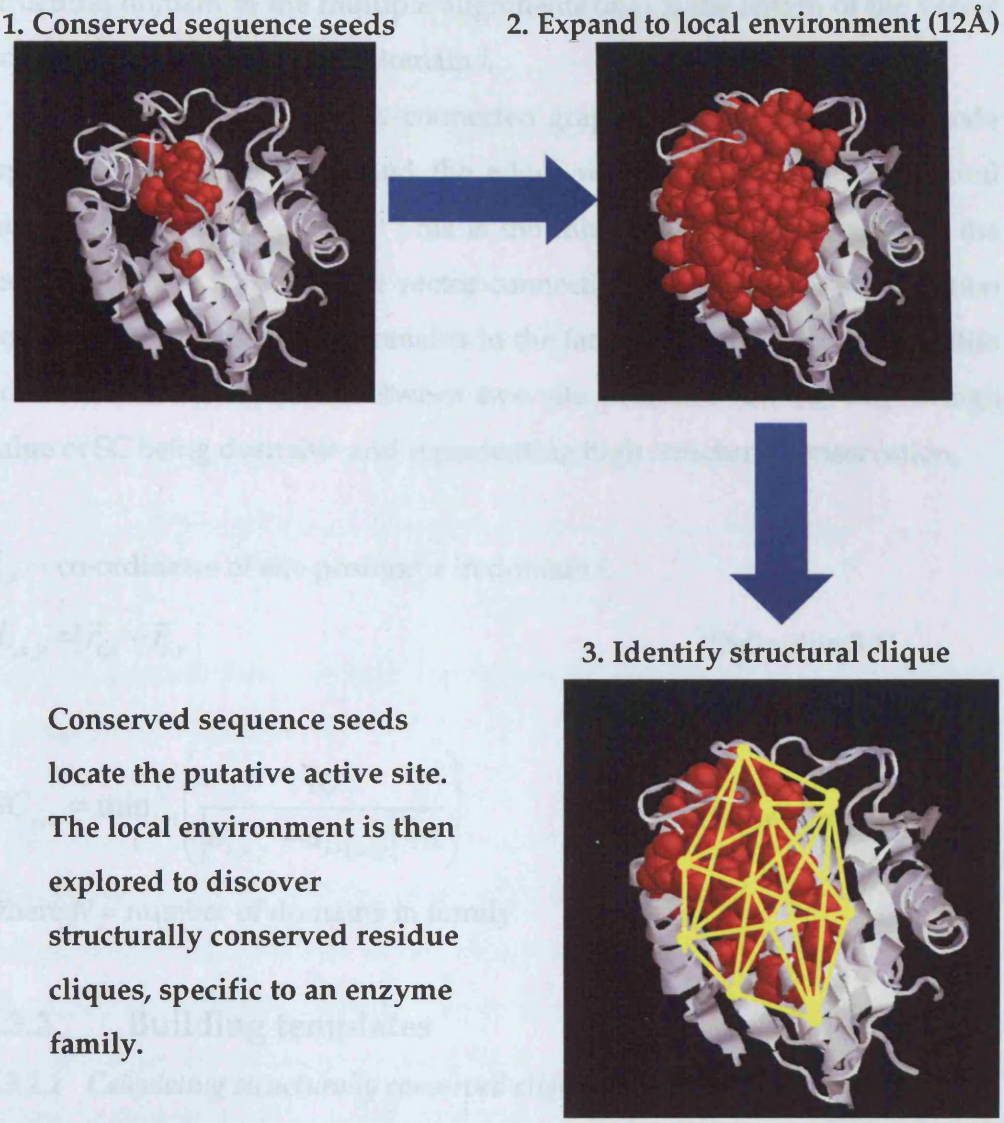


Figure 3.3 Structural representation of the major steps in the FLORA algorithm.

3.3.1.7 Calculating the structural conservation of the site positions

At this point in the algorithm, FLORA has predicted a putative functional site for each enzyme family and selected a set of *site positions*. The final step is to reduce this to a collection of structurally conserved positions, which can then be used to build a structural template associated with each enzyme function (Figure 3.3).

Vectors were calculated between all C_β atoms of the site positions in each structural domain in the multiple alignments ($d_{i,x,y}$ is the length of the vector between positions x and y in domain *i*,

(Definition 3.1). A fully-connected graph was built where each node represented a site position and the edges were labelled with a structural conservation measure, SC ( ). This is the minimum difference between the reciprocal of the length of the vector connecting the two alignment position nodes, *x* and *y*, across all N domains in the family. This essentially quantifies how variable the distance between two site positions can be, with a high value of SC being desirable and representing high structural conservation.

$\vec{r}_{i,x}$:= co-ordinates of site position *x* in domain *i*.

$$d_{i,x,y} = |\vec{r}_{i,x} - \vec{r}_{i,y}| \qquad \text{(Definition 3.1)}$$

$$SC_{x,y} = \min_{i=1}^{N}\left(\frac{10}{|d_{i,x,y} - d_{i+1,x,y}|+1}\right) \qquad \text{(Definition 3.2)}$$

where N = number of domains in family

3.3.2 Building templates

3.3.2.1 Calculating structurally conserved cliques of site positions

At this stage in the algorithm, each pair of site positions has been assigned a structural conservation score, *SC*, that represents the maximum variation observed across the multiple alignment. A logical approach would then be to select site positions for the template, which are above a given cut-off value for *SC*. However, Figure 3.4 demonstrates that even at relatively high cut-off values for *SC*, there is a vast range of template sizes when applied across all enzyme families. For example, using a cut-off of 1.9, the largest template contains 39 residues and the smallest only 3. This disparity causes problems as these large templates are slow to scan and those with few residues do not have sufficient power to effectively discriminate from false positives.

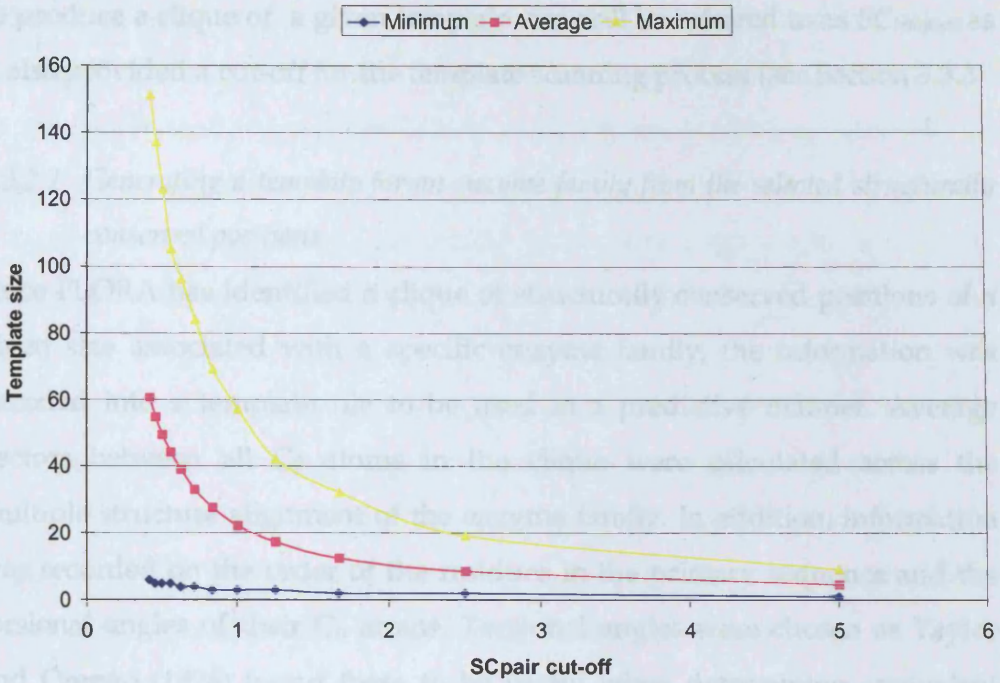


Figure 3.4 Minimum, maximum and mean size of templates generated over a range of SC cut-offs.

This result is not unexpected, as protein superfamilies exhibit different levels of structural variation within an enzyme family. The aim of FLORA was to capture a 3D configuration of residues that appears to be conserved within the functional site of an enzyme family. Given that the amount of variation is dependent on the plasticity of the fold, it was decided to optimise FLORA to obtain a template of a given size and develop specific cut-offs for each family. Therefore, a range of cut-off values for *SC* were explored for each enzyme family until a minimum template size was reached. This permitted the production of larger template for more flexible families.

To do this, a graph of site positions was constructed and edges were labelled by the value of SC. The Brøn-Kerbosch algorithm (Bron and Kerbosch, 1973) was then used to find the largest clique of positions in this graph. The value of SC cut-off was lowered until a template of the specified size was obtained. This variable was optimised in Section 3.3.4. The final cut-off that was used

to produce a clique of a given template size will be referred to as $SC_{template}$ as it also provided a cut-off for the template scanning process (see Section 3.3.3)

3.3.2.2 *Generating a template for an enzyme family from the selected structurally conserved positions*

Once FLORA has identified a clique of structurally conserved positions of a given size associated with a specific enzyme family, the information was encoded into a template file to be used in a predictive manner. Average vectors between all C_β atoms in the clique were calculated across the multiple structure alignment of the enzyme family. In addition, information was recorded on the order of the residues in the primary sequence and the torsional angles of their C_α atoms. Torsional angles were chosen as Taylor and Orengo (1996) found these to be useful when determining equivalent residues in structural alignment. Regardless of the algorithm used for scanning structural templates against novel proteins, it is useful to encode as much information in the template as possible to reduce the search space and potentially increase the fidelity of the matching.

3.3.3 FLORAScan – scanning the enzyme family test set domains against FLORA structural templates

3.3.3.1 *Template-matching algorithm*

A clique-matching algorithm, FLORAScan, was written to compare each template against new domains with the aim of identifying functional relatives. Graph-theoretical alignment methods that operate on the residue level are often slow as the graph involved is so large and highly connected. Although the problem is abated with smaller templates, comparisons with large domains can still be slow. The torsional angle and sequence order data in each template was used to reduce the search space and increase the speed of the graph matching algorithm.

For each comparison, vectors in the template were used to build a graph where the template positions were represented as nodes and the edges were

labelled with the length of the averaged vector calculated in the final stage of FLORA (Section 3.3.2.2). A similar graph was built from the C_β co-ordinates of the query domain, against which the template was being scanned. To assess the overlap between template residues and those in the query domain, a correspondence graph was calculated. Each node in this graph represented a template residue and a domain residue pair. Hence, the maximum size of the graph is equal to the number of template positions multiplied by the number of residues in the query domain. Nodes would then be connected by an edge if the distance between a pair of template residues was similar to a pair in the domain. As the speed of the clique detection algorithm (Bron and Kerbosch, 1973) is dependent on the number of edges in the graph, two initial conditions were added to reduce the size of the graph.

Firstly, edges were only constructed where domain residues shared the same sequence order as those in the template. For example, a node which corresponded to domain residue 42 (t_1) and template residue 51 (t_2) could be joined to a node representing domain residue 49 (d_1) and template residue 65 (d_2), as these are both vectors to residues further along the primary sequence. Secondly, a given template-domain residue pair in the correspondence graph must share comparable torsional angles (i.e. be within a *torsional cut-off*, which is optimised in Section 3.3.4).

The final condition for edge creation was a score based on distance similarity. The value of $SC_{template}$ that was used to produce the template for each enzyme family was implemented in FLORAScan. Another variable (*margin*) was implemented as an error tolerance, to be subtracted from $SC_{template}$ and allow each template to match positions in the query domain that may be correct, but where the inter-residue distances were larger than those observed in the alignment of the enzyme family. Hence, the distance between two template-domain residue pairs in the correspondence graph needed to satisfy (Definition 3.3.

$$\frac{1}{\left|{}^{template}r_{t_1t_2} - {}^{domain}v_{d_1d_2}\right| + 1} > SC_{template} - margin \qquad \text{(Definition 3.3)}$$

3.3.3.2 Scoring template matches to query domains

FLORA was designed to produce a template of residues conserved within all relatives of a specific enzyme family. Therefore, it should follow that when scanning these templates against new domains, template residues would constitute a complete sub-graph of the new domain in that family. However, the structural data is often incomplete, so FLORA may have selected residues in the template that are not always present in an enzyme family and hence may not be found in new functional relatives. Hence, it was decided to also explore the value of using a measure of overlap in scoring the match of a template to a query structure.

Hits were scored by RMSD, normalised by the number of matched residues:

$$FLORAScore = \frac{rmsd}{n+1}$$

where n = number of matched residues

The overlap is defined as:

$$Overlap = \frac{number \cdot of \cdot matched \cdot residues}{template \cdot size}$$

3.3.4 Parameter optimisation

The *dataset* described in Section 3.2.3 was used to optimise FLORA and FLORAScan to ensure that the algorithm was able to distinguish between structurally related domains with different functions, rather than simply detecting homologous relationships that can already by achieved effectively by methods such as SSAP. All 125 test domains were scanned against all 125 templates, generated by jack-knifing the *dataset*. Three parameters: the

template size, torsional angle cut-off, margin were optimised in turn. A range of overlap cut-offs was also explored for the template size optimisation.

3.3.4.1 Optimising the template size

Big templates tend to provide more structural information about residues that may be important for function, but they create larger graphs that are much slower to process with the clique detection algorithm. Conversely, small templates are fast to scan but highly likely to hit unrelated proteins. It was thus desirable to have as small a template as possible for each family, while retaining specificity.

Initial investigation showed that using a *margin* of 1.0 and a *torsional cut-off* of 100 found the correct enzyme family in the top 3 hits for the majority of enzyme families. Therefore, FLORA was used to build templates of sizes ranging between 10 and 28 using the latter cut-offs as defaults. An upper limit of 28 was chosen for practical purpose, as tests showed that templates above this size are very slow to scan with a negligible increase in performance. The performance was measured by calculating the number of domains in the data set that matched the correct template in the top 3 hits, when ranked by their FLORAScore. As FLORAScore is proportional to the RMSD of the residues in the matched clique, a small value indicates a better match.

Figure 3.5 shows a plot of the performance against the overlap cut-off for the range of template sizes, when taking the template for the enzyme family (built from the top seed cluster) to each domain in the test set. There is a clear trend that larger templates perform best, with a template size of 10 only ranking 38% of hits in the top 3 compared to 67% for a template size of 25. For most template sizes, the performance appears to drop when the overlap cut-off is set to 60 or above. However, for a template size of 16 or 19, the performance does rise by about 1% using an overlap cut-off of 50%. Given that the margin and torsional angle cut-offs were suboptimal at this stage, it

was decided to continue to evaluate the effect of using overlap to reduce the number of hits — this could also be applied in a post-processing stage, rather than making changes to the FLORA algorithm.

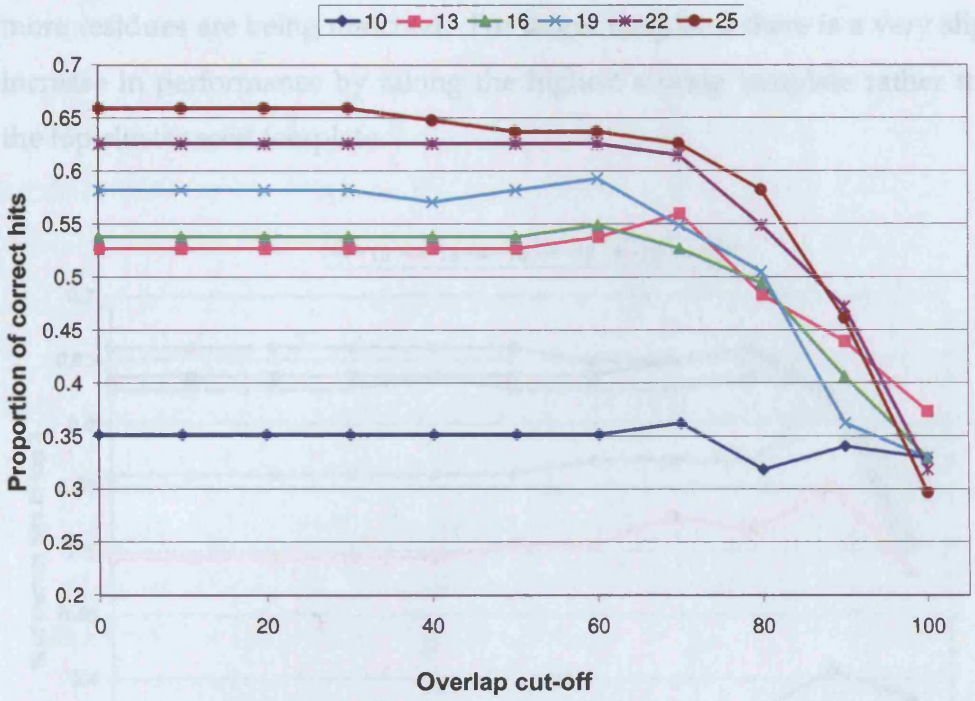


Figure 3.5 Performance (measured as percentage of correct hits in top 3) of FLORA over a range of overlap cut-offs, when varying the minimum template size. This was assessed by using the template from each enzyme family built from the selected seed cluster.

The performance of FLORA appeared to vary considerably with template size and at 65% was not as high as it was hoped compared to other published methods (Polacco and Babbitt, 2006; Laskowski *et al.*, 2005), so a different approach was explored. The program was instead used to build templates from all the seed clusters generated in Section 3.3.1.4. The jack-knifed data set was then re-scanned against *all* of these templates and the highest scoring template from each enzyme family was used to assign function to the test set domains.

Figure 3.6 shows an equivalent plot to Figure 3.5 but using the highest scoring cluster template for a given enzyme family to assign function to a test set domain. The preference for large templates is again evident, although there is a smaller drop in performance at higher overlap cut-offs, suggesting more residues are being matched. For larger templates there is a very slight increase in performance by taking the highest scoring template rather than the top cluster seed template.

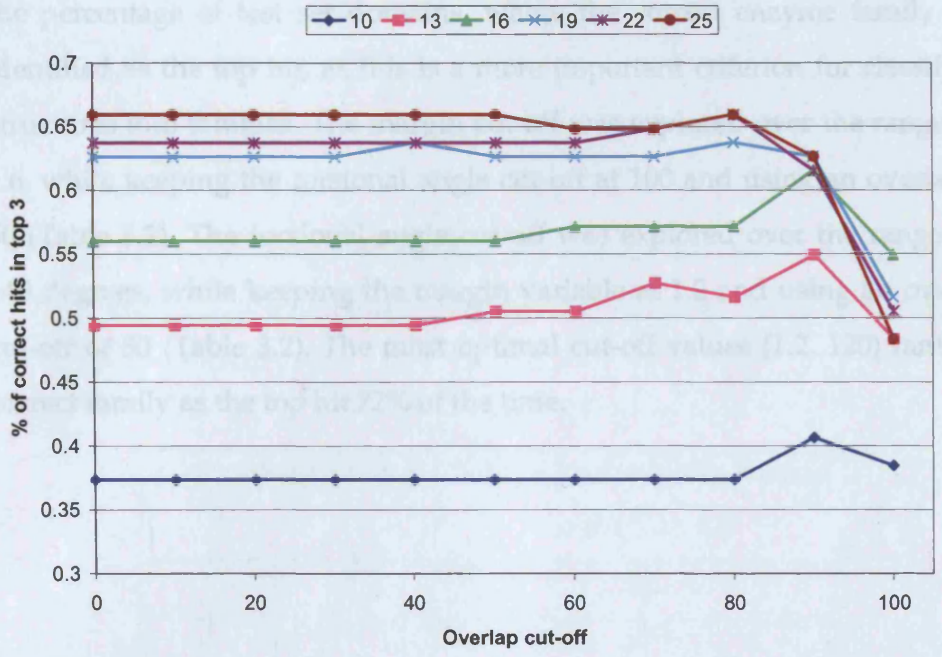


Figure 3.6 Performance (measured as percentage of correct hits in top 3) of FLORA over a range of overlap cut-offs, when varying the minimum template size. This was assessed by taking the best template match from each enzyme family to the test set domain.

Again, given that the *margin* and *torsional angle cut-offs* were still to be optimised, it was decided that taking the best template match for each enzyme family rather than the top seed cluster provided more consistent results and was less dependent on the overlap cut-off. Subsequent optimisations were undertaken by scanning all cluster templates for each enzyme family and taking the best match.

3.3.4.2 Optimising the margin and torsional cut-off

The error *margin* and *torsional cut-offs* used by FLORAScan affect the specificity of the template matching. If these are too liberal, the templates will match too many false positives. Conversely, if they are too conservative, genuine matches might not be recognised. A minimum template size of 25 was chosen and the best matched template for each enzyme family was used, as described above. This time the performance was measured by looking at the percentage of test set domains, where the correct enzyme family was identified as the top hit, as this is a more important criterion for classifying structures into families. The margin cut-off was explored over the range: 0 – 1.6, while keeping the torsional angle cut-off at 100 and using an overlap of 50 (Table 3.1). The torsional angle cut-off was explored over the range: 0 – 140 degrees, while keeping the margin variable at 1.2 and using an overlap cut-off of 50 (Table 3.2). The most optimal cut-off values (1.2, 120) rank the correct family as the top hit 72% of the time.

margin	% of correct matches ranked as the Top hit
0	30
0.2	32
0.4	41
0.6	60
0.8	63
1.0	67
1.2	72
1.4	70
1.6	69

Table 3.1 The performance of FLORA for finding the correct top hit over a range of margin cut-offs, while keeping the torsional angle cut-off at 100 and using an overlap of 50.

Torsional angle cut-off	% of correct matches ranked as the Top hit
0	52
20	57
40	58
60	64
80	65
100	67
120	72
140	70

Table 3.2 The performance of FLORA for finding the correct top hit over a range of torsional angle cut-offs, while keeping the margin variable at 1.2 and using an overlap of 50.

By analysing the results for each query domain, it appeared that a large proportion of the failed template matches were in the P-loop hydrolase superfamily (3.40.50.300). This is widely acknowledged to be the most diverse domain superfamily in the protein universe (Lee *et al.*, 2005). Closer inspection of the CORA alignments of its constituent enzyme families revealed that on average only 14% of the alignment of each family was ungapped positions (i.e. there was an equivalent residue in all domains). This meant that FLORA was often unable to build templates larger than around 10 residues, as there were not a sufficient number of fully-aligned positions in the expanded radius. Furthermore, Table 3.3 shows the values of $SC_{template}$ for different enzyme families in the data set and the P-loop superfamily (3.40.50.300) has the lowest conservation and hence the most permissive cut-off for template matching. Analysis of the P-loop enzyme families showed more than 3-fold differences in domain size. CORA would have problems aligning such diverse structures. The solution will be to sub-cluster the families into coherent structural sub-groups (SSGs) which has been used in other applications to help with this problem (Reeves *et al.*, 2006). If the P-loop superfamily is removed from the analysis, the top FLORA match was the correct enzyme family for 85% of the test set.

Enzyme family (C.A.T.H/Enzyme family)	$SC_{template}$
3.40.50.300 Phosphotransferases with an alcohol group as acceptor	0.474
3.40.50.300 Phosphotransferases with a phosphate group as acceptor	0.492
3.40.50.720 Oxidoreductases, acting on the CH-OH group of donors, with NAD(+) or NADP(+) as acceptor.	0.612
3.40.640.10 Aminotransferases	0.668
3.20.20.90 Intramolecular oxidoreductases, nterconverting aldoses and ketoses	0.563
3.20.20.90 Carboxy-lyases	0.653
3.40.640.10 Carbon-sulfur lyases	0.682
3.40.710.10 Cyclic amide hydrolases	0.864
3.90.550.10 Nucleotidyltransferases	0.877
3.40.50.720 Oxidoreductases, with NAD(+) or NADP(+) as acceptor	1.021
3.40.630.10 Aminopeptidases	1.089
3.40.630.10 Metallocarboxypeptidases	1.219
3.90.550.10 Hexosyltransferases	1.314
2.160.20.10 Glycosidases	1.349
3.40.710.10 Serine-type carboxypeptidases	1.523
2.160.20.10 Polysaccharide lyases	1.706
3.40.50.720 Carbohydrate isomerases	2.005
3.20.20.90 Oxidoreductases with oxygen as acceptor	2.365
3.40.50.1820 Ether hydrolases	3.107

Table 3.3 Values of $SC_{template}$ for different enzyme families in the data set, where a high value indicates good structural conservation.

3.3.5 Comparing the performance of FLORA to other methods for assigning function

3.3.5.1 Using PSI-BLAST to find functional homologues in the diverse data set

To put the performance of FLORA in detecting functional homologues in context, each domain in the data set was used as a query sequence for a PSI-

BLAST comparison, as PSI-BLAST is a method frequently used by biologists to assign function and used by other developers of structure-based function prediction algorithms to asses the value of their approach. For each query, the other domains within the same superfamily were embedded in the Uniref database, as detailed in Section 3.2.5. Only CATH domain-domain pairs were extracted from the final iteration of PSI-BLAST and ranked by their E-value. The P-loop hydrolase superfamily was included in this analysis.

Rank	FLORA	PSI-BLAST
1	72.0%	72.8%
2	2.1%	0%
3	2%	0%
4	1%	0%
5	1%	0.8%
>5 for FLORA/Not found by PSI-BLAST	18%	26.4%

Table 3.4 Rank of correct hit (same enzyme family) when scanning diverse domains using PSI-BLAST

Table 3.4 shows that PSI-BLAST is able to find the correct enzyme family as the top hit over 72% of the time, which is slightly higher than FLORA. However, FLORA finds 76.1% of functional relatives in the top 3 hits and PSI-BLAST does not find any correct hits for 26.4% of the query domains. The coverage might be improved by using more liberal cut-offs for PSI-BLAST, however this may also cause the profile to drift and pick up domains in the same superfamily whose function has diverged.

3.3.5.2 Using SSAP to find functional homologues in the diverse data set

SSAP is highly effective at recognising relatives at the superfamily level by global structural comparison. However, FLORA templates were designed to discriminate between enzyme families within diverse superfamilies as well as identify more distant homologues. The performance of SSAP in recognising functional homologues was tested by an all-vs-all comparison of

the 125 domains in the data set. Again, the P-loop hydrolase superfamily was included in this analysis. The results for each query domain were then ranked by the native SSAP score to see where a domain within the same enzyme family lay in the list of hits.

Table 3.5 shows that SSAP is able to find the correct enzyme family as the top hit nearly 90% of the time – this is markedly higher than the 72% achieved by FLORA. It appears in this instance that using a consensus local template for each enzyme family actually performs worse than simply finding the closest functional relative using global structure comparison. This is most likely due to the fact that FLORA has not correctly identified a motif that is able to discriminate between those relatives which have conserved their function during evolution and those that have diverged. Another reason might be that the enzyme families from which the templates were built are structurally diverse. In order to maintain the sensitivity of FLORA, it was often necessary to set quite liberal values for $SC_{template}$. Although this ensured that a given test domain was able to find its correct superfamily, it also decreased the specificity.

Rank	FLORA	SSAP
1	72.0%	89.6%
2	2.1%	3.2%
3	2%	0%
4	1%	4.8%
5	1%	1.6%
>5	18%	0%

Table 3.5 Rank of correct hit (same enzyme family) when comparing diverse domains using SSAP

3.3.5.3 Using SiteSeer to find functional homologues in the diverse data set

In order to compare the performance of FLORA with other local template methods, the SiteSeer program (Laskowski *et al.*, 2005) was applied to the diverse data set.

SiteSeer creates a large number of tri-peptide templates from the query structure and compares the templates to a library of other protein structures. The query structure is then superposed onto each matched structure based on the equivalent residues found by the template. The algorithm scores each match based on the sequence similarity of the local environment around the template region and converts the score to an E-value. The library structures are then ranked by the E-value of the most similar random template built from the query structure.

For this analysis, each domain in the diverse data set was compared using SiteSeer to produce an E-value score for each pair. Table 3.6 shows that SiteSeer is able to rank the domain with the correct function as the top hit in 80% of cases. It therefore outperforms FLORA by nearly 8%, although lags behind SSAP by 10%. This again suggests that it is difficult for a local template method to outperform global structure comparison, although SiteSeer comes closer than FLORA. An important area that SiteSeer exploits when scoring template matches that was not addressed by FLORA is that of the sequence similarity of the local environment around the template. Bartlett *et al.* (2002) showed that sequence similarity is higher in the active site than when calculated across the whole domain or protein. Future developments of FLORA to incorporate local sequence similarity are discussed in more detail in Section 3.4.

Rank	FLORA	SiteSeer
1	72%	80.0%
2	2%	6.4%
3	2%	3.2%
4	1%	1.6%
5	1%	1.6%
> 5	18%	6.8%

Table 3.6 Rank of correct hit (same enzyme family) when comparing diverse domains using SiteSeer

3.3.6 Generating a local scoring scheme from global SSAP alignments of domain pairs in the diverse data set

As discussed above, it appears that local templates were unable to assign the correct enzyme family to the test domains as effectively as global structure comparison. More specifically, that transferring annotation from the closest structural relative in a superfamily is able to correctly assign function to a domain in nearly 90% of cases. However, this does not tell us much about how function is conserved (i.e. which residues are important for function) and how to predict when it changes. It also relies on having a protein of similar function in the library of structures against which you are comparing the query.

To test whether adapting global structure comparison to focus on local similarities could discriminate better between domains in different enzyme families, a local scoring scheme was developed for comparing domain pairs in the data set, aligned by SSAP.

The CORA alignments from each enzyme family were analysed and all positions that did not contain gaps were selected (*CORA positions*). For each CORA position, the corresponding residues in each domain were noted and

annotated as *conserved residues*. SSAP was then used to align each domain pair in the diverse data set in the same way as in Section 3.3.5.2. The global SSAP score normalises the similarity of the vectors between aligned residues by the length of the largest protein, so it does not assign high scores to motif matches. It was hypothesised that in the case of determining functional equivalences a local scoring scheme might actually be more appropriate.

To develop a local scoring system, domains were aligned using SSAP but vector similarities were only summed over the conserved residues identified by CORA. The score was then normalised over the conserved residues. For each query domain, the results were ranked by this new score, denoted as *CORASCORE*. Table 3.7 shows that this approach (SSAP-CORA) is able to identify the correct top hit in 7% (79% vs 72%) more cases than FLORA, although it still falls short of the 89% identified by SSAP. This might suggest that it is useful to take account of indels, as in the SSAP global similarity score, when seeking the closest functional relative.

Rank	FLORA	SSAP-CORA
1	72%	79%
2	2%	4%
3	2%	0.7%
4	1%	0.2%
5	1%	5%
> 5	18%	10.7%

Table 3.7 Rank of correct hit (same enzyme family) when comparing diverse domains using FLORA and SSAP-CORA (CORASCORE)

To view the performance of the CORASCORE in a different way, a ROC curve was calculated to assess how well the CORASCORE was able to recognise all the functional homologues (i.e. domains in the same enzyme family) for a given cut-off. This was compared to the global SSAP score at low false positive rates.

Figure 3.7 shows that at low error rates the pair-wise score between two domains in the data set is actually better represented by the local CORASCORE, which focuses on conserved residues, rather than the global SSAP score. This might suggest that concentrating on residues that are conserved across an enzyme family is a better method for recognising some of the more diverse relatives in an enzyme family.

Most automatic methods for functional annotation rely on taking the highest scoring match. However, as with sequence profile methods, it is also important to have reliable score cut-offs that can be used to transfer function even from distant relatives. These thresholds can be derived from the ROC analysis. This idea is discussed in more detail in Section 3.4.

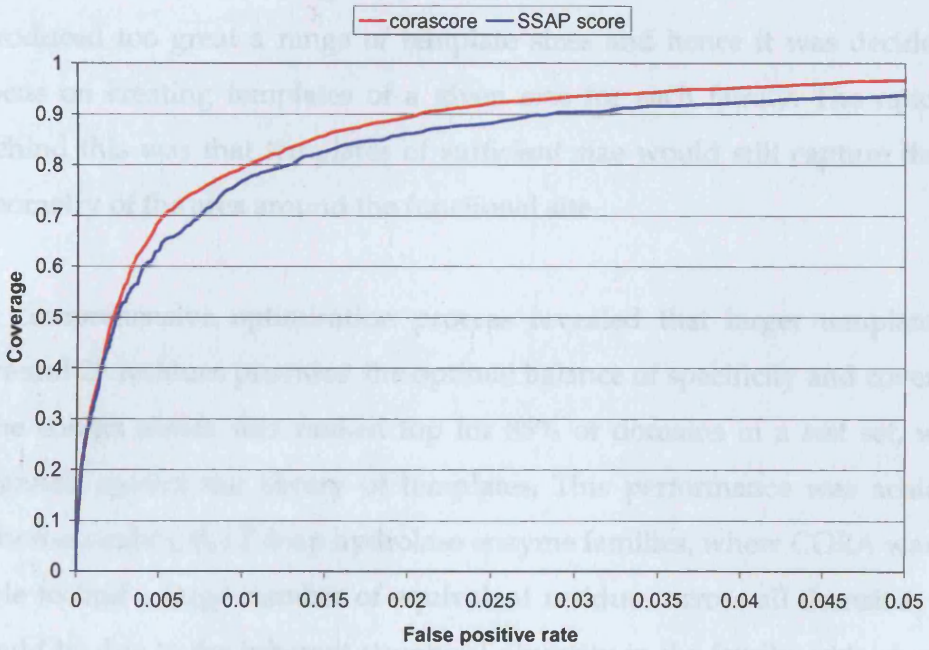


Figure 3.7 A Receiver-Operator Curve (ROC) comparing the ability of the local CORASCORE to discriminate between domains from the same enzyme family and false matches in the data set with respect to the global SSAP score.

3.4 Discussion

This work describes the development of a novel algorithm (FLORA) for generating structural templates to characterise enzyme families. Overall this work has established a protocol for classifying functional relatives into CATH enzyme families. Global structure comparison by SSAP was shown to recognise functional homologues in nearly 90% of the cases but provides not information on the functional sites. The FLORA method for identifying conserved functional sites was able to locate sites in 80% of the families tested and is therefore a valuable complement to using SSAP.

In FLORA, a template is built by selecting positions in the local environment of the predicted functional site, which are structurally conserved across a multiple alignment of an enzyme family. The optimisation process in Section 3.3.1.7 revealed that using an absolute measure of structural conservation produced too great a range of template sizes and hence it was decided to focus on creating templates of a given size for each family. The rationale behind this was that templates of sufficient size would still capture the 3D geometry of the area around the functional site.

A comprehensive optimisation process revealed that larger templates of around 25 residues provided the optimal balance of specificity and coverage. The correct match was ranked top for 85% of domains in a test set, when scanned against the library of templates. This performance was achieved when excluding the P-loop hydrolase enzyme families, where CORA was not able to find a large number of equivalent residues across all domains. This could be due to the inherent structural diversity in the family; indeed, some domain pairs had SSAP scores below 50 (with 100 being identical structures). However, it is also possible that one or more of the domains in the superfamily were annotated with an incorrect E.C. number, which would have caused CORA to produce an incorrect multiple alignment.

For the remaining 15% of domains where FLORA was unable to match the correct template as the top hit, the errors were mainly caused by matches to templates built from other enzyme families within the same superfamily. This suggests that the current implementation of FLORA was not always able to capture structural motifs for different functions, or that the FLORA templates were not focussing on the areas of the structure that are responsible for modifying the function. A slightly different approach would be to compare templates generated from different enzyme families within the same superfamily, to look for commonalities. These common superfamily motifs could then be removed from the clique-matching process and perhaps better focus on the family-specific regions of the structure. In addition, given that the problems with the P-loop hydrolase enzyme families appeared to lie with the original CORA alignment, it is possible that the domains were too structurally diverse to be accurately aligned. This problem could be addressed by clustering together more similar domains within each enzyme family and building templates from each sub-group. Although this would produce multiple templates per enzyme family, it might shed more light on whether the limitations lie with CORA or with the FLORA method presented here.

Preliminary work suggested that the FLORAScore was the most effective score for identifying the correct enzyme family for a given query domain. However, by looking at the data for the 15% that failed, it is clear that other scoring schemes may be better in some circumstances: for example, by combining the overlap with the FLORAScore or calculating the average score over all templates built from different seed clusters for a given enzyme family, rather than taking the best hit. Furthermore, the superior performance of SiteSeer over FLORA suggested that taking into account the sequence conservation to score template matches might also provide a useful discriminatory signal.

To put the work in context, a comparison of FLORA against both PSI-BLAST and SSAP showed that it was able to outperform PSI-BLAST if the top 3 hits are considered (76.1% for FLORA vs 72.1% for PSI-BLAST). However, PSI-BLAST was not able to find as great a percentage of functional homologues at a low error rate as SSAP. This demonstrates that global structure comparison remains powerful for detecting domains with similar functions. FLORA was still unable to outperform SSAP, which suggests that focussing solely on the functional site does not necessarily yield a significant improvement when seeking functional similarities between more distant evolutionary relatives. The work of Reeves *et al.* (2006) has shown that structural embellishments across a larger region of the structure can impact on function and global comparisons may capture this information more effectively. However, for function prediction a combination of using SSAP to find the closest functional relative and FLORA to predict the active site could provide useful complementary information.

3.5 Future Work

The relatively poor performance of FLORA compared to global structure comparison (SSAP) could be due to the current implementation or might be due to the fact that a more global similarity of domains must be taken into account to establish the closest functional relative in a superfamily. The SSAP-CORA (CORASCORE) method in Section 3.3.6 appeared to perform well and ROC curve analysis suggested that it was able to identify more functional relatives at a low error rate than SSAP. However, the CORASCORE only found the correct enzyme family as the top hit in 79% of cases, compared with 89% for SSAP. This might suggest that finding the closest functional relative is achieved more effectively by using a global method, yet when looking for a motif associated with all domains from a given enzyme family, it is useful to focus on local conservation patterns.

To take this work forward, it is planned to look in more detail at the conserved residues identified by SSAP-CORA and see if they can be further reduced based on their sequence conservation, local structural conservation and/or solvent accessibility. For each enzyme family, it should also be possible to down-weight the effect of residues that are conserved to maintain the protein fold in the superfamily, rather than being specific for function.

In addition, FLORA did not recognise the correct functional site in ~20% of the enzyme families in the data set. This leaves scope for using alternative methods, such as the evolutionary trace, which exploit phylogenic information (Lichtarge *et al.*, 1996) for identifying functional residues, followed by building templates based on the local structural environment.

Chapter 4 Improving *ab initio* structure predictions by assigning models to fold groups in CATH

4.1 Background

The ability to predict the tertiary structure of a protein directly from its sequence remains a significant goal of structural biology, as there is a large discrepancy between the number of available sequences and structure. Furthermore, structural data can be useful for understanding protein function. X-ray crystallography and NMR spectroscopy are the current methods of choice for experimental structure prediction. However, both approaches have limitations and cost implications, and hence cannot be applied indiscriminately to all genome sequences of interest. High-throughput methods can reduce the time and effort required, but highly flexible proteins and those which reside in cell membranes remain problematic. To facilitate this process, it is often necessary to modify the structure by mutating the sequence, although this risks moving further away from the native structure of the protein. Conversely, NMR is able to capture the intrinsic flexibility in a given protein by producing a series of models that fit the experimental data. However, current technology means that it is generally only possible to obtain models for small molecules (< 50 KDa). As a result, even with modern high-throughput methods, it is currently impractical to produce experimental structures for all known genes. Therefore, developing computational approaches to predict structure directly from a protein's sequence remains a useful complementary area of research as they provide a faster and cheaper alternative to experimental methods.

However, this does require developing a greater understanding of the complex interactions involved in protein folding.

4.1.1 *Ab initio* prediction of structure from sequence

The two greatest difficulties in predicting how a given sequence folds in three dimensions are the huge number of possible residue conformations available and how residues interact with one another to stabilise the protein structure. Algorithms that aim to predict protein structure *ab initio* require vast amounts of computational power and therefore even modelling small peptides becomes hugely time consuming. To combat this, many methods attempt to mimic the native folding process so that the search space can be collapsed at various stages in the algorithm. Each step seeks an energy minimum where the structure is at its most stable. However, exploring this energy landscape can still prove challenging as there is often no guarantee the algorithm will converge on the global energy minimum and may instead find local energy minima. As an alternative, several groups have chosen to exploit knowledge of known structures by using known conformations for small peptide fragments in conjunction with predicting physicochemical interactions from first principles (Simons *et al.*, 1997).

One way of building heuristics for *ab initio* methods is to predict structural features from the sequence, before attempting to model the whole fold. For example, predicting the secondary structure of each residue or the overall secondary structure content of the protein (protein class).

4.1.1.1 Predicting protein class

Many groups have endeavoured to predict the overall secondary structure content or protein class (e.g. mainly alpha, mainly beta, alpha-beta) based solely on amino acid composition (reviewed in Chou (2005)). The most accurate methods rely on machine learning algorithms (e.g. SVMs) and incorporate analysis of dipeptide/tripeptide fragments as well as

propensities of different residues to adopt certain secondary structures conformations (Rost and Sander, 1993; Eisenhaber *et al.*, 1996).

4.1.1.2 Predicting secondary structure

Predicting the secondary structure state (helix, strand or random coil) of individual residues in a sequence is the starting point for many structure prediction methods. It has long been known that some amino acids are more likely to be present in certain secondary structure elements than others (Chou and Fasman, 1974). For example, the pyrrolidine side chain of proline and the C_β atom of the preceding residue results in steric hindrance, which limits the use of proline in alpha helices. Chou and Fasman (1974) were the first to exploit this concept by analysing residue propensities in the small data set of protein structures that was available at the time. Although this method showed some predictive power, Garnier et al. (1978) showed that the performance could be substantially improved by looking at a given amino acid in context with its neighbouring residues in the sequence. They used information theory to analyse a 'window' of 16 residues to calculate a more accurate probability of the secondary structure state of each amino acid.

This approach can be taken a step further by looking for small patterns of residues in multiple alignments of related sequences that regularly coincide with specific secondary structure elements. For example, certain arrangements of hydrophilic and hydrophobic residues occur in helices where some of the side chains face the hydrophobic environment of the protein core, whereas others interact with the solvent. Furthermore, insertions in these alignments usually coincide with random coil regions, rather than conserved helices or strands. Comparing protein sequences across a family allows a more accurate assessment of residue propensities by distinguishing between genuine conservation and random mutations .

The most successful methods of secondary structure prediction have sought to combine conservation patterns and residue properties using machine

learning methods, such as neural networks. The PHD method (Rost *et al.*, 1994) was the first to use sequence profiles to train a neural network to accurately assign a secondary structure state to more than 70% of residues in a benchmark test set of sequences. Jones (1999) increased this performance to 77% in his PSIPRED method by improving the quality of the sequence profiles used to train the network.

4.1.1.3 Predicting residue contacts

Although predicting the number and type of secondary structure elements can give clues as to the overall fold and class of a protein, the tertiary structure may still adopt many different conformations. One way of restricting the conformational space is to predict interactions between residues in the chain. If a sufficient number of these residue contacts can be identified, then it is possible to generate a reasonable model of the tertiary structure.

Several groups (Taylor and Hatrick, 1994; Pollastri and Baldi, 2002) have shown that networks of residues act to stabilise a protein fold. Mutations at positions in spatial proximity are often subject to correlated mutations. That is, if one stabilising residue is mutated so that its physicochemical or stereochemical properties are changed then those amino acids with which it interacts may also change in order to avoid steric hindrance and the breaking of hydrogen/electrostatic bonds. By analysing mutation rates across protein families, it is possible to identify pairs of residues in sequences that are close to one another in 3D (Pollastri and Baldi, 2002).

Again, many groups have attempted to recognise these patterns of correlated mutations by training neural networks on multiple alignments of known sequence families. However, this has proved far more problematic than analogous methods of secondary structure prediction due to the vast number of related sequences required. One reason for this is that it is not just pairs of residues that interact, but networks of several residues that act to stabilise the

fold. As they are all dependent on one another, it means that two residues may have similar mutation rates but are not necessarily in contact in the structure. Hence, any identifiable sequence pattern may actually be specific to a given structural family, rather than observable across a wide range of proteins.

4.1.1.4 Predicting tertiary structure

Most approaches that predict tertiary structure *ab initio* directly from the sequence can be broken down into two discrete parts: a procedure for generating possible chain conformations and a potential energy function that assesses the likelihood that a given structure is adopted by the sequence on energetic grounds.

As previously mentioned, one of the biggest problems with structure prediction is the enormous number of possible conformations that could feasibly be adopted by a given protein chain. Two popular approaches for reducing this number are to either restrict the number of positions a given residue may occupy to discrete points in a 3D lattice (Hinds and Levitt, 1994; Park and Levitt, 1995) or constrain the range of permitted torsional angles between residues (Dandekar and Argos, 1994; Srinivasan and Rose, 1995). True *ab initio* methods will then assess the viability of each model based on the physicochemical properties of amino acids and their interactions e.g. size and charge. However, other prediction algorithms compare the properties of potential models with known structures (e.g. threading (Jones *et al.*, 1992)). Whilst this knowledge-based approach can improve the accuracy of the methods, it has the inherent limitation of only being able to provide models for sequences that adopt previously observed folds.

4.1.1.5 The Rosetta method of structure prediction

Over the past five years, The Baker group (Simons *et al.*, 1997; Simons *et al.*, 1999) have developed a structure prediction algorithm (*Rosetta*) that splits up the target protein sequence into small peptides, less than 10 residues in

length. The local interactions of each fragment are then modelled by observing sequence-similar regions in experimental structures. These fragments can then be concatenated to produce models that are consistent with reasonable hydrophobic burial, electrostatic interactions, main-chain hydrogen bonding and excluded volume. Structures which meet these requirements are then refined by minimizing the non-local interaction energy using Monte Carlo simulation. However, although this can help to eliminate models that do not form "protein-like" molecules, it can be difficult to gauge those which are most similar to the native structure. Hence, the algorithm produces a number of models, resulting from different random seed values, which aims to address the problem of finding non-optimal models from local energy minima in the optimisation procedure.

4.1.2 Assigning structural predictions to fold groups

Structure comparison methods have proved very successful in detecting distant structural relationships between experimentally derived structures (Orengo and Taylor, 1996; Holm and Sander, 1998; Kolodny *et al.*, 2005). Indeed, Chapter 2 described the ability of the CATHEDRAL algorithm to assign a putative fold to novel structures in the PDB by scanning against previously characterised representatives from the CATH database.

A previous collaboration between the CATH group and the De La Cruz *et al.* (de la Cruz *et al.*, 2002) explored the use of structure comparison for assigning a known fold to *ab initio* models generated by the Rosetta method. They found that the correct CATH fold could be recognised as the top hit using models within 6Å of the native structure, for half the data set. Although this result showed that structural comparison methods can still be applied to theoretical models, it was only tested on 4 proteins. Furthermore, it relied on a relatively slow structural comparison algorithm (SSAP) and was not able to determine automatically good models in advance.

Simons et al. (Simons *et al.*, 2001) took a similar approach by comparing their Rosetta models against the PDB using DALI (see Section 1). Although the closest relative in the PDB was only found for around 50% of models, for matches with a Z-score greater than 4, they showed that structural comparison methods were applicable for models that deviated from the native structure by as much as 7Å. They suggest that as *ab initio* methods improve, it may even be possible to recognise functional families for novel genes through an intermediate structure prediction stage.

4.1.2.1 Comparing protein structure models using MAMMOTH

In choosing structure comparison algorithms for matching *ab initio* models to fold groups in CATH or SCOP, an important consideration is how well the algorithm can cope with model structures in which the secondary structures are not well defined. A recent structure comparison method (MAMMOTH, (Ortiz *et al.*, 2002)) was specifically designed for comparing theoretical models with experimental structures. The algorithm was designed to focus purely on Cα co-ordinates, avoiding any dependence on primary sequence, secondary structure or contact maps. This can be especially important when using *ab initio* models where the latter two features may not be fully formed with respect to the native structure.

MAMMOTH calculates its alignments in four stages. Firstly, each protein structure is broken into heptapepide fragments. Each heptapeptide is then described by a set of unit vectors between successive Cα atoms and translated to the origin. Using standard minimisation technique (McLachlan, 1979), a rotation matrix and unit vector root mean square (URMS) is calculated between all fragments pairs and converted to a similarity score based on the expected URMS between two random sets of n unit vectors ($URMS^R$). Scores between all possible pairs of heptapeptides are then taken to populate a matrix, from which a global alignment is calculated using dynamic programming (Needleman and Wunsch, 1970). An overall structural similarity between two given structure is calculated using a

variant of the MaxSub algorithm to determine the percentage of corresponding residues (PSI) less than 4Å in 3D. The PSI is then converted into a P-value using a distribution of random structural alignments from a data set of unrelated SCOP domains. MAMMOTH is able to detect 50% of fold matches at the 99% confidence level, compared to 60% for DALI. Given its superior speed, the authors suggest this makes it a relatively accurate tool for structure comparison of large databases. It certainly lends itself to suggesting putative fold matches, which may then be aligned with a more accurate, computational intensive method.

4.2 Aims

The purpose of the method presented here was to build on the work of de la Cruz group in Barcelona, Spain (de la Cruz *et al.*, 2002) by developing a fast and novel protocol (MODMATCH) for determining the correct fold for a given target structure by comparing *ab initio* models from the Rosetta method to the CATH fold library. This work was undertaken in collaboration with Xavier de la Cruz.

The first objective was to reduce a large set of initial predictions (999 models per target structure) to a smaller sample, ideally of higher quality. This was to both increase the speed of the structure comparison and reduce the noise generated by erroneous hits between CATH library domains and bad models. The second aim was to optimise the accuracy of fold assignments by combining structural similarity scores from the MAMMOTH (Ortiz *et al.*, 2002) and SSAP (Taylor and Orengo, 1989) algorithms using a Support Vector Machine (SVM).

For this work, the MAMMOTH algorithm was utilised to identify putative folds from a CATH library which could then be more accurately aligned with SSAP. This is analogous to the approach presented in Chapter 2 in the implementation of CATHEDRAL, where GRATH was used to pre-select

similar CATH folds within multi-domain protein chains to be aligned by SSAP. However, CATHEDRAL was thought to be unsuitable for this work as it was not designed to handle low resolution models where secondary structures (which form the basis of the GRATH algorithm) may not be fully formed. The use of SSAP in this work as an accurate structure comparison method was thought to be an improvement on DALI (used by Simons *et al.*(1999)) because DALI relies on conserved contacts to align residues, which again may not necessarily be present in theoretical protein structure models. The overall goal was to improve the assignment of folds to *ab initio* models by developing a fast, accurate protocol whereby the *ab initio* models could be assigned a fold in the CATH database, in a similar fashion to the way experimental structures are classified.

4.3 Methods

This section describes the data sets used to benchmark the MODMATCH protocol and the details of the superposition of structures and models used in this method.

4.3.1 Dataset of *ab initio* structure predictions

A dataset was obtained from the Baker group (Simons *et al.*, 2001) of *ab initio* structure predictions for 34 single domain target structures. This spanned all of the three major CATH protein classes (mainly alpha, mainly beta, alpha-beta, few secondary structures) (Table 4.1). A total of 999 predictions were provided by the Rosetta method for each target, resulting in a total of 33966 models (34 x 999) that could potentially be scanned against the CATH database.

Class	Number of target structures
1 (mainly alpha)	14
2 (mainly beta)	7
3 (alpha-beta)	12
4 (few secondary structures)	1

Table 4.1 Class distribution of target structures in the data set

4.3.2 Comparing *ab initio* models to native structure

The quality of the models was assessed by superposing them onto their native structure. However, for a given target structure, the *Rosetta models* do not consistently predict conformation for all of the residues. Therefore, these residues were removed from the co-ordinate (PDB) file for each native structure before the superposition.

4.3.3 Superposition of models

For each target PDB, all 999 models were superposed (against one another and against their native structure) using their sequence and a Root Mean Squared Deviation (RMSD) was calculated on the C-alpha co-ordinates. From this, a SAS score ((Equation 4.1) was also determined, as this has been shown to be a useful discriminator of structural similarity across proteins of different sizes by accounting for the number of aligned residues (Kolodny *et al.*, 2005).

$$SAS = \frac{100 \times RMSD}{N}$$ (Equation 4.1 SAS score. N = aligned residues)

4.3.4 Representatives from CATH v2.6

A library of 6003 structures from CATH v2.6 was obtained by selecting representatives (SReps) from each cluster of 35% sequence identity relatives to provide a representative sample of domains for the fold assignment in the MODMATCH protocol. These will subsequently be referred to as the *library structures*. CATH folds are described by a code in the format: "Class.Architecture.Fold" (e.g. 1.10.10).

4.4 Protocol development and Results

The speed of the MODMATCH protocol was increased by reducing the number of Rosetta models that were required to be compared to the CATH

library. In addition, a scoring scheme was optimised using a Support Vector Machine to increase the accuracy of fold assignments for each target structure in the data set. This section is divided into three main parts:

1. Assessment of the relative performance of MAMMOTH and SSAP for identifying the correct fold group in a database search.
2. Exploring the correlation between the quality of *ab intio* models and protein class.
3. Development and optimisation of the MODMATCH protocol
    a. Selecting a reduced sample of models to search against the CATH library using MAMMOTH.
    b. Optimising the number of putative fold groups identified by the MAMMOTH search to scan again using the slower, more sensitive SSAP algorithm.
    c. Developing a new scoring scheme to predict the correct CATH fold for each target structure, by exploiting a Support Vector Machine (SVM) to combine alignment scores from MAMMOTH and SSAP.

4.4.1 Assessing the performance of the MAMMOTH structure comparison method as a fast filter for MODMATCH

It was hypothesised that the speed of the MODMATCH protocol could be enhanced by applying a fast initial search of the CATH library using MAMMOTH. Therefore, both SSAP and MAMMOTH were assessed to determine their comparative performance in detecting structures with similar folds.

Structure comparison methods have been shown to vary in performance for detecting fold similarities (Kolodny *et al.*, 2005). This is especially true for small domains, such as alpha helical bundles where the addition of one helix can change the overall fold. In order to investigate the performance that

could be expected on the Rosetta model data set, the equivalent native structure in the PDB for each target was scanned against the CATH library using MAMMOTH and SSAP. Comparisons were scored and ranked using the MAMMOTH Z-score and SSAP SAS score.

Table 4.2 shows that MAMMOTH and SSAP demonstrate comparable performance when seeking to match the correct fold in the CATH library. MAMMOTH finds the correct fold as the top hit for 23 out of 34 of the native structures, compared to 26 for SSAP. Given that MAMMOTH is around 50 times faster than SSAP, this is an impressive performance. However, the ROC curve analysis shown in Figure 4.1 suggests that overall SSAP is a better at recognising fold similarities when all SReps in the fold groups are considered, with coverage of 75% versus 58% for MAMMOTH at a 5% error rate. Combined with the ranking results, this supports the assertion that using SSAP to compare models against the CATH library would add value for fold prediction, after an initial MAMMOTH filter.

Rank	Mammoth	SSAP
1	23	26
2	3	1
3	0	1
4	0	0
5	1	0
6	1	0
7	0	0
8	1	1
9	0	0
10	1	0
> 10	1	2

Table 4.2 The frequency at which the correct fold appears when scanning the native structure against the CATH library using MAMMOTH/SSAP.

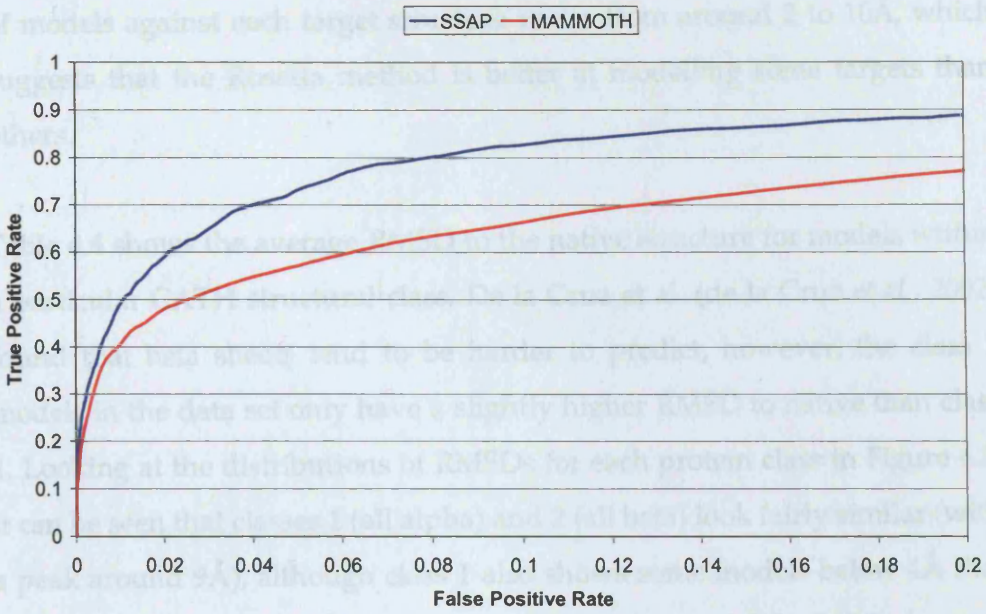


Figure 4.1 ROC curve analysis of MAMMOTH/SSAP for comparing native structures to the CATH library.

4.4.2 Exploring the correlation of model quality with protein class

The work of de la Cruz *et al.* (2002) suggested that the average quality of *ab initio* models varies with protein class. More specifically, that protein structures containing beta-sheets were harder to predict than those that contained mainly alpha helices. This is because helices are formed from local interactions, whereas beta sheets are more dependent on the tertiary structure of the protein. In order to explore whether the performance of the Rosetta modelling protocol varied with the class of the target structure and whether the MODMATCH protocol needed to be tuned for different classes of protein, all models were superposed onto their native structure as detailed in Section 4.3.2.

Figure 4.2 shows that *ab initio* predictions can vary considerably from the native structure, with only 16% of models possessing a RMSD less than 5Å. Furthermore, there is a substantial number of models with a RMSD greater than 9Å. However, looking at Table 4.3 it can be seen that the average RMSD

of models against each target structure varies from around 2 to 10Å, which suggests that the Rosetta method is better at modelling some targets than others.

Table 4.4 shows the average RMSD to the native structure for models within a particular CATH structural class. De la Cruz et al. (de la Cruz *et al.*, 2002) found that beta sheets tend to be harder to predict, however, the class 2 models in the data set only have a slightly higher RMSD to native than class 1. Looking at the distributions of RMSDs for each protein class in Figure 4.3, it can be seen that classes 1 (all alpha) and 2 (all beta) look fairly similar (with a peak around 9Å), although class 1 also shows some models below 4Å that are not seen in class 2. The distribution for class 3 (alpha-beta) shows a similar peak at 9Å, although the largest peak is at 6Å. It is hard to say whether this is peculiar to the data set used, or whether folds with a mixture of alpha and beta regions are easier to predict. The distribution for class 4 (few secondary structures) is narrow and initially appears to be different to the other classes; however, it is probably due to the fact that there is only one class 4 protein in the data set – its mean is still around 8Å (Table 4.4).

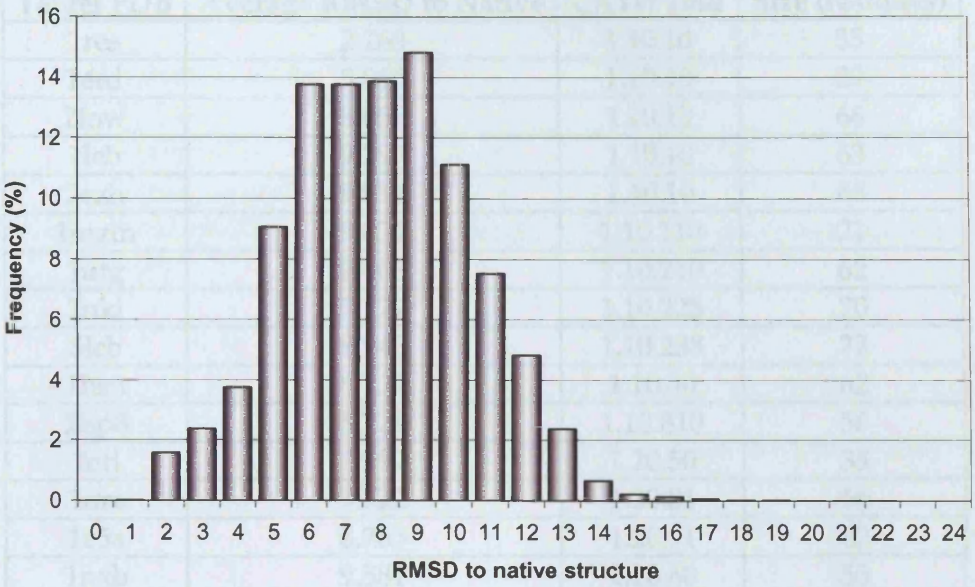


Figure 4.2 Distribution of RMSD for superpositions of all models against their native (experimental) structure.

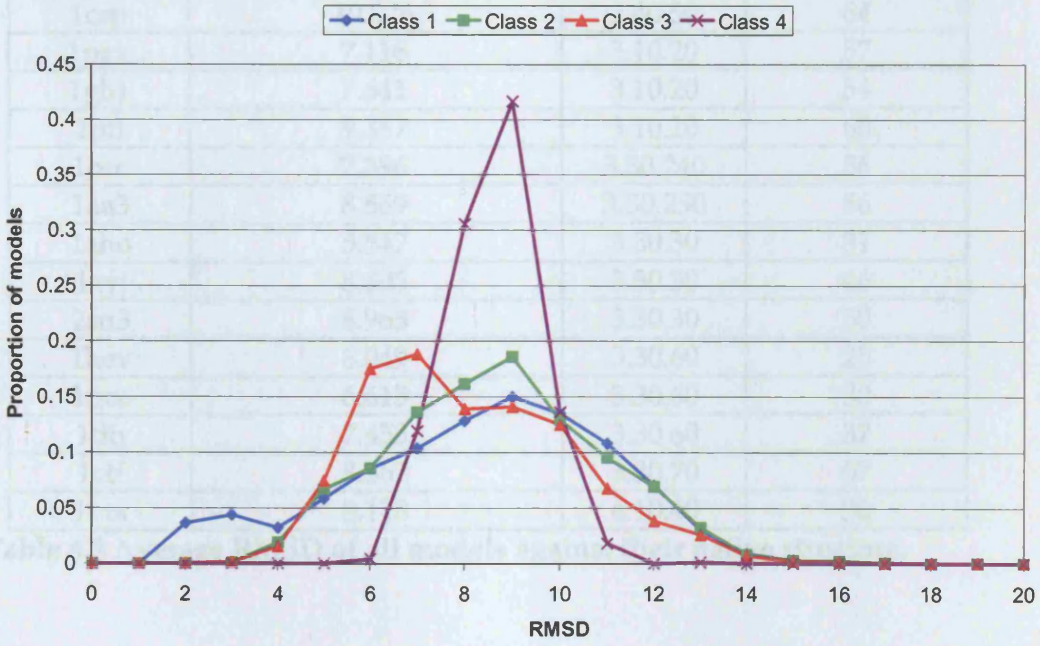


Figure 4.3 Distribution of RMSD values for models against their native structure, for different protein classes (according to CATH) in the data set

Target PDB	Average RMSD to Native	CATH Fold	Size (residues)
1res	2.263	1.10.10	35
1erd	5.850	1.10.10	29
2fow	8.030	1.10.10	66
1leb	9.297	1.10.10	63
2ezh	9.814	1.10.10	65
1mzm	9.071	1.10.110	71
1utg	9.605	1.10.210	62
1nkl	7.722	1.10.225	70
5icb	8.945	1.10.238	72
1hsn	8.129	1.10.30	62
2hp8	8.728	1.10.810	56
2erl	7.292	1.20.50	35
1nre	9.722	1.20.81	66
1c5a	8.765	1.20.91	62
1nxb	9.581	2.10.60	53
1tpm	8.110	2.10.70	41
2bds	6.581	2.20.20	21
1pft	5.373	2.20.25	36
1qyp	8.443	2.20.25	42
1sro	9.201	2.40.50	66
1csp	10.026	2.40.50	64
1pgx	7.116	3.10.20	57
1gb1	7.641	3.10.20	54
2ptl	9.357	3.10.20	60
1orc	7.396	3.30.240	56
1aa3	8.569	3.30.250	56
1aho	5.547	3.30.30	31
1ayj	8.405	3.30.30	46
2sn3	8.965	3.30.30	50
1hev	6.049	3.30.60	25
1pce	6.615	3.30.60	30
1tih	7.453	3.30.60	37
1ctf	8.362	3.30.70	67
1vtx	8.110	4.10.40	36

Table 4.3 Average RMSD of all models against their native structure.

Class	Average RMSD to native for targets in data set
1 (mainly alpha)	7.77
2 (mainly beta)	8.19
3 (alpha-beta)	7.62
4 (few secondary structures)	8.11

Table 4.4 Average RMSD to native for all models in a particular structural class in CATH

Overall, comparison of the quality of the models for the different protein classes did not show a difference in RMSD large enough to merit tuning the protocol for different protein classes.

4.4.3 Development and optimisation of the MODMATCH protocol

A new protocol (MODMATCH) was developed for increasing the speed and accuracy of identifying the correct fold in a search of the CATH library with a sample of *ab initio* models for a given target structure. This was achieved by performing a fast initial scan of the library using the MAMMOTH program to identify putative fold matches which were then explored further using the slower, yet more sensitive, SSAP algorithm. A further increase in speed was gained by reducing the number of models searched against the CATH library for a given target structure. The steps involved in the MODMATCH protocol are shown in Figure 4.4. The parameters M and N were optimised to increase the proportions of correct folds recognised, while maintaining the speed of the protocol. The optimisation procedure is described in the following sections.

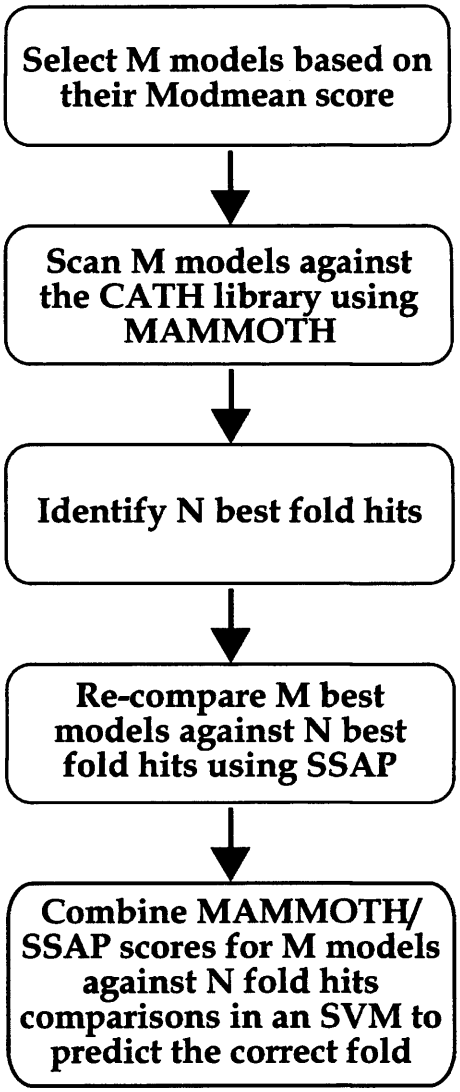


Figure 4.4 Outline of the protocol for scanning *Rosetta* models for a given target PDB structure, against the CATH fold library

4.4.3.1 Optimising the selection of representative models from each target structure

Reducing the set of 999 models for each target structure to a smaller set for structure comparison will increase the speed of the protocol and reduce the noise from bad models. This section describes the strategy used for selecting better models for the structure comparison stage. During the *ab initio* prediction optimisation, models are generated using a suitable energy function that seeks protein-like features. It was hypothesised that it would be possible to identify a subset of such models which would comprise a

majority of structurally similar predictions with a smaller number of outliers. More specifically, that the predictions that (on average) superposed best onto the other models, were most likely to be closer to the native structure.

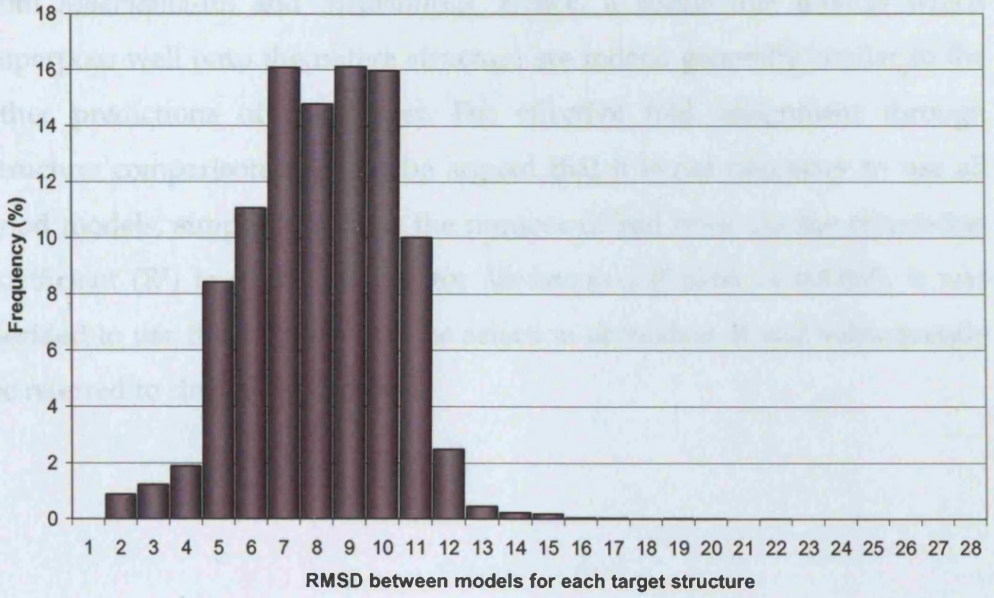
All models were superposed onto one another (as detailed in Section 4.3.3) and RMSD and SAS scores were calculated. Both scores were analysed to assess which would prove most useful for selecting models across all target structures in the data set. Figure 4.5 shows distributions of these scores for all models in the data set. It can be seen that the models are quite diverse in their similarity to one another, with the majority between 7Å and 10Å. As both RMSD and SAS distributions are approximately normal, calculating the average superposition score for a given model over all other models would give a meaningful measure of its relationship to these models for a particular target structure. (Equation 4.2 shows the calculation of this score, *Modmean*, for both RMSD and SAS scores.

$$Modmean_{RMSD} = \frac{1}{n}\sum_{i=1}^{n} RMSD(m_{sel}, m_i) \qquad \text{(Equation 4.2 b)}$$

$$Modmean_{SAS} = \frac{1}{n}\sum_{i=1}^{n} SAS(m_{sel}, m_i) \qquad \text{(Equation 4.2 b)}$$

Calculation of the average RMSD of a model to all other models; b) Calculation of the average SAS score of a model to all other models; where n = 998 and m_{sel} is the given model .

a)



b)

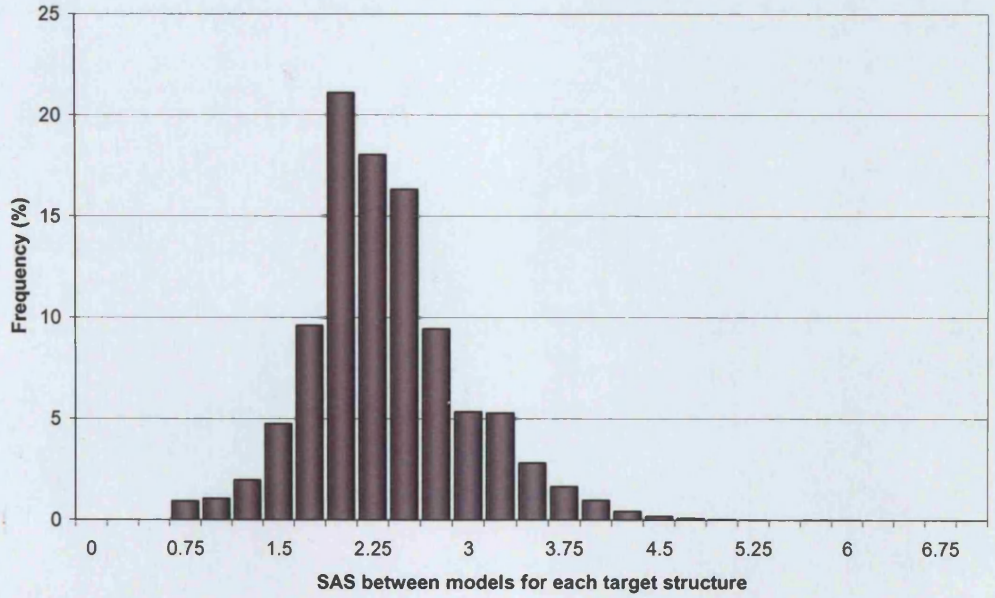
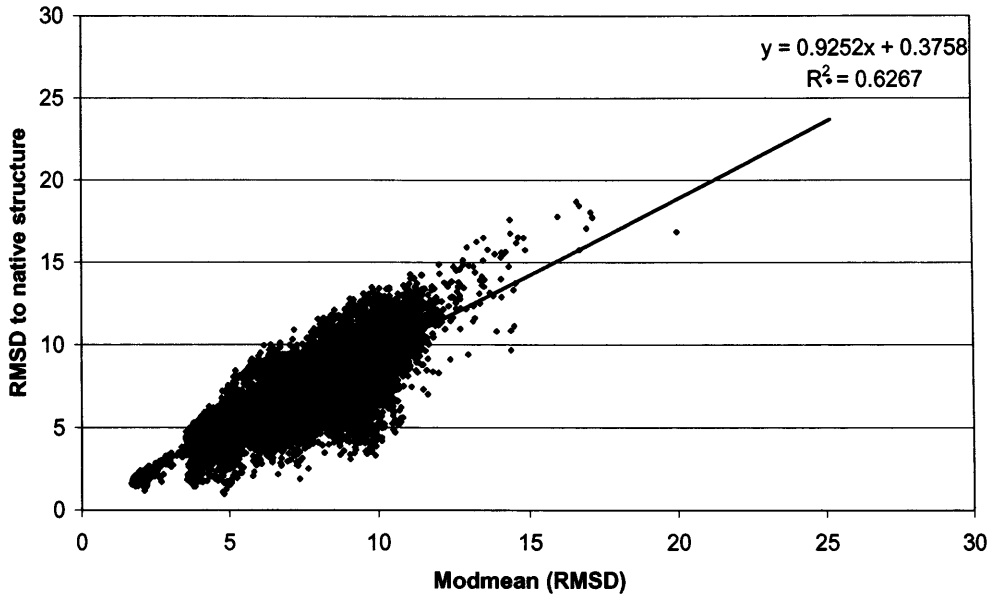


Figure 4.5 Distribution of (a) RMSD (b) SAS scores for superpositions between all models of each target PDB structure.

To explicitly test the hypothesis that better quality models will have lower values of *Modmean*, the *Modmean* scores were plotted against the RMSD and SAS scores obtained by comparing the models to the native structure (Figure

4.6). As can be seen there is a linear correlation, albeit not a strong one, for both $Modmean_{RMSD}$ and $Modmean_{SAS}$. Hence, it seems that models which superpose well onto the native structure are indeed generally similar to the other predictions of the target. For effective fold assignment through structure comparison, it could be argued that it is not necessary to use all good models, simply to reduce the number of bad ones. As the correlation coefficient (R^2) is slightly better for $Modmean_{SAS}$ (0.6768 vs 0.6267), it was decided to use this measure for the selection of models. It will subsequently be referred to simply as *Modmean*.

a)



b)

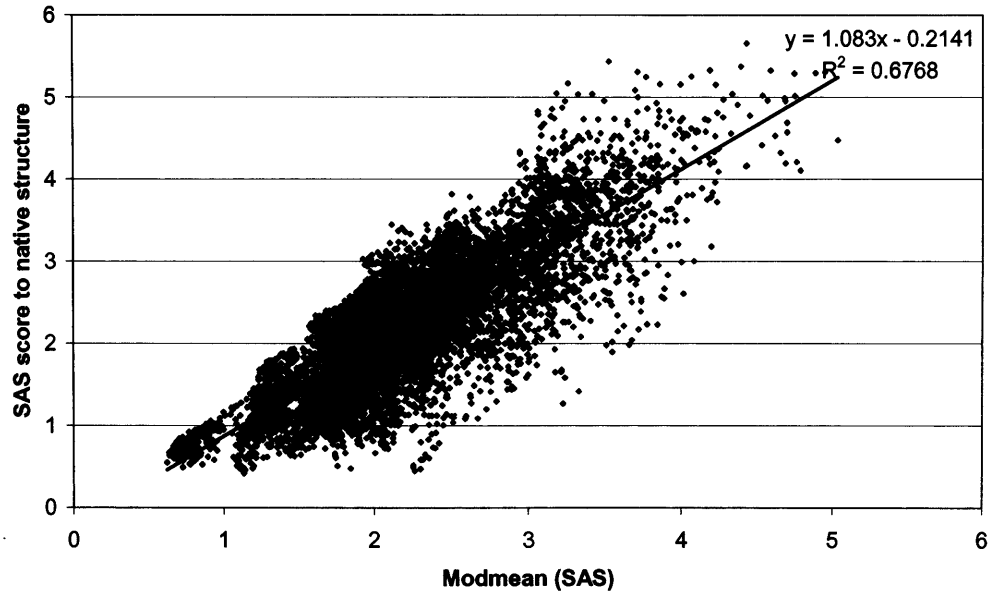


Figure 4.6 a) Plot of the RMSD score to the native structure for a given model against $Modmean_{RMSD}$ b) Plot of the SAS score to the native structure for a given model against $Modmean_{SAS}$.

4.4.3.2 Selecting a smaller sample of good quality models for each target structure

A varying number (M) of models were selected by using their *Modmean* score to see to what extent the model set could be reduced in numbers whilst ensuring that a sufficient number of good models remained.

For each target structure, the models were ranked by *Modmean* and the top M models in this list were selected, where M ranged between 1 and 150 (Figure 4.7). An analysis was made of the number of "good" models (i.e. those within the top 50 models when ranked by their SAS score to the native structure) to observe how many of these occurred within this set of N models. It can be seen from Figure 4.7 that the percentage of good models appears to increase linearly as more models are taken from the *Modmean* ranked list.

To calculate the number of good models that would be expected by chance, the list of models for each target structure was also sorted randomly. This process was repeated 1000 times and the average percentage of good models in this random set plotted in Figure 4.7, for each sample size as before. It is clear that ranking by *Modmean* does indeed enrich the sample set with good models. For example, a random sample of 50 models would only contain 5% of the good models on average, as opposed to 17% if a sample is selected using the *Modmean* score.

Since this seemed to be a reasonable proportion of 'good' models and to maintain the speed of the protocol, the top 50 *Modmean* ranked models (*Mod50*) for each target structure were selected for scanning against the CATH library.

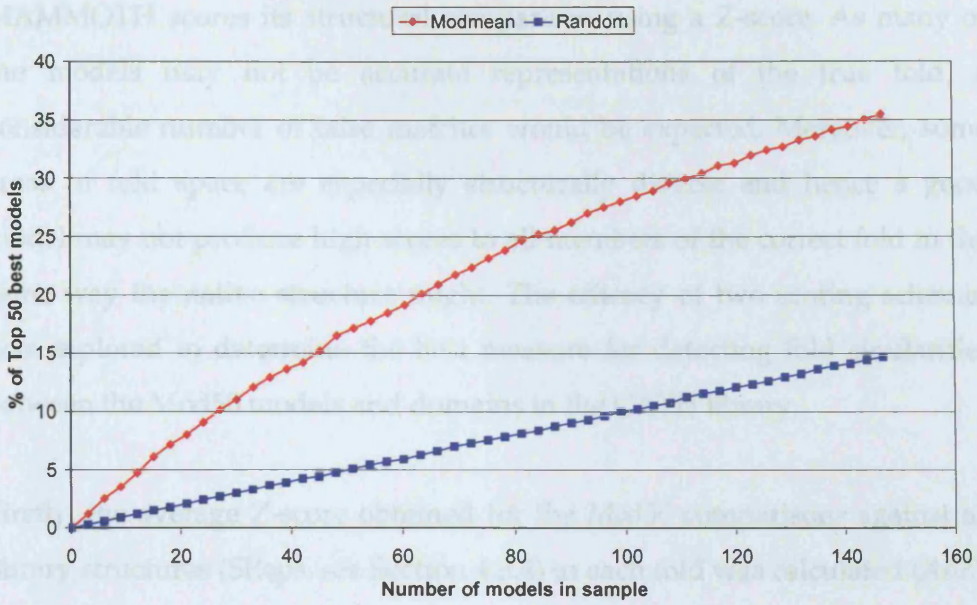


Figure 4.7 Models were ranked by their mean SAS score to other models ($Modmean_{SAS}$) and a sample of varying size was taken. The performance was assessed based on the percentage of "good models" (within the top 50 when ranked by their SAS score to the native structure). This was compared to selecting random models

4.4.3.3 Determining a reliable scoring scheme for the fast matching of the Mod50 models to the CATH library using MAMMOTH

Although MAMMOTH might not find the best alignment between two structures, it is a valuable tool for identifying putative fold matches that may then be analysed with a more accurate structural comparison method, such as SSAP, to give improved alignments and structural similarity measures. To optimise the performance of MAMMOTH as a filter, different scoring schemes were explored. A multi-processor computer farm was used to scan the Mod50 models for each target structure against the CATH domain library, using MAMMOTH. The aim was to discover the best score to discriminate between genuine fold matches and incorrect hits in the database.

MAMMOTH scores its structural comparison using a Z-score. As many of the models may not be accurate representations of the true fold, a considerable number of false matches would be expected. Moreover, some areas of fold space are especially structurally diverse and hence a good model may not produce high scores to all members of the correct fold in the same way the native structure might. The efficacy of two scoring schemes was explored to determine the best measure for detecting fold similarities between the Mod50 models and domains in the CATH library.

Firstly, the average Z-score obtained for the *Mod50* comparisons against all library structures (SReps, see Section 4.3.4) in each fold was calculated (*AvZ*). Secondly, the maximum Z-score (*MaxZ*) obtained between any model-SRep pair was determined for each fold. Both scoring schemes were examined using ROC curve analysis, where positive hits correspond to the CATH fold group in which the native structure was classified.

Figure 4.8 shows that the *MaxZ* score to each fold group appears to be a better discriminator of true matches than *AvZ* across the whole range of false positive rates. It shows coverage of 50% at a low error rate (5%) compared to only 40% for *AvZ*. This suggests that taking the highest scoring match for each fold group would provide the best route to determining the correct fold. This might be due to the inherent diversity of many fold groups and the fact that even a good model could give a low score to a relative within the correct fold group if it is very structurally different from the native target structure.

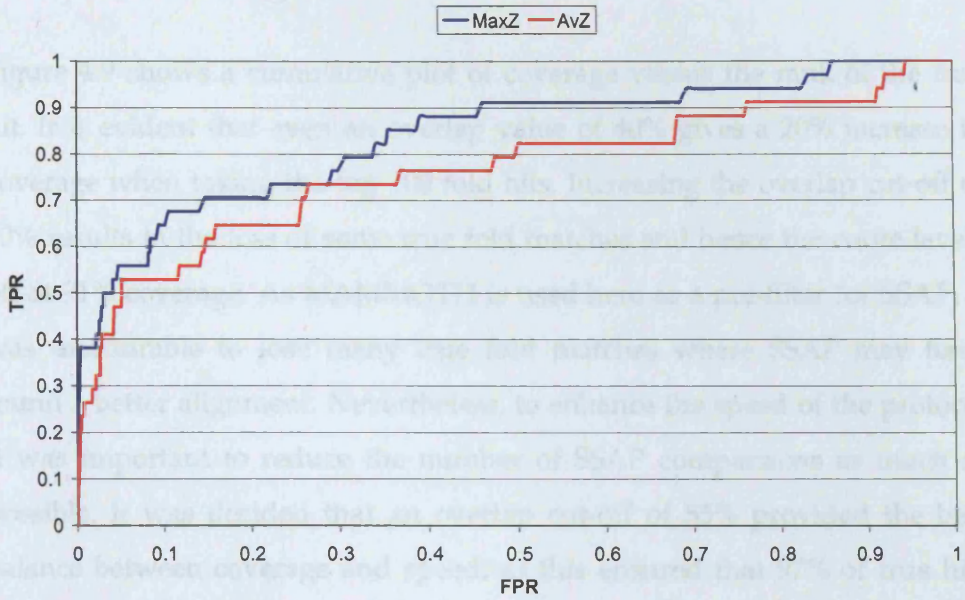


Figure 4.8 Comparison of MaxZ and AvZ scoring schemes for discovering fold matches using MAMMOTH. TPR = True positive rate or Coverage; FPR = False positive rate or coverage of domain pairs which are not in the same CATH fold

4.4.3.4 *Optimising the number of putative fold groups to re-compare against the Mod50 models using SSAP*

The protocol was designed to identify N putative fold groups, from which a representative structure (SRep) would be compared with SSAP against all the Mod50 models for each target. As discussed in Chapter 2, when selecting the most likely fold group matches to take forward to a SSAP scan, it is necessary to consider the degree of residue overlap involved in the model/fold match as well as the score (MaxZ). Matching smaller structural motifs may produce a reasonable score; however, it is not necessarily indicative of an overall fold similarity. Nevertheless, setting too stringent an overlap cut-off can cause genuine hits to be lost. Different overlap cut-offs were explored to see whether these could be used to reduce the number of fold hits that were taken forward for the SSAP comparison. For each target structure, the fold hits were ranked by their score (MaxZ) and a range of overlap cut-offs explored.

Figure 4.9 shows a cumulative plot of coverage versus the rank of the true hit. It is evident that even an overlap value of 40% gives a 20% increase in coverage when taking the top 100 fold hits. Increasing the overlap cut-off to 70% results in the loss of some true fold matches and hence the curve levels off at 91% coverage. As MAMMOTH is used here as a pre-filter for SSAP, it was undesirable to lose many true fold matches where SSAP may have found a better alignment. Nevertheless, to enhance the speed of the protocol it was important to reduce the number of SSAP comparisons as much as possible. It was decided that an overlap cut-off of 55% provided the best balance between coverage and speed, as this ensured that 97% of true hits were in the top 100 hits. It was felt that taking the top 100 fold hits forward for rescanning by SSAP provided a reasonable compromise between coverage and speed for the MODMATCH protocol.

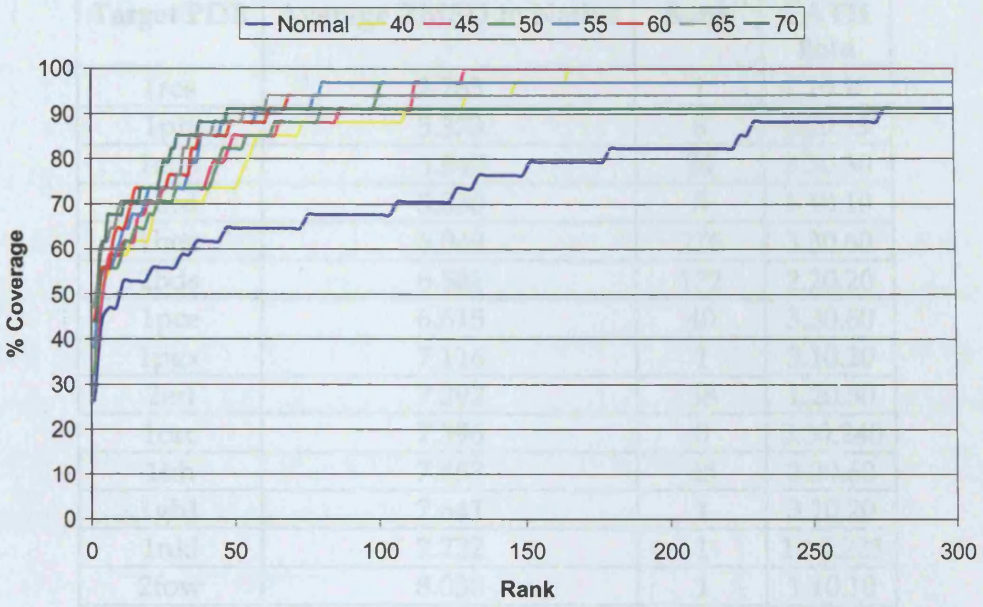


Figure 4.9 Cumulative coverage plot showing the MaxZ score performance at a range of overlap cut-offs (%). *Normal* denotes that no overlap cut-off was used.

Table 4.5 shows the rank (by MaxZ) of the correct fold in the MAMMOTH hit list for each target structure using an overlap cut-off of 55%, along with the average RMSD of the models to the native structure. It is perhaps surprising to note that targets where the average quality of models is good, such as 1aho and 1hev, rank the correct fold fairly low in the list. By contrast, target 2ezh has a generally poor selection of models, yet MAMMOTH ranks the correct fold as the top hit. This may be due to the fact that selecting models using the Modmean score fails for 1aho and 1hev and produces a bad sample, hence creating noise in the data set. A future improvement to MODMATCH might require exploring other strategies for reducing the model data set (see Discussion). Alternatively, it could be because these Rosetta models are built from a fragment library of known structures and hence highly populated folds like the *arc* repressor fold group (1.10.10) to which 2ezh belongs, could be better modelled.

Target PDB	Average RMSD to Native	Rank	CATH Fold
1res	2.263	1	1.10.10
1pft	5.373	8	2.20.25
1aho	5.547	24	3.30.30
1erd	5.850	8	1.10.10
1hev	6.049	276	3.30.60
2bds	6.581	172	2.20.20
1pce	6.615	40	3.30.60
1pgx	7.116	1	3.10.20
2erl	7.292	38	1.20.50
1orc	7.396	0	3.30.240
1tih	7.453	45	3.30.60
1gb1	7.641	1	3.10.20
1nkl	7.722	1	1.10.225
2fow	8.030	1	1.10.10
1tpm	8.110	23	2.10.70
1vtx	8.110	220	4.10.40
1hsn	8.129	1	1.10.30
1ctf	8.362	1	3.30.70
1ayj	8.405	3	3.30.30
1qyp	8.443	4	2.20.25
1aa3	8.569	1	3.30.250
2hp8	8.728	3	1.10.810
1c5a	8.765	1	1.20.91
5icb	8.945	1	1.10.238
2sn3	8.965	4	3.30.30
1mzm	9.071	1	1.10.110
1sro	9.201	1	2.40.50
1leb	9.297	1	1.10.10
2ptl	9.357	1	3.10.20
1nxb	9.581	2	2.10.60
1utg	9.605	318	1.10.210
1nre	9.722	1	1.20.81
2ezh	9.814	1	1.10.10
1csp	10.026	2	2.40.50

Table 4.5 Table showing rank of correct fold for each target PDB and the average RMSD to native.

4.4.3.5 *Re-comparing the Mod50 sample of ab initio models to the CATH library using SSAP*

To implement the second structure comparison stage in the protocol, SSAP was used to realign the Mod50 models against the best matched SRep from the top 100 fold groups (FoldHits100). This resulted in 5000 (50 x 100) comparisons per target structure. A SAS score was calculated for the match of each of the 50 models (Mod50), for a given query target structure, against each fold group (FoldHits100). Then, for each fold group the maximum SAS score (MaxSAS) was determined. This was compared with the MaxZ score (with no overlap cut-off) from the earlier MAMMOTH comparisons. The performance was again assessed using a ROC curve.

Figure 4.10 shows that both SSAP and MAMMOTH perform well, although SSAP actually performs better at low error rates (a 15% increase in coverage at a 5% error rate). This confirms that the superior performance of SSAP compared to MAMMOTH for identifying fold similarities shown in Section 4.4.1 is not confined to experimental structures.

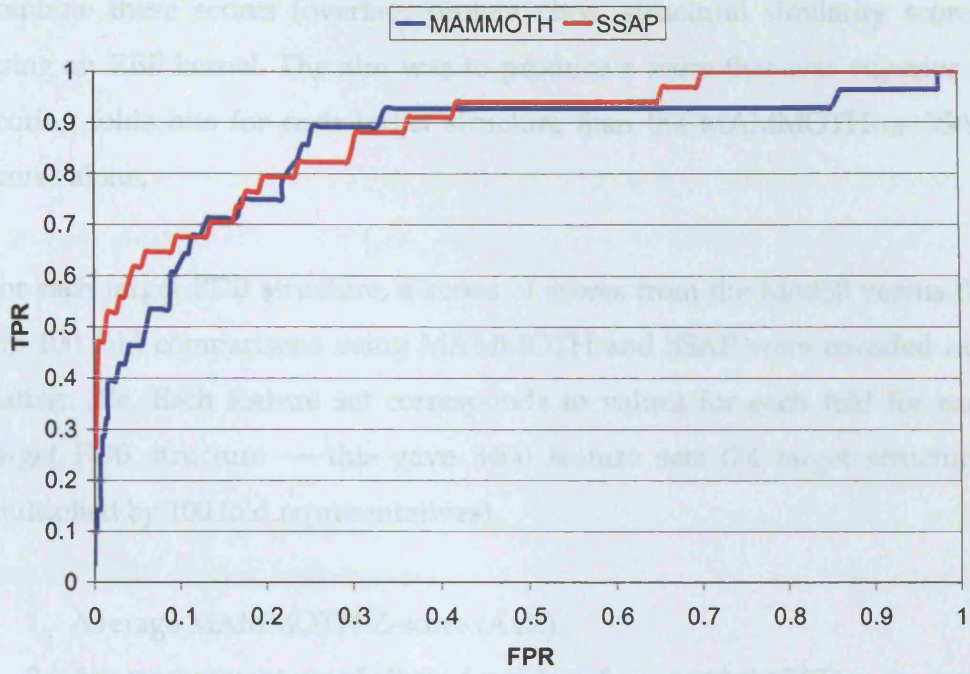


Figure 4.10 ROC curve analysis of SSAP and MAMMOTH for Mod50 vs. FoldHits100 comparisons.

4.4.3.6 Optimising the scoring scheme to predict the correct fold for ab initio models using an SVM

Although SSAP appears to be superior to MAMMOTH for identifying the correct fold when scanning the Rosetta models against the CATH library, it is conceivable that each method performs better with different types of structures. Furthermore, it has been shown that the performance of both algorithms can be improved by accounting for the percentage of aligned residues, as well as using different measures of structural similarity (SSAP MaxSAS and MAMMOTH MaxZ). To make use of all this information, a Support Vector Machine (SVM, see Section 1) was optimised to ascertain whether it could detect fold similarities more accurately by combining scores from MAMMOTH and SSAP alignments.

For each target structure, scores for each fold were extracted from the MAMMOTH and SSAP results, giving a set of structural similarity scores for

each fold in the CATH library. The SVMLight package was trained to combine these scores (overlap, protein class, structural similarity scores) using an RBF kernel. The aim was to produce a score that was superior at scoring folds hits for each target structure than the MAMMOTH or SSAP scores alone.

For each target PDB structure, a series of scores from the Mod50 versus the top 100 fold comparisons using MAMMOTH and SSAP were encoded in a pattern file. Each feature set corresponds to values for each fold for each target PDB structure — this gave 3400 feature sets (34 target structures multiplied by 100 fold representatives).

1. Average MAMMOTH Z-score (AvZ).
2. Average percentage of aligned residues from MAMMOTH.
3. Highest Z-score from MAMMOTH (MaxZ).
4. Highest percentage of aligned residues from MAMMOTH.
5. Average SSAP SAS score (AvSAS).
6. Average percentage of aligned residues from SSAP.
7. Lowest SAS score from SSAP (MaxSAS).
8. Highest percentage of aligned residues from SSAP.

Each feature in the pattern file was then scaled to values between 0 and 1, to avoid any bias towards a specific score in the SVM. It is more than possible that there might be some redundancy in the list of features above (for example, average and highest MAMMOTH Z-scores), which can affect the performance of other machine learning methods, such as neural networks. However, SVMs appear to not be affected by irrelevant or duplicated features (A. Lobley, personal communication).

Given that the training file only contained 34 true positives out of a total of 3400 patterns, an option in SVMLight that adds more weighting to misclassification of positive examples was used. This –j parameter was set to

108, which was the ratio by which the negative examples outweigh the positive hits.

A Radial Basis Function (RBF) kernel was chosen, which required optimisation of two parameters, C and γ. This optimisation was performed using a cross-validation approach, whereby the scores for a given target PDB were successively removed while the SVM was trained on the remaining targets — i.e. for a given optimisation cycle, data from 33 targets were used for training and the remaining target was used as the test set. For each pair of values for C and γ, the performance was measured as the area under a ROC curve (essentially the same graph as Figure 4.10 but using the SVM score) and averaged over all training sets.

Table 4.6 shows a selection of the optimisation results, sorted by the ROC curve area. A value of 0.0625 was taken for both C and γ in the final SVM model as this gave the maximal ROC area from range of optimisation values explored.

C	γ	ROC Area
0.0625	0.0625	0.8583
0.03125	0.125	0.8582
0.125	0.03125	0.8573
0.0625	0.03125	0.8558
0.03125	0.0625	0.8554
2	0.5	0.7986
8	0.25	0.7959
1	1	0.7902
2	1	0.7893
4	16	0.6517
8	16	0.6513
16	16	0.6510

Table 4.6 Optimisation of SVM parameters, C and γ, when training on MAMMOTH/SSAP alignment scores.

Results from the optimised SVM described above were compared against the performance of SSAP and MAMMOTH (Figure 4.11). It can be seen that the resulting score from the SVM produces a 15% increase in coverage over SSAP at a low false positive rate (5%). Moreover, that the SVM produces a superior score at all error rates, suggesting it is better at discriminating between true and false fold matches.

The overall objective of this protocol was to assign a fold to each of the 34 target structures. With this in mind, each fold was ranked by its SVM score to determine how often the top hit was the correct fold. This was compared to ranking folds by their MaxZ and MaxSAS scores for MAMMOTH and SSAP respectively. Table 4.7 shows that SSAP outperforms MAMMOTH by nearly 10%, when looking at the top hit. However, the SVM finds the correct fold 45.5% of the time as the top hit and coverage of 57.6% in the top 3 folds, compared with 51.5% and 42.4% for SSAP and MAMMOTH respectively. This suggests that an SVM is able to score structural similarity more effectively by combining measures of alignment quality (such as overlap and superposition scores), than MAMMOTH or SSAP are able to encapsulate in a single score.

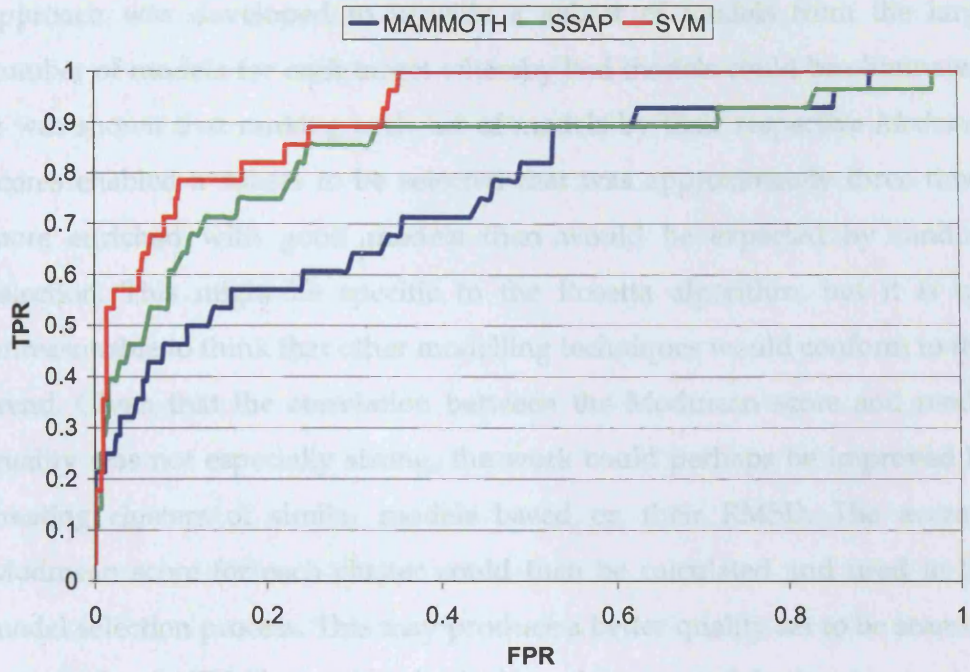


Figure 4.11 ROC curve comparison of SSAP, MAMMOTH and SVM scores for assessing the correct fold for model matches.

Rank	MAMMOTH	SSAP	SVM
1	33.3	42.4	45.5
2	6.1	6.1	6.1
3	3.0	3.0	6.1
4	3.0	3.0	6.1
5	3.0	3.0	0.0
6	6.1	3.0	3.0
7	0.0	3.0	0.0
8	3.0	0.0	0.0
9	0.0	0.0	0.0
10	3.0	0.0	0.0
>10	39.4	36.4	33.3

Table 4.7 Table showing percentage of correct folds when ranking hits by MAMMOTH, SSAP and SVM score

4.5 Discussion and Future Work

The MODMATCH protocol presented in this work was designed to provide a rapid and accurate means of assigning *ab initio* models from the Rosetta

method (Simons *et al.*, 1997) to fold groups in the CATH database. A novel approach was developed to provide a subset of models from the large number of models for each target whereby bad models could be eliminated. It was shown that ranking each list of models by their respective *Modmean* scores enabled a subset to be selected that was approximately three times more enriched with good models than would be expected by random selection. This might be specific to the Rosetta algorithm, but it is not unreasonable to think that other modelling techniques would conform to this trend. Given that the correlation between the Modmean score and model quality was not especially strong, the work could perhaps be improved by creating clusters of similar models based on their RMSD. The average Modmean score for each cluster could then be calculated and used in the model selection process. This may produce a better quality set to be scanned against the CATH library, by eliminating clusters models that have a low Modmean score. This will reduce the effect of large groupings of relatively poor models.

There did not appear to be a class-bias between the average qualities of the Rosetta models compared to their native structures. However, the average size of the models was not identical between classes and it could be argued that this should be taken into account, as larger structures can be harder to predict (Simons *et al.*, 2001). For future studies, it would be interesting to perform a more thorough analysis to see if, when length is taken into account, the all-beta structures do in fact produce lower quality models.

MAMMOTH was shown to be fast and reasonably accurate (55% coverage for a 5% error rate) for identifying folds in the CATH library that were in the same fold group to models in the data set. It was able to rank the correct fold in the top 5 matches for over 50% of the data, although the correct fold was below rank 100 for 4 of the targets. This performance was lower than obtained when using MAMMOTH with native structures, which suggests the drop was due to poor model quality. Despite the fact that MAMMOTH

was specifically designed for comparing theoretical models with experimental structures, it still does not perform as well as SSAP. The fact that the SVM score (which combined SSAP and MAMMOTH scores) showed superior performance to both SSAP and MAMMOTH suggests these structure comparison methods were complimentary to some extent. It would be interesting to investigate for which cases each method excels or fails at identifying the correct fold.

The overall protocol described here is both fast and suitable for large scale fold assignment for theoretical structures. The SVM score here is able to assign the correct fold at nearly 60% coverage with a 5% error, versus the 50% annotation achieved by Simons *et al.* (2001). As techniques improve for *ab initio* fold prediction, the strategy presented in this chapter could make the use of structure comparison a viable addition for genome annotation.

Chapter 5 Discussion and Conclusions

The aim of the work presented in this thesis was to develop a range of computational methods to improve the automated classification of protein structures into the CATH domain database. Where Chapter 2 dealt with detecting domain folds within the context of multi-domain proteins, Chapter 3 aimed to distinguish between functional sub-families within diverse evolutionary superfamilies. Chapter 4 applied structure comparison methods to theoretical ab initio models to predict the corresponding fold group in the CATH database.

Chapter 2 described the development of CATHEDRAL for assigning domain boundaries and folds to multi-domain proteins by exploiting the recurrence of folds in different multi-domain contexts. CATHEDRAL combines the power of two established structural comparison algorithms (GRATH and SSAP) to produce a fast and accurate protocol for fold recognition and domain assignment. On the data set used, CATHEDRAL found the correct domain boundaries within 15 residues in 86% of cases. However, it is unable to assign ~10% of the domains. These domains are often missed by GRATH due to their small size or because their secondary structures are poorly defined. Alternatively, although the correct domain is identified, it is distantly related to the query, so this structural variation results in incomplete alignments and hence erroneous domain boundaries. For these more difficult cases, it is essential that domain boundaries are manually verified so as not to propagate errors in the CATH database.

In order for CATHEDRAL to assign domain boundaries, it must perform a residue-based structural alignment (in this case using SSAP). Despite the fact

that automatic protein structure comparison and alignment can be traced back to the pioneering work of Rossmann and Argos in the 1970s, it is still an active area of research. It can be argued that one of the reasons for this is that structure comparison methods are applied to a vast number of different problems. For example, using algorithms to calculate the structural change that occurs on ligand binding requires a different emphasis to aligning all equivalent residues to determine how proteins have changed their 3D structure within an evolutionary superfamily. The latter problem of homologue detection can be performed very effectively using secondary structure methods such as GRATH (Harrison *et al.*, 2003) or SSM (Krissinel and Henrick, 2004) for more closely related proteins. However, within diverse superfamilies, creating alignments between proteins and assigning their significance can be problematic, as paralogous genes where function has changed substantially can show a large amount of structural variation, despite sharing a common fold (Reeves *et al.*, 2006).

On a practical level, even the most powerful structural alignment methods such as DALI, SSAP and STRUCTAL are optimised to achieve a balance between structural similarity and aligning all equivalent residues. As such, the resulting alignment might be limited to the conserved core of the fold, rather than determining all evolutionary equivalent parts of the structures. This is evinced by the fact that even DALI and SSAP are often unable to find a full alignment (50% of residues in 50% of protein pairs) when compared to manually curated data sets such as HOMSTRAD and BaliBase. Kolodny et al. (2005) suggested that no single structure alignment algorithm will always find the best alignment between two structures and hence it is better to apply several methods and choose the one that produces the best RMSD on superposition. However, their assessment did not take into account the fact that those alignments with higher RMSD might have actually aligned more biologically equivalent residues. Again, the choice of structural comparison algorithm depends on the application. A method that generally aligns fewer residues but is able to accurately assess the significance of the alignment to

detect homologues or fold relatives is useful for structural clustering and even classification. However, if one seeks to analyse how proteins have evolved within their structural superfamilies, it is vital to have as full an alignment as possible. Without this, important information on how domains have evolved new functions could be lost.

CATHEDRAL was benchmarked to detect the component folds within a multi-domain context. However, some argue (Kolodny *et al.*, 2006) that the idea of partitioning protein structure space into discrete fold groups is no longer appropriate. Indeed, there is evidence for a fold continuum, certainly within some areas of fold space (Harrison *et al.*, 2002). As such, benchmarking structure comparison methods in a binary fashion using SCOP or CATH, might unfairly penalise a method for finding genuine structural similarities that are not represented by the classification system. However, as was shown in Chapter 2, it is vital to consider the relative length of the structural overlap discovered by structure alignment, otherwise the alignment may simply represent the detection of super-secondary structure motifs that are present in a diverse range of folds and thus are not indicative of a meaningful homologous relationship or fold similarity. In fact, recent analysis of the CATH database has shown that the majority of "structural overlaps" detected by some structural comparison algorithms are actually the result of these common motifs occurring between small domains with less than 6 secondary structures (A. Cuff, unpublished data).

Although the CATHEDRAL algorithm was optimised to make CATH fold assignments to multi-domain chains, it is certainly possible that this might not always be necessary to correctly assign domain boundaries. In some cases, finding a similar fold could still allow CATHEDRAL to effectively detect the hydrophobic core of each domain. Indeed, looking for such structural cores are the basis of other domain prediction methods (e.g. (Swindells, 1995)). Nevertheless, as the fidelity of predicted domain boundaries in the query protein is dependent on the similarity to the

matched domain in CATH, these boundaries might still be of suspect quality. However, future work is underway to use a more intelligent system for refining domain boundaries in these cases. One of the most common sources of errors for all domain prediction methods is the lack of a concrete definition for a domain. It would be interesting to examine whether this problem could be alleviated by first assigning folds using CATHEDRAL and then adapting other approaches (e.g. (Swindells, 1995; Holm and Sander, 1994; Taylor, 1999)) to refine the boundaries. For the purposes of building sequence profiles, such as HMMs, for the analysis of genomic data, it is vital that these boundaries are correct. Furthermore, as the PDB expands, it will become increasingly difficult to manually classify structures into CATH and if CATHEDRAL can be relied on to make more accurate assignments, manual intervention will be restricted to novel folds and superfamilies. Other additions to CATHEDRAL could be to annotate multi-domain chains at the superfamily level, with the aim of aiding the assignment of protein function through domain architecture information.

Chapter 3 dealt with designing a novel algorithm (FLORA) to predict the functionally related protein domains in enzyme families from their structural similarities. By combining patterns of sequence conservation and solvent accessibility, the method was able to correctly predict the active site in ~80% of cases. However, the templates it selected from structurally conserved positions around this site did not always discriminate well between functional homologues within a superfamily, in comparison with finding the closest relative using global structure comparison (SSAP).

It could be argued that Designing algorithms to detect functional similarities between proteins one of the most difficult problems in bioinformatics, not least because a definition of protein function is highly context dependent. Finding close homologues to the query protein via sequence or structure similarity is often sufficient to transfer a whole range of functional similarities such as enzymatic activity, cognate ligands and biological

pathway information. However, as genes mutate randomly in different organisms, they often become quite structurally different from their ancestral protein. To predict at what point a mutation or indel will result in a change in function is problematic as it is highly dependent on where they occur. As a consequence, the vast majority of methods to predict function from structure focus on identifying changes and similarities within known or predicted functional sites. However, even within evolutionary families where proteins exhibit similar functional (e.g. enzyme) activities, they might have converged on a solution through different evolutionary pathways. For example, two related proteins might have a highly similar enzymatic function but utilise catalytic residues from slightly different parts of the structure (Todd *et al.*, 2002b) and therefore encompassing this function in a structural template might become problematic.

Given the difficulties associated with characterizing protein function when developing prediction methods, it is vital to clearly define the criteria on which novel algorithms are benchmarked. Even if one looks to group proteins by a common catalytic activity (as was done in Chapter 3), it is important to select a representative data set. Given the very different ways in which function changes across different superfamilies, it is important to include as many as possible to show that a novel method is able to work equally as effectively across the entire protein universe. A consistent benchmark is currently lacking in the literature; there is not currently a standard data set against which new methods should be compared. The most probable reason for this is that assembling such a data set requires time consuming manual analysis, especially to cover some of the largest protein families, such as the P-loop hydrolases or the aldolases.

Furthermore, it is also important for the community to arrive at a consensus as to the most pressing problems that need to be solved. The majority of structure-based prediction methods are justified on the basis that there are hundreds of new structures coming out of the structural genomics initiatives

(SGIs) and it is therefore important to identify their function in order to assess whether they should be prioritised for further experimental investigation. The broad aim of the SGIs was to put the vast majority, if not all, genomic sequences within the reach of homology modelling methods. The result being that every protein would have a structure, should we need to better understand its function for biotechnological or medical gain.

A successful protein function prediction algorithm should be able to rank the closest functional relatives at the top of a database search and also provide a reliable scoring function that is able to accurately discriminate between true and false matches. The latter problem is often far more difficult to solve due to the large structural variation observed in some protein superfamilies. However, it could be argued that a method which is able to identify the conserved residues that are particular to a specific biological function can tell us more about how proteins evolve. The P-loop hydrolase superfamily performed poorly using FLORA and this was most likely due to the structural diversity observed across each of its constituent enzyme families. As a consequence, CORA was unable to align a large number of residues across the multiple structure alignment. New local scoring schemes could be developed to increase the power of the CORASCORE for finding the closest functional relative and providing a normalised score cut-off that can be used to transfer function between all relatives within an enzyme family. As one of the problems with the FLORA method was optimising both the template construction and scanning procedures together, it would be interesting to use SSAP to align relatives within a superfamily and instead look at which residue positions are best able to identify domains with the same function.

Chapter 4 described the development of a new protocol (MODMATCH) for assigning ab initio predictions of structural domains to folds within the CATH database. The focus of the work was to ensure the method was accurate in its fold assignments by implementing a two-stage structure comparison process, following by machine learning using a SVM to combine

alignment scores. By reducing the ab initio models for each target to a sample set of models which on average were closer to the native structure (using the Modmean score), it was possible to drastically reduce the number of structure comparisons required. Although this latter process appeared to work effectively for the test data set as a whole, it would be interesting to look at targets where the Modmean score does not correlate with the quality of the models. For example, to analyse whether it is more effective for models that are generally close to the native structure, or whether it is particularly powerful for models where the mean RMSD to native is low due to a smaller number of outliers.

Over the past few years, substantial progress has been made in the field of ab initio structure prediction, especially using the approaches of the Baker group. Of particular interest is the work of Malmstrom et al. (Malmstrom *et al.*, 2007), that applied the Rosetta algorithm to small domains in the Yeast proteome for which a structure prediction could not be made using homology or fold recognition methods. The MAMMOTH structure comparison method was used to make putative assignments for these models to superfamilies in SCOP. A Bayesian approach was then used to combine these data with functional annotation predictions to confirm superfamily assignments. The authors' use of MAMMOTH is understandable given that it is such a fast algorithm and specifically designed to compare protein structure models against experimental structures, such as representatives from the SCOP database. However, as was demonstrated in Chapter 4, SSAP was shown to significantly outperform MAMMOTH for finding genuine structural relatives in the CATH database from Rosetta models. Given that the Baker group had previously used DALI to make such assignments (Simons, 2001), it is interesting that they chose MAMMOTH for their automated pipeline due to its superior speed. It could be argued that using a combination of MAMMOTH and SSAP could have increased the number of assignments from Rosetta models.

Overall, this thesis has shown that global structure comparison methods can be modified and integrated into novel algorithms to assign domain boundaries, inherit functional annotations and make fold predictions from theoretical models. Despite this, there is still scope for improvement to the basics of structure comparison methods. Ye and Godzik (Ye and Godzik, 2003), Shatsky et al. (Shatsky *et al.*, 2004) and Menke et al. (Menke *et al.*, 2008), have all developed methods based on the idea that structural alignments should incorporate a degree of flexibility to allow for conformational changes. As some structures continue to be released from the structural genomics projects with little functional annotation, as well as the prospect of homology detection via structure prediction, there is likely to be increasing focus on predicting function from structure. Although local motif methods continue to be important, there is certainly scope for further utilising global structure alignment in novel ways in order to improve methods' ability to detect homologous genes and better understand the relationship between protein structure and function.

Bibliography

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

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

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

4. Apweiler, R., Attwood, T. K., Bairoch, A., Bateman, A., Birney, E., Biswas, M., Bucher, P., Cerutti, L., Corpet, F., Croning, M. D., Durbin, R., Falquet, L., Fleischmann, W., Gouzy, J., Hermjakob, H., Hulo, N., Jonassen, I., Kahn, D., Kanapin, A., Karavidopoulou, Y., Lopez, R., Marx, B., Mulder, N. J., Oinn, T. M., Pagni, M., Servant, F., Sigrist, C. J. & Zdobnov, E. M. (2001). The InterPro database, an integrated documentation resource for protein families, domains and functional sites. *Nucleic Acids Res.* **29**, 37-40.

5. Apweiler, R., Bairoch, A., Wu, C. H., Barker, W. C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., Magrane, M., Martin, M. J., Natale, D. A., O'Donovan, C., Redaschi, N. & Yeh, L. S. (2004). UniProt: the Universal Protein knowledgebase. *Nucleic Acids Res.* **32**, D115-D119.

6. Artymiuk, P. J., Poirrette, A. R., Grindley, H. M., Rice, D. W. & Willett, P. (1994). A graph-theoretic approach to the identification of three-dimensional patterns of amino acid side-chains in protein structures. *J. Mol. Biol.* **243**, 327-344.

7. 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. & Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.* **25**, 25-29.

8. 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. & Zygouri, C. (2003). PRINTS and its automatic supplement, prePRINTS. *Nucleic Acids Res.* **31**, 400-402.

9. Bairoch, A. (2000). The ENZYME database in 2000. *Nucleic Acids Res.* **28**, 304-305.

10. Barker, J. A. & Thornton, J. M. (2003). An algorithm for constraint-based structural template matching: application to 3D templates with statistical analysis. *Bioinformatics.* **19**, 1644-1649.

11. Bartlett, G. J., Porter, C. T., Borkakoti, N. & Thornton, J. M. (2002). Analysis of catalytic residues in enzyme active sites. *J. Mol. Biol.* **324**, 105-121.

12. Bateman, A., Birney, E., Cerruti, L., Durbin, R., Etwiller, L., Eddy, S. R., Griffiths-Jones, S., Howe, K. L., Marshall, M. & Sonnhammer, E. L. (2002). The Pfam protein families database. *Nucleic Acids Res.* **30**, 276-280.

13. Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L., Studholme, D. J., Yeats, C. & Eddy, S. R. (2004). The Pfam protein families database. *Nucleic Acids Res.* **32**, D138-D141.

14. Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J. & Wheeler, D. L. (2006). GenBank. *Nucleic Acids Res.* **34**, D16-D20.

15. Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. & Bourne, P. E. (2000). The Protein Data Bank. *Nucleic Acids Res.* **28**, 235-242.

16. Bron, C. & Kerbosch, J. (1973). Algorithm 457 -- finding all cliques of an undirected graph. *Commun. Assoc. Comput. Mach* **16**, 575.

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

18. Chothia, C. (1992). Proteins. One thousand families for the molecular biologist. *Nature.* **357**, 543-544.

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

20. Chou, K. C. (2005). Progress in protein structural class prediction and its impact to bioinformatics and proteomics. *Curr. Protein Pept. Sci.* **6**, 423-436.

21. Chou, P. Y. & Fasman, G. D. (1974). Prediction of protein conformation. *Biochemistry.* **13**, 222-245.

22. Corpet, F., Gouzy, J. & Kahn, D. (1998). The ProDom database of protein domain families. *Nucleic Acids Res.* **26**, 323-326.

23. Dandekar, T. & Argos, P. (1994). Folding the main chain of small proteins with the genetic algorithm. *J. Mol. Biol.* **236**, 844-861.

24. Dayhoff, M. O. (1978). Atlas of Protein Sequence and Structure.

25. de la Cruz, X., Sillitoe, I. & Orengo, C. (2002). Use of structure comparison methods for the refinement of protein structure predictions. I. Identifying the structural family of a protein from low-resolution models. *Proteins* **46**, 72-84.

26. Doolittle, R. F. (1986). *Of URFs and ORFs: a primer on how to analyze derived amino acid sequences* University Science Books, Mill Valley, CA, USA.
Ref Type: Book, Whole

27. Doolittle, R. F. (1990). Searching through sequence databases. *Methods Enzymol.* **183**, 99-110.

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

29. Eisenhaber, F., Frommel, C. & Argos, P. (1996). Prediction of secondary structural content of proteins from their amino acid composition alone. II. The paradox with secondary structural class. *Proteins.* **25**, 169-179.

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

31. Garnier, J., Osguthorpe, D. J. & Robson, B. (1978). Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. Mol. Biol.* **120**, 97-120.

32. Gough, J. (2002). The SUPERFAMILY database in structural genomics. *Acta Crystallogr. D. Biol. Crystallogr.* **58**, 1897-1900.

33. Grant, A., Lee, D. & Orengo, C. (2004). Progress towards mapping the universe of protein folds. *Genome Biol.* **5**, 107.

34. Grindley, H. M., Artymiuk, P. J., Rice, D. W. & Willett, P. (1993). Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism algorithm. *J. Mol. Biol.* **229**, 707-721.

35. Hadley, C. & Jones, D. T. (1999). A systematic comparison of protein structure classifications: SCOP, CATH and FSSP. *Structure.* **7**, 1099-1112.

36. Hardin, C., Pogorelov, T. V. & Luthey-Schulten, Z. (2002). Ab initio protein structure prediction. *Curr. Opin. Struct. Biol.* **12**, 176-181.

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

38. Harrison, A., Pearl, F., Sillitoe, I., Slidel, T., Mott, R., Thornton, J. & Orengo, C. (2003). Recognizing the fold of a protein structure. *Bioinformatics.* **19**, 1748-1759.

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

40. Hinds, D. A. & Levitt, M. (1994). Exploring conformational space with a simple lattice model for protein structure. *J. Mol. Biol.* **243**, 668-682.

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

42. Holm, L. & Sander, C. (1994). Parser for protein folding units. *Proteins.* **19**, 256-268.

43. Holm, L. & Sander, C. (1996). Mapping the protein universe. *Science.* **273**, 595-603.

44. Holm, L. & Sander, C. (1997). Dali/FSSP classification of three-dimensional protein folds. *Nucleic Acids Res.* **25**, 231-234.

45. Holm, L. & Sander, C. (1998). Touring protein fold space with Dali/FSSP. *Nucleic Acids Res.* **26**, 316-319.

46. Hubbard, S. J. & Thornton, J. M. (1993). NAccess.

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

48. Hulo, N., Sigrist, C. J., Le, S., V, Langendijk-Genevaux, P. S., Bordoli, L., Gattiker, A., De Castro, E., Bucher, P. & Bairoch, A. (2004). Recent improvements to the PROSITE database. *Nucleic Acids Res.* **32**, D134-D137.

49. Janssen, P., Audit, B., Cases, I., Darzentas, N., Goldovsky, L., Kunin, V., Lopez-Bigas, N., Peregrin-Alvarez, J. M., Pereira-Leal, J. B., Tsoka, S. & Ouzounis, C. A. (2003). Beyond 100 genomes. *Genome Biol.* **4**, 402.

50. Jones, D. T. (1999). Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol.* **292**, 195-202.

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

52. Jones, S., Stewart, M., Michie, A., Swindells, M. B., Orengo, C. & Thornton, J. M. (1998). Domain assignment for protein structures using a consensus approach: characterization and analysis. *Protein Sci.* **7**, 233-242.

53. Karplus, K., Katzman, S., Shackleford, G., Koeva, M., Draper, J., Barnes, B., Soriano, M. & Hughey, R. (2005). SAM-T04: what is new in protein-structure prediction for CASP6. *Proteins.* **61 Suppl 7:135-42.**, 135-142.

54. Kinoshita, K. & Nakamura, H. (2004). eF-site and PDBjViewer: database and viewer for protein functional sites. *Bioinformatics.* **20**, 1329-1330.

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

56. Kolodny, R., Petrey, D. & Honig, B. (2006). Protein structure comparison: implications for the nature of 'fold space', and structure and function prediction. *Curr. Opin. Struct. Biol.* **16**, 393-398.

57. Koppensteiner, W. A., Lackner, P., Wiederstein, M. & Sippl, M. J. (2000). Characterization of novel proteins based on known protein structures. *J. Mol. Biol.* **296**, 1139-1152.

58. Krause, A., Stoye, J. & Vingron, M. (2000). The SYSTERS protein sequence cluster set. *Nucleic Acids Res.* **28**, 270-272.

59. Krissinel, E. & 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.

60. Laskowski, R. A., Hutchinson, E. G., Michie, A. D., Wallace, A. C., Jones, M. L. & Thornton, J. M. (1997). PDBsum: a Web-based database of summaries and analyses of all PDB structures. *Trends Biochem. Sci.* **22**, 488-490.

61. Laskowski, R. A., Luscombe, N. M., Swindells, M. B. & Thornton, J. M. (1996). Protein clefts in molecular recognition and function. *Protein Sci.* **5**, 2438-2452.

62. Laskowski, R. A., Watson, J. D. & Thornton, J. M. (2005). Protein function prediction using local 3D templates. *J. Mol. Biol.* **351**, 614-626.

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

64. Lichtarge, O., Bourne, H. R. & Cohen, F. E. (1996). An evolutionary trace method defines binding surfaces common to protein families. *J. Mol. Biol.* **257**, 342-358.

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

66. Liu, Z. P., Wu, L. Y., Wang, Y., Chen, L. & Zhang, X. S. (2007). Predicting gene ontology functions from protein's regional surface structures. *BMC. Bioinformatics* **8**, 475.

67. Madej, T., Gibrat, J. F. & Bryant, S. H. (1995). Threading a database of protein cores. *Proteins* **23**, 356-369.

68. Malmstrom, L., Riffle, M., Strauss, C. E., Chivian, D., Davis, T. N., Bonneau, R. & Baker, D. (2007). Superfamily assignments for the yeast proteome through integration of structure prediction with the gene ontology. *PLoS. Biol.* **5**, e76.

69. Marchler-Bauer, A., Addess, K. J., Chappey, C., Geer, L., Madej, T., Matsuo, Y., Wang, Y. & Bryant, S. H. (1999). MMDB: Entrez's 3D structure database. *Nucleic Acids Res.* **27**, 240-243.

70. McLachlan, A. D. (1979). Gene duplications in the structural evolution of chymotrypsin. *J. Mol. Biol.* **128**, 49-79.

71. Menke, M., Berger, B. & Cowen, L. (2008). Matt: local flexibility aids protein multiple structure alignment. *PLoS. Comput. Biol.* **4**, e10.

72. Miller, R. T., Jones, D. T. & Thornton, J. M. (1996). Protein fold recognition by sequence threading: tools and assessment techniques. *FASEB J.* **10**, 171-178.

73. Mirny, L. & Shakhnovich, E. (2001). Evolutionary conservation of the folding nucleus. *J. Mol. Biol.* **308**, 123-129.

74. Mizuguchi, K., Deane, C. M., Blundell, T. L. & Overington, J. P. (1998). HOMSTRAD: a database of protein structure alignments for homologous families. *Protein Sci.* **7**, 2469-2471.

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

76. Needleman, S. B. & 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.

77. Novotny, M., Madsen, D. & Kleywegt, G. J. (2004). Evaluation of protein fold comparison servers. *Proteins* **54**, 260-270.

78. Orengo, C. A. (1999). CORA--topological fingerprints for protein structural families. *Protein Sci.* **8**, 699-715.

79. Orengo, C. A., Brown, N. P. & Taylor, W. R. (1992). Fast structure alignment for protein databank searching. *Proteins* **14**, 139-167.

80. Orengo, C. A., Jones, D. T. & Thornton, J. M. (1994). Protein superfamilies and domain superfolds. *Nature* **372**, 631-634.

81. Orengo, C. A., Michie, A. D., Jones, S., Jones, D. T., Swindells, M. B. & Thornton, J. M. (1997). CATH--a hierarchic classification of protein domain structures. *Structure.* **5**, 1093-1108.

82. Orengo, C. A. & Taylor, W. R. (1993). A local alignment method for protein structure motifs. *J. Mol. Biol.* **233**, 488-497.

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

84. Ortiz, A. R., Strauss, C. E. & Olmea, O. (2002). MAMMOTH (matching molecular models obtained from theory): an automated method for model comparison. *Protein Sci.* **11**, 2606-2621.

85. Park, B. H. & Levitt, M. (1995). The complexity and accuracy of discrete state models of protein structure. *J. Mol. Biol.* **249**, 493-507.

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

87. Pawlowski, K. & Godzik, A. (2001). Surface map comparison: studying function diversity of homologous proteins. *J. Mol. Biol.* **309**, 793-806.

88. Pearl, F., Todd, A., Sillitoe, I., Dibley, M., Redfern, O., Lewis, T., Bennett, C., Marsden, R., Grant, A., Lee, D., Akpor, A., Maibaum, M., Harrison, A., Dallman, T., Reeves, G., Diboun, I., Addou, S., Lise, S., Johnston, C., Sillero, A., Thornton, J. & Orengo, C. (2005). The CATH Domain Structure Database and related resources Gene3D and DHS provide comprehensive domain family information for genome analysis. *Nucleic Acids Res.* **33**, D247-D251.

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

90. Polacco, B. J. & Babbitt, P. C. (2006). Automated discovery of 3D motifs for protein function annotation. *Bioinformatics.* **22**, 723-730.

91. Pollastri, G. & Baldi, P. (2002). Prediction of contact maps by GIOHMMs and recurrent neural networks using lateral propagation from all four cardinal corners. *Bioinformatics.* **18 Suppl 1:S62-70.**, S62-S70.

92. Ponting, C. P., Schultz, J., Milpetz, F. & Bork, P. (1999). SMART: identification and annotation of domains from signalling and extracellular protein sequences. *Nucleic Acids Res.* **27**, 229-232.

93. Porter, C. T., Bartlett, G. J. & 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.

94. Rangwala, H. & Karypis, G. (2005). Profile-based direct kernels for remote homology detection and fold recognition. *Bioinformatics.* **21**, 4239-4247.

95. Rangwala, H. & Karypis, G. (2006). Building multiclass classifiers for remote homology detection and fold recognition. *BMC. Bioinformatics.* **7**, 455.

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

97. Richardson, J. S. (1981). The anatomy and taxonomy of protein structure. *Adv. Protein Chem.* **34:167-339.**, 167-339.

98. Rison, S. C. & Thornton, J. M. (2002). Pathway evolution, structurally speaking. *Curr. Opin. Struct. Biol.* **12**, 374-382.

99. Rost, B. (1997). Protein structures sustain evolutionary drift. *Fold. Des* **2**, S19-S24.

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

101. Rost, B. & Sander, C. (1993). Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.* **%20;232**, 584-599.

102. Rost, B., Sander, C. & Schneider, R. (1994). PHD--an automatic mail server for protein secondary structure prediction. *Comput. Appl. Biosci.* **10**, 53-60.

103. Sali, A. & Blundell, T. L. (1990). Definition of general topological equivalence in protein structures. A procedure involving comparison of properties and relationships through simulated annealing and dynamic programming. *J. Mol. Biol.* **%20;212**, 403-428.

104. Sasin, J. M., Godzik, A. & Bujnicki, J. M. (2007). SURF'S UP! - protein classification by surface comparisons. *J. Biosci.* **32**, 97-100.

105. Sasson, O., Vaaknin, A., Fleischer, H., Portugaly, E., Bilu, Y., Linial, N. & Linial, M. (2003). ProtoNet: hierarchical classification of the protein space. *Nucleic Acids Res.* **31**, 348-352.

106. Shatsky, M., Nussinov, R. & Wolfson, H. J. (2004). FlexProt: alignment of flexible protein structures without a predefinition of hinge regions. *J. Comput. Biol.* **11**, 83-106.

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

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

109. Shulman-Peleg, A., Nussinov, R. & Wolfson, H. J. (2005). SiteEngines: recognition and comparison of binding sites and protein-protein interfaces. *Nucleic Acids Res.* **33**, W337-W341.

110. Sibanda, B. L. & Thornton, J. M. (1991). Conformation of beta hairpins in protein structures: classification and diversity in homologous structures. *Methods Enzymol.* **202**, 59-82.

111. Siddiqui, A. S. & Barton, G. J. (1995). Continuous and discontinuous domains: an algorithm for the automatic generation of reliable protein domain definitions. *Protein Sci.* **4**, 872-884.

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

113. Simons, K. T., Kooperberg, C., Huang, E. & Baker, D. (1997). Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *J. Mol. Biol.* **268**, 209-225.

114. Simons, K. T., Ruczinski, I., Kooperberg, C., Fox, B. A., Bystroff, C. & Baker, D. (1999). Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins. *Proteins.* **34**, 82-95.

115. Simons, K. T., Strauss, C. & Baker, D. (2001). Prospects for ab initio protein structural genomics. *J. Mol. Biol.* **306**, 1191-1199.

116. Srinivasan, R. & Rose, G. D. (1995). LINUS: a hierarchic procedure to predict the fold of a protein. *Proteins.* **22**, 81-99.

117. Stark, A. & Russell, R. B. (2003). Annotation in three dimensions. PINTS: Patterns in Non-homologous Tertiary Structures. *Nucleic Acids Res.* **31**, 3341-3344.

118. Swindells, M. B. (1995). A procedure for the automatic determination of hydrophobic cores in protein structures. *Protein Sci.* **4**, 93-102.

119. 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. & Natale, D. A. (2003). The COG database: an updated version includes eukaryotes. *BMC. Bioinformatics.* **4**, 41.

120. Taylor, W. R. (1999). Protein structural domain identification. *Protein Eng.* **12**, 203-216.

121. Taylor, W. R. & Hatrick, K. (1994). Compensating changes in protein multiple sequence alignments. *Protein Eng.* **7**, 341-348.

122. Taylor, W. R. & Orengo, C. A. (1989). Protein structure alignment. *J. Mol. Biol.* **208**, 1-22.

123. Thompson, J. D., Plewniak, F. & Poch, O. (1999). BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics.* **15**, 87-88.

124. Tian, W. & Skolnick, J. (2003). How well is enzyme function conserved as a function of pairwise sequence identity? *J. Mol. Biol.* **333**, 863-882.

125. Todd, A. E., Marsden, R. L., Thornton, J. M. & Orengo, C. A. (2005). Progress of structural genomics initiatives: an analysis of solved target structures. *J. Mol. Biol.* **%20;348**, 1235-1260.

126. Todd, A. E., Orengo, C. A. & Thornton, J. M. (2002a). Sequence and structural differences between enzyme and nonenzyme homologs. *Structure. (Camb.)* **10**, 1435-1451.

127. Todd, A. E., Orengo, C. A. & Thornton, J. M. (2002b). Sequence and structural differences between enzyme and nonenzyme homologs. *Structure.* **10**, 1435-1451.

128. Torrance, J. W., Bartlett, G. J., Porter, C. T. & Thornton, J. M. (2005). Using a library of structural templates to recognise catalytic sites and

explore their evolution in homologous families. *J. Mol. Biol.* **347**, 565-581.

129. Valdar, W. S. (2002). Scoring residue conservation. *Proteins.* **48**, 227-241.

130. Valdar, W. S. & Thornton, J. M. (2001). Conservation helps to identify biologically relevant crystal contacts. *J. Mol. Biol.* **313**, 399-416.

131. Veretnik, S., Bourne, P. E., Alexandrov, N. N. & Shindyalov, I. N. (2004). Toward consistent assignment of structural domains in proteins. *J. Mol. Biol.* **339**, 647-678.

132. Vladimir N.Vapnik (1995). *The Nature of Statistical Learning Theory* Springer.
Ref Type: Book, Whole

133. Wallace, A. C., Borkakoti, N. & Thornton, J. M. (1997). TESS: a geometric hashing algorithm for deriving 3D coordinate templates for searching structural databases. Application to enzyme active sites. *Protein Sci.* **6**, 2308-2323.

134. Wangikar, P. P., Tendulkar, A. V., Ramya, S., Mali, D. N. & Sarawagi, S. (2003). Functional sites in protein families uncovered via an objective and automated graph theoretic approach. *J. Mol. Biol.* **326**, 955-978.

135. Wu, C. H., Apweiler, R., Bairoch, A., Natale, D. A., Barker, W. C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., Magrane, M., Martin, M. J., Mazumder, R., O'Donovan, C., Redaschi, N. & Suzek, B. (2006). The Universal Protein Resource (UniProt): an expanding universe of protein information. *Nucleic Acids Res.* **34**, D187-D191.

136. Yao, H., Mihalek, I. & Lichtarge, O. (2006). Rank information: a structure-independent measure of evolutionary trace quality that improves identification of protein functional sites. *Proteins.* **65**, 111-123.

137. Ye, Y. & Godzik, A. (2003). Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics* **19 Suppl 2**, ii246-ii255.

138. Yeats, C., Maibaum, M., Marsden, R., Dibley, M., Lee, D., Addou, S. & Orengo, C. A. (2006). Gene3D: modelling protein structure, function and evolution. *Nucleic Acids Res.* **34**, D281-D284.