Structural Information and Hidden Markov Models for Biological Sequence Analysis

JEANETTE TÅNGROT

Umeå Centre for Molecular Pathogenesis
and
Department of Computing Science
Umeå University

Umeå Centre for Molecular Pathogenesis
and
Department of Computing Science
Umeå University
SE-901 87 Umeå, Sweden

jeanette@cs.umu.se

ISBN 978-91-7264-369-7
ISSN 0348-0542
UMINF 08.05

Printed by Print & Media, Umeå University, 2008

Till Olof

Abstract

Bioinformatics is a fast-developing field, which makes use of computational methods to analyse and structure biological data. An important branch of bioinformatics is structure and function prediction of proteins, which is often based on finding relationships to already characterized proteins. It is known that two proteins with very similar sequences also share the same 3D structure. However, there are many proteins with similar structures that have no clear sequence similarity, which make it difficult to find these relationships.

In this thesis, two methods for annotating protein domains are presented, one aiming at assigning the correct domain family or families to a protein sequence, and the other aiming at fold recognition. Both methods use hidden Markov models (HMMs) to find related proteins, and they both exploit the fact that structure is more conserved than sequence, but in two different ways.

Most of the research presented in the thesis focuses on the structure-anchored HMMs, saHMMs. For each domain family, an saHMM is constructed from a multiple structure alignment of carefully selected representative domains, the saHMM-members. These saHMM-members are collected in the so called "midnight ASTRAL set", and are chosen so that all saHMM-members within the same family have mutual sequence identities below a threshold of about 20%. In order to construct the midnight ASTRAL set and the saHMMs, a pipe-line of software tools are developed. The saHMMs are shown to be able to detect the correct family relationships at very high accuracy, and perform better than the standard tool Pfam in assigning the correct domain families to new domain sequences. We also introduce the FI-score, which is used to measure the performance of the saHMMs, in order to select the optimal model for each domain family. The saHMMs are made available for searching through the FISH server, and can be used for assigning family relationships to protein sequences.

The other approach presented in the thesis is secondary structure HMMs (ssHMMs). These HMMs are designed to use both the sequence and the predicted secondary structure of a query protein when scoring it against the model. A rigorous benchmark is used, which shows that HMMs made from multiple sequences result in better fold recognition than those based on single sequences. Adding secondary structure information to the HMMs improves the ability of fold recognition further, both when using true and predicted secondary structures for the query sequence.

Kort sammanfattning på svenska

Bioinformatik är ett område där datavetenskapliga och statistiska metoder används för att analysera och strukturera biologiska data. Ett viktigt område inom bioinformatiken försöker förutsäga vilken tredimensionell struktur och funktion ett protein har, utifrån dess aminosyrasekvens och/eller likheter med andra, redan karaktäriserade, proteiner. Det är känt att två proteiner med likande aminosyrasekvenser också har liknande tredimensionella strukturer. Att två proteiner har liknande strukturer behöver dock inte betyda att deras sekvenser är lika, vilket kan göra det svårt att hitta strukturella likheter utifrån ett proteins aminosyrasekvens.

Den här avhandlingen beskriver två metoder för att hitta likheter mellan proteiner, den ena med fokus på att bestämma vilken familj av proteindomäner, med känd 3D-struktur, en given sekvens tillhör, medan den andra försöker förutsäga ett proteins veckning, d.v.s. ge en grov bild av proteinets struktur. Båda metoderna använder s.k. dolda Markov modeller (hidden Markov models, HMMer), en statistisk metod som bland annat kan användas för att beskriva proteinfamiljer. Med hjälp en HMM kan man förutsäga om en viss proteinsekvens tillhör den familj modellen representerar. Båda metoderna använder också strukturinformation för att öka modellernas förmåga att känna igen besläktade sekvenser, men på olika sätt.

Det mesta av arbetet i avhandlingen handlar om strukturellt förankrade HMMer (structure-anchored HMMs, saHMMer). För att bygga saHMMerna används strukturbaserade sekvensöverlagringar, vilka genereras utifrån hur proteindomänerna kan läggas på varandra i rymden, snarare än utifrån vilka aminosyror som ingår i deras sekvenser. I varje proteinfamilj används bara ett särskilt, representativt urval av domäner. Dessa är valda så att då sekvenserna jämförs parvis, finns det inget par inom familjen med högre sekvensidentitet än ca 20%. Detta urval görs för att få så stor spridning som möjligt på sekvenserna inom familjen. En programvaruserie har utvecklats för att välja ut representanter för varje familj och sedan bygga saHMMer baserade på dessa. Det visar sig att saHMMerna kan hitta rätt familj till en hög andel av de testade sekvenserna, med nästan inga fel. De är också bättre än den ofta använda metoden Pfam på att hitta rätt familj till helt nya proteinsekvenser. saHMMerna finns tillgängliga genom FISH-servern, vilken alla kan använda via Internet för att hitta vilken familj ett intressant protein kan tillhöra.

Den andra metoden som presenteras i avhandlingen är sekundärstruktur-HMMer, ssHMMer, vilka är byggda från vanliga multipla sekvensöverlagringar, men också från information om vilka sekundärstrukturer proteinsekvenserna i familjen har. När en proteinsekvens jämförs med ssHMMen används en förutsägelse om sekundärstrukturen, och den beräknade sannolikheten att sekvensen tillhör familjen kommer att baseras både på sekvensen av aminosyror och på sekundärstrukturen. Vid en jämförelse visar det sig att HMMer baserade på flera sekvenser är bättre än sådana baserade på endast en sekvens, när det gäller att hitta rätt veckning för en proteinsekvens. HMMerna blir ännu bättre om man också tar hänsyn till sekundärstrukturen, både då den riktiga sekundärstrukturen används och då man använder en teoretiskt förutsagd.

Preface

The thesis consists of the five papers listed below and an introductory part. In the introductory part, some biological and bioinformatics background is given, as well as more detailed descriptions of important resources and techniques used in the thesis. The final chapters of the introductory part describe the research presented in the papers, including an overview of the contributions, summaries of the individual papers and concluding remarks.

List of papers

Paper I Jeanette Tångrot, Bo Kågström and Uwe H. Sauer. Accurate Domain Identification with Structure-Anchored Hidden Markov Models, saHMMs. *Technical report UMINF 07.12*, Department of Computing Science, Umeå University. Submitted to *Proteins: Structure, Function and Bioinformatics* (2008)

Paper II Jeanette Tångrot, Lixiao Wang, Bo Kågström and Uwe H Sauer. FISH - Family Identification of Sequence Homologues Using Structure Anchored Hidden Markov Models.[1] *Nucleic Acids Research*, 34, Web Server Issue, pp. W10-W14 (2006)

Paper III Jeanette Tångrot, Lixiao Wang, Bo Kågström and Uwe H. Sauer. Design, Construction and Use of the FISH Server.[2] PARA 2006, *Lecture Notes in Computer Science*, LNCS4699, pp. 647-657 (2007)

Paper IV Jeanette Tångrot, Bo Kågström and Uwe H. Sauer. Combinatorial Selection Improves HMM Performance. *Technical report UMINF 07.14*, Department of Computing Science, Umeå University. (2008)

Paper V Jeanette Hargbo[3] and Arne Elofsson. Hidden Markov Models That Use Predicted Secondary Structures For Fold Recognition.[4] *PROTEINS: Structure, Function and Genetics*, 36, pp. 68-76 (1999)

[1] Reprinted by permission of **Nucleic Acids Research**, Oxford University Press, 2006.

[2] Reprinted by permission of Springer-Verlag, Berlin Heidelberg, 2007.

[3] Now Jeanette Tångrot

[4] Reprinted by permission of **Proteins: Structure, Function, and Genetics** © copyright 1999 Wiley-Liss, Inc., A Wiley Company.

Conference Presentations

Results from the thesis work have been presented at several conferences and workshops, including the annual SocBiN conference Bioinformatics (2000, 2001, 2002, 2006, 2007), the Swedish Bioinformatics Workshop for PhD students and Postdocs (2000, 2001, 2005), and Umeå Bioinformatics Symposium (2002, 2003).

Other publications

Outside the thesis work, and in addition to the papers listed above, Jeanette Tångrot has co-authored the following publications:

- Magnus O. Lindberg, Jeanette Tångrot, Daniel E. Otzen, Dmitry A. Dolgikh, Alexi V. Finkelstein and Mikael Oliveberg. Folding of Circular Permutants with Decreased Contact Order: General Trend Balanced by Protein Stability. *Journal of Molecular Biology* Vol. 314, No. 4, pp. 891-900, 2001.

- Magnus Lindberg, Jeanette Tångrot and Mikael Oliveberg. Complete Change of the Protein Folding Transition State Upon Circular Permutation. *Nature Structural Biology* Vol. 9, No. 11. pp. 818-822, 2002.

Acknowledgements

As I have been working in a cross-disciplinary field, I have had the advantage of working with two main supervisors, Bo Kågström at the Department of Computing Science and Uwe Sauer at UCMP. Even though this is largely outside the scope of his research area, Bosse has always read and commented drafts handed to him, and given constructive critisism. Uwe has been of great help in writing our manuscripts. He has also provided many interesting ideas, taking my research forward, and guided me in the research process.

Arne Elofsson, Stockholm, is the one who introduced me to bioinformatics. He also supervised my Master's thesis and is the co-author of one of the papers.

For practical reasons, I have spent most of my time at the computing science department. I want to express my gratitude to all my collegues at the department for making it a nice working environment, even though I cannot mention you all by name. Special thanks to the administrative and technical staff for helping me out with major or minor problems.

However, there are still a few names I want to mention. Gunilla Wikström has acted like a bit of a mentor for me, which I appreciate. Pedher Johansson helped me finish this thesis by providing LaTeX templates and support. Åke Sandgren at the HPC2N has helped me a lot with the parallel systems.

Although I have only spent occational periods of time at the UCMP, I have always felt welcome when I am there. I want to acknowledge the former and present members of the X-ray group for providing a fiendly working environment. A special thank to Lixiao Wang, my first bioinformatics collegue, and to Marek Wilcynski, who makes the computers work. A very special thank also to Anders Karlsson, for nice chats, and to Stefan Bäckström and Fredrik Ekström for all your encouragement and support.

Part of this research was conducted using the resources of High Performance Computing Center North (HPC2N). Financial support has been jointly provided by the Department of Computing Science and Umeå Centre for Molecular Pathogenesis (UCMP).

On a more personal level I want to thank my parents, Eva-Britt and Leif, for always believing in me. And, of course, I want to acknowledge Olof, my partner in life, who encouraged me to start this work, and made it possible to finish it. I also want to mention our children, Emelie, Natanael and Disa – thank you just for being who you are!

Umeå, April 2008
Jeanette Tångrot

Contents

CHAPTER 1
Introduction

This chapter presents a brief introduction to the thesis, including a motivation for the research and a description of the research goals.

1.1 Background and Motivation

The sequencing of the human and other genomes are generating huge amounts of biological data to analyse. To fully explore the information gathered, all genes have to be located and their roles in the cells have to be determined. For all proteins to be completely characterized, we want to know their three-dimensional (3D) structures, their molecular and cellular functions, their interactions with each other and other molecules, and how they are regulated. The function of a protein is the same as its role in the organism, for example, as a building block making up the very walls of the cells or pumps that transport other molecules in and out of the cells, as a helper molecule that makes some chemical reaction go faster, or as a signal sending messages between different cells. Due to the vast amount of proteins, it is not feasible to study each molecule in each genome experimentally. Instead, the characteristics of a newly sequenced protein is usually derived by sequence and/or structure comparison to already characterized proteins. Also, to determine the 3D structure of a protein might be problematic, and the procedures used are time-consuming. If possible, it is preferable to use computational methods to guide the experimental approaches.

One commonly used tool for the comparison of protein sequences is profile hidden Markov models (HMMs, Chapter 5), which have proven to be very powerful at recognising new members of protein families (see for example [94], [111]). Often, a HMM is constructed to model a protein sequence family of interest. The HMM can then be used to search genomes for previously unannotated members of that family, or a sequence can be searched against a database of HMMs to find which model fits the sequence best, and thereby locate the family it most likely belongs to.

Proteins with sequences that are very similar, are known to also have similar structures, with some rare exceptions (mostly in the case of structural plasticity [51]). However, similarity in structure says little about how similar, or dissimilar, the corresponding sequences are. During evolution, the structure of a protein is conserved to a much higher degree than is its amino acid sequence. The reason is that some amino acids, or combinations of amino acids, can perform similar tasks in the protein, and therefore can be substituted for each other without changing the conformation of the protein. For example, at some positions in the sequence it may be sufficient to have a reasonably small amino acid for the chain to fold correctly. In protein chains, some parts are more important than others for the correct folding of the chain, and these fragments are typically buried in the core of the structure. Often, residues located on the surface of the molecule are less important.

For a protein to function correctly, the most important issues are the conformation of the chain of amino acids, and that a few crucial residues are in their correct positions.

All this has the effect that two proteins, very similar in structure and possibly performing similar tasks in the cell, might differ a lot in their amino acid sequences. This makes it difficult to identify relationships based on the sequences only.

To construct profile hidden Markov models for protein families, a multiple sequence alignment is needed for each model to build. Usually, the models are based purely on sequence alignments, which means that proteins that differ too much in sequence from the proteins the HMM was based upon, will never be found by the model, even if they are very similar in structure. Therefore, several attempts have been made to use structural information, both together with HMMs and with other methods, to be able to detect these relationships (for some examples, see Sections 7.1.7 and 7.2.3).

1.2 Research Goals and Scope

In this thesis, HMMs are used to locate which family a given sequence most likely belongs to. Given the fact that structure is more conserved than sequence during evolution, I aim at investigating how structural information can be added to HMMs, and whether this improves their ability to locate distant relationships that would otherwise be missed. By distant relationships, I in this case mean that the proteins are similar in structure, even though their amino acid sequences differ significantly. I consider two novel approaches to include structural information into the HMMs. First, our structure-anchored HMMs, saHMMs, are presented. These HMMs use structure alignments as the base for the models. Then the secondary structure HMMs, ssHMMs, are described. These use secondary structure information in addition to sequence information when scoring sequences against the HMMs.

A second goal of this thesis is to investigate whether a limited number of representative sequences, with low mutual sequence identities, are sufficient for characterising a protein domain family. Often, it is assumed that the more sequences available for building a model of a protein family, the better is a resulting model. However, as I use structure alignments it is possible to align also very divergent sequences, which reduces the need for multiple sequences in order to obtain enough information to model the family.

A third goal of the thesis is to develop a pipe-line of software tools to automatically update the collection of saHMMs as more structures are solved. This makes it possible to maintain an up-to-date collection of saHMMs, which is needed for the fourth goal – to make the saHMMs available for searching through a web server.

Of course, the result of a search for similar structures is only a first step towards the characterization of a new protein. The next step is to make a correct alignment of the two proteins, in order to be able to draw reliable conclusions about the 3-dimensional structure. However, the construction of optimal query-target alignments is outside the scope of this thesis. Also, different members within structurally related families might have very different functions, even though the structures are similar. Hence, the identification of distant relationships mainly provides clues that can be used for guiding and enabling experiments to focus on the most likely function of the protein.

1.3 Research Environment and Process

This thesis is in the area of Bioinformatics (see Chapter 3), and more precisely within fold and family recognition. In other words, the ultimate goal is to use computational methods to find the family or fold for a protein sequence. Bioinformatics is inherently cross-disciplinary, which made it natural to perform the PhD-work jointly at the Department of Computing Science and Umeå Centre for Molecular Pathogenesis (UCMP). Part of the work was also carried out at the Department of Biochemistry at Stockholm University, in a group that later was one of the founders of Stockholm Bioinformatics Centre.

The approach taken for the saHMMs in order to include structure information is most natural. For these I use multiple structure alignments as the base for regular hidden Markov models. However, in order to maximize the sequence diversity within the protein families, I construct a representative set of sequences for each domain family. This set is then used for the structure alignment the saHMM is based on. The representatives within each family are chosen so that their mutual sequence identities are very low, and so that the best possible structures are used. This quite elaborate selection procedure forms a major part of the method.

For the ssHMMs, the approach is different. In order to include secondary structure information in the HMMs, a change is made in the very architecture of the models. In this way, the secondary structure information literally affects how a sequence is scored against the model.

The two approaches are described in more detail in Chapter 7.

1.4 Organization of the Thesis

The rest of this thesis is organized in the following way. The first chapters form an introductory part, which gives some essential background to non-experts in Biology or Bioinformatics. Due to the cross-disciplinary nature of the thesis, it cannot be expected that the common reader is familiar with all the terms and techniques used in the thesis. Therefore, in Chapter 2 some basic concepts of molecular biology are described, followed by a very brief overview of the area of Bioinformatics in Chapter 3. Chapters 4–6 describe important resources and techniques used in the thesis. In Chapter 4, some important databases are described. Chapter 5 presents profile hidden Markov models and how they are used in Bioinformatics. A number of protein structure alignment methods, with emphasis on those used in the thesis, are treated in Chapter 6.

The last two chapters focus on the research presented in the thesis. Chapter 7 contains an overview of the main contributions of the thesis, together with a survey of related work and how this thesis relates to that work. Finally, Chapter 8 gives short summaries of the individual papers included in the thesis. This chapter also discusses some computational aspects and presents ideas for future work.

CHAPTER 2
Biological Background

The information carriers in all living organisms are the strains of deoxyribonucleic acid (DNA). All the genetic material, i.e. the description of every part of our cells and ourselves (in a biological sense), is stored in the DNA, which is located in the cell nucleus of every cell in the body. This information is then transferred to finally construct the proteins, the molecules that perform most of the work in the cells. The process from DNA to protein is illustrated in Figure 1, where the so called “central dogma of molecular biology” is depicted. The central dogma captures, in a very simplified way, the flow of information in the cell. DNA consists of four types of bases, commonly called A (for adenine), C (cytosine), G (guanine), and T (thymine), which are connected into strands. In Bioinformatics, sequences of these letters, corresponding to the genetic information, are investigated and compared. The DNA is stored as double helices, where two strands are twisted around each other. The two strands are connected by base-pairing, such the A binds to T and C binds to G. Whenever an A is seen on one strand, a T appears on the other, meaning that the two strands are each others complement, and that each strand contains all the information stored. When cells divide, for example during embryonic development, the DNA is replicated to produce two identical double helices (see Figure 1). During replication, each of the two strands act as a template for a new molecule. During transcription, parts of the DNA is translated into mRNA (messenger ribonucleic acid), consisting of the bases A, C, G, and U (uracil). There is a one-to-one correspondence between the DNA bases and the mRNA bases, and usually the mRNA is a simple copy of part of the DNA, with all T’s replaced by U’s. The mRNA sequences too are interesting from a biological perspective, since they represent molecules that actually perform tasks in the cell, apart from the DNA that mainly stores all information. The mRNA is then used as a template for proteins, which are produced during translation. Proteins are built from 20 kinds of amino acids, often represented by 20 letters (see Figure 2). There is a three-to-one correspondence between RNA and protein, with three RNA-bases, called a codon, representing one amino acid in the protein. There are between one and six codons coding for each amino acid, depending on the particular amino acid. The 20 genetically encoded amino acids each have different characteristics. They all have a common base (coloured red in Figure 2), where they are linked together to form the protein chain. This chain of amino acids forms the so-called backbone of the protein. Very short stretches of connected amino acids are called peptides. To the common base, each kind of amino acid has a unique side-chain connected, which

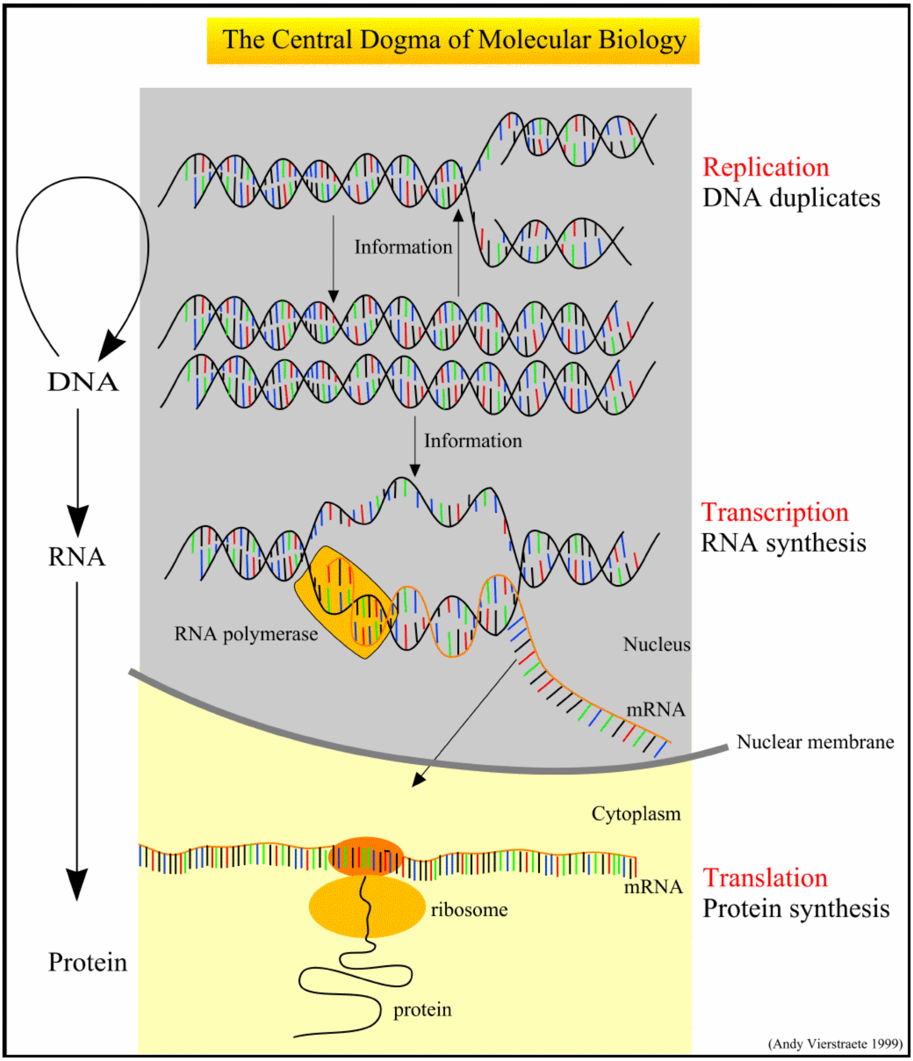


FIGURE 1: The central dogma of molecular biology. The picture shows the flow of information in the cell, from DNA to protein. The picture is kindly provided by Andy Vierstraete (http://allserv.rug.ac.be/~avierstr/tif.html).

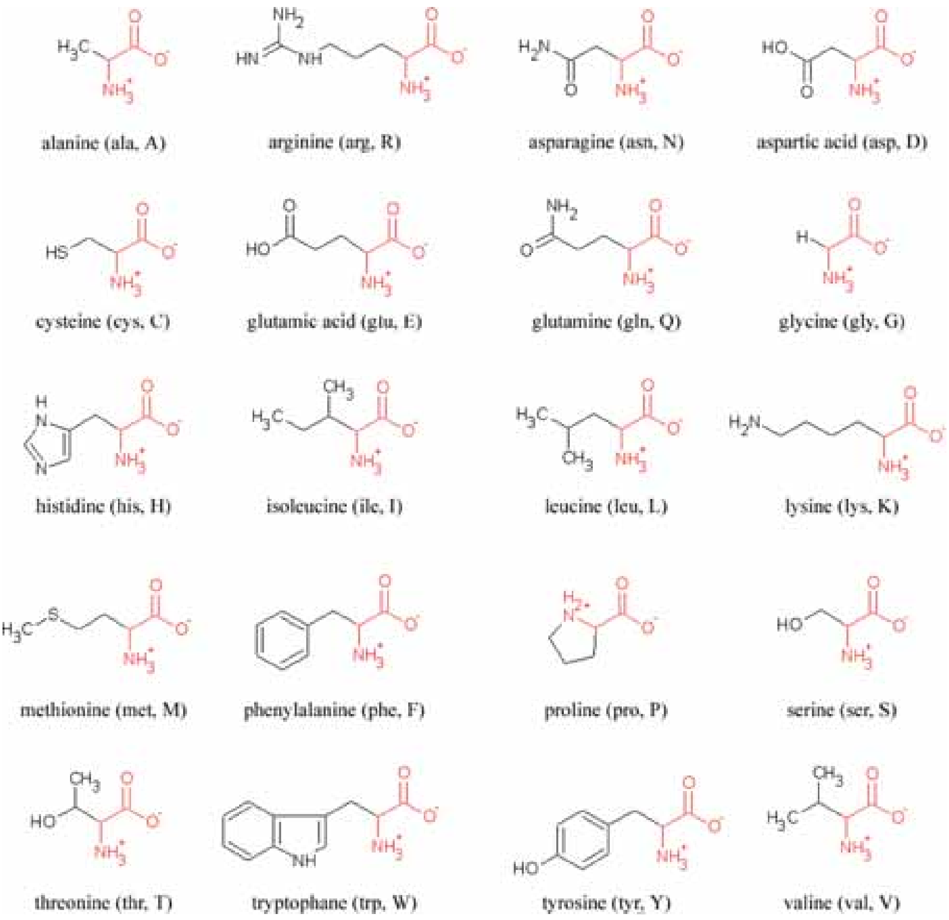
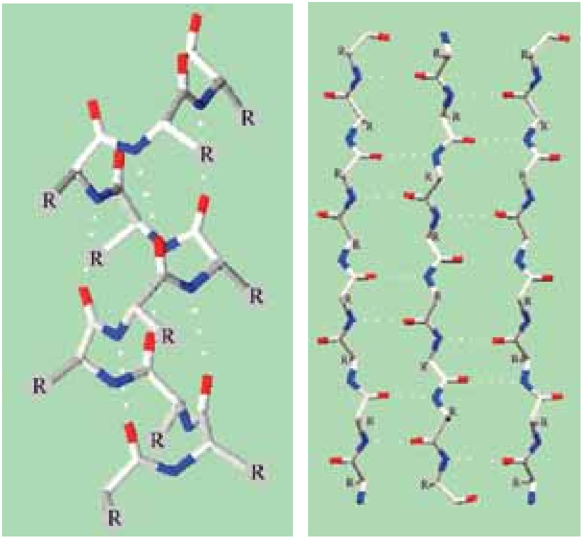


FIGURE 2: The twenty genetically encoded amino acids. The common parts of all amino acids, which are connected to form the backbone of the protein, are coloured red. In parenthesis are the three-letter and one-letter codes for the amino acids.

gives the amino acids their different properties. The 20 amino acids can be divided into groups with similar properties, for example hydrophilic/hydrophobic (water loving/water avoiding), neutral/charged or small/large.

In the context of a protein chain, the amino acids are called residues. The protein chain folds into a well-defined 3D structure, determined by the actual sequence of amino acids. It is the chemical properties of the amino acids that determine the shape of the protein molecule.

The overall structure of a protein is defined at different levels. The pure sequence of amino acids, represented by a sequence of letters, is called the primary structure of



(a) (b)

FIGURE 3: The two most common types of secondary structures. Only the main chain (backbone) is shown, the side chains are indicated by filled grey circles. (a) An alpha helix. (b) A beta sheet consisting of three anti-parallel strands.

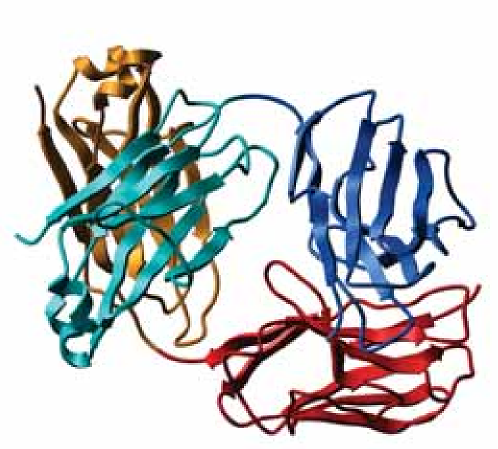


FIGURE 4: A dimer (two aggregated molecules) of the immunoglobulin light chain. The chain folds into two separate domains, coloured blue and cyan, respectively, that mainly consist of beta strands. The red and gold domains are the other chain, located in a different direction.

the protein, or simply the sequence. Parts of the chain fold locally to form so called secondary structure elements. There are two major kinds of secondary structures: alpha helices, and beta strands that form beta sheets (see Figure 3). The alpha helices are often shown as extended spirals in pictures of proteins, while the beta strands are shown as arrows. The secondary structures can group to form super-secondary structures or motifs. Two examples are beta hair-pins, consisting of two anti-parallel beta strands and a short loop connecting them, and the beta-alpha-beta motifs, consisting of two parallel beta strands with an alpha helix in between them. The super-secondary structures are packed together to form domains, that in turn pack to form the tertiary structure of the protein. In general, proteins fold so that amino acids which do not like water are located in the inside of the protein, and form the so called hydrophobic core, while residues that easily interact with water are found on the outside of the protein. Exceptions are, for example, residues important for the interaction with other molecules. A protein domain is a region of the protein that has its own hydrophobic core, and that interact relatively little with the rest of the protein. Domains can often fold independently of other parts of the protein. In Figure 4, an example of a protein with two domains is illustrated.

Sometimes, several protein chains pack together to form complexes, that build up the so called quaternary structure. Very large collections of proteins, possibly packed together with RNA or DNA, are called macromolecular assemblies. One example of such an assembly is the ribosome, which produces new proteins from an mRNA template.

The particular packing and orientation of the secondary structure elements, and the location of residues important for the structure and/or function of a protein, is called the fold of the protein. In Figure 5, an example of the different levels of protein folding is shown. Protein structures can be displayed in a number of ways. In Figure 6, this is illustrated by five different representations of the same protein.

The amino acid sequence of a protein can easily be determined from its corresponding DNA, and the sequence of DNA is routinely determined experimentally. The 3D structure of a protein can be determined by experimental methods such as X-ray crystallography, Nuclear Magnetic Resonance (NMR) and electron microscopy (EM) reconstruction. As of the 1st of April 2008, 49974 structures are known and deposited in the Protein Data Bank (PDB, see Section 4.2), and the number is increasing exponentially.

Currently, it is not possible to determine the structure from the sequence only, although there has been some success in folding small proteins (see Section 3.7). The most common way to find the structure of a new protein is to compare it to proteins with known structures and predict a conformation based on sequence similarity. From a sequence comparison of two proteins, similarities can be found between proteins from different organisms or between two proteins in the same organism. These similarities often indicate a common evolutionary origin, in other words, the proteins are homologous. Given sequence homology, it is possible to deduce similarity in structure and perhaps even in function. Homology between two proteins from the same organism means that a gene has been duplicated, and that during time, differences have been introduced by mutations and reorganisations. During time, the proteins

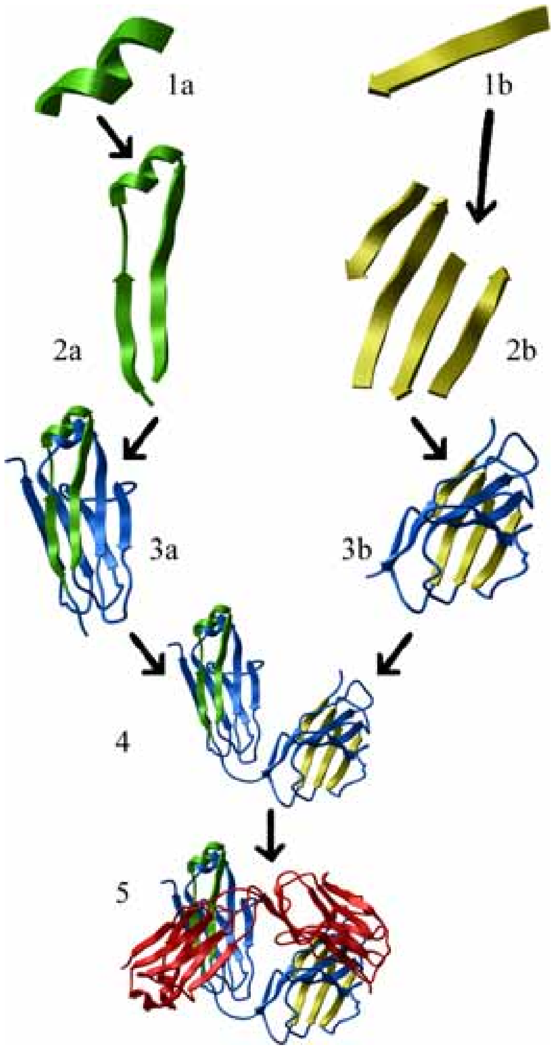


FIGURE 5: The different levels of protein folding. At the top of the figure, two secondary structures are shown, one alpha helix (1a) and one beta strand (1b). In (2a), the helix is packed with two beta strands to form a beta-alpha-beta motif, that in turn joins more strands to form a complete protein domain (3a). The strand in (1b) is packed with more strands, and together they form a beta sheet (2b). In (3b) the sheet together with another sheet form a second domain. The two domains together form the complete folded protein (4), that interact with another identical protein chain to form a dimer (5).

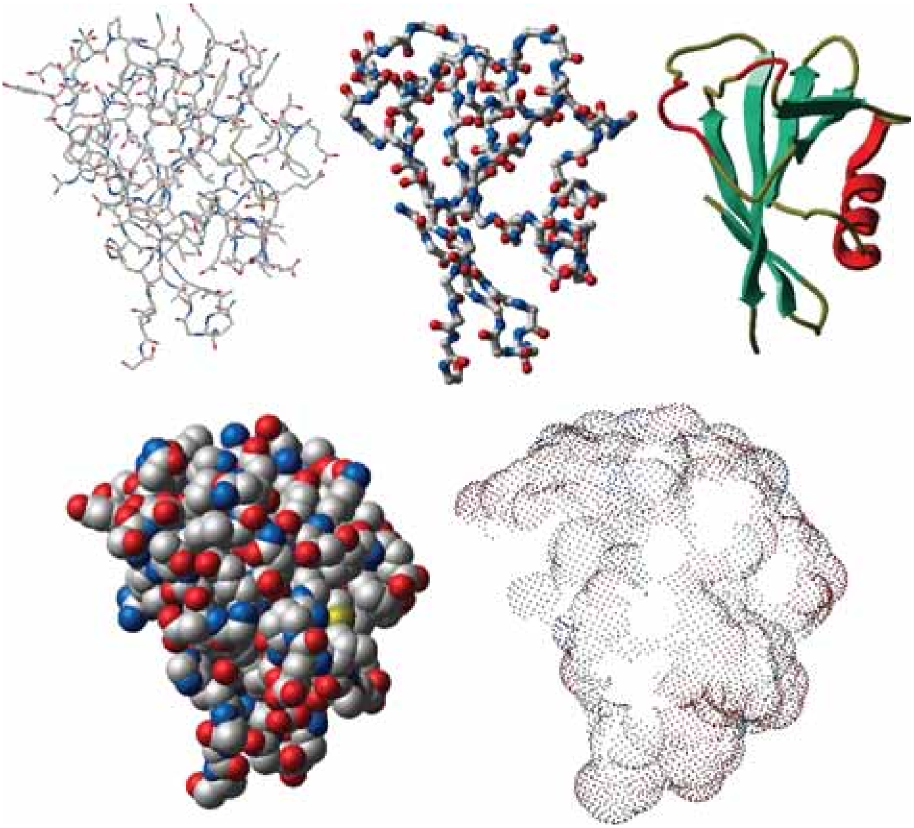


FIGURE 6: Five different representations of the same protein chain (anti-platelet protein from leech, PDB code 1i8n). At the top left, the molecule is shown as lines between atoms, which are represented by the colour of the line. At the middle of the top row, only the backbone of the molecule is shown, now with sticks instead of lines and the atoms represented by balls. At the top right, the backbone of the protein is represented with ribbons, where helices are shown as extended spirals and beta strands as arrows. The bottom left of the figure shows the protein using a space fill representation, i.e., each atom in the molecule is represented by a sphere, where the radius corresponds to the Van der Waals distance (the closest any other atom can be without contact). The bottom right shows the area of the protein that is accessible to water molecules, and is perhaps the most true picture of the protein from any other molecules point of view.

have developed in divergent directions to perform two different, but probably similar, functions. This is the main strategy for the evolution of new genes and more complex organisms.

However, not all similarities are indicators of common ancestry. Some similarity may also be introduced by convergent evolution to a similar 3D structure, resulting in analogous proteins. Analogy appears when two proteins performing the same task in different organisms have evolved similar properties, without having a common ancestor, simply because those properties make the proteins more suitable for the task. A clear distinction between homologous and analogous proteins is difficult to obtain because functional relatedness is hard to prove. A distinction can in some cases be made based on similarity in side-chain directions [79].

There exist extensive resources for retrieval and comparison of proteins on the Internet. For example, there are databases containing protein and DNA sequences, including the complete genomes of several organisms. The PDB contains all currently known protein structures. Protein structures are also classified in a number of ways, see Chapter 4. Much information and many databases are available at the National Center for Biotechnology Information, NCBI, through the Entrez search engine[1]. EMBL-EBI (European Bioinformatics Institute) offers the SRS (Sequence Retrieval System)[2], which provides access to hundreds of databases and applications. The Biology Workbench[3] at San Diego Supercomputer Center (SDSC) offers a similar environment for browsing databases and applying analysis and modelling tools to retrieved data. These are only a few of the available resources for data retrieval and analysis.

[1] http://www.ncbi.nlm.nih.gov/Entrez/

[2] http://srs.ebi.ac.uk/

[3] http://workbench.sdsc.edu/

CHAPTER 3

Bioinformatics

Bioinformatics is a very broad area of research, with that in common that it uses computational methods to analyse and structure biological data, and from this make theoretical predictions about biological processes. Much of the research in Bioinformatics is multidisciplinary, and includes computing science, statistics, and structural and molecular biology.

It is unclear when and where the term "Bioinformatics" first appeared. However, computers were used in the field of biology long before the term appeared, but then under names as Computational Biology, Biocomputing or Biostatistics. Already in the 1960's, the first computer programs were constructed to analyze protein sequences [64]. The foundation of Bioinformatics was laid along with the construction of biological sequence databases. The first bioinformatics "database" was gathered in 1965, when Dayhoff et al. compiled the first *Atlas of Protein Sequence and Structure* [31], a book containing all sequence data available at that time, including sequence alignments. As the number of and the sizes of databases grew, new tools for searching these became available, see for example Sections 3.6 and 6.1.1.

Today, there are several branches of Bioinformatics, some of which are briefly described below. The list of branches presented here is most likely not complete, and some people would probably claim that some important aspect is missed or that some things listed below not at all belong to Bioinformatics in its true sense. This illustrates the fact that there still is no consensus on the definition of Bioinformatics. However, most people would probably agree on the description given above, even though many choose to narrow it further.

3.1 Genomics and Phylogenetic Trees

An important part of Bioinformatics is the analysis and comparison of genes and genomes. For example, in order to show the evolutionary relationships between different organisms, phylogenetic trees can be constructed based on their genomes (e.g., [96]). These kinds of trees can also be constructed for individual genes, showing how they have evolved and how they are related. Stochastic grammars have been used to determine evolutionary relationships between biological sequences, and to find a common ancestor (e.g., [70]).

As mentioned in Chapter 2, in the DNA each amino acid is represented by one or more combinations of three bases, so-called codons. However, all codons coding for a specific amino acid are not equally abundant. The patterns of codon usage differ between organisms and between genes in the same organisms, and can be studied to find similarities and differences. The amounts of the bases C and G in genes also differ between organisms and genes, and give information about the history of a gene, the level of expression, and about evolution.

Having the human and other genomes sequenced, it is important to locate the parts of this huge amount of DNA that are genes that code for proteins. The major parts of the human and other large genomes do not code for any gene, and the functions of these regions remain unclear. Methods have been and are developed to find the start and stop of the protein coding sequence, and other patterns characteristic for genes (e.g., [135], [100]). It is also interesting to locate regulatory regions, which for example govern when and how often a gene is transcribed (e.g., [113]), or to find possible cleavage sites, where the final protein is cleaved to remove for example signal sequences after they have been used (e.g., [147]). In comparative genomics, the complete genomes of organisms are compared, in order to find for example conserved protein coding genes or regulatory elements. Comparing genomes can also help to locate and understand the nature of functional DNA that do not code for protein or RNA (e.g., [116]).

During and after translation, some proteins are transported out of the cell or to the mitochondria, the energy fabrics of the cell. These proteins have signal peptides, making it possible to transport them to the correct location. Neural networks have been used to, based on the sequence, locate this signal and determine where a protein should be located [41]. Some proteins are inserted into the membrane surrounding the cell. Hidden Markov models (HMMs, Chapter 5) are used to determine whether a protein is a membrane protein or not, and which parts of the protein are inside the cell, which parts are inserted into the membrane, and which are outside of the cell [136].

Statistics in different forms can be used to study genetic diseases. Healthy and sick people are compared on a genetic level, and genetic properties are determined (e.g., [42]).

3.2 Study of RNA

As RNA has a very important role in the cell, both as an information carrier between DNA and final protein, and as an important actor on its own, the theoretical study of RNA sequences has grown during the recent years. Much effort is put into the prediction of RNA secondary and tertiary structures, which are formed due to base pairing and other interactions within the RNA chain. For example, stochastic context-free grammars are used to make these predictions [59]. Fold libraries are also developed, containing typical fold fragments, and algorithms are developed to design molecules with a given secondary structure, in order to be able to produce self-assembling RNA structures for certain purposes (e.g., [8]). Short fragments of RNA, containing only about 20 nucleotides, form so called microRNAs. These have proved to be important in the regulation of protein expression. The microRNA binds the mRNA of its target protein, and down-regulates the expression with the help of protein complexes. Some research in Bioinformatics is aimed at the identification of microRNA targets (e.g., [125]). Other research that could give insight into the gene regulation and processing is to be able to locate the sequence regions recognised by the RNA splicing machinery (e.g., [155]).

3.3 Study of Protein Function

The study of protein function on a larger scale, both experimentally and theoretically, has grown increasingly important. To study chemical modifications, the binding of cofactors, interactions between proteins, etc., is called proteomics (compare to genomics, the study of the genome). Bioinformatics is needed when it comes to the analysis of experimental data. For example, patterns from mass spectrometry experiments can be compared to databases in order to find what the sample contains (e.g., [9]). Approaches are also made to predict, for example, which parts of a protein that interact with other molecules (e.g., [84]).

A field that has grown into a research area of its own is the use of microarrays in functional genomics. Microarrays are small arrays, where several different DNA strands are attached. These are used to study gene expression in different types of cells and under different conditions. When a gene is expressed, mRNA is produced with DNA as a template, and the RNA in turn is used as a template to build protein molecules, see also Chapter 2. Not all genes are expressed in all cells, and a single gene is only expressed in a given cell when it is needed. The particular mRNA molecules present in a cell at a certain time can be captured using the DNA arrays, thus capturing information about which genes are currently expressed in the cell. The mRNA has the ability to base pair with the DNA, due to the chemical similarity between DNA and RNA. It is this ability that is used in the technique. The RNA molecules bound can be detected and the strength of the signal is a measure of the amount of RNA in the cells.

Bioinformatics is used when processing and analysing the data. Often, gene expression is studied under different conditions, for example the expression of genes in starving cells can be compared to that in cells under normal conditions. In the experiments, what is interesting is the difference in expression, not the actual expression levels. Scientists are looking for genes that are up-regulated or down-regulated (i.e., expressed more or less than under normal conditions), in order to find patterns in the expression of different genes. In this way, it is possible to, for example, locate genes that belong to a common pathway and cooperate to perform a certain task in the cell, since these proteins should be expressed in a similar way. The very amount of data makes it a difficult task to find patterns between genes.

To find patterns in gene expression, several different approaches have been used, such as graph theory [149], self-organising maps (SOM) [146], the singular value decomposition (SVD) [72], and fuzzy clustering based on neighbourhood approximation [49].

3.4 Biological Interactions and Networks

In order to understand how biological processes work, methods have been developed for predicting which proteins interact in a cell, e.g., [133]. Based on available information on protein function and interactions, metabolic networks can be constructed, showing which proteins are parts of which pathways, and how the pathways are connected. Work is also done to predict metabolic networks, and to make models of signal transduction and other important processes in the cell (e.g., [30], [104], [7]). The modelling of metabolic networks, how biological molecules interact, and even of complete cells, falls under the label *Systems Biology*, and is nowadays a research area of its own.

3.5 Databases and Information Searches

Several databases containing biological data are available via the Internet, some of which are discussed in Chapter 4. These databases might store raw data as well as annotated, or literature references. Also, new databases are created by developing new algorithms to, for example, cluster proteins into structural or functional families (e.g., [110], [55]). Some researchers focus on annotating the raw data and constructing cross-links to create new, value added databases (e.g., [18], [23]). Research is also done to combine several databases and/or to index web pages, in order to make it possible to find all data relevant from just one or a few searches, and to quickly find other, related information (e.g., [71]).

3.6 Biological Sequence Analysis

A classical branch of Bioinformatics is the analysis of biological sequences, such as DNA and protein sequences. Comparisons of sequences most often involve sequence alignments, where one sequence is matched as good as possible to another sequence. From a sequence alignment it is possible to determine characteristics common to the two sequences, such as conserved amino acids or conserved properties such as size or charge of the aligned residues. More information of a whole protein family can be gained from multiple sequence alignments, where many sequences are aligned simultaneously. Several methods have been developed for the alignment of multiple sequences, see for example [105], [39] for reviews. Some examples of approaches used are dynamic programming (see Section 6.1.1) in different forms [27], the divide and conquer strategy, where the alignment is divided into small manageable parts [139], genetic algorithms, which use the analogy of genetic mutations and recombination to find the best alignment [106], and to progressively align the sequences, i.e. to add one at a time following some schema (e.g., [45], [107]).

The key to making alignments are the use of scoring matrices to determine similarity between amino acids. Two of the most comonly used series of scoring matrices are the PAM [32] and BLOSUM [67] matrices.

3.7 Prediction of Protein Structure and Function

The ultimate goal of much work in Bioinformatics is to be able to predict the structure, and perhaps even the function, of a protein, based on its amino acid sequence. This is needed since, in general, it is very expensive, difficult, and time consuming to determine the structure of a protein experimentally. For certain proteins, it is even impossible using current techniques. The function of a protein is also hard to determine, especially if one has no clues about what the role of the protein could be. If one can predict the structure of a protein, the structure gives clues about possible functions of the molecule, and together with other techniques it might be possible to predict the function of the protein. This prediction can in turn make a base for constructing tailored experiments to determine the true function of the protein. The scope of this thesis is in the area of structure prediction, or rather fold and family recognition, see below. To predict function based on sequence and structure is still a largely unexplored area, much due to the need for good structure prediction methods to base the work on.

There are many approaches to structure prediction. The most direct, and perhaps most difficult, approach is to make *ab initio* prediction. This means to try to calculate the fold of a protein based on its sequence and knowledge about the amino acids' chemical properties, using different energy functions (e.g., [90], [152]). In a way, this is equivalent to simulating in the computer how the folding of the protein sequence is done in the cell. Another *ab initio* approach is the Frankenstein monster model method, which combines small parts from many different known structures, finding the combination of structural parts that seems to fit the sequence best [131], [80]. Here, steric and chemical properties of the amino acids making up the sequence are considered to find the best combination of structural parts. If one succeeds in constructing an efficient *ab initio* method, this yields important insight into the natural folding process and which parameters or properties are important for defining the particular fold a certain protein adopts.

Other methods for structure prediction use already known structures as templates to deduce the fold of a given sequence. A common method is threading, where the sequence is “threaded” through a number of structures, in order to find the one that fits the steric and chemical properties of the chain the best (e.g., [85], [127], [141]). In homology modelling, the sequence is fitted to the sequence of a protein with known structure, and a possible structure is determined based both on the fit of the sequences and on the known structure (see for example [54] for an overview). For this procedure to be possible, a way to locate the closest homologue with known structure is needed, and it is necessary to be able to fit the sequences in a biologically sensible way. This area, often called fold recognition, is one of the classic fields within Bioinformatics. Much work in this area is based on sequence alignments in one way or the other.

Alignment methods have been developed, that are able to find in a database the sequence that fits a given query sequence best, i.e., that gives the best alignment between the two (e.g., [112], [5]). More sophisticated methods have also been developed, that try to model the sequences of a whole protein family, to be able to assign the query to a group of related proteins even if the relationships are distant. Two examples of such models are profiles and hidden Markov models (see Chapter 5). To automatically cluster and classify proteins into families, methods based on for example graphs [81] and Markov clustering [43] have been used.

Every second year, the Critical Assessment of Structure Prediction (CASP[1]) takes place, where methods for structure prediction are tested on real targets. Predictors are invited to submit predictions on sequences whose structures are due to be released, but that are not known to the predictors at the time of prediction. This blind test makes it possible to make a fair comparison between the best methods available today. The results from the latest assessments show that expert evaluation and intervention in the prediction procedure still is superior to purely automatic methods, but that the gap is decreasing. However, human predictors often start with models generated from some automatic server, why the quality of server predictions highly affects the quality of predictions in general. By comparing the results from all experiments since the start in 1994, it is clear that steady improvement in performance is made [83]. The CAFASP (Critical Assessment of Fully Automated Structure Prediction)[2] experiment runs in parallel with CASP, and is designed to evaluate the performance of fully automated services on the blind targets provided in CASP. Since 1999, LiveBench[3] performs automatic evaluation of publicly available automatic servers on a more regular basis. New targets, obtained from the PDB, are submitted to the servers on a weekly basis, and the results are evaluated. To participate, the servers must delay the updating of their structural libraries by one week. In general, it seems like the combination of results from several different methods give the best results.

The ssHMMs presented in Paper V of this thesis were used in combination with other methods in CASP3, with some success (see Section 7.2.2). The saHMMs have not been used in any of these assessments. In order to participate, the methods must generate an appropriate alignment to a template sequence with known structure, which is not within the scope of this thesis.

An area related to structure prediction is the prediction of interactions and interaction sites in proteins, including binding sites (e.g., [19], [26]). Another related area is the prediction of *unstructured* regions (e.g., [28], [50], [65]). The existence of intrinsically disordered regions in some proteins plays an essential role in their functions, and may also prevent experimental structure determination of the remaining regions of the proteins. The prediction of unstructured regions is one of the sections in the CASP experiments.

[1] http://predictioncenter.gc.ucdavis.edu/

[2] http://www.cs.bgu.ac.il/ dfischer/CAFASP5/index.html

[3] http://bioinfo.pl/meta/livebench.pl

CHAPTER 4

Databases and Protein Classifications

In this chapter, some of the most common databases relevant for this work are described, with focus on databases containing classifications of proteins. A longer listing of useful biological databases can be found in Baxevanis [15], including a short description of each database. A more detailed description of each database is provided through the Nucleic Acids Research web site[1]. The annual database issue of the Nucleic Acids Research gives an up to date overview of the most important databases, as well as completely new ones.

4.1 Sequence Databases

There are three important databases storing genetic information, i.e. nucleotide databases containing DNA and RNA sequences. GenBank[2] is the NIH (National Institute of Health, USA) genetic sequence database. GenBank is an annotated collection of all publicly available DNA sequences and is maintained at the National Center for Biotechnology Information (NCBI). The EMBL Nucleotide Sequence Database[3] (sometimes called EMBL-Bank) is the main resource of nucleotide sequences in Europe, and is maintained at the European Bioinformatics Institute (EBI), which is a part of the European Molecular Biology Laboratory (EMBL). The third collection of nucleotide sequences can be found in the DNA database of Japan (DDBJ)[4]. The three databases cooperate, and exchange new and updated database records on a daily basis. Each database entry is given a unique accession number, making it possible to refer to a specific gene sequence. The main sources of DNA, and also RNA sequences, are submissions from individual researchers, genome sequencing projects and patent applications.

One of the main sources of protein sequence information has historically been the Swiss-Prot Protein Knowledgebase (SWISS-PROT)[5]. SWISS-PROT is a curated

[1] http://nar.oxfordjournals.org/

[2] http://www.ncbi.nlm.nih.gov/Genbank/index.html

[3] http://www.ebi.ac.uk/embl/index.html

[4] http://www.ddbj.nig.ac.jp/Welcome-e.html

[5] http://www.ebi.ac.uk/swissprot/

protein sequence database, which aims at providing a high level of annotation, as little redundancy as possible, and a high level of integration with other databases. SWISS-PROT is maintained by the Swiss Institute for Bioinformatics (SIB) together with the EBI. The SWISS-PROT release of March 2008 contains 359942 sequence entries.

The TrEMBL database (Translated EMBL)[6] contains the translations of all coding sequences present in the EMBL Nucleotide Sequence Database. In other words, the DNA sequences in EMBL-Bank that code for a protein are translated into the corresponding protein sequence. Only sequences which are not yet integrated into SWISS-PROT are stored in TrEMBL. A subset of TrEMBL, called SP-TrEMBL, contains sequences that eventually will be incorporated into SWISS-PROT. PIR (Protein Information Resource)[7] produces the Protein Sequence Database (PSD), which contains protein sequences that are functionally annotated.

To collect the information in these three databases, the United Protein Databases (UniProt)[8] project was formed in 2002 by joining the forces of the SWISS-PROT, TrEMBL and PIR protein database activities.

The UniProt Knowledgebase combine the three collections of data in a single protein database, divided into two sections; UniProtKB/Swiss-Prot and UniProtKB/-TrEMBL. UniProt contais two more components; the UniRef databases providing clustered sets of sequences, and UniParc providing non-redundant protein sequences with links to all sources and versions of these sequences.

4.2 PDB

The RSCB Protein Data Bank[9] (PDB, [17]) is a collection of structural data of proteins and other biological macromolecules. In the PDB, all protein structures are stored in an organised way, and all entries are assigned a unique PDB accession code. The data in the individual structure files is ordered according to the PDB format, making it easy to parse and extract specific information.

The world wide PDB, wwPDB[10] [16], was established in 2003 as a cooperation between the RCSB PDB, described above, MSD-EBI[11] (the Macromolecular Structure Database at the European Bioinformatics Institute), and PDBj[12] (PDB Japan). In 2006, the BMRB[13] (Biological Magnetic Resonance Data Bank, harbouring NMR spectroscopic data) joined the wwPDB. The goal of the wwPDB is to maintain a single, world-wide, publicly available archive of macromolecular structural data with uniform content and format of the data files. However, the individual wwPDB sites present the data using their own views and tools.

[6] http://www.ebi.ac.uk/trembl/index.html

[7] http://pir.georgetown.edu/

[8] http://www.uniprot.org

[9] http://www.rcsb.org/pdb/

[10] http://www.wwpdb.org/

[11] http://www.ebi.ac.uk/msd/

[12] http://www.pdbj.org/

[13] http://www.bmrb.wisc.edu/

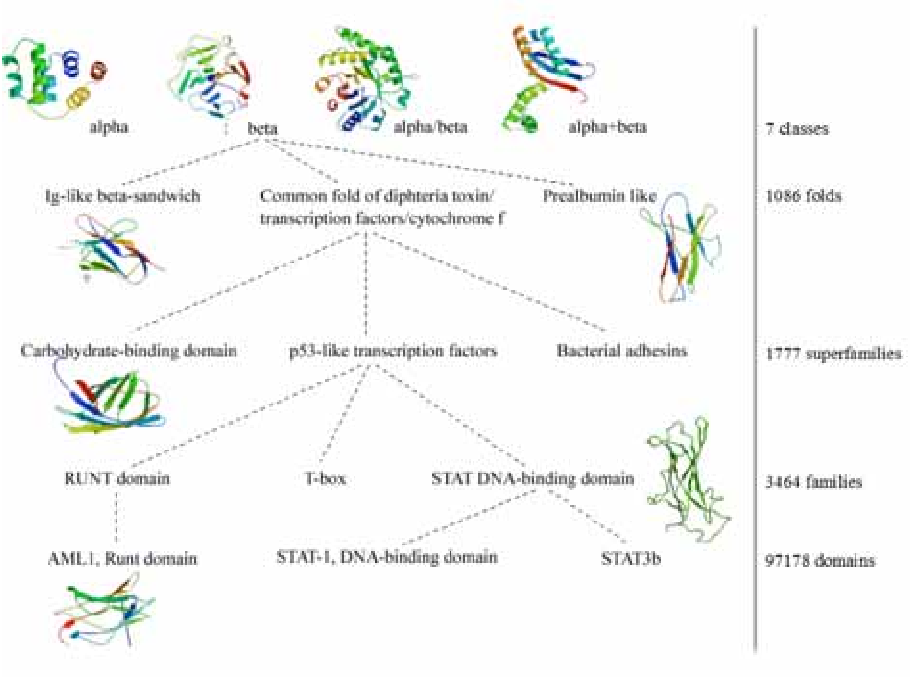


FIGURE 7: A schematic picture of the SCOP classification, together with the number of entries at each level, as of version 1.73 (to the right). Pictures of protein structures are obtained from the PDB.

4.3 SCOP

In the Structural Classification of Proteins (SCOP, [101]), all proteins with known structures are divided into groups based on different levels of similarity. The classification is made at the domain level (see Chapter 2), meaning that different parts of a single protein may appear in multiple families in the classification, and even in different classes. The aim is to capture evolutionary relationships between protein domains. In SCOP, a domain is defined as an evolutionary unit, either observed in isolation in nature or together with different domains in different multidomain proteins.

In Figure 7, a schematic picture of the SCOP classification is shown. The lowest level of the classification contain the actual protein domains (at the bottom of Figure 7), sorted by species. Protein domains that are very similar in structure, and with experimentally determined similarities in function, are put into the same family, the next higher level. Especially, domains having a sequence identity of 30% or more are assigned to the same family. Families of proteins with similar structures, but uncertain similarity in function, are part of the same superfamily. One level higher is the fold, where superfamilies with roughly the same arrangement of secondary structures and the same topology are grouped together. The highest level in the SCOP hierarchy is the class level, where folds consisting of the same kinds of secondary structure elements are grouped into the same class. Apart from the four main classes shown in Figure 7 – all alpha-helices, all beta-sheets, and the two kinds of mixtures of alpha and beta – there exist three more true classes; multidomain proteins, membrane proteins, and small proteins. There are also four additional classes containing peptides, low resolution structures, and other groups of proteins that could not be included in the actual classification. These are not considered as true classes.

SCOP includes all proteins in the PDB until the date they started working on the current release of SCOP, and most of the proteins whose structures have been published but not included in the PDB. The database is curated, meaning that the classification of the protein domains is determined manually by a group of experts. The investigation is done using both visual inspection and comparison of structures. Automatic tools are used to speed up the classifications. Sequence comparison can be used to group domains with high sequence similarity to the same family, while structural alignments are used to suggest a fold for a protein of interest, even though manual inspection must be used to verify the result and choose an appropriate superfamily and family for the domain. The manual check of the classification is the reason why the SCOP database often is used as the gold standard for grouping of similar protein domains.

The current version of SCOP (version 1.73) was released in November 2007 and contains 97178 domains divided into 3464 families.

4.3.1 ASTRAL

The ASTRAL Compendium [25] is a collection of sequences for the domains classified in SCOP, derived from their respective PDB files. The sequences can be retrieved filtered according to different criteria such as sequence identity or BLAST [5] E-value. The compendium also provides the extracted coordinates of single SCOP domains, as well as predicted domains from PDB structures not yet classified in SCOP.

4.4 CATH

In CATH [110], protein domain structures are classified into five levels: protein class (C), architecture (A), topology (T), homologous superfamily (H), and sequence family (S). The classification is, as far as possible using current techniques, done automatically, with the goal of completely automatic classification in the future. The database classifies single structural domains, so multidomain proteins are divided into separate domains using an automatic procedure. In those cases where the procedure fails, the domain borders are determined manually.

The class (C-level) describes the content of α helices and β sheets in the structures. There are four classes: mainly α, mainly β, $\alpha - \beta$, and a special class grouping all domains with low secondary structure content. The class of a domain is determined by an automatic procedure, which examines the secondary structure composition of one representative for each sequence family. The architecture (A-level) describes the general arrangement of secondary structures, and is determined manually, while the topology (T-level) further groups the structural domains based on the overall fold. The fold describes the number and arrangement of secondary structures, and the connectivity between them. The homologous superfamilies (H-level) group domains by high structural similarity and similar functions. The T- and H-levels are determined by structural comparison of representative proteins using the SSAP program [142], with different cut-offs for the two levels. For a protein to belong to a certain homologous superfamily, it must also have a common function to the other members in the superfamily. Function is determined from SWISS-PROT, the PDB file or literature.

At the lowest level (S-level, sequence family), protein domains with high sequence similarity (more than 35% identical) are clustered. These domains are assumed to have very similar structures and functions. The sequence similarity is determined by pairwise comparisons using the Needleman-Wunsch algorithm [102], and the sequences are clustered into families by single linkage cluster analysis.

From the PDB [17], only NMR structures and crystal structures with 3.0Å resolution or better are selected. The domains are sorted so that low resolution, native X-ray structures are first and mutant NMR-structures become last. The domain listed highest is chosen as representative for the sequence family in the classification.

In addition to the actual classification, the database contains derived data such as structural alignments and family templates. Also, for each structure in CATH, a number of graphical representations are provided, together with a report containing information from the PDB file, domain boundary data and functional data.

CHAPTER 5

Hidden Markov Models in Bioinformatics

5.1 Hidden Markov Models in Bioinformatics

Hidden Markov models have been used for a number of purposes within the area of Bioinformatics. Perhaps the most common use is for protein fold or family recognition, where profile HMMs (see Section 5.3) are used extensively. In addition, a type of hidden Markov model based on structural features, instead of the sequence features used by profile HMMs, has been described for these purposes [4]. The use of profile HMMs have also been developed by the invention of profile-profile comparison techniques (e.g., [124]). However, HMMs have also been used to locate transmembrane regions in proteins [82], to find signal peptides [103], to model interaction sites [47], and for many other applications.

In this chapter, the profile hidden Markov models, which are used throughout this work, are described in detail, with focus on the HMMER[1] [37] implementation. For a more complete description of hidden Markov models and their use in molecular biology, see for example [11] or [36].

5.2 Multiple Sequence Alignments and Profiles

One often used technique to identify common sequence characteristics within a protein family is to construct a multiple sequence alignment, see Figure 8 for an example. In a multiple sequence alignment, several sequences are fitted on top of each other, such that the amino acids placed above each other in the alignment are as similar as possible. To make the sequences fit better, gap symbols, shown as dashes in Figure 8, might be inserted at some positions in one or more of the sequences. These are needed when a sequence contains insertions or deletions with respect to the consensus of all aligned sequences. From the aligned sequences, a consensus sequence can be derived, which aims at representing a "typical" sequence of the group.

A more elaborate way to describe a protein family is to construct a position-specific scoring matrix, or profile, from the multiple sequence alignment [60]. In a

[1] http://hmmer.janelia.org/

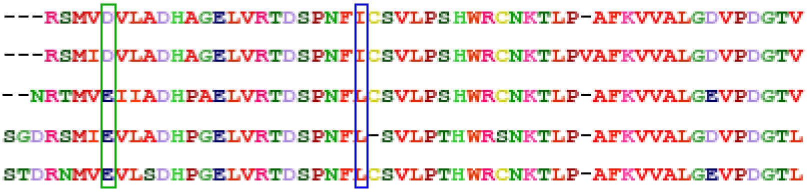

---RSMVDVLADHAGELVRTDSPNFICSVLPSHWRCNKTLP-AFKVVALGDVPDGTV
---RSMIDVLADHAGELVRTDSPNFICSVLPSHWRCNKTLPVAFKVVALGDVPDGTV
--NRTMVEIIADHPAELVRTDSPNFLCSVLPSHWRCNKTLP-AFKVVALGEVPDGTV
SGDRSMIEVLADHPGELVRTDSPNFL-SVLPTHWRSNKTLP-AFKVVALGDVPDGTL
STDRNMVEVLSDHPGELVRTDSPNFLCSVLPTHWRCNKTLP-AFKVVALGEVPDGTL

FIGURE 8: An example of a multiple sequence alignment. Here, five fictitious peptide sequences are shown aligned to each other. The amino acids are coloured according to their chemical properties.

profile, each position in the alignment is associated with a score for matching each of the 20 amino acids to that position, or to have insertions/deletions. For example, in column eight, marked with a green box, in the alignment seen in Figure 8, the amino acids ‘D’ (aspartic acid) and ‘E’ (glutamic acid) would have high scores, while all other amino acids would have lower, or even negative, scores.

Using the position specific scores, it is possible to score a collection of sequences against the matrix in order to find high scoring sequences that most likely are additional family members.

PSI-BLAST

PSI-BLAST (Position-Specific Iterated BLAST) [6] is an extension of BLAST (Basic Local Alignment Search Tool) [5], a commonly used method for pairwise comparison of biological sequences and for finding relationships between, for example, two protein sequences. PSI-BLAST uses a position-specific scoring matrix similar to the profiles described above, but without the column for gap penalties. The position specific scoring matrix is automatically constructed from the alignments resulting from a BLAST run. The BLAST search is then repeated using the matrix instead of the query sequence, and the procedure is iterated using the new results acquired in each run. PSI-BLAST has proven to be sensitive to weak sequence similarities [89].

5.3 Profile Hidden Markov Models

A similar approach as the profiles, but with a formal probabilistic basis and a consistent theory behind insertion and gap scores, is to use profile hidden Markov models to model protein families (e.g., [38]).

In a Markov model, a probability is assigned to symbols in a sequence, based on which symbols are seen in the preceding positions in the sequence. In a general case, a ‘sequence’ can be any sequence of symbols or events. The order of the Markov model is the number of preceding symbols the probabilities are based on. A simple first order Markov model of a *protein* sequence would be a set of arrays a_k, one for each amino acid, with the probabilities $P(i \mid k)$ of seeing amino acid i after amino acid k in the sequence. The probability that an observed protein sequence belongs to

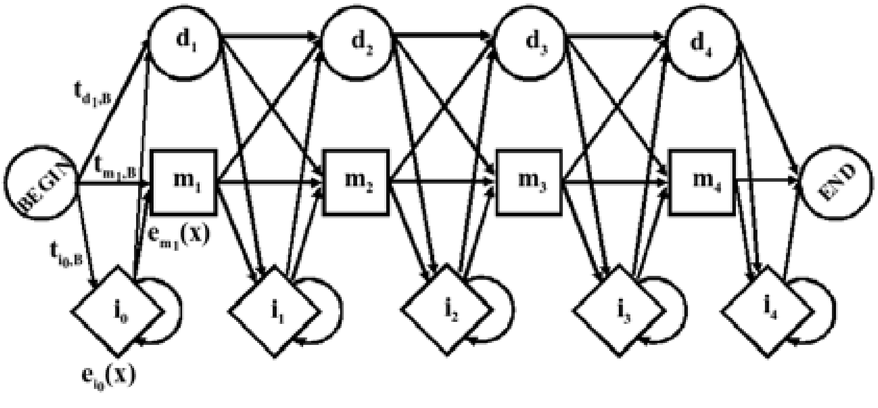


FIGURE 9: A schematic picture of a HMM. Circles symbolize delete states, squares are match states, and insert states are pictured as diamonds. Arrows between the states indicate the possible transitions. A transition probability *t* is associated with each transition. Match and insert states are associated with emission probabilities *e*. These are only shown for the first few states in the figure.

the model would then be the product of the probabilities for each amino acid in the sequence, treating the first amino acid as a special case, since it is not preceded by any other residue. These kind of models work well in some occasions, but they do not give much information about the sequences they model.

A slightly more complicated, but also more informative, way to model a group of sequences is to use profile hidden Markov models (profile HMMs). Hereafter, the terms profile HMM and HMM will be used interchangeably, unless otherwise stated. In short, a profile HMM is a statistical model of a multiple sequence alignment, where probabilities are assigned to each amino acid at each position in the alignment, and to the transitions between positions. The analogy to multiple sequence alignments makes it possible to draw conclusions about the group of sequences that are modelled, making hidden Markov models more appealing than the simple Markov model described above. The HMM can, like multiple alignments and profiles, be used to locate structurally or functionally important residues, since they are conserved in the sequences and consequently receives high probabilities in the HMM. The HMMs are also useful for finding other sequences, similar to the ones modelled.

Some of the advantages with HMMs, compared to for example simple profiles, are position specific scores for amino acids and for the penalties for insertions and deletions. In many other methods, a single gap penalty is chosen regardless of where in the sequence a gap is inserted. This does not model true sequences very well, since it is much more likely to find insertions or gaps in loop areas than in an alpha helix, for example. Other advantages are that the HMMs are built on a formal probabilistic basis, and that less skill and manual interventions are required for using HMMs than for profiles.

A profile HMM consists of a collection of states of three kinds (Figure 9): *match states*, which correspond to the positions in the consensus sequence, *insert states*, which model insertions with respect to the consensus, and *delete states*, which represent deletions with respect to the consensus. The match and insert states emit symbols, in this case amino acids, with a certain probability $e_i(x)$ that symbol x is emitted from state i. The delete states are silent, not emitting any symbols. There are also transitions between the different states (arrows in Figure 9), and a probability t_{ij} to move from state j to state i is associated with each transition. The insert states have self transitions, i.e. transitions back to themselves, to allow for arbitrary length insertions relative to the consensus sequence. To model the beginning and end of a sequence belonging to the alignment, two special states that do not emit any symbols are added in the first and last positions of the HMM. A HMM of length N has N match states with corresponding delete states, and $N+1$ insert states in between the match states.

Using the emission and transition probabilities, a sequence can be emitted by the HMM. Assume that the sequence "TLVSM" is observed. This sequence can be emitted by the HMM in Figure 9 in a number of ways. One possible sequence of states resulting in the observed sequence is $m_1 \rightarrow m_2 \rightarrow m_3 \rightarrow m_4 \rightarrow i_4$. That is, to go from the begin state to match state m_1 emitting symbol "T", then move on to state m_2 emitting symbol "L", to state m_3 emitting an "V", to state m_4 emitting an "V" and finally go to state i_4 emitting symbol "M" before going to the end state. Another possibility is the state sequence $d_1 \rightarrow d_2 \rightarrow d_3 \rightarrow d_4 \rightarrow i_3 \rightarrow i_3 \rightarrow i_3 \rightarrow i_3 \rightarrow i_3$, skipping all match states and emitting all symbols from state i_3 by using the transition back to itself. Yet another possibility is $i_0 \rightarrow m_1 \rightarrow d_2 \rightarrow m_3 \rightarrow i_3 \rightarrow m_4$, as is illustrated in Figure 10. All the possible state sequences have different probabilities, but there is no way to tell which state sequence emitted the observed sequence – the state sequence is hidden for us. That is why hidden Markov models are called *hidden*.

5.3.1 The Plan7 Architecture for HMMs

The HMMs in HMMER2.2g (http://hmmer.janelia.org/), which is the HMM implementation used throughout the work in Papers I–IV, do not look exactly as described above. Instead, the Plan7 architecture, illustrated in Figure 11, is used. The basics are the same as described earlier, with a number of match states corresponding to consensus positions, associated insert and delete states, and transitions between the states. Unlike the previously described architecture, Plan7 does not have any transitions, in any direction, between insert and delete states. This reduction of transitions from 9 to 7 for each node, i.e. each match state with associated insert and delete states, is one of the reasons for the name Plan7. The B and E states are, as above, states used to enter and exit the main model. The special states S, N, J, C and T control which kind of alignment the model is most likely to generate. The S and T states are start and termination states, respectively. None of them emit any symbols. The N state is used to model unaligned N-terminal sequence, in other words the beginning of the sequence. Every time it makes a transition to itself, a symbol is emitted. The same holds for the C state, which models C-terminal sequence not aligned to the actual model. These two states make it possible to model local alignments with respect to the sequence, for

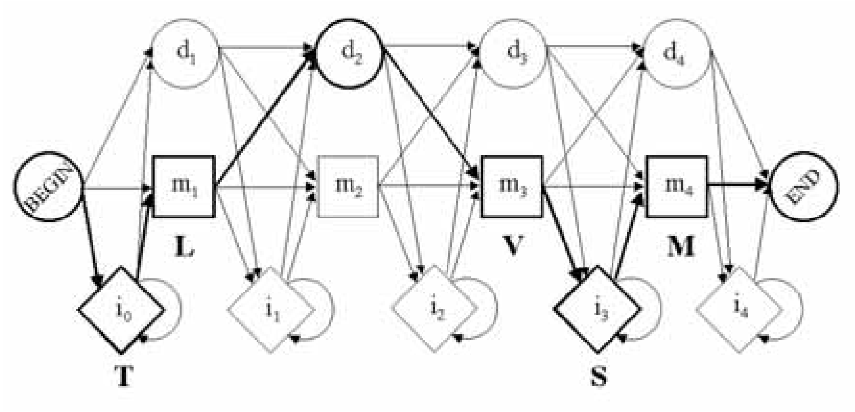


FIGURE 10: An example of how the sequence “TLVSM” can be generated by the HMM illustrated in Figure 9. The bold arrows and states show the path followed to generate the sequence. The bold letters are the symbols emitted at each match and insert state the path passes through. Together, these symbols form the sequence.

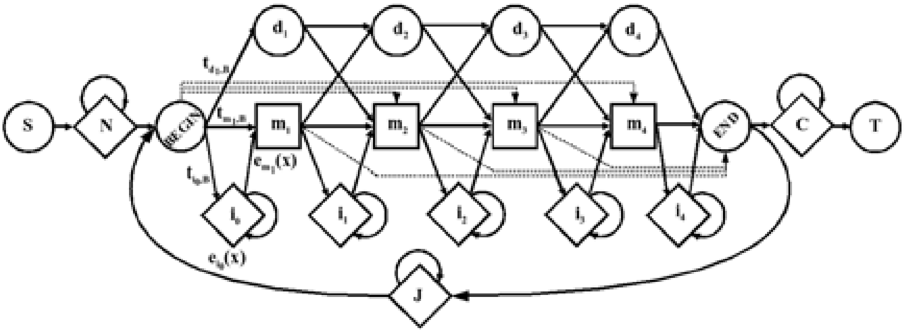


FIGURE 11: The Plan7 architecture used in HMMER2.0 and later. See text for details.

example a single domain in a multidomain protein, as the parts of the sequence not aligned to the main model are captured by the N and C states. The J state is used to model regions in between two matching domains in a sequence. A protein sequence containing two domains belonging to the modelled family, with flanking sequence regions, would therefore start in the S state and then go to the N state where the N-terminal sequence region is emitted. As the first of the two domains starts, it will be modelled by the main model, starting with the B state and ending with the E state. The region between the two domains is described by the J state, after which the sequence enters the main model again to model the second domain. The region following the second domain is finally modelled by the C state, before the T state is entered and the process is terminated.

The dotted arrows in Figure 11 illustrate transitions between the B state and match states, and between match states and the E state. These make it possible to model local alignment with respect to the main model. Using one of the dotted transitions, it is possible to skip some of the match states in the beginning and/or end of the model, without having to pass through a number of delete states. The alignment mode is determined by the actual values for the transitions between the special states, and is decided when building the model. If one wishes more than one type of alignment mode, several HMMs have to be constructed for the same sequences.

5.3.2 Scoring Sequences and HMMs

To score a sequence versus the HMM is the same as finding the probability that a certain HMM generated an observed sequence, i.e. to determine how likely it is that a sequence s is related to the sequences modelled by the HMM. If the probability that the HMM generated the sequence is high, then it is also very likely that the observed sequence is related to the group of sequences that are modelled by the HMM.

A sequence $s = x_1 \dots x_L$ with length L, following the state path $q = q_0 \dots q_{N+1}$ through a HMM μ with N states, has the probability

$$P(s \mid q,\mu) = \prod_{i=1}^{N+1} t_{q_i,q_{i-1}} \prod_{j=1}^{N} e_j(x_{l(j)}), \tag{5.1}$$

where $l(j)$ is the index in the sequence for symbol x at state q_j. Equation 5.1 is simply the product of the probabilities of going from one state to the other, i.e. the transitions t, and the probabilities e of emitting the symbols of the sequence at the given states. To calculate the probability of the HMM emitting the sequence, we have to choose a suitable path for the sequence. The most common approaches are to sum over all possible paths, or to take the path which has the highest probability. To sum over all possible paths can be expressed as:

$$P(s \mid \mu) = \sum_q P(s \mid q,\mu). \tag{5.2}$$

However, to compute the probabilities for all possible paths is often too computationally exhausting, especially when there are many models to compare the sequence against. The path with the highest probability is called the *Viterbi path* [11]:

$$P(s \mid \mu) = \max_q P(s \mid q,\mu). \tag{5.3}$$

Strictly, it is not the probability $P(s \mid \mu) = P(s = x \mid x \text{ is generated by model } \mu)$ that is interesting, since it describes the probability of seeing sequence s in a collection of sequences generated by the given model. Instead, the question of interest is to find the probability, given a sequence s, that this sequence is generated by the model: $P(x \text{ is generated by model } \mu \mid x = s) = P(\mu \mid s)$. To calculate this probability, Bayes' rule can be used:

$$P(\mu \mid s) = \frac{P(s \mid \mu)P(\mu)}{P(s)}. \tag{5.4}$$

To avoid computing the unknown probabilities $P(\mu)$ and $P(s)$, the question is slightly twisted; instead of calculating the probability that the model generated the sequence, the odds that the sequence was generated by model μ rather than model η is calculated:

$$\frac{P(\mu \mid s)}{P(\eta \mid s)} = \frac{P(s \mid \mu)}{P(s \mid \eta)} \frac{P(\mu)}{P(\eta)}. \tag{5.5}$$

Here, η is generated as a null model that tries to fit all sequences in the universe of sequences, for example a sequence database. The relative probability $P(\mu)/P(\eta)$ of the two models can be estimated as the expected number of hits divided by the number of sequences scored.

Scoring in HMMER2.0

In HMMER2.0 and later releases, the two models μ and η are considered equiprobable, so the relative probability is set to 1. The null model in HMMER2 is a single insert state that can make transitions back to itself, and a dummy end state equal to the END state in the actual model. The insert state of the null model emits symbols according to a distribution equal to the average amino acid composition in SWISS-PROT34. The score reported by HMMER is the logarithm of the right hand side in Equation (5.5) – a log-odds score. To correct for bias in sequence composition, HMMER2.0 and later actually uses a second null model in addition to the simple one described above. This model is useful for HMMs modelling sequences with unusual sequence compositions, preventing unrelated sequences with the same unusual composition from receiving unreasonably high scores.

E-values

In addition to the raw scores, an E-value is reported from HMMER2.0 and later. The E-value is an expectation value; it is the expected number of sequences in the database that score higher than or equal to the reported score $S = y$, but that are *not* related to the model. This is the same as the number of hits with a score greater than or equal to y in a database of size N, that one can expect just by chance. By default, the E-value in HMMER is calculated as an analytic upper bound, roughly equal to $\varepsilon = Nz^{-y}$, where z is the base of the logarithm, in this case 2 [12]. More accurate values can be obtained by calibrating the HMM before using it for sequence searches. When calibrating the model, an extreme value distribution $P(S < y) = \exp(-e^{-\lambda(y-\mu)})$ is fitted to the scores generated by the model, and the E-value can then be calculated as $\varepsilon = N \cdot P(S \geq y)$. The scores the distribution is fitted to are generated from a Monte Carlo simulation of a sequence database. To find the parameters λ and μ, the log likelihood is maximized. That is, the maximum of the logarithm of the likelihood of obtaining the simulated scores from a distribution defined by λ and μ is determined:

$$\max_{\lambda,\mu}\{\log P(y_1, \ldots, y_n \mid \lambda, \mu)\}. \tag{5.6}$$

To find the maximum, the zeroes of the partial derivatives with respect to λ and μ are found using a Newton-Raphson algorithm. In practice, only the right tail of the histogram of scores is fitted, because the left tail (the low scoring sequences) does not obey the extreme value distribution. The right tail, around $\varepsilon = 1$, empirically fits the distribution quite well, and since this is the region of interest it is recommended that the models are calibrated.

According to the HMMER User's Guide (http://hmmer.janelia.org/), E-values of 0.1 or less in general represent significant hits.

5.3.3 Aligning a Sequence to a HMM

Aligning a sequence to the HMM is the same as finding the state sequence, or path, through a given HMM, that generated the observed sequence. If one finds that path, one also has the optimal alignment of the sequence to the model, as well as the optimal alignment to other sequences generated by the HMM. The solution is to find the path that gives the highest probability for the sequence, as given by Equation (5.1).

5.3.4 Constructing a HMM

To find the parameters of the HMM, i.e. the transition and emission probabilities, is sometimes called the *training problem*. If an there exist an alignment of the sequences in the family to model, it is rather a question of building a HMM, not training. In this case, the consensus positions, i.e. positions in the alignment where most sequences have an amino acid and not a gap symbol, are set to match states. All gaps with respect to the consensus are counted as delete states, and all insertions correspond to symbols emitted by insert states. The transition probabilities can be calculated by simply counting the number T_{ij} of observed transitions from one state, j, to another state, i, divided by the total number of transitions from that state:

$$t_{ij} = \frac{T_{ij}}{\sum_{i'} T_{i'j}}. \tag{5.7}$$

The emission probabilities are calculated as

$$e_j(x) = \frac{E_j(x)}{\sum_{x'} E_j(x')}, \tag{5.8}$$

where $E_j(x)$ is the number of occurrences of symbol x at position j.

As an example, we consider the alignment in Figure 8, and the column marked with a blue box, before the one with an conserved 'C' in the middle of the alignment. If this column is a match state, there are four transitions from it to the next match state (the next column with the conserved 'C'). There is also one transition from this state to a delete state, since the fourth sequence has a gap instead of the conserved 'C'. There are no transitions to insert states at this position. This means that the transition probabilities from this match state become $t_{mm} = 4/(4+1+0) = 0.80$, $t_{dm} = 1/(4+1+0) = 0.20$ and $t_{im} = 0$.

At this position in the multiple alignment, there are two symbols 'I' (isoleucine) and three symbols 'L' (leucine). This means that the emission probabilities at this position become $e(\mathrm{I}) = 2/(2+3) = 0.40$ and $e(\mathrm{L}) = 3/(3+2) = 0.60$. All other emission probabilities are equal to zero.

For the insert states, background frequencies are often used for the emission probabilities. It is assumed that the symbols in insertions are more or less random, so that the probability of emitting an ‘A’ should be the same as the frequency of an ‘A’ in the universe of protein sequences. The reason for this assumption is that the number of observations often is too small to determine all the parameters, especially in inserts. Also, inserts are by nature not very conserved within a family.

A serious problem with this raw calculation of probabilities is the risk of overfitting the model to the data. If the HMM fits the data too well, it will only recognise the sequences used when building the model, and no related sequences. In the worst case, a sequence differing from the observed sequences in just one, single position can obtain a probability of zero, since this very amino acid has not been observed at that position. To handle this problem, so called *pseudocounts* are added to the raw counts. In this way, all possible symbols will be assigned a probability greater than zero at all positions, even if they are not observed, making it possible to generate and recognise sequences that differ slightly from the training sequences. Also, in the case of proteins, one knows from alignment of homologous proteins that some substitutions of amino acids are more likely than others. For example, tyrosine and phenylalanine often occur in the same place in an alignment, while they both rarely substitute for proline. Knowing that phenylalanine and tyrosine often substitute for each other, a small count can be added to one of them each time the other is observed, increasing the probability for both amino acids.

In a group of related sequences, there are often many similar sequences belonging to the same sequence family, and a few more unique ones. To obtain a good model of all sequences in the family, not just the majority of very similar ones, the few unique ones should have a higher weight. This can be achieved by using tree-based weighting, where sequences with few neighbours on the same branch are given higher weights.

If no alignment is given, the model has to be trained from the raw data, i.e. from a set of unaligned sequences. First, a random alignment is produced, most simply by aligning the first residue of each sequence and then aligning all the others without gaps until the end of the sequences. From this random alignment the parameters can be calculated to create an initial model. All sequences are then aligned to the model, resulting in a new alignment which can be used to calculate new parameters. The procedure is then iterated until the alignment and parameters converge. To avoid being trapped in a local minimum, with a suboptimal alignment, a few variations in this procedure are implemented. However, in HMMER2.0 and later, the training of HMMs is not implemented at all, since sequence alignment tools such as ClustalW give much better alignments, resulting in better HMMs, than the HMM training.

In this work, we use alignments based on structural superimposition as the base for building structure anchored HMMs (saHMMs), and alignments derived from the HSSP database [121] for the secondary structure HMMs (ssHMMs), see Chapter 7 for further details.

Dirichlet mixtures

In the default settings of HMMER2.2g, Dirichlet mixtures are used to define the pseudocounts to add at each position, in order to avoid overfitting.

Let **p** be a probability vector, containing a possible distribution over the twenty amino acids. That is, element p_i in the vector is the probability of amino acid i, $p_i \geq 0$ and $\sum_i p_i = 1$. A Dirichlet density ρ is a statistical density over all probability vectors, meaning that it gives high probabilities to some distributions (probability vectors) of amino acids, and low to others. For example, a certain Dirichlet density may give high probability to distributions where one single amino acid dominates, i.e. to conserved distributions. Other densities might give high probabilities to distributions where amino acids that share a common feature, such as hydrophobicity or size, dominate, while even others favour distributions where no particular kind of amino acid dominates.

For a particular **p**, the value of the density is

$$\rho(\mathbf{p}) = \frac{\prod_{i=1}^{20} p_i^{\alpha_i - 1}}{Z}, \tag{5.9}$$

where Z is a constant that makes ρ sum to unity, and α_i are the parameters of the density.

A Dirichlet mixture is a mixture of Dirichlet densities. The individual densities ρ_j are called components of the mixture, and each component is associated with a mixture coefficient q_j, that functions as a weight for the component. The mixture coefficients sum to 1. A Dirichlet mixture ρ with l components has the form

$$\rho = q_1\rho_1 + \ldots + q_l\rho_l. \tag{5.10}$$

At each position of the alignment, the probability of each amino acid is calculated based on the observed number of occurrences in that column. Pseudocounts are added from each component ρ_j of the Dirichlet mixture, each contributing with different number of counts depending on the particular density. The pseudocounts from each component are scaled according to how likely it is that the individual component has produced the observed data. The result is that the final probability distribution at each position reflects the most likely probability distribution given the observed data, and is not based solely on the raw counts of amino acids.

The mixture used in HMMER2.0 and later versions is a nine-component mixture, where the parameters (q_j, α_j) are estimated based on the multiple sequence alignments in the Blocks database [66]. The Blocks database contains ungapped alignments of protein segments, which correspond to highly conserved regions of proteins.

CHAPTER 6

Structure Alignment Methods

In this chapter, some approaches for structure alignment of protein molecules are described, in particular methods designed to align several molecules simultaneously. A structure alignment is a matching of residues, in different molecules, that are equivalent from a structural point of view, instead of a matching based on the identity of the residues, as in sequence alignments (see Section 5.2). These equivalences can be used to make an optimal superimposition of the structures, by minimizing the root mean square deviation (RMSD) of the equivalenced residues. Often, superimpositions are used as a step in the process of finding the structural equivalences.

There are several reasons for being interested in structure alignments of proteins, instead of pure sequence alignments. The most obvious reason is to study which residues are really at equivalent positions in the folded protein structure, and thereby locate residues that are important for the function and/or folding of the protein. Structural alignments can also help to detect distant evolutionary relationships that are difficult, or even impossible, to find from sequence information alone. In this thesis, structure alignments are used instead of sequence alignments in order to obtain alignments of sequences with highly similar structures, but where the mutual sequence identities are very low.

6.1 Alignment and Superimposition of Protein Structures

To find matching residues in two or more protein structures is not a trivial task, since the proteins can differ in size or have slightly different angles between their secondary structure elements, and still have the same overall fold. Even if the similarity is obvious by eye, it is difficult to parameterize it and make a computer find the matching residues automatically.

If the structures are superimposed as rigid bodies, the centre of the superposition might be quite well defined, while the further away you are from the centre, the further apart are the structures, even though the basic shape is the same. For example, two helices that are situated at the same position with respect to the other elements of the protein in two structures, might be parallel but still some distance apart in the superposition, since one protein could be more loosely connected or have longer loops than the other. This kind of situation makes it very difficult to determine which residue in one protein corresponds to which in the others, especially in an automatic approach. An

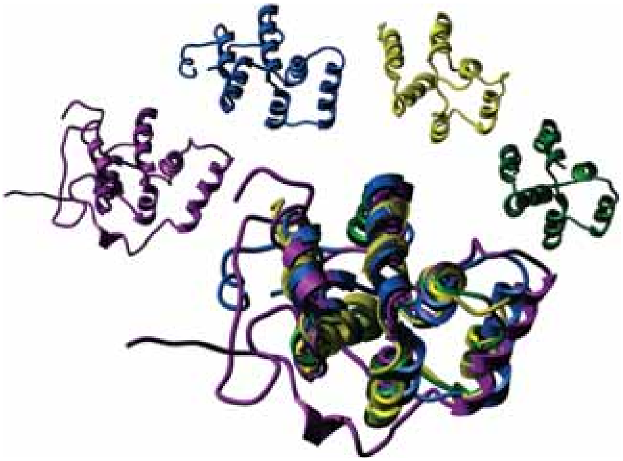


FIGURE 12: An example of four superimposed structures belonging to the DEATH domain family. Above the superposition smaller images are shown for the individual structures. The four protein domains are: d1d2zb_ (magenta), d1d2za_ (blue), d3ygsp_ (yellow) and d1cy5a_ (green), following the SCOP nomenclature.

example of such a situation is the rightmost helix in Figure 12, where a superposition of four similar domain structures is displayed. Here, the blue and the magenta helices obviously are equivalent, but since they are tilted in slightly different directions, they only overlap perfectly on a few residues in the middle, while the ends of the helices are quite distant.

Several methods have been developed to compare protein structures. Most methods developed are designed to compare just two proteins at a time, and almost all multiple methods use pairwise alignments as a starting point. Methods for structural alignments are reviewed in for example [53], [40], and [77].

A very common approach to find matching residues in the proteins to align, is to use dynamic programming (e.g., [142], [120], [69], [44], [148], [74], [138]). Dynamic programming (see Section 6.1.1) finds the optimal solution for the superposition of two structures, conditioned on the scoring function optimized during the process. This scoring function is also the main difference between the above methods. Some alternative scoring functions are to compare intra-protein distances [69], to combine and compare features such as surface accessible area, secondary structure and sequence information [74], to minimize the “soap area” between the backbones of the two structures [44] and to compare the discrete curvature of the backbones [148].

Several methods have been developed that represent the secondary structure elements as vectors, and find the best matching between those as a first step in the alignment procedure ([91], [140], [128], [3], [132], [150], [93]). The reason for this choice is to reduce search space for the initial alignment, and to ensure biologically relevant alignments since the secondary structures are the building blocks of the structures.

Genetic algorithms have also been used to find initial equivalences [140].

For the actual rotation and translation to superimpose the structures, most methods use some kind of iterative least squares procedure that minimizes the RMSD between equivalenced residues (e.g., [91], [118], [148], [74], [140], [128], [3], [132]). Most often, the structures are treated as rigid bodies, however, some approaches accept more flexible alignments (e.g., [154]). Often, equivalenced residues are found using nearest neighbours or dynamic programming. These equivalences are then superimposed, and the procedure is iterated until either the RMSD, the equivalenced residues, or both have converged. Another method to find the optimal superposition and/or equivalences is Monte Carlo optimization [69], [93].

There are a few methods with more "unique" approaches, that use hashing to find common submotives [86], search all possible combinations of rotations and translations to find the maximum number of matched C_α [34], or assemble structurally similar fragment pairs using combinatorial extension [129].

6.1.1 Dynamic Programming

Dynamic programming is one of the most common methods to optimally align two sequences, whether it is DNA, protein or an abstract sequence of structural features. Dynamic programming is also used extensively in approaches to align multiple sequences, however, in these cases the method quickly becomes computationally exhausting. Since the method is so commonly used, the basic procedure is shortly described in this section.

Dynamic programming is a general method that guarantees a mathematically optimal alignment of two linear sequences, given a scoring function and penalties for insertions or deletions (see Section 5.2). The scoring function is often given as a scoring matrix – a table of scores for matches and mismatches between all sequence symbols. Often, there are two kinds of penalties for generating an insertion/deletion; a gap opening and a gap extension penalty. The gap opening penalty is used when opening a new gap in a sequence, while the gap extension penalty is used for extending the gap, i.e. inserting multiple gap symbols in one of the sequences. The gap extension penalty is usually lower than the gap opening penalty, since it is more biologically reasonable to extend an existing gap than to open a new one.

Dynamic programming was first introduced in molecular biology by Needleman and Wunch [102]. The heuristic measure of homology introduced in that paper has since then been developed into a true measure of the distance between sequences, as illustrated in the Smith-Waterman algorithm [134]. The Needleman-Wunch algorithm is designed for constructing global alignments, where one complete sequence is aligned to another complete sequence. The Smith-Waterman algorithm, on the other hand, is designed for local alignments, where parts of one sequence is aligned to a subsequence of the other. This makes it possible to find alignments between only parts of the sequences, which is the biologically more common situation. The method has also been optimized for time and memory usage [56], and the method is often slightly modified to fit a given application. However, the basic idea of dynamic programming is the same in all cases, why the Smith-Waterman algorithm is described below in more detail, to illustrate the method.

The Smith-Waterman algorithm

The Smith-Waterman algorithm [134] is used to find similarities between two long sequences, by locating a pair of segments (one from each sequence) such that the pair has a higher similarity than any other pair of segments. The similarity is calculated using a similarity measure $s(a,b)$ between elements a and b in sequences $A = a_1a_2\dots a_n$ and $B = b_1b_2\dots b_m$. Introducing gaps in one of the sequences is penalised with a penalty w_k, dependent on the number of gaps, k. An $(n+1)\times(m+1)$ similarity matrix H is constructed to find the most similar pair of segments, where the element H_{ij} can be seen as the similarity of the two segments ending at positions a_i and b_j, respectively. To start, the similarities between an empty position b_0 first in B and all sequence positions in A are set to 0. These represent segments where the beginning of sequence B matches internal positions in sequence A, for example if $B = abc$ is matched to $A = xxabc$. The equivalent holds for an empty position a_0 matched to B. Hence:

$$H_{k0} = H_{0l} = 0 \text{ for } 0 \le k \le n \text{ and } 0 \le l \le m. \tag{6.1}$$

The other elements in H are then chosen as the maximum similarity given by one of the following four possible combinations of sequence elements:

1. If a_i is matched to b_j, the similarity is calculated as $H_{ij} = H_{i-1,j-1} + s(a_i,b_j)$.
2. If a_{i-k} is matched to b_j, so that a_i is at the end of a deletion of length k (k gaps are inserted after position b_j, and a_i is matched to gap number k), then the similarity is calculated as $H_{ij} = H_{i-k,j} - w_k$.
3. If a_i is matched to b_{j-l}, so that b_j is at the end of a deletion of length l (l gaps are inserted after position a_i, and b_j is matched to gap number l), then the similarity is calculated as $H_{ij} = H_{i,j-l} - w_l$.
4. If $s(a,b)$ can give negative values, 0 is included to avoid negative similarities. The number 0 means no similarity.

In summary, element H_{ij} is determined as:

$$H_{ij} = \max \left\{ \begin{array}{c} H_{i-1,j-1} + s(a_i,b_j) \\ \max_{1\le k\le i}\{H_{i-k,j} - w_k\} \\ \max_{1\le l\le j}\{H_{i,j-l} - w_l\} \\ 0 \end{array} \right\}. \tag{6.2}$$

The pair of segments giving the highest possible similarity, i.e. the optimal alignment, is found by locating the largest element H_{ij}, and then backtracking the calculations to find the other matrix elements leading to this value. The backtracking procedure ends when a zero matrix element is found. In this way, the most similar segments from the two sequences and their alignment are found. To find alternative matching segments, the next largest element, not in the same path as the largest element, should be located.

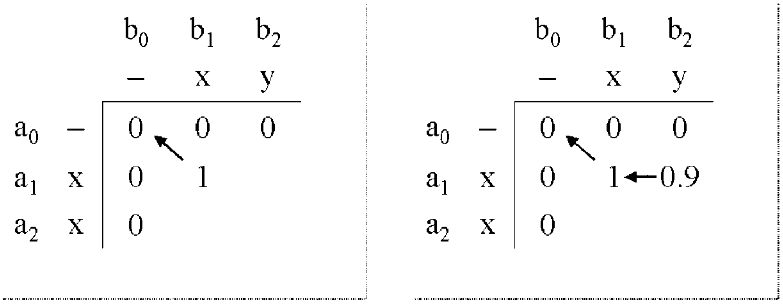


FIGURE 13: The construction of the similarity matrix in the example. After the first step, the first row and column are set to 0. Left: The result after calculating element $H_{1,1}$. Right: The result when element $H_{1,2}$ is calculated and added. The arrows show which other element each calculated element is based on.

The algorithm is illustrated with the following example. Assume that we have a simple similarity measure

$$s(a,b) = \begin{cases} 1 & \text{if} a = b \\ 0 & \text{otherwise} \end{cases} \tag{6.3}$$

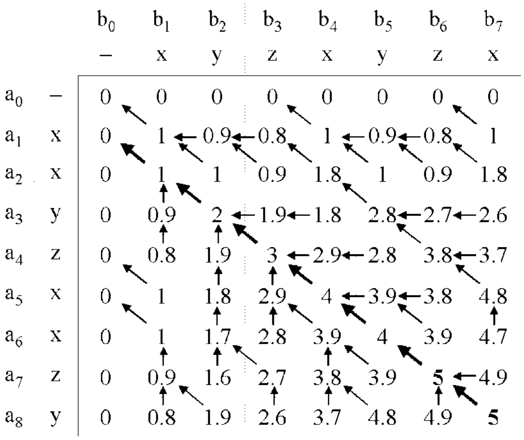
and a gap penalty $w_k = 0.1 \cdot k$. Given sequences $A = xxyzxxzy$ and $B = xyzxyzx$, the similarity matrix H can be calculated. In this case, $n = 8$ and $m = 7$.

First, all elements in the first row and the first column are set to 0. Then, we continue to calculate element $H_{1,1}$, which represents the similarity of two segments ending at positions $x_{a1}x_{b1}$, where x_{a1} is the x at position 1 in sequence A and x_{b1} is the x at position 1 in sequence B. If x_{a1} is matched to x_{b1} (alternative 1 above), then the similarity is $H_{0,0} + s(a_1, b_1) = 0 + s(x,x) = 0 + 1 = 1$. If x_{a1} is at the end of a deletion (alternative 2), the similarity is $\max_{1 \leq k \leq 1}\{H_{1-k,1} - w_k\} = \max_{1 \leq k \leq 1}\{H_{1-k,1} - 0.1 \cdot k\}$. In this case, $k = 1$ is the only option, since the index $1 - k$ should be equal to or greater than zero (no negative indices!). Hence, we obtain a similarity of $H_{0,1} - 0.1 \cdot 1 = 0 - 0.1 = -0.1$. Correspondingly, if x_{b1} is at the end of a deletion (alternative 3), we obtain a similarity of -0.1. Alternative 4 above is not relevant in this case, since $s(a,b)$ never yields negative values. To find element $H_{1,1}$, we take the maximum of all these values (cf. Equation (6.2)):

$$H_{1,1} = \max \left\{ \begin{array}{c} H_{0,0} + s(x_{a1}, x_{b1}) \\ \max_{1 \leq k \leq 1}\{H_{1-k,1} - w_k\} \\ \max_{1 \leq l \leq 1}\{H_{1,1-l} - w_l\} \end{array} \right\} = \max\{1, -0.1, -0.1\} = 1.$$

We find that $H_{1,1} = 1$, a value derived from element $H_{0,0}$ (see Figure 13, left).

If we move on to element $H_{1,2}$, alternative 1 gives the similarity $H_{0,1} + s(x_{a1}, y_{b2}) = 0 + 0 = 0$, and alternative 2 gives the similarity -0.1, as derived above for $H_{1,1}$. Alternative 3 can give two values for the similarity, one for $l = 1$ and one for $l = 2$, of



		b_0	b_1	b_2	b_3	b_4	b_5	b_6	b_7
		–	x	y	z	x	y	z	x
a_0	–	0	0	0	0	0	0	0	0
a_1	x	0	1	0.9	0.8	1	0.9	0.8	1
a_2	x	0	1	1	0.9	1.8	1	0.9	1.8
a_3	y	0	0.9	2	1.9	1.8	2.8	2.7	2.6
a_4	z	0	0.8	1.9	3	2.9	2.8	3.8	3.7
a_5	x	0	1	1.8	2.9	4	3.9	3.8	4.8
a_6	x	0	1	1.7	2.8	3.9	4	3.9	4.7
a_7	z	0	0.9	1.6	2.7	3.8	3.9	5	4.9
a_8	y	0	0.8	1.9	2.6	3.7	4.8	4.9	5

FIGURE 14: The final similarity matrix resulting from the example. Arrows indicate from which element each value is derived, and are used to backtrack the calculations to obtain the matching sequence segments. Bold arrows represent the optimal path through the matrix, i.e. the alignment of the two segments having the highest similarity.

which we want to choose the largest: $\max\{H_{1,2-1} - w_1, H_{1,2-2} - w_2\} = \max\{H_{1,1} - 0.1 \cdot 1, H_{1,0} - 0.1 \cdot 2\} = \max\{1 - 0.1, 0 - 0.2\} = 0.9$, derived from element $H_{1,1}$. Element $H_{1,2}$ is the maximum of all three alternatives:

$$H_{1,2} = \max \left\{ \begin{array}{c} H_{0,1} + s(x_{a1}, y_{b2}) \\ \max_{1 \leq k \leq 1}\{H_{1-k,2} - w_k\} \\ \max_{1 \leq l \leq 2}\{H_{1,2-l} - w_l\} \end{array} \right\} = \max\{0, -0.1, 0.9\} = 0.9,$$

which is derived from element $H_{1,1}$ (see Figure 13, right).

In this way, all the elements in the matrix can be calculated, see Figure 14. In the figure, the arrows indicate which previous element each value is based on. When the similarity matrix H is filled, the largest element is located, representing the pair of fragments with the largest similarity. In this case, elements $H_{7,6} = H_{8,7} = 5$ contain the largest value. By following the arrows we can backtrack the calculations from $H_{8,7}$ to $H_{7,6}$ to $H_{6,5}$ to ..., all the way to element $H_{1,0}$, which has the value 0. This yields the aligned sequence segments

$$\begin{array}{cccccccc} x & x & y & z & x & x & z & y \\ - & x & y & z & x & y & z & x, \end{array}$$

which contain no gaps, except for the initial one, and two mismatches, at positions six and eight in the alignment.

6.2 Multiple Structure Alignment

In the work presented in this thesis, we want to make use of alignments of *multiple* structures. In a multiple structure alignment, preferably all input structures should be aligned simultaneously, or at least the order of adding the structures to the alignment should not affect the final result.

To construct multiple structural alignments, the most common approach is to perform pairwise alignments and add proteins to the alignment based on a guide tree (e.g., [118], [120], [92]), pairwise similarity scores (e.g., [109], [151]) or Monte Carlo optimization [63]. Other methods align all proteins to a pivot structure, which might be a consensus structure or a chosen representative structure (e.g., [52], [148], [108], [153], [156]).

There are only a few software tools available that attempt to align all input molecules simultaneously. MUSTA [87] uses a hash table to find conserved submotives, which are then used to find the optimal transformations. MASS [35] initially detects pairs of conserved secondary structure elements, also using hash tables, to construct local alignments, which are then refined using the atomic coordinates. Also MultiProt [126] and MUSTANG [78], both described below, do simultaneous multiple structure alignments.

In the following, three methods for structure alignment are described in more detail. STAMP is the software that was first used in this work. Until recently, it was one of the very few methods available for multiple structure alignment that produces actual residue matches. The two other tools described are MultiProt and MUSTANG, where the later now is used in our method instead of STAMP for producing structure-anchored sequence alignments.

6.3 STAMP

STAMP (Structural Alignment of Protein Sequences) [118] aligns several sequences based on their structural similarity. A guide tree based on pairwise comparisons is used to determine the order in which the structures are aligned.

An overview of the procedure STAMP uses is shown in Figure 15. To start, STAMP needs the structures to be reasonably superimposed, a superimposition which is refined in the procedure, and which is used to construct the guide tree. The structural domains are then superimposed in the order indicated by the pre-calculated tree. First, a matrix is calculated, containing the likelihoods of structure equivalence between each residue in one domain and each residue in the other. The optimal way through the matrix is found, see below, resulting in a list of equivalent residues with corresponding C_α positions. These positions are used to calculate the transformation (translation and rotation) of one structural domain that gives the lowest RMSD with respect to the other. The domain is transformed, resulting in a new set of coordinates, and the calculations are repeated until convergence. STAMP then repeats the procedure for the next pair to be superimposed.

The initial multiple superimposition or multiple sequence alignment needed by STAMP can be produced by (i) constructing a multiple sequence alignment of the

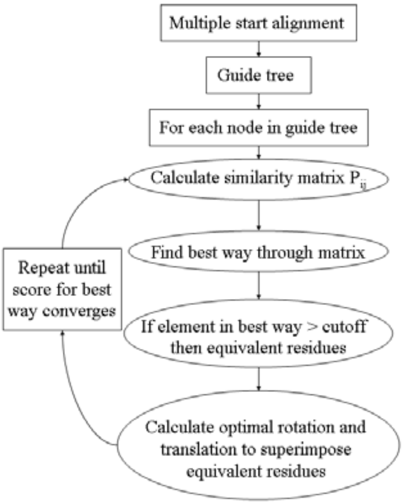

Multiple start alignment
Guide tree
For each node in guide tree
Calculate similarity matrix P_{ij}
Find best way through matrix
Repeat until score for best way converges
If element in best way > cutoff then equivalent residues
Calculate optimal rotation and translation to superimpose equivalent residues

FIGURE 15: The STAMP procedure for alignment and superposition of protein structures. See text for details.

domains to be superimposed, (ii) constructing a simple alignment where the sequences are stacked on top of each other with no gaps, starting from the N-terminus, or (iii) pairwise superimposition of all structural domains against one of them, after which the superimpositions are converted to a multiple sequence alignment.

The initial alignment is used to construct a tree that guides the order of superimposition. For each pair of domains to be superimposed, the k positions aligned for this pair in the initial alignment, with no gaps, are rotated and translated to minimize the RMSD for the pair. The RMSD is a measure of the distance between equivalent atoms in the two molecules, and consequently measures how well the two molecules are superimposed, and how similar they are in structure. If N proteins are to be aligned, the RMSD values for each of the possible $N(N-1)/2$ pairs are used to construct the guide tree, such that pairs with low RMSD are close to each other in the tree, while pairs with high RMSD are further away from each other. To construct the tree, each molecule is assigned to its own subset. Then the two subsets with the lowest RMSD are joined together, and the length of the branch is set as the distance. These two subsets are then treated as a single subset when the process is iterated, until all subsets have been joined to a single set, the tree. When two subsets containing more than one molecule are compared, the average RMSD values from all possible pairings of molecules from the two subsets are used to determine the distance.

The structural domains are then superimposed starting from the leaves in the tree, superimposing a pair of domains at each node until reaching the root of the tree. In this way, the most similar domains are compared first, leaving the comparison and alignment of more distantly related domains until later in the procedure. At internal nodes, where more than two domains are to be superimposed, average values are used for domains belonging to the same branch of the tree.

The actual superimposition in each node starts by calculating a structure equivalence matrix for the two domains to be superimposed. For each residue i in domain A, the probability P_{ij} of structural equivalence to residue j in domain B is calculated as:

$$P_{ij} = \exp(-\frac{d_{ij}^2}{2E_1^2})\exp(-\frac{s_{ij}^2}{2E_2^2}), \tag{6.4}$$

where d_{ij} is the distance between the C_α atoms of residues i and j, s_{ij} is a measure of their conformational similarity, and E_1 and E_2 are constants. If A contains m residues and B contains n, this results in a $m \times n$ matrix. In case more than two domains are to be superimposed in a node, domains on the same branch are kept fixed with respect to each other, and the average P_{ij} for all possible combinations is computed for each position ij. For example, if domains A and B superimposed on one branch are to be compared to domains C and D from the other branch, then all possible combinations are A-C, A-D, B-C, and B-D. If a comparison is made to a gap, a neutral value of 0 is used.

The best way through the matrix, i.e. the path that yields the highest score S, is determined using a modified Smith-Waterman algorithm [134], [118] (see also Section 6.1.1). The score S is calculated as the sum of P_{ij} values along the path. The path corresponds to the best possible set of equivalent residues. From this set, the pairs having a P_{ij} larger than a threshold T are used to obtain two sets of equivalenced C_α positions.

The two sets of equivalent C_α positions can be seen as two sets A and B of k vectors $\mathbf{a}_i$ and $\mathbf{b}_i$ ($i = 1,\ldots,k$), where k is the number of equivalent positions. Each vector $\mathbf{a}_i$ and $\mathbf{b}_i$ contain three elements, representing the x-, y-, and z-coordinates of the residue at position i. Given these two sets, the optimal superimposition is found by determining a rotation matrix $\mathbf{R}$, and a translation $\mathbf{t}$ which, when applied to set A, yield a transformed set of coordinates $\tilde{\mathbf{a}}_i$ which minimizes the RMSD with respect to set B. In nodes where several domains are compared, the average C_α coordinates for domains belonging to the same branch are used.

The domain is transformed using the calculated rotation $\mathbf{R}$ and translation $\mathbf{t}$, resulting in a new set of coordinates that can be used to calculate a new distance matrix according to Equation (6.4). The calculations are repeated until the score S does not change more than 0.1% compared to the previous iteration. STAMP then moves on to the next node and pair of domains/averaged domains to be superimposed, until reaching the root where all structural domains are superimposed.

6.4 MUSTANG

MUSTANG (MUltiple STructural AligNment AlGorithm) [78] is a relatively recent tool for multiple structure alignment. According to the authors, it tries to extend the spirit of DALI [69], which is one of the most widely used applications for pairwise structure alignments. Instead of comparing the actual coordinates of the proteins, the methods find similarities between distance matrices computed from the structures. The distance matrix of a protein structure comprises all pairwise distances between the C_α atoms in the structure. This matrix contains all information needed to reconstruct the protein structure, except for the chirality[1] of the molecule.

In DALI, the distance matrices of the two molecules to align are first systematically compared to find all matching hexapeptide-hexapeptide contact patterns, which are stored in a pair list. All scores are calculated based on similarities in the internal distances stored in the distance matrices. The highest scoring pairs are used in the next step, the actual alignment, which is produced by Monte Carlo optimization. A number of seed alignments are constructed from all triplets of non-overlapping hexapeptides in the pair list. The seed alignments are extended using overlapping contact pairs, and the highest scoring alignments are optimized in parallel. The optimization consists in extending the alignment based on overlapping contact pairs and trimming the alignments by removing negatively scoring matches. The procedure continues with expansion and trimming until the score no longer improves. Finally, the best alignment is refined.

In a similar way, MUSTANG uses similarity in patterns of residue-residue contacts within the proteins, as well as local structural topology, to align the C_α atoms in a set of protein structures. First, scores of pairwise correspondences are determined, which are used for pairwise structural alignments. The scores of all residue-residue correspondences are then recalculated in the context of multiple structures. Finally, all structures are progressively aligned along a guide tree, using the recalculated scores.

For the first phase, where pair-wise residue-residue scores are determined, complete distance matrices are calculated for all structures to align. Then a list of all maximal similar substructures is compiled for each pair of structures. A similar substructure is a pair of equal length fragments, one from each structure, with a length of at least l_{min}, that can be superimposed with an RMSD of at most ε. The MUSTANG authors empirically determined the values $l_{min} = 6$ and $\varepsilon = 1.75$Å to give the best results. A *maximal* similar substructure is a pair of similar fragments that are not contained in longer fragments with the same N-terminus, i.e. which do not start at the same position as a longer fragment.

The list of maximal similar substructures for a pair of structures is compiled by superimposing all possible combinations of fragments of length l_{min} from the two structures. In case the RMSD is at most ε, the fragment pair is extended by adding positions to the C-terminus, i.e. to the ends of the fragments, until the fragments no longer can be superimposed with small enough RMSD.

From the list, rough pairwise similarity scores are derived for all maximal fragment pairs $m^{ii'}_{jj'}(l)$, where i and i' are the structures to align, j and j' are the start of the respective fragments, and l is the length of the fragments. The score $w^{ii'}_{j+t,j'+t}(0 \leq t \leq l-1)$ of every correspondence $m^{ii'}_{jj'}(l)$ is calculated as:

[1] The chirality of a molecule is its "handedness". Compare to a left and a right hand – they are identical with respect to internal distances between for example fingers, but are each others mirror images.

$$w^{ii'}_{j+t,j'+t} = \sum_{\forall 0 \leq p \leq l-1} \sum_{\forall p+1 \leq q \leq l-1} \phi(i,i',j,j',p,q) + \sum_{\forall 0 \leq q \leq l-1} \phi(i,i',j,j',t,q). \quad (6.5)$$

The function ϕ is a slight modification of the similarity function used in DALI, and is based on values in the distance matrices:

$$\phi(i,i',j,j',x,y) = \begin{cases} (\theta - |d^{i}_{j+x,j+y} - d^{i'}_{j'+x,j'+y}|/d^{*}) \cdot \omega(d^{*}), & x \neq y, \\ 0, & x = y. \end{cases} \quad (6.6)$$

Here, d^{i}_{jk} is the distance between residues j and k in structure i, d^{*} is the average of the distances $d^{i}_{j+x,j+y}$ and $d^{i'}_{j'+x,j'+y}$, θ is a constant and $\omega(d)$ is an envelope function (see [78] and [69]). The similarity scores are used to derive pairwise global structure alignments by dynamic programming (see Section 6.1.1) without gap penalties.

These pairwise alignments are used to prune the list of maximal similar substructures, in order to limit its length. Only those pairs of fragments that are close to any of the correspondences in the pairwise alignment of the structures are kept in the list.

Using the pruned list of fragments for each pair of structures i and i', the pairwise residue-residue scores are then recalculated. Each pair of maximal fragment pairs on the form $m^{ii'}_{jj'}$ and $m^{ii'}_{kk'}$ are jointly superimposed, and in case the resulting RMSD is at most $\varepsilon' = 6.5$Å, the score is updated using a modified version of Equation (6.5).

In phase two, pairwise structural alignments are generated using the recalculated scoring matrix and dynamic programming. The alignments are used in phase three, the extension phase, to generate a new scoring matrix with scores that are calculated in the context of multiple structures.

For all correspondences in a pairwise alignment, the score in the new matrix is set to the same value as in the previous matrix. Next, transitive correspondences between every pair of structures i and i' through every other structure j are detected, and the scores are updated according to these. I.e., if residue x in structure i is aligned to residue y in structure j, which in turn is aligned to residue z in structure i', the score $w^{ii'}_{x,z}$ is increased in order to reflect this transitive correspondence. The more intermediate structures supporting the alignment of a pair of residues, the higher the score of matching these two residues.

Finally, the recalculated scores for each pair of structures are used in the progressive alignment phase where the multiple structure alignment is generated. The multiple alignment is assembled along a binary guide tree, where the structures to align are the leaves, which are connected in a way reflecting their structural similarities. The tree is constructed using the neighbour-joining method [119], which starts from a star-like tree and iteratively pairs the nodes that gives the smallest sum of branch lengths, until the tree is binary. The branch lengths are calculated based on the distances between nodes, which in this case are derived from normalized alignment scores of the pairwise structure alignments. Starting from the leaves in the guide tree, a multiple alignment is generated by aligning the two subalignments in each node in a pairwise fashion.

From the correspondences in the multiple structure alignment, a structure superimposition is also constructed by minimizing the sum of the RMSD for the residues that are aligned in all structures.

6.5 MultiProt

In our evaluations of structure alignment software (see Section 7.1.3), we also included MultiProt [126], which is one of the few additional methods that perform simultaneous multiple structure alignments.

MultiProt derives alignments from simultaneous superimposition of input molecules, and the method does not require that all molecules participate in the alignment. Instead, an ensemble of alignments is reported, with alignments for all possible number of input molecules. MultiProt uses the pivoting technique, meaning that all molecules are aligned to a pivot molecule. By selecting each molecule in turn as the pivot, all solutions can be detected. In the first stage, all possible structurally similar fragment pairs are detected between the pivot molecule and all the other structures to align. A structurally similar fragment pair is a pair of fragments, one from the pivot and one from another molecule, with maximal length and an RMSD of at most ε.

All possible combinations of structurally similar fragments between two or more molecules are then detected. It is required that the pivot molecule participates in the alignment, but no requirement is placed on including *all* of the input molecules. From each set of structurally similar fragments, only one fragment is selected for each molecule, in case there is more than one. The fragment is chosen so that the transformation which optimally superimposes the fragment on the pivot, also gives the largest global structural alignment, with an RMSD of at most ε, with the pivot molecule. In this way, a set of rotations and translations is calculated, based on the similar fragments, so that all aligned molecules are superimposed on the pivot. Finally, the largest structural cores between the 3D-transformed molecules are detected. This is done iteratively, by applying the transformations, determining the multiple structural correspondences, and calculating new transformations based on these new correspondences. In the default settings, this procedure is repeated three times. The solutions are scored based on alignment size and the multiple RMSD of the alignment, calculated as the average of the RMSDs of the structural cores of each molecule and the pivot. Longer alignments and smaller RMSDs give higher scores. Solutions are grouped based on the number of aligned molecules.

Chapter 7
Contributions and Related Work

As discussed in the introduction (Chapter 1), one of the aims in this thesis is to investigate how structural information can be included in HMMs, and how this affects the performance of the models. We hypothesize that structural information will improve the HMMs ability to recognize and model remote relationships, and make the models more accurate.

In the first four papers, the structure-anchored HMMs, saHMMs, are presented and investigated. These HMMs use structure alignments as the base for the models. In the last paper, the secondary structure HMMs, ssHMMs, are described. These use secondary structure information in addition to sequence information when scoring sequences against the HMMs.

7.1 Structure-Anchored HMMs (saHMMs), Papers I-IV

Sequences more than 20-30% identical are often uncomplicated to align to each other. However, below this limit the quality of an alignment cannot be guaranteed, because the significance of an alignment is no higher than that of an alignment of two random sequences [121] (see Section 7.1.2). This presents a problem in case one wants to build models of groups of very dissimilar sequences, as sequence alignments are needed to find similarities to other proteins, and in particular to construct hidden Markov models.

Our approach is to use structural alignments of known structures, where the residue equivalences are determined based on the structural environment of the residues rather than on the identity of the actual amino acids. This makes it possible to align low identity protein sequences, as long as their structures are similar enough. Also, sequence alignments based on pure sequence information and statistical methods might differ significantly from those constructed based on structure. Thus, the use of structure alignments is a way to include structural information in the models, when the structure-based sequence alignments are used to build HMMs. The resulting structure-anchored HMMs are presumably better at recognising even very distant relatives of the protein family.

7.1.1 Outline of the Method

We have developed a method that use structurally similar, low sequence identity representatives of SCOP protein domain families, to construct so called structure-anchored

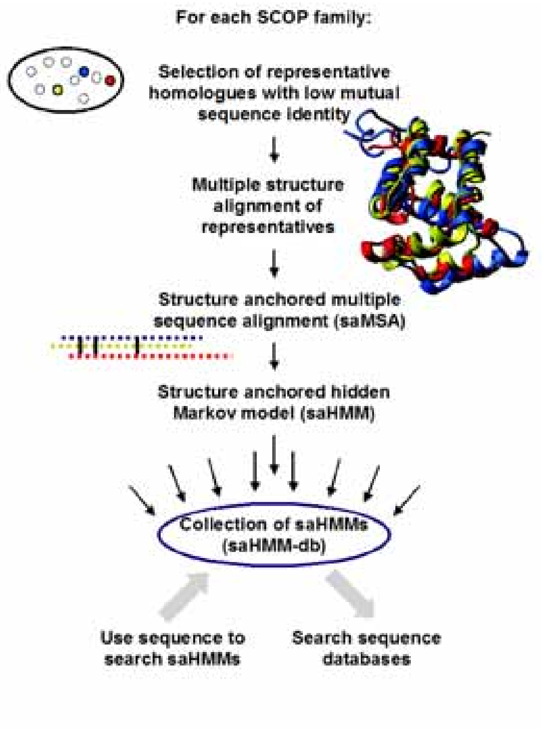

For each SCOP family:
Selection of representative homologues with low mutual sequence identity
Multiple structure alignment of representatives
Structure anchored multiple sequence alignment (saMSA)
Structure anchored hidden Markov model (saHMM)
Collection of saHMMs (saHMM-db)
Use sequence to search saHMMs
Search sequence databases

FIGURE 16: A schematic illustration of our method. For each family, we start by selecting a representative set of protein domains with low mutual sequence identities. These are then structurally aligned, and the structure-based multiple sequence alignment is used to build a structure anchored hidden Markov model (saHMM) for the family. The resulting database of saHMMs can be used to obtain useful information. See text for details.

hidden Markov models, saHMMs, for assigning family relationships to protein sequences. The main steps of our method are illustrated in Figure 16.

First, only those sequences in a family that have very low sequence identity with respect to each other are selected as representatives for that particular family of protein domains. This selection is done in order to avoid bias towards sequences common in the family. The structures of the selected protein domains are then multiply aligned, i.e. a multiple sequence alignment is generated based on which residues are structurally equivalent in the structures, and thus are close to each other in space when the structures are superimposed. One structure alignment is constructed for each family. The resulting structure-anchored multiple sequence alignment is presumably better than what could be achieved from aligning the sequences based on sequence information only, especially in the case of low sequence identity. Finally, the structure-anchored multiple sequence alignment is used to build a structure-anchored HMM representing the family.

The construction of one model for each protein family yields a whole database of saHMMs, which in turn can be searched with sequences to find similarities. If one has a particularly interesting sequence, this can be searched against the database to find which saHMM fits the sequence best, and thus which family the sequence most likely belongs to. If, on the other hand, one is particularly interested in a certain protein family, the corresponding saHMM can be used to search sequence databases or newly sequenced genomes for more members of the family. In both cases, the fact that the saHMMs are built from structure-anchored alignments, means that matching a sequence to an saHMM also matches the sequence to the corresponding structure.

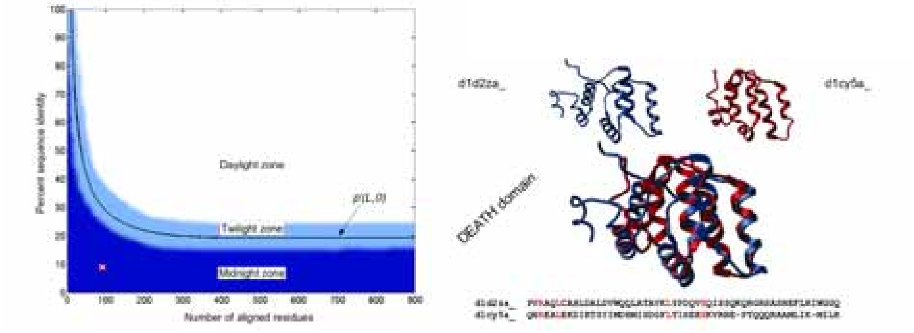
7.1.2 The Midnight ASTRAL Set

The first step in the process of building saHMMs involves the definition of groups of protein domains with similar structures, and to select representatives from each group. We chose to use the family level in the SCOP classification (Section 4.3) as groups of structurally related protein domains, and exploit the ASTRAL compendium (Section 4.3.1) in order to obtain the 3D coordinates of individual domains.

In the PDB (Section 4.2), and consequently in SCOP, there is a high degree of redundancy [20]. Both databases are biased towards proteins that crystallize or are suitable for NMR experiments, and they also contain structures which are the result of mutagenesis studies, where the effect single mutations have on the final structure is investigated. This means that some proteins have a huge number of entries, only differing in single positions, while the majority of proteins only have one entry. As a consequence, some families in SCOP contain lots of domains, while others only have one or two members. The number of families in superfamilies, and the number of superfamilies in folds are also skewed, but not in a correlated way.

To avoid obtaining an alignment biased towards sequences very common in the family, and in order to maximize the sequence diversity of the representatives from each family, we decided to use only sequences with mutual sequence identities below a certain limit. The limit was defined as the border to the so called *twilight zone*, described by the HSSP-curve presented by Mika and Rost [97], [117], and illustrated in Figure 17(a). The twilight zone is the border where the percentage sequence identity between two aligned protein sequences no longer is useful to determine whether the two proteins are related or not. The actual curve that defines the border to the twilight zone differs depending on the data it is based on, and on slightly different definitions between authors, but the basic idea is the same.

If all known proteins are pairwise aligned, the resulting sequence identity can be plotted versus the alignment length. In such a plot, pairs of non-related proteins will have low sequence identities over mostly short alignment lengths, while related proteins often have higher sequence identities and longer alignments. A curve can be defined so that protein pairs falling above the curve always are homologous proteins. Around the curve, the number of unrelated pairs rapidly increases, and below the curve most of the protein pairs are not related at all. The equation used in this thesis for the HSSP-curve is [97]:



(a) (b)

FIGURE 17: (a) The HSSP-curve, which is the border to the so called twilight zone. Marked with a red cross is the position in the plot of the two DEATH domains shown superimposed in (b), together with the corresponding structural alignment. When the percentage sequence identity is plotted to the length of the alignment of two protein sequences, related proteins fall above this curve. The further into the twilight zone one gets, the less likely it is that the two proteins are related.

$$p^I(L,n) = n + \begin{cases} 100 & for \quad L \leq 11, \\ 480 \cdot L^{-0.32 \cdot \left(1+e^{-L/1000}\right)} & for \quad 11 < L \leq 450, \\ 19.5 & for \quad L > 450. \end{cases} \tag{7.1}$$

Here, $p^I(L,0)$ is the cut-off percentage of residues identical over an alignment length of L residues that is required for concluding that two proteins are homologous, and n is the distance, in percentage points, from the curve. The equation is based on the HSSP-curve originally defined by Sander and Schneider [121], whose principal functional dependency was later shown to follow from statistics [2]. The parameters of the equation are visually selected so that it excludes most false positives, i.e. most pairs falling above this curve really are related. Two constraints are used; to reach 100% at an alignment length of 11, as shorter alignments do not reveal much about structural similarity, and to level out at around 20% sequence identity for longer alignments.

The equation implies that for short alignments the sequence identity has to be very high for the two sequences to be considered related, while for longer alignments even quite low percentage identities are significant. For building the saHMMs we use $n = 0$, except in a comparison of the saHMMs resulting from $n = 0$ and $n = -10$ (see Section 7.1.6).

Pairs of proteins falling under the HSSP-curve are most likely not related. However, there exist some pairs in this area that are related, as illustrated by the two DEATH domains shown in Figure 17(b). These domains are structurally related, even

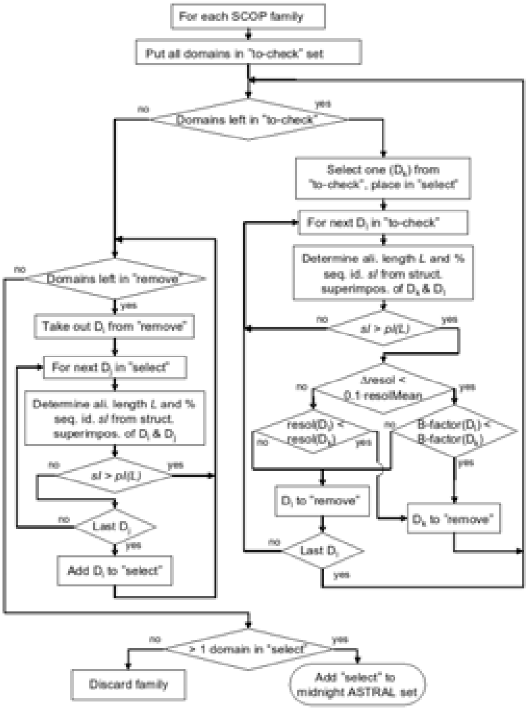

For each SCOP family
Put all domains in "to-check" set
no
yes
Domains left in "to-check"
Select one (D_k) from "to-check", place in "select"
For next D_i in "to-check"
Determine ali. length L and % seq. id. sI from struct. superimpos. of D_k & D_i
no
yes
sI > pI(L)
Δresol < 0.1 resolMean
no
yes
resol(D_i) < resol(D_k)
no
yes
no
B-factor(D_i) < B-factor(D_k)
yes
D_i to "remove"
D_k to "remove"
no
Last D_i
yes
no
Domains left in "remove"
yes
Take out D_i from "remove"
For next D_j in "select"
Determine ali. length L and % seq. id. sI from struct. superimpos. of D_i & D_j
no
yes
sI > pI(L)
no
Last D_j
yes
Add D_i to "select"
no
yes
> 1 domain in "select"
Discard family
Add "select" to midnight ASTRAL set

FIGURE 18: A flowchart showing our procedure for selection of representative protein domains from each SCOP family. Each member in a family is compared to each other member, by constructing a pairwise structural alignment. The length of the structure alignment and the resulting pairwise sequence identity is calculated, and if the numbers fall above the curve in Equation (7.1), i.e. the two proteins are too similar, one of them is removed. The protein kept is the one with the highest (best) resolution, or, if the resolutions are similar, the protein with the best mean B-factor.

though their mutual sequence identity is very low. One motivation for this thesis is to be able to detect such related pairs that fall under the curve.

Our procedure to select representatives for each family is illustrated in Figure 18. The representatives for each family are chosen by taking all proteins belonging to the same SCOP family and comparing them pairwise. For each pair of domains, a structure alignment is constructed, and the resulting alignment length and pairwise sequence identity is calculated. The number of structural comparisons is limited by removing one of the domains in a pair from further consideration, if the structure alignment reveals a sequence identity above the identity cut-off for that alignment length, as defined in Equation (7.1). The domain with the best resolution is kept, if one of them is solved at a higher resolution than the other. In case the resolutions are within 10% of the mean value of the two resolutions, the protein domain with the best mean B-factor is kept. The mean B-factor is calculated as the average of the temperature factors, or B-factors, of all C_α atoms in a domain, and is an additional measure of the quality of the structure. If the mean B-factors are equal, one domain is chosen randomly. To guarantee high quality structures for the structural alignments, only X-ray structures with a better resolution than 3.6Å are chosen, and all structures with worse resolutions, or determined using NMR or any other technique, are discarded. In case the structure alignment fails, the two domains are treated as very similar, in order to remove one of them and thus avoid problems to align the complete set of selected domains in later steps.

After going through the first round of selection, all removed protein domains are checked against all left, to insure that only sequences with too high sequence identities are discarded. The rationale behind this second comparison is that in the process of removing proteins, it is possible that a domain A is removed due to high sequence identity to domain B. If B later is removed due to high identity to domain C, it might be the case that A and C have a mutual sequence identity below the threshold. Hence, A and C must be compared, and in case the identity is equal to or less than $p^I(L,0)$ both domains should be kept.

The set of selected protein domains is named the *midnight ASTRAL set*, as we use domain structures obtained from the ASTRAL compendium (Section 4.3.1) that fall within the “midnight zone”, i.e. that have mutual sequence identities below the twilight zone.

The creation of the midnight ASTRAL set is treated in Papers I and III. In Paper III, a second algorithm for selecting representative domains is evaluated. In this second algorithm, all-against-all pairwise structure alignments are constructed within each family. However, as the number of pairwise comparisons grows quadratically with the number of domains in a family, we find that it is not feasible to use this algorithm for regular updates of the collection of saHMMs.

7.1.3 Construction of Structure-Anchored Sequence Alignments

Initially, we chose to use the STAMP software [118] (Section 6.3) for the construction of structure alignments. STAMP is used in Paper II, which is chronologically the first paper treating the saHMMs. The reason for choosing STAMP was that it produced the best sequence alignments among the software tools tested. The first choice was MAPS (G. Lu, personal communication), since this method superimposes multiple structures simultaneously. MAPS makes multiple structure alignments based on the same ideas as the pairwise method TOP (protein TOPological comparison) [91], which represents the secondary structure elements as vectors. Based on these, TOP makes a first fit, which is iteratively refined using rigid body transformations. MAPS starts with the pairwise alignments produced by TOP and minimizes the total RSMD between all structures. MAPS produces very nice superimpositions when looking at the structures in 3D. However, it only presents very short stretches of aligned residues, those that are really close in space, and therefore the method was abandoned. STAMP produces longer sequence alignments, and also has the benefit that the output can be easily parsed to a format suitable to construct HMMs using HMMER (see Section 7.1.4).

To construct a multiple structure alignment, STAMP needs an initial alignment to start from. We use SCAN, included in the STAMP package, which generates an initial alignment by scanning all structures to align against a given template structure, generating a multiple alignment from the resulting pairwise alignments. We use the domain that is of median length as template structure. In the few cases where STAMP nevertheless fails to align the domains in the family, we chose to use MAPS to generate a better initial alignment in the form of superimposed structures.

Until recently, there did not exist many publicly available software tools for multiple alignment of protein structures. The only real alternative to STAMP and MAPS was the multiple version of SSAP [142], which was not readily available to us. However, in the last few years several new methods to produce structure alignments have been presented.

Comparison of multiple structure alignment methods

In order to evaluate which multiple structure superimposition program best suits our purpose, we compare the quality of the saHMMs resulting from STAMP, MUSTANG [78](Section 6.4) and MultiProt [126](Section 6.5). For the comparison we select eight of the larger SCOP families, two from each of the four major SCOP classes. The selected families are listed in Table 7.1, and each has at least ten representatives in the midnight ASTRAL set constructed from SCOP 1.69 using STAMP and SCAN as described above.

We find that the choice of structure alignment program substantially influences the quality of the resulting saHMMs (see Table 7.1). Judging from the eight example families, MUSTANG performs slightly better than MultiProt. The saHMMs generated using MUSTANG find more family members than those based on MultiProt, at the cost of a few more false positives. Both methods clearly outperform STAMP. Moreover, MUSTANG can produce structure-anchored sequence alignments in multiple sequence format, msf, an accepted input format for HMMER, which makes it ideal for our automated saHMM construction pipeline. We therefore decided to replace STAMP with MUSTANG for all further structure superimpositions and saMSA extraction, including the results presented in Papers I, III and IV.

7.1.4 Construction of Family saHMMs

Only SCOP families with two or more representatives in the midnight ASTRAL set are used, since at least two structures are needed to construct an alignment. We call the representatives selected for a given family the *saHMM-members* of that family, as they are used to construct the saHMM. The saHMM-members in each family are structurally aligned as described in the previous section. The resulting structure anchored multiple sequence alignments are thereafter used to produce the final saHMMs with HMMER2.2g (http://hmmer.janelia.org/), using standard parameters. These parameters are derived to work well in most cases, and to optimize them individually for

SCOP id/ sunid	Domain name	Nr of family members	STAMP	Mustang	MultiProt
a.1.1.2/ 46463	Globins	973	958 0	973 2	973 0
a.3.1.1/ 46627	Monodomain cyto-chrome c	210	70 0	209 53	126 10
b.1.1.1/ 48727	Ig V set domains	1691	303 0	1685 34	1685 17
b.60.1.1/ 50815	Retinol binding protein-like	172	150 0	168 0	167 0
c.1.8.3/ 51487	Beta-glycanases	336	287 0	306 23	235 0
c.37.1.1/ 52541	Nucleotide and nucleoside kinases	244	134 0	234 4	230 0
d.108.1.1/ 55730	N-acetyl transferase, NAT	98	91 0	97 1	91 0
d.169.1.1/ 56437	C-type lectin domain	243	144 0	243 0	243 0

TABLE 7.1: Comparison of structure alignment methods. For each family, the upper value is the number of family members found and the lower value is the number of false positives.

each family is outside the scope of this thesis (see also Section 8.8).

In HMMER2.0 and later, Dirichlet mixtures are used by default for prior information. The mixtures make the models perform better when using only very few sequences and, consequently, not much sequence information is available. The Dirichlet mixtures improve the model's ability to recognize remote homologues (e.g., [22]) by giving each match state emission probabilities with the most likely probability distribution, based on the observed data (see Section 5.3.4). One of the reasons for the success of the saHMMs is the use of Dirichlet mixtures, which allow us to use very few sequences as the base for our models.

The type of HMM was chosen to be optimal to find alignments and/or hits that are global with respect to the HMM and local with respect to the query sequence, i.e. to match the complete saHMM, but allowing matches to only parts of the sequence. All HMMs are calibrated to obtain fitted E-values (see Section 5.3.2)

Using MUSTANG, we were able to construct saHMMs for about 30% of the families in SCOP version 1.69, covering 65% of the SCOP domains belonging to true classes. Due to the exponential increase of deposited 3D structures, the number of saHMMs is likely to increase, which will make the saHMMs cover more of the protein space. Also, as the number of structures known for each protein family grows, more members can be included in the individual models.

7.1.5 Alternative Implementations

In our implementation, we use the SCOP classification (see Section 4.3) to separate protein domains into families. SCOP is a highly reliable classification, since it is manually curated by experts, and therefore is a very good base for the HMMs. However, this is also the drawback of using this classification. The existence of the database relies on a few people, and the inclusion of new protein structures in the database cannot be done immediately. There exist some automatically created databases, but the exact classification of domains depends on the method used. The most natural alternative to the SCOP classification would be to use CATH, a similar database that is built on semi-automatic clustering of the proteins (see Section 4.4). Using CATH would make the method less dependent on A. Murzin and co-workers, who runs SCOP, but on the other hand SCOP is usually seen as a more reliable classification of protein structures because of the human expertise. This far, CATH has been less straightforward to use, and SCOP is the commonly used database in similar studies.

We chose to use the family level in the SCOP classification as the base for our HMMs. As the majority of our false matches are within the correct superfamily, the accuracy of the saHMMs might increase by working on the superfamily level, especially by pooling the results of all families in the same superfamily. In Paper I, this approach is investigated. We find that both the coverage and the accuracy of the saHMMs increase, with a slightly more marked increase in accuracy, when pooling the results on the superfamily level. The accuracy approaches 100%, i.e. the number of false hits is essentially zero. However, as we know that the vast majority of the matches are correct on the family level, we decided against combining the saHMMs into superfamilies, and instead report the results on the family level. This strategy has the advantage that the structural and functional information is more specific, at the same time as we know that in case there are false matches, they are most likely within the correct superfamily.

The structure-based sequence alignment can be extended to include sequences for which the structures are not determined, but which are similar in sequence to one or more of the saHMM-members. In that case, each of the saHMM-members would be searched independently against some sequence database using BLAST [5] or some other sequence alignment tool. The so found sequences would then be aligned by sequence to the saHMM-member used as query, and consequently to the other saHMM-members through the structure-anchored alignment. An extended structure-anchored multiple sequence alignment would give the saHMM more family specific sequence information, and decrease the need for general prior information. However, it is not clear how many additional sequences to add for each saHMM-member, how much the number of sequences added per saHMM-member might differ without introducing bias, which we take great care to remove, or what is a reasonable level of sequence similarity for the additional sequences. For these reasons, no extension is made of the multiple structure alignment, which makes the saHMMs well balanced with respect to the sequence variability within the family. In Paper IV, we show that also a very limited number of saHMM-members often result in excellent performance of the saHMM. It is even the case that limiting the number of saHMM-members further can result in better-performing saHMMs. From Papers I, II and III it is clear that the saHMMs, which are built from a low number of carefully selected protein domains, for some families outperform the HMMs in Pfam, which are built from alignments that include a large number of similar sequences.

7.1.6 Performance Evaluation of the saHMMs

The evaluation of the saHMMs is largely based on matching all sequences in a test set against the collection of saHMMs, each saHMM representing one protein domain family, in order to assign families to the query sequences. In addition, each saHMM is matched to the set of sequences in order to evaluate its ability to accurately locate family members. As the test set we construct a subset of SCOP, containing all domain sequences from families with an saHMM, except for the saHMM-member sequences.

The performance at a given E-value threshold e can be evaluated with respect to the following two criteria: the *coverage*, which is expressed as the percentage of all sequences in the test set that are matched with the correct saHMM with an E-value less than or equal to e, and the *accuracy*, which stands for the percentage of all hits with an E-value of at most e that are correct. We count as correct hits, also called true positives (*tp*) matches, those between a sequence and an saHMM from the same family. All other hits are considered as false positives (*fp*).

In order to take both the coverage and the number of false positives into consideration in a single score, we introduce the Family Identification score, *FI-score*. The FI-score is calculated as $(tp - fp)/N$, where N is the total number of sequences that should be matched. The FI-score can be at most 1.0, for perfect performance. However, in case the number of false positives exceeds the number of correct matches, the score may become negative. This makes the FI-score penalize false positives more than other scores, such as the F-measure [88]. The F-measure is the weighted harmonic mean of the coverage and accuracy, and is often used for binary classification problems.

The performance is also evaluated by plotting the number of Errors Per Query (EPQ) versus coverage, at a range of E-value cut-offs e. The EPQ at a given e is calculated as the total number of false positives divided by the total number of queries. In the case of sequence searches versus saHMMs, the number of queries is equal to the number of sequences used for searches. In the case of searching saHMMs versus sequences, the number of queries corresponds to the number of saHMMs, in other words, the number of families. Note that for these searches, an error per query corresponds to the number of false positives per domain family, and not per sequence. As the SCOP families in the test set harbour on average 53 sequences, this means that an EPQ of one corresponds to one false positive per 53 sequences. The coverage is in both cases calculated as described above. The advantage of these graphs over Receiver Operating Characteristic (ROC) plots [157], [61], which are often used in medicine, is that the EPQ versus coverage plots are better suited to visualize a high degree of accuracy and a vast background of non-homologues, while they communicate essentially the same information as the ROC curves [21].

Ability to Identify Family Relationships

When the sequences in the test set are used to search against the collection of saHMMs with an E-value cut-off of 0.01, we find that the coverage exceeds 96% and that the hits are highly family specific, with an accuracy of about 95%. Counting only the highest scoring hit for each search gives 98.5% coverage and an accuracy of 99.2%. Also the EPQ versus coverage plots show that the test set can be matched to the saHMMs with few errors per query and very high coverage.

When instead searching with the saHMMs against the test set, and using an E-value cut-off of 0.1, both the coverage and accuracy is 95%. Of the false positive hits, more than 99% are matches within the correct superfamily.

The performance of the saHMMs is compared to the results obtained using PSI-BLAST (see Section 5.2), which is a common approach to locate sequences belonging to the same family as the query. For the comparison, all sequences in the midnight ASTRAL set are used, one at a time, as queries in PSI-BLAST searches. First, PSI-BLAST (blastpgp v. 2.2.13) is run for five iterations versus the NCBI non-redundant database, nr, (downloaded March 30, 2006). The resulting position specific scoring matrices, PSSMs, one for each saHMM-member, are thereafter used to search SCOP version 1.69. The results obtained for all saHMM-members within the same family are pooled, in order to produce results comparable to those from searches of saHMMs versus SCOP sequences. As the pooled PSSM matches can contain duplicates, we consider only non-redundant matches. In case of two or more hits to the same sequence, we keep the match with the lowest E-value.

The graph in Figure 19(a) shows that the saHMMs are able to identify family members within SCOP with few errors per query and high coverage. For example, the coverage is about 91% at an EPQ of two, which corresponds to an accuracy of 96

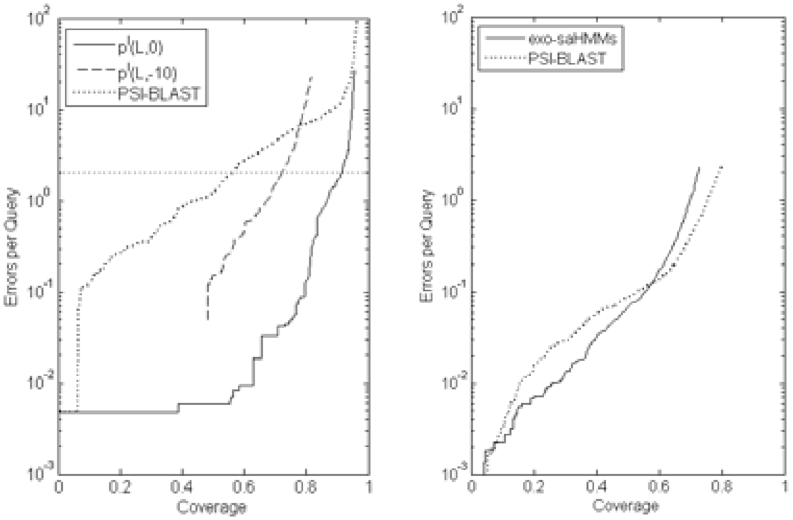
The results demonstrate that at a given coverage, the saHMMs based on $p^I(L,0)$ are able to accurately identify family members with clearly less errors per query compared to PSI-BLAST PSSMs. It seems that a few diverse, well aligned sequences can perform better than the PSSMs built from a large number of sequences without any restrictions on diversity.

Performance of saHMMs based on $p^I(L,-10)$

We constructed a set of saHMMs based on $p^I(L,-10)$ in order to investigate the effect of an even lower cutoff on sequence identity within the saHMM.

In Table 7.2, results are shown for both collections of saHMMs, and for searches both with sequences versus HMMs and HMMs versus sequences. The table shows that the saHMMs based on $p^I(L,-10)$ are more accurate, with less errors per query, but at the cost of a reduced coverage. This can also be observed in the graphs in Figure 19(a), where the set of saHMMs based on $p^I(L,-10)$ at a given EPQ obtain lower coverages than the saHMMs based on $p^I(L,0)$.

These results clearly show that, even though the mutual sequence identities of the domains used to build the saHMMs are exceedingly low, the models correctly describe the essential characteristics of the family and are able to identify the majority of the family members with high accuracy.



(a)

(b)

FIGURE 19: EPQ versus coverage plots. Note that the scale on the y-axis is logarithmic. (a) Results from searches with saHMMs against the test set, using E-value cut-offs between zero and ten. The solid and the dashed curves show the results for the selection based on $p^I(L,0)$ and $p^I(L,-10)$ respectively, whereas the dotted curve illustrates the results of searches with PSI-BLAST PSSMs against SCOP sequences. (b) Results from searches with low identity sequences versus exo-saHMMs. The plot also contains the results obtained with PSI-BLAST PSSMs (dotted line).

Performance of saHMMs built from two saHMM-members

At present, the number of sequences per SCOP domain family in the midnight ASTRAL set is not evenly distributed. In fact, more than half of the families are represented by only two saHMM-members. One would expect that the performance of an saHMM is affected by the number of constituent saHMM-members, and that the saHMMs built from two saHMM-members do not perform as well as saHMMs constructed from more sequences. In the following, we analyze how these saHMMs perform, considering the limited amount of family specific information contributed by only two sequences. Of all the saHMMs built from two sequences, we consider only the 448 that represent SCOP families with three or more members, since they allow us to find at least one additional family member. When the sequence identity of the two sequences used to build an saHMM is plotted as a function of alignment length, one notices that many pairs have sequence identities well below the threshold $p^I(L,0)$ and are distributed over alignment lengths of 300 residues or less (Figure 20(a)). Surprisingly, most of the saHMMs built from two sequences are able to score significant hits to all of their family members. The ability to find family members is only marginally

		Nr of sequences, resp. saHMMs	accuracy (%)	coverage (%)	fp w/i correct superfamily (%)	hits outside correct superfamily † (%)	sequences resp. saHMMs w/o hit (%)
test-set vs. saHMMs	$p^I(L,0)$	40877	94.6	96.3	99.4	0.03	3.6
	$p^I(L,-10)$	28745	98.5	71.7	99.3	0.1	28.2
saHMMs vs. test-set	$p^I(L,0)$	831	94.9	95.0	99.5	0.03	0.6
	$p^I(L,-10)$	491	98.3	65.5	99.4	0.01	1.2

†Here we have calculated the percent of the false positive matches, fp, that are hits within the correct superfamily.

TABLE 7.2: The performance of the two sets of saHMs. $p^I(L,0)$ represents the saHMMs based on the midnight ASTRAL set, $p^I(L,-10)$ represents saHMMs from the set based on $p^I(L,-10)$.

affected by the percent identity of the two sequences used to construct the saHMM. This can be seen in Figure 20(b), where we plotted the percent sequence identity against the coverage. The histogram along the right hand axis corresponds to the distribution of sequence identities for saHMMs that find 100% of their family members in SCOP test-set. When we plot the percentage of found family members as a function of alignment length (Figure 20(c)) we can not detect any correlation between the alignment length and the coverage. The histogram on the top of the graph shows the distribution of alignment lengths for saHMMs that are able to detect 100% of their family members. As can be seen in Figure 20(d) the saHMMs either find all their family members, or, if not all members are found, no false positive hits are obtained. In addition, 86.4% of the saHMMs find all their family members with at most 0.5% false positives. The histogram at the top shows the distribution of false positive hits for saHMMs that find 100% of their family members. Note the peak at the zero percent false positives, which, due to space limitations, is not to scale.

Performance on Low Sequence Identity Homologues

The way the midnight ASTRAL set sequences are selected implies that each sequence in the test set has a pairwise sequence identity above $p^I(L,0)$ with respect to at least one sequence in the midnight ASTRAL set. In order to investigate the ability of the saHMMs to match low sequence identity homologues, whose identity is equal to or less than $p^I(L,0)$ when compared to the saHMM-members, we construct exclude-one-saHMMs, exo-saHMMs. For families with at least three saHMM-members, exo-saHMMs are generated by excluding one representative sequence at a time and building new saHMMs from the structure alignment of the remaining domains. In this way, we obtain a collection of n exo-saHMMs for a family with n saHMM-members.

The excluded sequences are used, one at a time, to query the collection of saHMMs, where the full family saHMM is exchanged with the exo-saHMM that lacks that query sequence. The search results show that 66% of the excluded sequences are matched

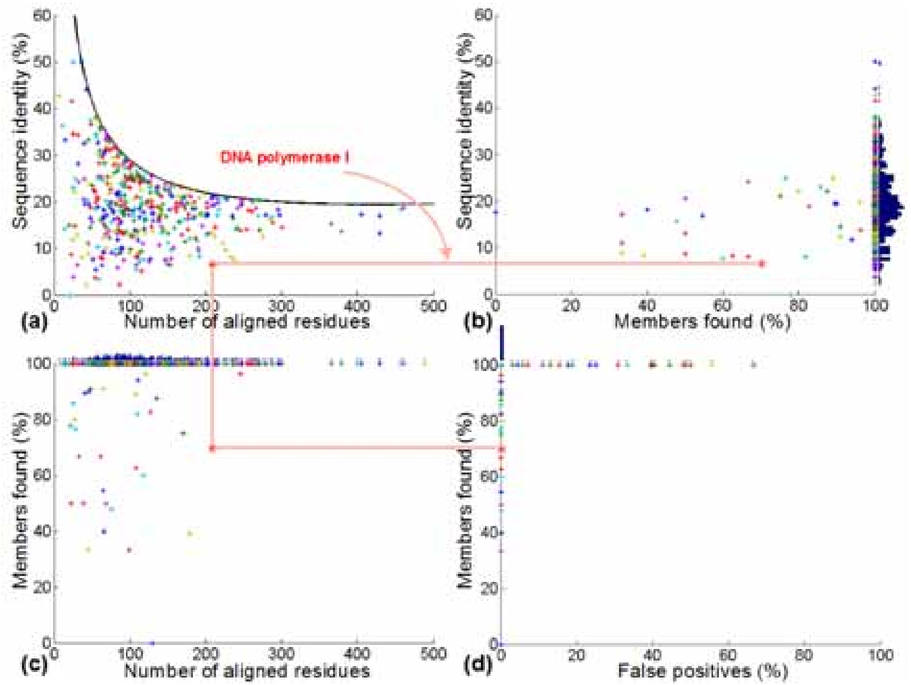


FIGURE 20: Results for families with two saHMM-members. Each star represents one of the 448 domain families with two saHMM-members and at least three family members in SCOP 1.69. (a) Percent sequence identity as a function of alignment length, together with the limiting $p^I(L,0)$ curve marked in black. (b) Percent sequence identity as a function of percent coverage. (c) Percent coverage as a function of alignment length. (d) Percent coverage as a function of percent false hits. A star in one subfigure has a corresponding star in the other subfigures, coloured the same way and representing the same family. As an example, the four stars representing the family of DNA polymerase I domains are connected by pink lines. Note that, in the majority of the cases, the coverage is high at the same time as the percent of false positives is low.

to the corresponding exo-saHMM, when considering top hits. The accuracy of these matches is 76%. If we tolerate matches within the correct superfamily, the accuracy increases to 83%. Similarly, we evaluate the ability of each of the exo-saHMMs to find the missing sequence among the excluded sequences. The results vary significantly among families. For some families, all of the exo-saHMMs find their excluded sequence, while for other families none do. In total, 30% of the sequences are identified by the exo-saHMMs from which they were excluded, with an accuracy of 78%. Of the false positive matches, 93% fall within the correct superfamily. These results show that also low sequence identity homologues, which are very hard to identify, can be detected with high accuracy.

For comparison, we test the ability of PSI-BLAST to correctly assign a sequence to its family, even at low sequence identity. As before, all sequences in the midnight ASTRAL set are used as queries. In this case, a query sequence is counted as assigned to the correct family if it obtains a match to at least one other family saHMM-member. Matches to sequences outside the correct family, but within the midnight ASTRAL set, are counted as false positives. It should be noted that PSI-BLAST PSSMs have an advantage over the exo-saHMMs in this comparison. Since the PSSMs are derived from searches in the NCBI's nr-database, they can contain sequences with a mutual identity exceeding $p^I(L,0)$ when compared to the query sequences. In this way, the PSSMs might contain bridging sequences, with a sequence identity above $p^I(L,0)$ to both the query sequence and another saHMM-member within the same family, which can facilitate the PSSMs' ability to find these sequences. In Figure 19(b), the EPQ values are plotted versus the coverage for searches of low identity sequences versus exo-saHMMs. The plot also contains the results obtained with PSI-BLAST PSSMs. For proper sequence annotations it is important to consider only reliable matches, i.e. matches with a low error rate. As can be seen in the figure, the exo-saHMMs have fewer errors per query than the PSSMs up to about 58% coverage, where the two curves cross at an EPQ value of 0.12. Below this value, the exo-saHMMs achieve a higher coverage, at a given EPQ, than the PSI-BLAST PSSMs. This demonstrates that the exo-saHMMs perform better at a low error rate, despite the inbuilt advantage of the PSSMs.

Performance Test on New Sequences

We also assess the ability of the saHMMs to assign the correct domain family memberships to newly sequenced proteins. This was done for two sets of saHMMs, first a collection based on SCOP 1.61 and STAMP, then a collection based on SCOP 1.69 and using MUSTANG for the structure alignment. For the first collection of saHMMs, we used the domain sequences that are contained in SCOP 1.69 (released July 2005) but not in SCOP 1.61 (released Nov. 2002) as query sequences. For the second collection, we use the domain sequences that are present in SCOP version 1.71 but not in version 1.69 to search against the saHMMs. Counting only top hits, 74% of the sequences with an saHMM in the first collection obtained a correct match, with an accuracy of 94%. At a comparable E-value cut-off, the corresponding numbers are 85% coverage and 99% accuracy for the second collection. This shows that the saHMMs can be used to accurately identify the correct family relationships to new sequences. Even though the second set of sequences is smaller, it is also clear that the shifting to MUSTANG greatly improved the saHMMs.

Of the sequences for which our collections lacks an saHMM, in both cases about 98% do not find a match at all, and 2% find the correct superfamily. This demonstrates that our saHMMs are very domain family specific.

In order to evaluate the ability to detect low sequence identity homologues we select, for each domain family and for each collection of saHMMs, those sequences in the set of new sequences that have sequence identities equal to or less than $p^I(L,0)$ compared to the saHMM-members. Even though the sequence identity is very low, 62% obtained top hits to the correct family saHMMs in the first set, a number that increase to 69% for the second set.

Comparison to Pfam

The two sets of saHMMs are compared to the corresponding releases of Pfam [137] (see Section 7.1.9). The saHMMs based on SCOP 1.61 are compared with Pfam_ls HMMs version 7.8 (released November 2002), and the saHMMs based on SCOP version 1.69 with the Pfam_ls HMMs version 19.0 (released Nov. 2005). The classification of domains in Pfam is not identical to that of SCOP. Therefore we have mapped the two different Pfam versions onto the corresponding SCOP versions. The relationships between corresponding families in the two databases are determined by finding the SCOP classification of PDB sequences that are part of Pfam-A seed alignments. Among the "new" sequences defined above, we select as queries those with both a HMM in Pfam and an saHMM. For the older collections of HMMs, based on the 2002 releases, the correct family relationships are detected as top hits for 83% of the sequences using the saHMMs and for 87% of the sequences using Pfam. The matches are not completely overlapping, and as many as 812 of the relationships detected by the saHMMs are not found by Pfam. Interestingly, 79 of these relationships are matches to low identity sequences, despite the possibility that some of the query sequences could have a sequence identity above $p^I(L,0)$ to Pfam-A seed sequences. Using the collections of HMMs based on the 2005 releases, the correct family relationships are detected as top hits for 94% of the sequences using the saHMMs and for 88% using Pfam. Hence, this time the saHMMs perform clearly better than the corresponding Pfam HMMs. One reason for this improvement in the saHMMs is the switch from STAMP to MUSTANG for the multiple structure alignments. Of the relationships missed by Pfam, 77 can be counted as hits within the midnight zone, since they have an identity of at most $p^I(L,0)$ compared to the saHMM-members.

HMM Logos [123] can be used to visualize the probability distributions within a HMM, including the amino acid probabilities and the probabilities of entering differnet states of the model. We use HMM Logos to compare the parameters of a number of saHMMs representing the V-set domain family (sunid 48727) with those of the corresponding Pfam HMM. The saHMMs are built from varying number of saHMM-members, but all represent the same family. We find that the pattern of conserved residues is very similar for the saHMMs and the Pfam HMM, despite the considerably lower number of sequences used when building the saHMMs, at most 28, compared to the 121 used for the Pfam HMM. Even the best performing saHMM of those built from just two saHMM-members show the same basic pattern as the other HMMs.

Using the saHMMs for Annotating Sequences

In order to show the use of the saHMMs for annotating protein sequences, we searched the National Center for Biotechnology Information, NCBI, for human proteins labelled 'unknown'. Of the 2905 sequences found (March 2006), 590 proteins can be matched to at least one saHMM with an E-value not exceeding 0.01. The classic Zinc-finger domain family receives with 196 hits by far the most matches, distributed over 31 different proteins. For 18 of these proteins, the NCBI annotation is incomplete in the sense that none or not all of the Zn-finger domains are identified in the sequence entry. For an additional seven proteins the NCBI annotation might not be correct, since we find hits with very low E-values for classic Zn-finger domains in sequence regions where the NCBI annotation suggests other domains.

When the search for ’unknown’ sequences was repeated in November 2007, the search resulted in 1986 sequences, of which 232 can be matched to at least one of the saHMMs. Also this time, the classic Zinc-finger domain family receive the most matches, and for 17 of the found Zn-finger proteins, the NCBI annotation is either incomplete or possibly incorrect.

Obviously, the saHMMs can be used to detect relationships missed by other methods.

Improving Performance by Combinatorial Selection

By examining the FI-scores for saHMMs built from varying numbers and combinations of saHMM-members within the same family, we find that it is not neccessarily the case that the saHMM built from the maximum number of saHMM-members performs the best. For a given number of saHMM-members, the performance can vary considerably depending on the combination of members used. We therefore introduce the concept of combinatorial selection, in order to increase the number of members found and reduce the number of false positive matches for the worst performing family saHMMs.

For families with saHMMs that find less than 65% of their respective family members, we build saHMMs with different number and combinations of saHMM-members, and select for each family the saHMM that achieves the highest FI-score. In this way, we are able to increase the average FI-score of these saHMMs from 0.298 to 0.649, with an increase in average coverage from 42% to 65%.

7.1.7 Other Approaches Using Structures and HMMs

The last few years, the concept of using structural information to improve hidden Markov models has been investigated in a number of ways. One of the first studies was performed by Al-Lazikani *et al.* [1], who use a structural alignment of SH2 domains, in combination with multiple sequence alignments of closely related sequences, to build a hidden Markov model. The HMM is capable of successfully detecting SH2 domains, which was encouraging for the further development of our approach. Later, Griffiths-Jones and Bateman [62] built HMMs from protein structure alignments obtained from the HOMSTRAD database [98], and compared these with HMMs built from sequence-based alignments of the same proteins. The authors find that although the structures improve the quality of the alignments, they do not significantly increase the ability of the resulting HMMs to find sequence homologues. No filtering of sequences is made based on sequence identity, meaning that their alignments contain sequences with identities covering almost the entire range up to 100%.

Recently, three more elaborate approaches have been presented. In the first study, HMMs are built based on structural alignments of sequences with less than 35% sequence identity within CATH superfamilies [130]. In the second study [122], sequences with BLAST E-values below 10^{-3} are structurally aligned within SCOP superfamilies. In both these cases, the structural alignments are extended with multiple sequence alignments of closely related sequences. Also in the third study [24] the

HMMs are based on SCOP superfamilies, but only using sequences with less than 10% sequence identity. All three studies compare the performance of the resulting structure-based HMMs to single-member HMMs, built from single seed sequences in an iterative approach. The conclusions are that structure improves alignment quality, and might help in detecting very remote homologues, although no overall improvement in homologue detection can be found. In particular, Casbon and Saqi [24] find that at low levels of false positives, the structure-based HMMs are able to detect more relationships than single-member HMMs.

The work presented in this thesis differs from the work presented above in that we use the more specific SCOP families instead of superfamilies. We also use strict selection criteria in order to include only sequences with high resolution structures and with low mutual sequence identities *within each family*. That is, we only consider the sequence identities of sequences participating in the same saHMM. We also use relatively few sequences in our alignments, only the selected low identity sequences, and make no extensions with closely related sequences. This makes the alignments well balanced with respect to the sequence variability within the family. The selected domains are simultaneously and multiply aligned, based purely on structure. A structure alignment of a few carefully selected sequences is apparently sufficient to construct an saHMM capable of recognising family members with high coverage and accuracy. In addition, we have made our saHMMs available for searching through the FISH server (Section 7.1.8).

The major ideas in the thesis were first presented in a technical report [145] included in the licentiate thesis by the author [144]. The thesis was presented in 2003, prior to the publication of the last three studies discussed above.

7.1.8 The FISH Server

From a user's perspective, the use of the saHMMs should be like entering a sequence into a black box, and out comes the name of the domain family the sequence most likely belongs to, some measure of the quality of the match, as well as a list of representative members of the family.

This "black box" is implemented in the FISH server, which is described in Papers II and III, and is found at http://babel.ucmp.umu.se/fish/. FISH stands for Family Identification using Structure-anchored HMMs. The organization of the FISH server input and results pages is outlined in Figure 21.

The FISH server enables the user to enter one or more query sequences and also provides the option to select an E-value cut-off for the results. The E-value corresponds to the probability that the similarity found is random. The closer to zero the E-value is, the less likely it is that the match is random, i.e., the more likely it is that the query sequence really is related to the sequences the saHMM is built from (see also Section 5.3.2).

Inside the "black box" the sequence is searched against the collection of saHMMs, using hmmpfam in the HMMER-package. In the server, the the most recent HMMER release, HMMER2.3.2, is used. The result of this search is a list of names of saHMMs that match the sequence, sorted in increasing order by the E-value of the match. The

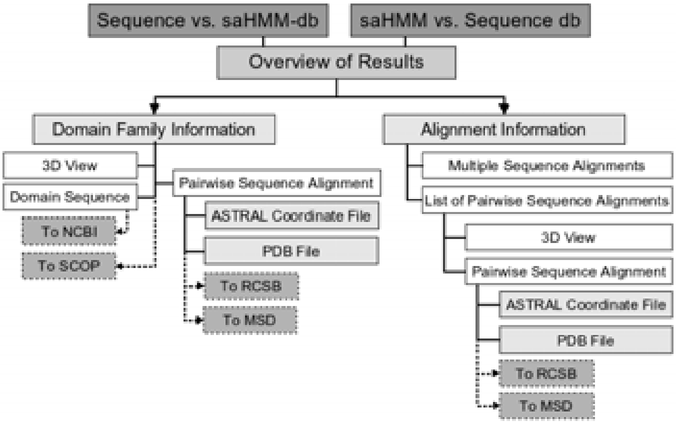
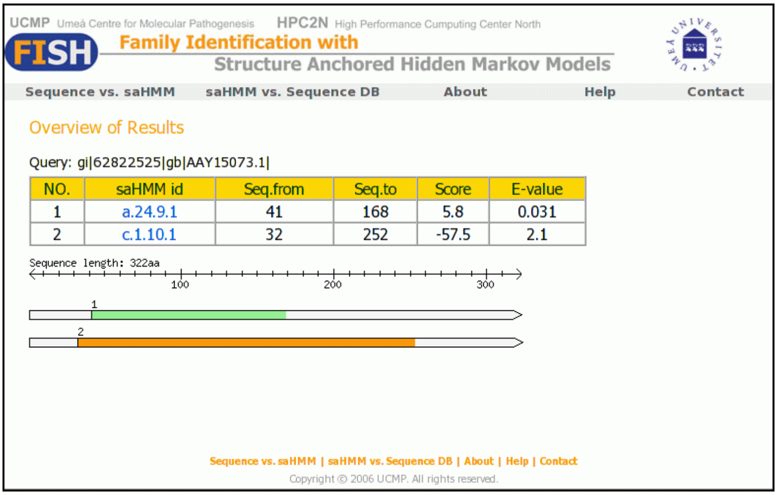


FIGURE 21: Organization of the input and result pages of the FISH server. The information included in the result pages is similar for a search with a query sequence versus the collection of saHMMs and for a search with an saHMM versus a sequence database. The information available can be roughly divided into domain family information (left branch) and alignment information (right branch). The domain family information includes SCOP classification, the sequences and 3D structures of the saHMM-members, and pairwise sequence alignments of the query to each member. The alignment information provides multiple and pairwise alignments of the query sequence to the saHMM-members and to a consensus sequence extracted from the saHMM. All alignments are anchored to the saHMM. Links are provided to relevant databases.

user is presented with a list of matches, up to the chosen E-value cut-off (see Figure 22(a) for an example). As proteins often contain more than one structural domain (see Chapter 2), one might expect several matches, to different parts of the query sequence. When selecting an entry from the list, family specific information for that match is displayed, see Figure 22(b). This information includes the name of the domain family and its place in the SCOP classification, the names of all saHMM-members, and details specific for this particular match, such as the percent sequence identity of the query sequence when it is aligned to each saHMM-member. It is also possible to view the structure of each saHMM-member in an interactive Java window. In addition, the user can view pairwise comparisons of the query sequence and each saHMM-member, as well as a multiple sequence alignment of the query and all saHMM-member sequences in different formats.

A second possibility is to use a single saHMM to search a sequence database, or a whole genome, looking for proteins that harbour a certain domain. The FISH server provides the option to select an individual saHMM for searching against a number of sequence databases. Currently, SWISS-PROT, TrEMBL, and the non-redundant database, nr, from NCBI are available for searching. In addition, the user has the option to upload his/her own sequence database, and query it with individual saHMMs.


UCMP Umeå Centre for Molecular Pathogenesis
HPC2N High Performance Cumputing Center North
FISH
Family Identification with
Structure Anchored Hidden Markov Models
Sequence vs. saHMM
saHMM vs. Sequence DB
About
Help
Contact

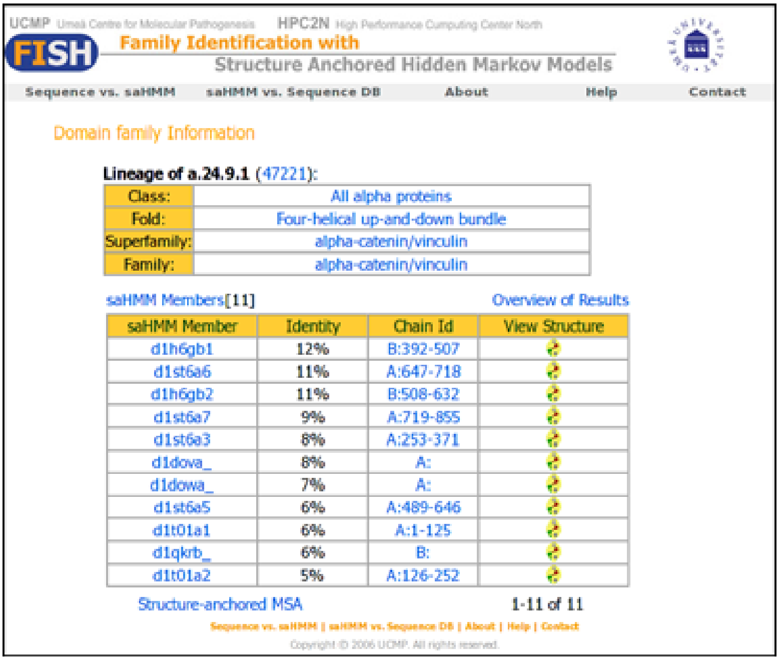
Overview of Results

Query: gi|62822525|gb|AAY15073.1|

NO.	saHMM id	Seq.from	Seq.to	Score	E-value
1	a.24.9.1	41	168	5.8	0.031
2	c.1.10.1	32	252	-57.5	2.1

Sequence length: 322aa
100
200
300
1
2
Sequence vs. saHMM | saHMM vs. Sequence DB | About | Help | Contact
Copyright © 2006 UCMP. All rights reserved.

(a)


UCMP Umeå Centre for Molecular Pathogenesis
HPC2N High Performance Cumputing Center North
FISH
Family Identification with
Structure Anchored Hidden Markov Models
Sequence vs. saHMM
saHMM vs. Sequence DB
About
Help
Contact

Domain family Information

Lineage of a.24.9.1 (47221):

Class:	All alpha proteins
Fold:	Four-helical up-and-down bundle
Superfamily:	alpha-catenin/vinculin
Family:	alpha-catenin/vinculin

saHMM Members[11] Overview of Results

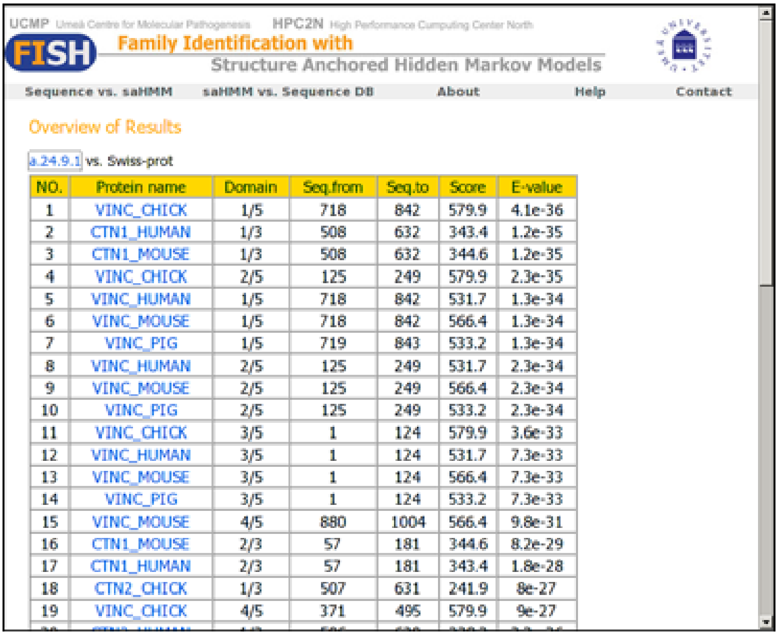
saHMM Member	Identity	Chain Id	View Structure
d1h6gb1	12%	B:392-507	●
d1st6a6	11%	A:647-718	●
d1h6gb2	11%	B:508-632	●
d1st6a7	9%	A:719-855	●
d1st6a3	8%	A:253-371	●
d1dova_	8%	A:	●
d1dowa_	7%	A:	●
d1st6a5	6%	A:489-646	●
d1t01a1	6%	A:1-125	●
d1qkrb_	6%	B:	●
d1t01a2	5%	A:126-252	●

Structure-anchored MSA 1-11 of 11

Sequence vs. saHMM | saHMM vs. Sequence DB | About | Help | Contact
Copyright © 2006 UCMP. All rights reserved.

(b)

FIGURE 22: (a) Example of results obtained through the FISH server when using a query sequence to search the collection of saHMMs. The coloured arrows show which parts of the query sequence that are matched. The colour of the arrow illustrates the significance of the hits, where green is significant and orange is not significant at all. In this example, the query sequence used is AAY15073, a human protein labelled ’unknown’ by the NCBI. (b) Family specific information obtained when clicking on the top match of the results shown in (a).


UCMP Umeå Centre for Molecular Pathogenesis
HPC2N High Performance Computing Center North
FISH
Family Identification with
Structure Anchored Hidden Markov Models
Sequence vs. saHMM
saHMM vs. Sequence DB
About
Help
Contact

Overview of Results

a.24.9.1 vs. Swiss-prot

NO.	Protein name	Domain	Seq.from	Seq.to	Score	E-value
1	VINC_CHICK	1/5	718	842	579.9	4.1e-36
2	CTN1_HUMAN	1/3	508	632	343.4	1.2e-35
3	CTN1_MOUSE	1/3	508	632	344.6	1.2e-35
4	VINC_CHICK	2/5	125	249	579.9	2.3e-35
5	VINC_HUMAN	1/5	718	842	531.7	1.3e-34
6	VINC_MOUSE	1/5	718	842	566.4	1.3e-34
7	VINC_PIG	1/5	719	843	533.2	1.3e-34
8	VINC_HUMAN	2/5	125	249	531.7	2.3e-34
9	VINC_MOUSE	2/5	125	249	566.4	2.3e-34
10	VINC_PIG	2/5	125	249	533.2	2.3e-34
11	VINC_CHICK	3/5	1	124	579.9	3.6e-33
12	VINC_HUMAN	3/5	1	124	531.7	7.3e-33
13	VINC_MOUSE	3/5	1	124	566.4	7.3e-33
14	VINC_PIG	3/5	1	124	533.2	7.3e-33
15	VINC_MOUSE	4/5	880	1004	566.4	9.8e-31
16	CTN1_MOUSE	2/3	57	181	344.6	8.2e-29
17	CTN1_HUMAN	2/3	57	181	343.4	1.8e-28
18	CTN2_CHICK	1/3	507	631	241.9	8e-27
19	VINC_CHICK	4/5	371	495	579.9	9e-27

FIGURE 23: Example of results obtained through the FISH server when searching an saHMM versus a sequence database. Here, the saHMM of SCOP family a.24.9.1 (sunid 47221), alpha-catenin/vinculin domains, is used to search SWISS-PROT.

For practical reasons, the size of the uploaded database is currently limited to 2 MB.

The output of such a search is similar to that from a search with a sequence versus the collection of saHMMs, except that instead of a list of families that most likely fits a query sequence, the user obtains a list of those protein sequences in the database that most likely belong to the family modelled by the saHMM, see Figure 23. For each match, the user is presented with the corresponding sequence entry. It is also possible to view sequence alignments of the matched sequence and the saHMM-members, both pairwise and multiple. The alignments are all anchored on the saHMM. Information about the saHMM used for searching, and the corresponding domain family, is also available.

The Architecture of the FISH Server

Much of the information provided through the FISH server, for example all information given about the domain families, is fixed, and can therefore be stored in a database. When a query is submitted by a user, the information relevant for the obtained matches is extracted from the database. An overview of the FISH server architecture is shown in Figure 24.

All information about individual saHMMs and saHMM-members, and the corresponding domain families, is imported from flat file databases and stored in relational

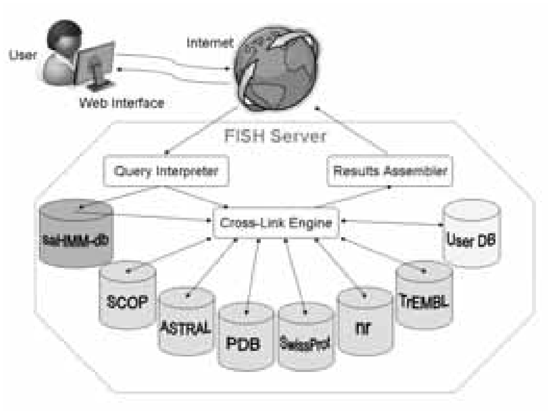


FIGURE 24: Overview of the FISH server architecture. A query is initialized by the user via the web interface. The query is processed by the query interpreter, using the collection of saHMMs. The cross-link engine integrates the results of the query with information from the associated databases [SCOP, ASTRAL, PDB, nr (NCBI), SWISS-PROT and TrEMBL]. The results assembler compiles the search results and presents them to the user via the web interface.

databases. We use MySQL[1], implemented on a Linux platform. The information is crosslinked for easy access to all relevant information about a given saHMM (see Figure 25). The user inputs a query through a web interface, which is written in perl, PHP, and JavaScript. A query interpreter analyses the input, and uses the collection of saHMMs to perform the requested searches. The cross-link engine then merges the results from the query with relevant information extracted from the associated databases. Finally, the results assembler presents the outcome of a search to the user through the user interface. The results are stored on the server for 24 hours, and can be sent to the user by e-mail in the form of a www-link.

A search with an saHMM versus SWISS-PROT takes anything from 15 minutes up to about nine hours, depending on the saHMM chosen. To search through TrEMBL, which is about ten times larger, takes considerably longer. As a service to the user, in order to minimize the waiting time, we have pre-calculated the searches of all saHMMs versus SWISS-PROT, TrEMBL and nr using an E-value cut-off of 100. When the user requests a search with a given saHMM and database, these results can immediately be extracted and presented up to the E-value choice of the user. The computations were done in parallel, by searching the databases with several saHMMs concurrently. Up to 20 processors were used in parallel on the HPC2N Linux cluster Seth.

[1] http://www.mysql.com/

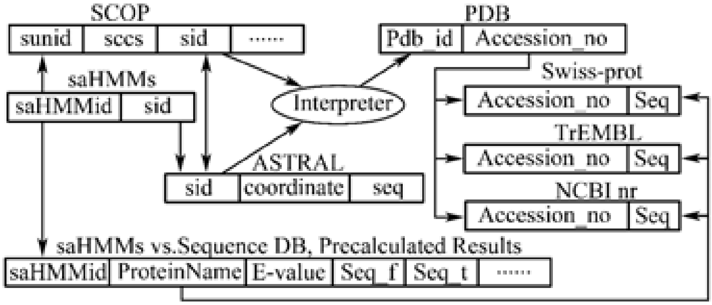


FIGURE 25: Schematic view of the database cross-linking used in the FISH server.

7.1.9 Related Internet Resources

In this section, some available web resources based on HMMs are briefly presented. Pfam is a collection of sequence-based HMMs that is commonly used to find sequence family relationships. In SUPERFAMILY, the HMMs are based on sequence domains with known structure, however, they use single-sequence HMMs and classify sequence domains on the SCOP superfamily level. PALI is a collection of structure alignments of SCOP families, from which phylogenetic trees and PSSMs are derived. The collection is mentioned here since they use SCOP families and the same structure alignment tool as we do. In all cases, HMMs are built from sequences with no cut-off on sequence identity. All groups also try to include as many sequences as possible in their alignments.

Pfam

Pfam[2] [137] is a semi-automatically created database of multiple sequence alignments of protein domain families. The families are defined based on clear common ancestry and sequence similarity. The database is purely sequence-based, but is mentioned here since it uses HMMs to define families and construct alignments. Pfam contains two sets of alignments with corresponding HMMs; PfamA and PfamB.

The base of Pfam is a collection of high quality seed alignments. The initial members of a seed alignment are collected from a number of sources, including structural alignments, SWISS-PROT (see Section 4.1) and published alignments. The sequences are aligned by an automatic alignment method, most often ClustalW [143], and checked manually. From each seed alignment a HMM is built, which in turn is used to search a non-redundant collection of sequences from SWISS-PROT and SP-TrEMBL (see Section 4.1), called Pfamseq, for additional members. The seed is updated with selected sequences until all known members are found. These are aligned to the HMM to construct a full alignment of the family. Where available, structural information is used to ensure that each Pfam family corresponds to just one structural domain. The seed alignment, the HMM built from it, the full alignment and some

[2] http://pfam.cgb.ki.se

annotation and cross-references to other families make up Pfam-A. Pfam-B is a less reliable collection of multiple sequence alignments, initially constructed by automatically clustering the rest of pfamseq, i.e. all sequences not included in Pfam-A. In later releases [13], [14], Pfam-B has been constructed from all protein domain families in the ProDom database, not included in Pfam-A. ProDom is an automatically generated database of protein domain families [29]. The latest addition to Pfam is the clustering of Pfam families into clans, based on related structures and functions, as well as similarities in the respective HMMs [46].

Superfamily

SUPERFAMILY[3] [57], [95] is a library of HMMs representing SCOP superfamilies. The HMMs are built from seed sequences with less than 95% sequence identity, based on pairwise BLAST alignments. Each seed sequence is used to construct one HMM, using WU-BLASTP (http://blast.wustl.edu) to search a non-redundant sequence database to obtain an initial alignment. A HMM is built from this alignment, and the HMM is searched against a sequence database to obtain additional related sequences. After four iterations, a final HMM is built using the SAM programs [73], which is a suite of programs similar to HMMER. Each seed sequence gives rise to one HMM, resulting in one or more HMMs representing the same superfamily. The HMMs are available for searching on the server, where sequence annotations of a number of genomes are provided as well.

PALI

PALI[4] (Phylogeny and ALIgnment of homologous protein structures) [10] is a database of structure-based sequence alignments and phylogenetic trees for each SCOP family. For each family, the database provides a multiple structural alignment, all possible pairwise alignments, and two phylogenetic trees; one based on structure similarity and the other on similarity of aligned residues. The structural alignments were previously constructed using STAMP ([118], Section 6.3), but are now generated by MUSTANG ([78], Section 6.4)

Also, in later versions of PALI [58], sequences homologous to the members are aligned to the family, and Position Specific Scoring Matrices (PSSMs) are constructed based on these enriched alignments. The alignments and PSSMs are available in the database.

7.2 Secondary Structure HMMs (ssHMMs), Paper V

At the second CASP process in 1996 (see Section 3.7), five groups were selected for the best performance in the threading category. Among these groups, one used predicted secondary structures [115], another used hidden Markov models [76], and a third group used a hidden Markov model that only used secondary structure and matched a predicted secondary structure against this model [33]. The success of using

[3] http://supfam.org/SUPERFAMILY/

[4] http://pauling.mbu.iisc.ernet.in/ pali/

HMMs and the idea of using predicted secondary structures made it a natural step to try to combine these two methods, as we have done in Paper V.

We constructed hidden Markov models that use both the amino acid sequence and secondary structure information simultaneously, so called secondary structure HMMs (ssHMMs).

7.2.1 Implementation of the ssHMMs

The ssHMMs match and insert states have, in addition to the emission probabilities for amino acids (Section 5.3), also associated an emission probability for secondary structures. This means that even though the actual sequence symbol does not match the HMM, a position can obtain higher probability if the secondary structure matches that of the model. The secondary structures for query sequences, which of course are not known, are predicted using some secondary structure prediction method before the search.

In order to implement the ssHMMs, the program package HMMER[5], version 1.8.4, was modified to include secondary structure information both when building a HMM for a protein family, and when matching an amino acid sequence to a HMM.

In an ordinary profile HMM, a sequence $s = x_1 \ldots x_L$ following the path $q = q_0 \ldots q_N$ through model μ has the probability

$$P(s \mid q,\mu) = \prod_{i=1}^{N+1} T(q_i \mid q_{i-1}) \prod_{i=1}^{N} P(x_{l(i)} \mid q_i). \tag{7.2}$$

Here, $T(q_i \mid q_{i-1})$ is the probability of a transition from state q_{i-1} to q_i, $l(i)$ is the sequence index of amino acid x in state q_i, $P(x_{l(i)} \mid q_i)$ is the probability of observing amino acid $x_{l(i)}$ in state q_i, and N is the number of states in the path. See also Section 5.3.2.

The ssHMM has an extra distribution of emission probabilities associated with each insert and match state, describing the probability of observing the secondary structures E (beta), H (alpha), or L (loop), see Figure 26.

When matching a sequence to the ssHMM, in each match or insert state the model gives the probability of observing the given amino acid, as before. However, in addition to this it gives a second probability for the secondary structure assigned to that position. In this way, the probability for the sequence becomes higher if its secondary structure is the same as in the modelled family. The total probability for a sequence $s = x_1 \ldots x_L$ with the secondary structure $ss = y_1 \ldots y_L$, given the path $q = q_0 \ldots q_N$ and model μ, is:

$$P(s, ss \mid q,\mu) = \prod_{i=1}^{N+1} T(q_i \mid q_{i-1}) \prod_{i=1}^{N} P(x_{l(i)} \mid q_i) \prod_{i=1}^{N} P(y_{m(i)} \mid q_i), \tag{7.3}$$

where $y_{m(i)}$ is the secondary structure seen in state q_i. The emission probabilities of the secondary structures are determined in a similar way as the amino acid emission probabilities when building the ssHMM.

[5] http://hmmer.janelia.org/

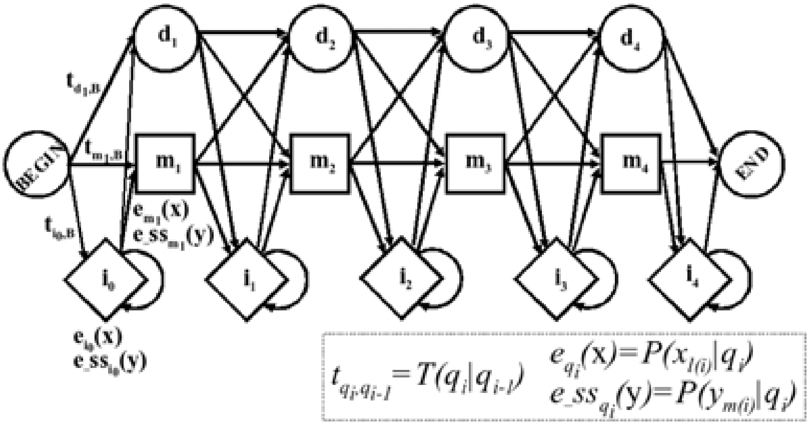


FIGURE 26: The architecture of the ssHMM. In addition to amino acid emission probabilities, the match states of the ssHMM also have associated secondary structure emission probabilities, compare with Figure 9.

At the beginning of the training, all secondary structures are assumed to occur at equal probabilities. Thus, even if a position is found in only one secondary structure type, the other secondary structure types will also have a small probability of occurrence.

In our implementation, the secondary structures are given the same weight as the amino acids. In our benchmark we found that a few families caused a very large part of the false positives obtained using the ssHMMs. The majority of these matches were between different families that all consisted of a various number of alpha helices. It seems plausible that the contribution from the secondary structure was ranked too high in comparison with the contribution from the sequence in these cases. For the ssHMMs to perform the best, and to obtain scores following the extreme value distribution assumed when calculating E-values, the weighting needs to be fine-tuned.

7.2.2 Performance Evaluation of the ssHMMs

The benchmarking of the ssHMMs is based on matching all proteins in a test set against all other proteins of the test set, in order to assign a correct fold to the query sequences.

A library of ssHMMs was built from the sequences and secondary structures of a representative set of all proteins with known structures. We use the pdb40 data set of SCOP version 1.37, which contain a subset of SCOP (Section 4.3) where no protein domains have more than 40% sequence identity to any other member of the data set [21]. For each of the 1130 proteins in the data set, all closely related sequences in SWISS-PROT are found via the HSSP database [121]. On average, 26 sequences are collected for each protein. However, many of these sequences are identical or almost identical to the original sequence. The secondary structure is assumed to be the same for all proteins within a group. The multiple sequence alignment from HSSP, together with the secondary structure, is used to build an ssHMM. For comparison with the original HMM method, HMMs not using the secondary structure are also created, as are HMMs and ssHMMs using substitution matrices for prior information. Finally, another set of HMMs, ignoring multiple sequence alignments, is created.

When a protein is matched against an ssHMM, its secondary structure is needed in addition to its sequence. The secondary structure was obtained in two different ways. First, the correct secondary structure was used, since this was known for all proteins in our test set. Second, the secondary structure predicted by predator [48] was used. This most closely resembles how the ssHMMs would be used to annotate protein sequences in a real case.

In this study, we have focused on proteins that have the same fold but belong to different families, according to SCOP. Two proteins that are classified into the same fold have the same secondary structure elements in a similar topological arrangement, while two proteins that belong to the same family have a clear common evolutionary origin. In our benchmark, all proteins are matched to the HMMs of all other proteins, and for each pair the folds and families are recorded. If the two proteins belong to the same family, we eliminate them from further consideration, because this indicates that they are homologous and thereby not a good test of fold recognition.

Two different criteria are used to analyse the performance of a method; at what rank the first true hit was found, and specificity-sensitivity plots [114]. The *sensitivity* measures the model's ability to find all members of the same fold, and is the same as the coverage (see Section 7.1.6). The *specificity*, also called accuracy, measures the probability that a pair of sequences with a score greater than a certain threshold really belong to the same fold, and is calculated as the fraction of all matches above the threshold that are correct. The sensitivity is plotted as a function of specificity, where each point in the plot corresponds to a certain score. The main advantage of the specificity-sensitivity plots over the rank is that they describe the ability of a method to find *all* pairwise matches in the benchmark.

We find that the sensitivity of a hidden Markov model is increased when the secondary structure is included, both when using the true secondary structures and when using predicted ones (see Figure 27(a)). Also the fraction of the possible hits that were ranked in first place is increased when adding the secondary structures.

We also show that the sensitivity at a given specificity is increased for models built from multiple sequences compared to models built from just one sequence. The number of sequences placed at rank one is more than doubled when building models from multiple sequence alignments.

In HMMER1.8, the prior information used for emission probabilities do not include any information about which substitutions are most likely. If the protein family is large enough and diverse enough this should not be a problem. However, in our benchmark, we have many small families with low diversity. We find that using a substitution matrix, which contains information about how likely it is to exchange an amino acid for another, as the prior when building the models, increases the sensitivity significantly. In fact, the use of a substitution matrix helps more than the use of multiple sequence alignments. However, in both cases the secondary-structure-based

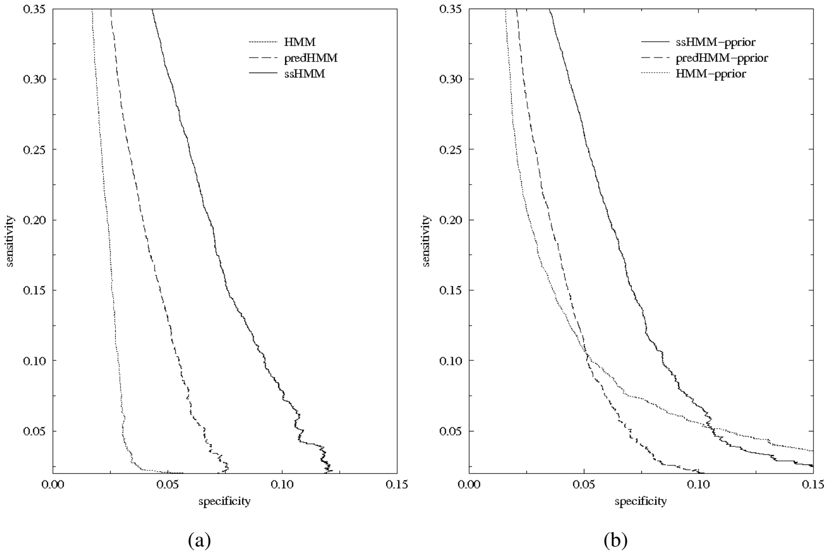


FIGURE 27: Specificity-sensitivity plots showing the performance of different types of HMMs. (a) The effect of adding secondary structure information. The sensitivity of a hidden Markov model is increased when the secondary structure is included, both when using predicted structures (dashed line) and the true secondary structure (bold line). (b) A comparison of HMMs that use multiple sequence information as well as substitution matrices, with and without secondary structure information.

HMMs place more correct sequences at high ranks than the ordinary HMMs. In later releases of HMMER, the more advanced Dirichlet mixtures (Section 5.3.4) are used for prior information as default.

We conclude that using multiple sequence alignment, predicted secondary structures, and a substitution matrix improves the performance of the HMMs (see Figure 27(b)).

The ssHMM method, together with other methods and manual judgement, were used for blind predictions in the CASP3 process [99]. Our best prediction was T0071 (Alpha adaptin ear domain), in which, using ssHMMs, we were able to identify the first 125 residues as an Ig-like fold. We were also able to produce a rather good alignment, with 21 out of 125 residues correctly aligned.

7.2.3 Other Approaches Using Secondary Structure and HMMs

There have been a few other approaches to integrate secondary structure information into hidden Markov models. Di Francesco *et al.* [33] also use predicted secondary structure of query proteins to achieve fold recognition. However, they build their HMMs from alignments of experimentally derived secondary structures only, and do not include any sequence information. Later, Karchin *et al.* [75] presented an approach slightly more similar to Paper V. However, instead of matching a query or target sequence against a library of HMMs, they score a template library of amino acid sequences with known secondary structures against the target HMM. They use SAM-T2K to train a HMM from a single target sequence. A neural net is used to predict secondary structure probabilities for the target protein, based on the amino acid emission probabilities of the HMM. The secondary structure probabilities are then added as secondary structure emission probabilities in the match states of the HMM.

CHAPTER 8

Summary of Papers and Outline of Future Work

In this chapter, a brief summary is given of each paper in the thesis. Some computational aspects are mentioned, and possible approaches for future work are outlined.

8.1 Paper I: Structure-Anchored HMMs

In Paper I, the novel *structure-anchored HMM* (saHMM) method is presented. The saHMMs are hidden Markov models based on alignments derived from the matching of structurally equivalent positions in protein structures. These kinds of alignments are assumed to be more biologically correct than those based solely on sequence and simple statistics, especially for sequences with very low sequence identities. The saHMMs are built using a careful selection of representative protein domains, where it is ensured that no domain sequence is more than about 20% identical to any other representative domain in the same family. This is to guarantee sequence diversity among the domains chosen as representatives for each family.

First, we show that the saHMMs are able to accurately identify members of the families they represent. Using the saHMMs, we can assign the correct family to the vast majority of our test sequences, and most of the few false matches are still within the correct superfamily.

In a comparison with PSI-BLAST, we find that the saHMMs are much more accurate in their domain assignments. Even when evaluating the ability to correctly identify sequences with very low sequence identities to any sequence used for model building, the saHMMs are more accurate and have a higher coverage than PSI-BLAST at an acceptable number of errors per query. Using the sequences added between SCOP releases 1.69 and 1.71 as queries for HMMs corresponding to SCOP 1.69 result in 94% correct assignments by the saHMMs. This number is higher than the 88% obtained using the corresponding Pfam HMMs. We also show that the saHMMs can be used to annotate protein sequences that previously lack annotations.

8.2 Papers II & III: The FISH Server and the Midnight ASTRAL Set

Paper II introduces the FISH server, where the saHMMs can be accessed for searching. In the paper, the saHMMs are briefly introduced, and the architecture and use of the server are described. We also show that the saHMMs are able to correctly assign family relationships for a majority (74%) of sequences added in SCOP 1.69 compared to SCOP 1.61, which was used to build the saHMMs for the benchmark. Of the sequences with a very low sequence identity to those used to build saHMMs, 62% could be assigned to the correct domain family, despite the low sequence identity. In addition, we find in an analogous benchmark that the saHMMs perform similarly to Pfam HMMs, and that 813 of the sequences correctly assigned to a domain family by the saHMMs could not be assigned to a family by Pfam. This shows that the FISH server is complementary to Pfam.

The FISH server is also treated in Paper III, where the use and the design of the server is described in more detail. However, the main focus of this paper is on the creation of the midnight ASTRAL set, i.e. the selection of representative domains for each SCOP family. Two algorithms, both slightly modified from those presented in [68], are evaluated; one trying to minimize the number of pairwise structural comparisons, and one aimed at maximizing the number of representatives. We find that the second algorithm is more than an order of magnitude slower than the first one, due to the all-against-all structural comparisons within each domain family. We therefore conclude that even though the second algorithm results in a slightly larger midnight ASTRAL set, it is not feasible to use this algorithm for regular updates. Hence, we decide to use the first algorithm for the construction of the saHMMs.

8.3 Paper IV: The FI-score and Combinatorial Selection

In Paper IV, we explore a way to improve the worst performing saHMMs. Here, the *Family Identification score*, FI-score, for measuring the performance of the saHMMs is introduced. The FI-score takes both the coverage and the accuracy into account in a combined score, ranging from negative (more false matches than correct ones) to 1 (perfect performance). We use the FI-score to rank saHMMs that represent the same domain family, but that are built from different number and combinations of saHMM-members. We find that also saHMMs with a very low number of saHMM-members can perform remarkably well. It is also obvious that it is not necessarily the complete set of saHMM-members that yields the best performing saHMM, instead some combination of a subset of saHMM-members might result in the best model. We exploit this fact and choose the number and combination of saHMM-members that yield the best saHMM for the families with the worst performing saHMMs in the database. In this way, the average FI-score of the saHMMs with less than 65% coverage was increased from 0.298 to 0.649. We also compare the parameters of selected saHMMs representing the Ig V-set domain family, as well as the corresponding Pfam HMM, by using HMM Logo plots [123]. We find that the pattern of conserved residues in well performing saHMMs are very similar to that of the Pfam HMM, despite the considerably lower number of sequences used when constructing the saHMMs.

8.4 Paper V: Secondary Structure HMMs

In Paper V, a different type of novel HMMs, *secondary structure HMMs* (ssHMMs), is described and evaluated. The ssHMM is a combined HMM, taking both the sequences and the secondary structures of the proteins into account. Here, the actual architecture of the standard HMM is modified. If the secondary structure of the query sequence matches that of the HMM, the score for that match is increased, even if the particular amino acid at that position does not fit well. For a sequence whose structure is unknown, which would be the matter in a real case, the secondary structure of course has to be predicted, using some secondary prediction method, before it can be compared to the HMM.

In this paper, we also present a more rigorous benchmark than was used in most previous studies, and show that the use of HMMs made from multiple sequences results in better fold recognition than that obtained by HMMs based on only single sequences and scoring matrices. Adding secondary structure information to the HMMs improves the ability of fold recognition further, both when using true and predicted secondary structures for the query sequence.

8.5 Conclusions

In this thesis, two approaches are used to add structural information to hidden Markov models. The novel structure-anchored HMMs use structure alignments of selected representative domains to model SCOP domain families. The representative domains are selected so that they have high quality crystal structures and low mutual sequence identities. We find that these few, carefully selected, representative structures are sufficient to create HMMs for family recognition with high coverage and accuracy. We find that the saHMMs are very family specific, and are able to distinguish between members of the families they represent and members of other families, even within the same superfamily. Also, most of the few false predictions made by the saHMMs are still within the correct superfamily.

The saHMMs are in our tests able to assign the correct family to more sequences than are Pfam HMMs, despite the limited number of representative sequences for each family. In addition, we find that the saHMMs are more accurate than PSI-BLAST when locating members of a given family. When investigating the ability of the two methods to assign the correct family to remote family members, with low sequence identity to the other members of the family, we find that the saHMMs are able to make more assignments at low error rates.

We introduce the FI-sore, which is used to score the performance of saHMMs resulting from different number and combinations of saHMM-members within the same family. We show that it is possible to improve the performance of the worst performing saHMMs by selecting the best combination of saHMM-members for each domain family.

Through the structural information associated with a hit to an saHMM it might be possible to build comparative models using the saHMM-members as template structures. This provides a starting point for further computational and experimental analysis such as mutagenesis studies, identification of active sites and interaction surfaces, and could possibly assist in drug design.

The saHMMs are made publicly available for searching through the FISH server, where a user can select to submit a query sequence for searching the collection of saHMMs, or choose an saHMM for searching against a sequence database. In order to construct the midnight ASTRAL set, i.e. the selection of representative domains, and build saHMMs from the structure alignments of these domains, a nearly automatic pipe-line of software tools has been developed. This pipe-line facilitates the update of the FISH server with new SCOP releases.

The second approach used in this thesis to include structural information in HMMs is to modify the architecture of the HMM, in order to consider both the sequence and the secondary structure of a protein when scoring it against the model. We find that the novel ssHMMs, which take both sequence and secondary structure into account, are better than comparable methods for fold recognition. We could also confirm the assumption that HMMs built from multiple sequences perform better than HMMs built from single sequences.

8.6 Computational Aspects

To generate and test the two kinds of hidden Markov models, extensive calculations were needed. The individual calculations are, taken by themselves, of moderate sizes. However, the construction of the midnight ASTRAL set for the saHMMs require pairwise structural comparisons of all members within each family, to make sure that none of the saHMM-members has a higher than allowed sequence identity when compared to any other saHMM-member within the same family. Using MUSTANG, the selection of representative domains within a single family can take more than 90 CPU hours using the faster algorithm. However, most families are finished within less than 30 minutes. The procedure was parallelized by running several families concurrently. For the evaluation of the performance of the saHMMs, each of the saHMMs was used to search the collection of roughly 67000 SCOP sequences, and each of the sequences was used to query the collection of saHMMs. Also this was performed in parallel, to make the computation times feasible.

All computations concerning the saHMMs were done using up to 20 processors on the HPC2N Linux cluster Seth. The compute nodes on Seth are AMD AthlonMP2000+ with 1GB of memory per dual node, connected in a high-speed SCALI network.

An almost entirely automatic “pipe-line” was developed, using primarily perl, to go from raw SCOP classification of domains, through the selection of the midnight ASTRAL set, to the construction of saHMMs.

For the construction and evaluation of the ssHMMs, we had no access to parallel resources.

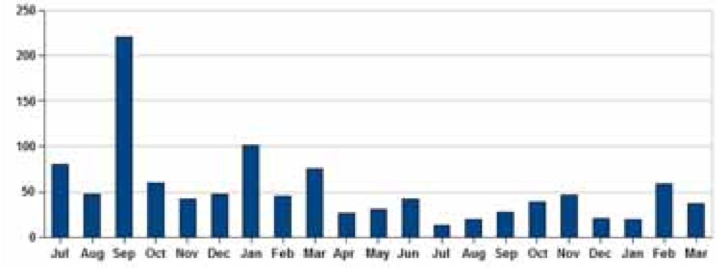


FIGURE 28: Histogram showing the number of sequences submitted to the FISH server each month from July 2006 to March 2008.

8.7 User Statistics for the FISH Server

The saHMMs can be accessed through the FISH server, which is described in Papers II and III of this thesis, and is available at http://babel.ucmp.umu.se/fish/. The server has at the time of writing been running for 21 full months since the publication of Paper II, and has this far received visitors from 31 countries. From outside Sweden, 128 unique visitors have been recorded (as of April 1st, 2008), and on average 39 sequences have been submitted to the server per month. The number of sequences submitted each month is illustrated in Figure 28.

8.8 Future Work

Since the saHMMs are based on structure anchored sequence alignments, the alignment of a query sequence to a structure-based alignment of members representative of a domain family gives important clues about the putative structure of the query, and about secondary structure elements in particular. Hence, the saHMMs can be used to draw conclusions about the structure of an unknown protein.

The saHMMs could be optimized further by tuning the parameters used when building the HMMs with HMMER. For example, the effect of the prior information used should be evaluated, and the standard Dirichlet mixtures could possibly be exchanged for an updated mixture or other prior information. Also, instead of using a standard E-value cut-off to determine significance of the hits, each saHMM should be associated with an individual cut-off.

Apart from being used to assign the correct family to protein sequences, the actual parameter values of the saHMMs could be studied in more detail to identify regions and residues important for a particular domain fold. For example, highly conserved amino acids could be extracted based on the emission probabilities.

The ssHMMs should be remade, using a newer version of HMMER – the standard HMM implementation used as the base for the ssHMMs. Also, the actual scoring and the weighting of the secondary structure information with respect to sequence information have to be evaluated and developed further.

One suggestion for futute work is to use the saHMM approach, with structural alignments of a carefully selected set of representative family members, together with the (updated) ssHMM-implementation, in order to obtain accurate HMMs that are able to detect very remote sequence relationships.

References

[1] B. Al-Lazikani, F. B. Sheinerman, and B. Honig. Combining multiple structure and sequence alignments to improve sequence detection and alignment: Application to the SH2 domains of Janus kinases. *Proc. Natl. Acad. Sci. USA*, 98(26):14796–14801, 2001.

[2] N. Alexandrov and V. Soloveyev. Statistical significance of ungapped sequence alignments. In R. Altman, A. Dunker, L. Hunter, and T. Klein, editors, *HICCS' 98: Pacific Symposium on Biocomputing' 98*, pages 463–472, 1998.

[3] N. N. Alexandrov and D. Fischer. Analysis of topological and nontopological structural similarities in the PDB: New examples with old structures. *PROTEINS: Structure, Fiunction and Genetics*, 25(3):354–365, 1996.

[4] V. Alexandrov and M. Gerstein. Using 3D hidden Markov models that explicitly represent spatial coordinates to model and compare protein structures. *BMC Bioinformatics*, 5:2, 2004.

[5] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215:403–410, 1990.

[6] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.

[7] B. Andreopoulos, A. An, X. Wang, M. Faloutsos, and M. Schroeder. Clustering by common friends finds locally significant proteins mediating modules. *Bioinformatics*, 23(9):1124–1131, 2007.

[8] M. Andronescu, A. P. Fejes, F. Hutter, H. H. Hoos, and A. Condon. A new algorithm for RNA secondary structure design. *Journal of Molecular Biology*, 336:607–624, 2004.

[9] V. Bafna and N. Edwards. SCOPE: a probabilistic model for scoring tandem mass spectra against a peptide database. *Bioinformatics*, 17:S13–S21, 2001.

[10] S. Balaji, S. Sujatha, S. Sai Chetan Kumar, and N. Srinivasan. PALI - a database of Phylogeny and ALIgnment of homologous protein structures. *Nucleic Acids Research*, 29(1):61–65, 2001.

[11] P. Baldi and S. Brunak. *Bioinformatics: The Machine Learning Approach, 2nd edition*. The MIT Press, Cambridge, Massachusetts, 2001.

[12] C. Barrett, R. Hughey, and K. Karplus. Scoring hidden Markov models. *Comput. Applic. Biosci.*, 13(2):191–199, 1997.

[13] A. Bateman, E. Birney, R. Durbin, S. R. Eddy, K. L. Howe, and E. L. L. Sonnhammer. The Pfam protein families database. *Nucleic Acids Research*, 28(1):263–266, 2000.

[14] A. Bateman, L. Coin, R. Durbin, R. D. Finn, V. Hollich, S. Griffiths-Jones, A. Khanna, M. Marshall, S. Moxon, E. L. Sonnhammer, D. J. Studholme, C. Yeats, and S. R. Eddy. The Pfam protein families database. *Nucleic Acids Research*, 32:D138–D141, 2004.

[15] A. D. Baxevanis. The molecular biology database collection: 2003 update. *Nucleic Acids Research*, 31(1):1–12, 2003.

[16] H. M. Berman, K. Henrick, and H. Nakamura. Announcing the worldwide protein data bank. *Nature Structural Biology*, 10(12):980, 2003.

[17] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Research*, 28(1):235–242, 2000.

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

[19] J. R. Bradford, C. J. Needham, A. J. Bulpitt, and D. R. Westhead. Insights into protein–protein interfaces using a bayesian network prediction method. *Journal of Molecular Biology*, 362(2):365–386, 2006.

[20] S. E. Brenner, C. Chothia, and T. J. P. Hubbard. Population statistics of protein structures: lessons from structural classifications. *Current Opinion in Structural Biology*, 7:369–376, 1997.

[21] S. E. Brenner, C. Chothia, and T. J. P. Hubbard. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Natl. Acad. Sci. USA*, 95:6073–6078, 1998.

[22] M. P. Brown, R. Hughey, A. Krogh, I. S. Mian, K. Sjölander, and D. Haussler. Using Dirichlet mixture priors to derive hidden Markov models for protein families. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 1:47–55, 1993.

[23] A. Caprera, B. Lazzari, A. Stella, I. Merelli, A. R. Caetano, and P. Mariani. GoSh: a web-based database for goat and sheep EST sequences. *Bioinformatics*, 23(8):1043–1045, 2007.

[24] J. Casbon and M. A. S. Saqi. On single and multiple models of protein families for the detection of remote sequence relationships. *BMC Bioinformatics*, 7:48, 2006.

[25] J.-M. Chandonia, G. Hon, N. S. Walker, L. L. Conte, P. Koehl, M. Levitt, and S. E. Brenner. The ASTRAL compendium in 2004. *Nucleic Acids Research*, 32:D189–D192, 2004.

[26] Y.-C. Chen, Y.-S. Lo, W.-C. Hsu, and J.-M. Yang. 3D-partner: a web server to infer interacting partners and binding models. *Nucleic Acids Research*, 35:W561–W567, 2007.

[27] R. Chenna, H. Sugawara, T. Koike, R. Lopez, T. J. Gibson, D. G. Higgins, and J. D. Thompson. Multiple sequence alignment with the clustal series of programs. *Nucleic Acids Research*, 31(13):3497–3500, 2003.

[28] K. Coeytaux and A. Poupon. Prediction of unfolded segments in a protein sequence based on amino acid composition. *Bioinformatics*, 21(9):1891–1900, 2005.

[29] F. Corpet, F. Servant, J. Gouzy, and D. Kahn. ProDom and ProDom-CG: tools for protein domain analysis and whole genome comparisons. *Nucleic Acids Research*, 28(1):267–269, 2000.

[30] M. W. Covert and B. O. Palsson. Constraints-based models: Regulation of gene expression reduces the steady-state solution space. *Journal of Theoretical Biology*, 221:309–325, 2003.

[31] M. Dayhoff, R. Eck, M. Chang, and M. Sochard. *Atlas of Protein Sequence and Structure*. National Biomedical Research Foundation, Silver Spring, Maryland, 1965.

[32] M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt. *Atlas of Protein Sequence and Structure*, volume 5, chapter 22, pages 345–352. National Biomedical Research Foundation, Washington, 1978.

[33] V. Di Francesco, V. Geetha, J. Garnier, and P. J. Munson. Fold recognition using predicted secondary structure sequences and hidden Markov models of protein folds. *PROTEINS: Structure, Function, and Genetics*, 29(S1):123–128, 1997.

[34] K. Diederichs. Structural superposition of proteins with unknown alignment and detection of topological similarity using a six-dimensional search algorithm. *PROTEINS: Structure, Function and Genetics*, 23(2):187–195, 1995.

[35] O. Dror, H. Benyamini, R. Nussinov, and H. Wolfson. MASS: multiple structural alignment by secondary structures. *Bioinformatics*, 19:i95–i104, 2003.

[36] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998.

[37] S. R. Eddy. Hidden Markov models. *Current Opinion in Structural Biology*, 6:361–365, 1996.

[38] S. R. Eddy. Profile hidden Markov models. *Bioinformatics*, 14(9):755–763, 1998.

[39] R. C. Edgar and S. Batzoglou. Multiple sequence alignment. *Current Opinion in Structural Biology*, 16(3):368–373, 2006.

[40] I. Eidhammer, I. Jonassen, and W. R. Taylor. Structure comparison and structure patterns. *Journal of Computational Biology*, 7(5):685–716, 2000.

[41] O. Emanuelsson, H. Nielsen, S. Brunak, and G. von Heijne. Predicting subcellular localization of proteins based on their N-terminal amino acid sequence. *Journal of Molecular Biology*, 300:1005–1016, 2000.

[42] N. S. Enattah, T. Sahi, E. Savilahti, J. D. Terwilliger, L. Peltonen, and I. Järvelä. Identification of a variant associated with adult-type hypolactasia. *Nature Genetics*, 30(2):233–237, 2002.

[43] A. J. Enright, S. Van Dongen, and C. A. Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, 30(7):1575–1584, 2002.

[44] A. Falicov and F. E. Cohen. A surface of minimum area metric for the structural comparison of proteins. *Journal of Molecular Biology*, 258:871–892, 1996.

[45] D.-F. Feng and R. F. Doolittle. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *Journal of Molecular Evolution*, 25:351–360, 1987.

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

[47] T. Friedrich, B. Pils, T. Dandekar, J. Schultz, and T. Müller. Modelling interaction sites in protein domains with interaction profile hidden Markov models. *Bioinformatics*, 22(23):2851–2857, 2006.

[48] D. Frishman and P. Argos. Seventy-five percent accuracy in protein secondary structure prediction. *PROTEINS: Structure, Function, and Genetics*, 27(3):329–335, 1997.

[49] L. Fu and E. Medico. FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC Bioinformatics*, 8:3, 2007.

[50] O. V. Galzitskaya, S. O. Garbuzynskiy, and M. Y. Lobanov. Prediction of amyloidogenic and disordered regions in protein chains. *PLoS Computational Biology*, 2(12):e177, 2006.

[51] H. H. Gan, R. A. Perlow, S. Roy, J. Ko, M. Wu, J. Huang, S. Yan, A. Nicoletta, J. Vafai, D. Sun, L. Wang, J. E. Noah, S. Pasquali, and T. Schlick. Analysis of protein sequence/structure similarity relationships. *Biophysical Journal*, 83:2781–2791, 2002.

[52] M. Gerstein and M. Levitt. Comprehensive assessment of automatic structural alignment against a manual standard, the scop classification of proteins. *Protein Science*, 7:445–456, 1998.

[53] J.-F. Gibrat, T. Madej, and S. H. Bryant. Surprising similarities in structure comparison. *Current Opinion in Structural Biology*, 6:377–385, 1996.

[54] K. Ginalski. Comparative modeling for protein structure prediction. *Current Opinion in Structural Biology*, 16(2):172–177, 2006.

[55] S. Gong, G. Yoon, I. Jang, D. Bolser, P. Dafas, M. Schroeder, H. Choi, Y. Cho, K. Han, S. Lee, H. Choi, M. Lappe, L. Holm, S. Kim, D. Oh, and J. Bhak. PSIbase: a database of Protein Structural Interactome map (PSIMAP). *Bioinformatics*, 21(10):2541–2543, 2005.

[56] O. Gotoh. An improved algorithm for matching biological sequences. *Journal of Molecular Biology*, 162:705–708, 1982.

[57] J. Gough, K. Karplus, R. Hughey, and C. Chothia. Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structures. *Journal of Molecular Biology*, 313:903–919, 2001.

[58] V. S. Gowri, S. B. Pandit, P. S. Karthik, N. Srinivasan, and S. Balaji. Integration of related sequences with protein three-dimensional structural families in an updated version of PALI database. *Nucleic Acids Research*, 31(1):486–488, 2003.

[59] L. Grate, M. Herbster, R. Hughey, D. Haussler, I. Mian, and H. Noller. RNA modeling using Gibbs sampling and stochastic context free grammars. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 2:138–146, 1994.

[60] M. Gribskov, A. D. McLachlan, and D. Eisenberg. Profile analysis: Detection of distantly related proteins. *Proc. Natl. Acad. Sci. USA*, 84:4355–4358, 1987.

[61] M. Gribskov and N. L. Robinson. The use of reciever operating characteristic (ROC) analysis to evaluate sequence matching. *Computers and Chemistry*, 20(1):25–33, 1996.

[62] S. Griffiths-Jones and A. Bateman. The use of structure information to increase alignment accuracy does not aid homologue detection with profile HMMs. *Bioinformatics*, 18(9):1243–1249, 2002.

[63] C. Guda, E. Scheeff, P. Bourne, and I. Shindyalov. A new algorithm for the alignment of multiple protein structures using Monte Carlo optimization. *Proceedings of the Pacific Symposium on Biocomputing*, 6:275–286, 2001.

[64] J. B. Hagen. The origins of bioinformatics. *Nature Reviews Genetics*, 1:231–236, 2000.

[65] P. Han, X. Zhang, R. Norton, and Z. Feng. Predicting disordered regions in proteins based on decision trees of reduced amino acid composition. *Journal of Computational Biology*, 13(10):1723–1734, 2006.

[66] S. Henikoff and J. G. Henikoff. Automated assembly of protein blocks for database searching. *Nucleic Acids Research*, 19(23):6565–6572, 1991.

[67] S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*, 89:10915–10919, 1992.

[68] U. Hobohm, M. Scharf, R. Schneider, and C. Sander. Selection of representative protein data sets. *Protein Science*, 1(3):409–417, 1992.

[69] L. Holm and C. Sander. Protein structure comparison by alignment of distance matrices. *Journal of Molecular Biology*, 233:123–138, 1993.

[70] I. Holmes. A probabilistic model for the evolution of RNA structure. *BMC Bioinformatics*, 2004.

[71] P. Hong, S. Zhong, and W. H. Wong. Towards ubiquitous bio-information computing: Data protocols, middleware, and web services for heterogeneous biological information integration and retrieval. *Proc IEEE Symposium on Bioinformatics and Bioengineering (BIBE)*, pages 57–64, 2004.

[72] D. Horn and I. Axel. Novel clustering algorithm for microarray expression data in a truncated SVD space. *Bioinformatics*, 19(9):1110–1115, 2003.

[73] R. Hughey and A. Krogh. Hidden Markov models for sequence analysis: extension and analysis of the basic method. *Comput. Applic. Biosci.*, 12(2):95–107, 1996.

[74] J. Jung and B. Lee. Protein structure alignment using environmental profiles. *Protein Engineering*, 13(8):535–543, 2000.

[75] R. Karchin, M. Cline, Y. Mandel-Gutfreund, and K. Karplus. Hidden Markov models that use predicted local structure for fold recognition: alphabets of backbone geometry. *PROTEINS: Structure, Function, and Genetics*, 51(4):504–514, 2003.

[76] K. Karplus, K. Sjölander, C. Barrett, M. Cline, D. Haussler, R. Hughey, L. Holm, and C. Sander. Predicting protein structure using hidden Markov models. *PROTEINS: Structure, Function, and Genetics*, 29(S1):134–139, 1997.

[77] P. Koehl. Protein structure similarities. *Current Opinion in Structural Biology*, 11:348–353, 2001.

[78] A. S. Konagurthu, J. C. Whisstock, P. J. Stuckey, and A. M. Lesk. MUSTANG: A multiple structural alignment algorithm. *PROTEINS: Structure, Function, and Bioinformatics*, 64(3):559–574, 2006.

[79] W. A. Koppensteiner, P. Lackner, M. Wiederstein, and M. J. Sippl. Characterization of novel proteins based on known protein structures. *Journal of Molecular Biology*, 296:1139–1152, 2000.

[80] J. Kosinski, I. A. Cymerman, M. Feder, M. A. Kurowski, J. M. Sasin, and J. M. Bujnicki. A "FRankenstein's monster" approach to comparative modeling: merging the finest fragments of Fold-Recognition models and iterative model refinement aided by 3D structure evaluation. *PROTEINS: Structure, Function and Genetics*, 53(S6):369–379, 2003.

[81] A. Krause, J. Stoye, and M. Vingron. Large scale hierarchical clustering of protein sequences. *BMC Bioinformatics*, 6:15, 2005.

[82] A. Krogh, B. Larsson, G. von Heijne, and E. L. L. Sonnhammer. Predicting transmembrane protein topology with a hidden Markov model: Application to complete genomes. *Journal of Molecular Biology*, 305:567–580, 2001.

[83] A. Kryshtafovych, Č. Venclovas, K. Fidelis, and J. Moult. Progress over the first decade of CASP experiments. *PROTEINS: Structure, Function, and Bioinformatics*, 61(S7):225–236, 2005.

[84] I. B. Kuznetsov, Z. Gou, R. Li, and S. Hwang. Using evolutionary and structural information to predict DNA-binding sites on DNA-binding proteins. *PROTEINS: Structure, Function, and Bioinformatics*, 64(1):19–27, 2006.

[85] R. H. Lathrop and T. F. Smith. Global optimum protein threading with gapped alignment and empirical pair score functions. *Journal of Molecular Biology*, 255:641–665, 1996.

[86] N. Leibowitz, Z. Y. Fligelman, R. Nussinov, and H. J. Wolfson. Automated multiple structure alignment and detection of a common substructural motif. *PROTEINS: Structure, Function and Genetics*, 43(3):235–245, 2001.

[87] N. Leibowitz, R. Nussinov, and H. J. Wolfson. MUSTA - a general, efficient, automated method for multiple structure alignment and detection of common motifs: Application to proteins. *Journal of Computational Biology*, 8(2):93–121, 2001.

[88] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12, 1994.

[89] E. Lindahl and A. Elofsson. Identification of related proteins on family, superfamily and fold level. *Journal of Molecular Biology*, 295:613–625, 2000.

[90] A. Liwo, M. Khalili, and H. A. Scheraga. Ab initio simulations of protein-folding pathways by molecular dynamics with the united-residue model of polypeptide chains. *Proc. Natl. Acad. Sci. USA*, 102(7):2362–2367, 2005.

[91] G. Lu. TOP: a new method for protein structure comparisons and similarity searches. *Journal of Applied Crystallography*, 33:176–183, 2000.

[92] D. Lupyan, A. Leo-Macias, and A. R. Ortiz. A new progressive-iterative algorithm for multiple structure alignment. *Bioinformatics*, 21(15):3255–3263, 2005.

[93] T. Madej, J.-F. Gibrat, and S. H. Bryant. Threading a database of protein cores. *PROTEINS: Structure, Function and Genetics*, 23(3):356–369, 1995.

[94] M. Madera and J. Gough. A comparison of profile hidden Markov model procedures for remote homology detection. *Nucleic Acids Research*, 30(19):4321–4328, 2002.

[95] M. Madera, C. Vogel, S. K. Kummerfeld, C. Chothia, and J. Gough. The SUPERFAMILY database in 2004: additions and improvements. *Nucleic Acids Research*, 32:D235–D239, 2004.

[96] G. Magiorkinis, E. Magiorkinis, D. Paraskevis, A. Vandamme, M. V. Ranst, V. Moulton, and A. Hatzakis. Phylogenetic analysis of the full-length SARS-CoV sequences: Evidence for phylogenetic discordance in three genomic regions. *Journal of Medical Virology*, 74:369–372, 2004.

[97] S. Mika and B. Rost. UniqueProt: creating representative protein sequence sets. *Nucleic Acids Research*, 31(13):3789–3791, 2003.

[98] K. Mizuguchi, C. M. Deane, T. L. Blundell, and J. P. Overington. HOMSTRAD: A database of protein structure alignments for homologous families. *Protein Science*, 7:2469–2471, 1998.

[99] J. Moult, T. Hubbard, S. Bryant, K. Fidelis, and J. Pedersen. Critical assessment of methods of proteins structure predictions (CASP): round II. *PROTEINS: Structure, Function, and Genetics*, 29(S1):2–6, 1997.

[100] K. Munch and A. Krogh. Automatic generation of gene finders for eukaryotic species. *BMC Bioinformatics*, 7(263), 2006.

[101] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, 247:536–540, 1995.

[102] S. B. Needleman and C. D. Wunch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48:443–453, 1970.

[103] H. Nielsen and A. Krogh. Prediction of signal peptides and signal anchors by a hidden Markov model. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 6:122–130, 1998.

[104] R. A. Notebaart, F. H. van Enckevort, C. Francke, R. J. Siezen, and B. Teusink. Accelerating the reconstruction of genome-scale metabolic networks. *BMC Bioinformatis*, 7:296, 2006.

[105] C. Notredame. Recent progresses in multiple sequence alignment: a survey. *Pharmacogenomics*, 3(1):131–144, 2002.

[106] C. Notredame and D. G. Higgins. SAGA: sequence alignment by genetic algorithm. *Nucleic Acids Research*, 24(8):1515–1524, 1996.

[107] C. Notredame, D. G. Higgins, and J. Heringa. T-COFFEE: a novel method for fast and accurate multiple sequence alignment. *Journal of Molecular Biology*, 302:205–217, 2000.

[108] M. E. Ochagavía and S. Wodak. Progressive combinatorial algorithm for multiple structural alignments: Application to distantly related proteins. *PROTEINS: Structure, Function, and Bioinformatics*, 55(2):436–454, 2004.

[109] C. A. Orengo. CORA - topological fingerprints for protein structural families. *Protein Science*, 8:699–715, 1999.

[110] C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton. CATH - a hierarchic classification of protein domain structures. *Structure*, 5:1093–1108, 1997.

[111] J. Park, K. Karplus, C. Barrett, R. Hughey, D. Haussler, T. Hubbard, and C. Chothia. Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. *Journal of Molecular Biology*, 284:1201–1210, 1998.

[112] W. R. Pearson and D. J. Lipman. Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA*, 85:2444–2448, 1988.

[113] L. A. Pennacchio, G. G. Loots, M. A. Nobrega, and I. Ovcharenko. Predicting tissue-specific enhancers in the human genome. *Genome Research*, 17:201–211, 2007.

[114] D. W. Rice and D. Eisenberg. A 3D-1D substitution matrix for protein fold recognition that includes predicted secondary structure of the sequence. *Journal of Molecular Biology*, 267:1026–1038, 1997.

[115] D. W. Rice, D. Fischer, R. Weiss, and D. Eisenberg. Fold assignments for amino acid sequences of the CASP2 experiment. *PROTEINS: Structure, Function, and Genetics*, 29(S1):113–122, 1997.

[116] E. Rivas, R. J. Klein, T. A. Jones, and S. R. Eddy. Computational identification of noncoding RNAs in E. coli by comparative genomics. *Current Biology*, 11:1369–1373, 2001.

[117] B. Rost. Twilight zone of protein sequence alignments. *Protein Engineering*, 12:85–94, 1999.

[118] R. B. Russell and G. J. Barton. Multiple protein sequence alignment from tertiary structure comparison: Assignment of global and residue confidence levels. *PROTEINS: Structure, Function and Genetics*, 14(2):309–323, 1992.

[119] N. Saitou and M. Nei. The neighbour-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, 4(4):406–425, 1987.

[120] A. Sali and T. L. Blundell. Definition of general topological equivalence in protein structures: A procedure involving comparison of properties and relationships through simulated annealing and dynamic programming. *Journal of Molecular Biology*, 212:403–428, 1990.

[121] C. Sander and R. Schneider. Database of homology-derived protein structures and the structural meaning of sequence alignment. *PROTEINS: Structure, Function and Genetics*, 9(1):56–68, 1991.

[122] E. D. Scheeff and P. E. Bourne. Application of protein structure alignments to iterated hidden Markov model protocols for structure prediction. *BMC Bioinformatics*, 7:410, 2006.

[123] B. Schuster-Böckler, J. Schultz, and S. Rahmann. HMM Logos for visualization of protein families. *BMC Bioinformatics*, 5:7, 2004.

[124] J. Söding. Protein homology detection by HMM-HMM comparison. *Bioinformatics*, 21(7):951–960, 2005.

[125] P. Sethupathy, M. Megraw, and A. G. Hatzigeorgiou. A guide through present computational approaches for the identification of mammalian microRNA targets. *Nature Methods*, 3(11):881–886, 2006.

[126] M. Shatsky, R. Nussinov, and H. J. Wolfson. A method for simultaneous alignment of multiple protein structures. *PROTEINS: Structure, Function, and Bioinformatics*, 56(1):143–156, 2004.

[127] J. Shi, T. L. Blundell, and K. Mizuguchi. FUGUE: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. *Journal of Molecular Biology*, 310:243–257, 2001.

[128] E. S. C. Shih and M.-J. Hwang. Protein structure comparison by probability-based matching of secondary structure elements. *Bioinformatics*, 19(6):735–741, 2003.

[129] I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Engineering*, 11(9):739–747, 1998.

[130] I. Sillitoe, M. Dibley, J. Bray, S. Addou, and C. Orengo. Assessing strategies for improved superfamily recognition. *Protein Science*, 14:1800–1810, 2005.

[131] K. T. Simons, R. Bonneau, I. Ruczinski, and D. Baker. Ab initio protein structure prediction of CASP III targets using ROSETTA. *PROTEINS: Structure, Function, and Genetics*, 37(S3):171–176, 1999.

[132] A. P. Singh and D. L. Brutlag. Hierarchical protein structure superposition using both secondary structure and atomic representations. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 5:284–293, 1997.

[133] M. Singhal and H. Resat. A domain-based approach to predict protein-protein interactions. *BMC Bioinformatics*, 8:199, 2007.

[134] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[135] V. Solovyev and A. Salamov. The gene-finder computer tools for analysis of human and model organisms genome sequences. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 5:294–302, 1997.

[136] E. L. Sonnhammer, G. von Heijne, and A. Krogh. A hidden Markov model for predicting transmembrane helices in protein sequences. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 6:175–182, 1998.

[137] E. L. L. Sonnhammer, S. R. Eddy, and R. Durbin. Pfam: A comprehensive database of protein domain families based on seed alignments. *PROTEINS: Structure, Function and Genetics*, 28(3):405–420, 1997.

[138] D. M. Standley, H. Toh, and H. Nakamura. Detecting local structural similarity in proteins by maximizing number of equivalent residues. *PROTEINS: Structure, Function, and Bioinformatics*, 57(2):381–391, 2004.

[139] J. Stoye. Multiple sequence alignment with the divide-and-conquer method. *Gene*, 211:GC45–GC56, 1998.

[140] J. D. Szustakowski and Z. Weng. Protein structure alignment using a genetic algorithm. *PROTEINS: Structure, Function and Genetics*, 38(4):428–440, 2000.

[141] W. R. Taylor, K. Lin, D. Klose, F. Fraternali, and I. Jonassen. Dynamic domain threading. *PROTEINS: Structure, Function, and Bioinformatics*, 64(3):601–614, 2006.

[142] W. R. Taylor and C. A. Orengo. Protein structure alignment. *Journal of Molecular Biology*, 208:1–22, 1989.

[143] J. D. Thompson, D. G. Higgins, and T. J. Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, 22(22):4673–46800, 1994.

[144] J. Tångrot. *The Use of Structural Information to Improve Biological Sequence Similarity Searches*. Licentiate thesis, Umeå University, Umeå, Sweden, 2003.

[145] J. Tångrot, B. Kågström, and U. H. Sauer. Structure anchored HMMs (saHMMs) for sensitive sequence searches. Technical Report UMINF 03.18, Department of Computing Science, Umeå University, Sweden, 2003.

[146] K. Torkkola, R. M. Gardner, T. Kaysser-Kranich, and C. Ma. Self-organizing maps in mining gene expression data. *Information Sciences*, 139(1–2):79–96, 2001.

[147] M. Wang, J. Yang, and K.-C. Chou. Using string kernel to predict signal peptide cleavage site based on subsite coupling model. *Amino Acids*, 28:395–402, 2005.

[148] T. D. Wu, S. C. Schmidler, T. Hastie, and D. L. Brutlag. Regression analysis of multiple protein structures. *Journal of Computational Biology*, 5(3):585–595, 1998.

[149] Y. Xu, V. Olman, and D. Xu. Clustering gene expression data using a graph-theoretic approach: an application of minimum spanning trees. *Bioinformatics*, 18(4):536–545, 2002.

[150] A.-S. Yang and B. Honig. An integrated approach to the analysis and modeling of protein sequences and structures. I. Protein structural alignment and a quantitative measure for protein structural distance. *Journal of Molecular Biology*, 301:665–678, 2000.

[151] A.-S. Yang and B. Honig. An integrated approach to the analysis and modeling of protein sequences and structures. III. A comparative study of sequence conservation in protein structural families using multiple structural alignments. *Journal of Molecular Biology*, 301:691–711, 2000.

[152] J. Yang, W. Chen, J. Skolnick, and E. Shakhnovich. All-atom ab initio folding of a diverse set of proteins. *Structure*, 15(1):53–63, 2007.

[153] J. Ye and R. Janardan. Approximate multiple protein structure alignment using the sum-of-pairs distance. *Journal of Computational Biology*, 11(5):986–1000, 2004.

[154] Y. Ye and A. Godzik. Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics*, 19:ii246–ii255, 2003.

[155] G. Yeo and C. B. Burge. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *Journal of Computational Biology*, 11(2–3):377–394, 2004.

[156] T. Zhou, L. Chen, Y. Tang, and X. Zhang. Aligning multiple protein structures by deterministic annealing. *Journal of Bioinformatics and Computational Biology*, 3(4):837–860, 2005.

[157] M. H. Zweig and G. Campbell. Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. *Clinical Chemistry*, 39(4):561–577, 1993.

I

Paper I

Accurate Domain Identification with Structure-Anchored Hidden Markov Models, saHMMs*

Jeanette Tångrot[1,2], Bo Kågström[2,3] and Uwe H. Sauer[1]

[1] Umeå Centre for Molecular Pathogenesis,
[2] Department of Computing Science
and
[3] High Performance Computing Center North
Umeå Universty
S-901 87 Umeå, Sweden
{jeanette,bokg}@cs.umu.se, uwe@ucmp.umu.se

Abstract: The speed of DNA sequencing has increased the discrepancy between the number of known gene products, and the knowledge of their function and structure. Proper annotation of protein sequences is therefore crucial if the missing information is to be deduced from sequence-based similarity comparisons. These comparisons become very difficult as the pairwise identities drop to very low values. In order to increase the accuracy of sequence searches, we exploit the fact that the three-dimensional structures of proteins are much more conserved than their sequences. Based on structure-anchored multiple sequence alignments of low identity sequences we have constructed a collection of 850 structure-anchored hidden Markov models, saHMMs, each representing one domain family. A search of SCOP sequences versus our saHMMs shows that 95% of the matches are to the correct family. Of the few hits outside the family, almost all fall within the correct superfamily. A comparison with PSI-BLAST shows that the saHMMs have consistently lower errors per query at a given coverage. Evaluating the ability of the saHMMs to correctly identify new family members by searching with sequences from a new version of SCOP resulted in 99% accuracy and 85% coverage. In a similar evaluation we find that the saHMMs performed about 6% better than the corresponding Pfam_ls HMMs. Furthermore, of 1986 human protein sequences labelled “unknown” in the NCBI protein database, we were able to annotate 232 proteins with 530

*From UMINF 07.12, 2008

non-overlapping domains belonging to 102 different domain families.

Our results demonstrate that saHMMs, which are derived from multiple structure alignments of a few carefully selected homologous sequences with low mutual sequence identities, result in a versatile and reliable tool for identification of domains in protein sequences. With the aid of saHMMs, homology on the family level can be assigned, even for distantly related sequences. The saHMMs have the added benefit that all matches are associated with multiple high quality crystal structures, which will support accurate structure annotation. The saHMMs are freely available for querying via the FISH server at http://babel.ucmp.umu.se/fish/.

Accurate Domain Identification with Structure-Anchored Hidden Markov Models, saHMMs

Jeanette E. Tångrot[1,2], Bo Kågström[2,3] and Uwe H. Sauer[1 §]

[1]Umeå Centre for Molecular Pathogenesis (UCMP), Umeå University, SE-901 87 Umeå, Sweden

[2]Department of Computing Science, Umeå University, SE-901 87 Umeå, Sweden

[3]High Performance Computing Center North (HPC2N), Umeå University, SE-901 87 Umeå, Sweden

§ to whom correspondence should be addressed

E-mail addresses
JET: jeanette@cs.umu.se;
BK: bokg@cs.umu.se;
UHS: uwe@ucmp.umu.se

Abstract

The speed of DNA sequencing has increased the discrepancy between the number of known gene products, and the knowledge of their function and structure. Proper annotation of protein sequences is therefore crucial if the missing information is to be deduced from sequence-based similarity comparisons. These comparisons become very difficult as the pairwise identities drop to very low values. In order to increase the accuracy of sequence searches, we exploit the fact that the three-dimensional structures of proteins are much more conserved than their sequences.

Based on structure-anchored multiple sequence alignments of low identity sequences we have constructed a collection of 850 structure-anchored hidden Markov models, saHMMs, each representing one domain family. A search of SCOP sequences versus our saHMMs shows that 95% of the matches are to the correct family. Of the few hits outside the family, almost all fall within the correct superfamily. A comparison with PSI-BLAST shows that the saHMMs have consistently lower errors per query at a given coverage. Evaluating the ability of the saHMMs to correctly identify new family members by searching with sequences from a new version of SCOP resulted in 99% accuracy and 85% coverage. In a similar evaluation we find that the saHMMs performed about 6% better than the corresponding Pfam_ls HMMs. Furthermore, of 1986 human protein sequences labelled "unknown" in the NCBI protein database, we were able to annotate 232 proteins with 530 non-overlapping domains belonging to 102 different domain families.

Our results demonstrate that saHMMs, which are derived from multiple structure alignments of a few carefully selected homologous sequences with low mutual sequence identities, result in a versatile and reliable tool for identification of domains in protein sequences. With the aid of saHMMs, homology on the family level can be assigned, even for distantly related sequences. The saHMMs have the added benefit that all matches are associated with multiple high quality crystal structures, which will support accurate structure annotation. The saHMMs are freely available for querying via the FISH server at http://babel.ucmp.umu.se/fish/.

Introduction

Finding a needle in a haystack is a relatively simple task compared to finding a particular pin in a stack of needles. A similar situation arises when one attempts to find pairs of homologous sequences with very low percent sequence identity after optimal alignment. The task is difficult in view of the fact that for a certain alignment length, *L*, the sequence identities of homologous pairs are virtually undistinguishable from the sequence identities of randomly picked sequences as the pairwise sequence identities decrease from about 20% towards zero.

One motivation for developing a method capable of identifying true matches with a high degree of certainty is the aim to accurately assign a newly sequenced gene product to a structural family, which might support the assignment of function. Ongoing genome sequencing projects produce an exponentially increasing amount of new sequences. As it is not feasible to study every newly sequenced protein experimentally, a common approach has been to deduce the characteristics from their sequence relationships to already characterized proteins. Due to their modularity, proteins can harbour many domains. It is advisable to characterize their constituent domains rather than the protein as a whole.

One way to determine the level of sequence identity is by inference from pairwise sequence alignments. However, it is much more recommendable to use multiple sequence alignments, MSAs, to extract pairwise identities. From the MSAs, one can also calculate sequence profiles, or build statistical models in the form of profile hidden Markov models, HMMs, thus improving on the ability to detect remotely related sequences. Park *et al.* showed that HMMs outperform other methods, in particular methods based on pair-wise alignments. It is crucial that the HMMs are built from reliable multiple sequence alignments, in order to best represent the families of sequences they model.

Most MSA programs use alignment steps that are based on sequence information, statistical analysis and certain evolutionary models, often neglecting structural information. Sequences with mutual identities above 20% - 30%, depending on their alignment length, can be aligned by standard alignment tools. However, existing methods show weaknesses when the mutual sequence identities fall below a soft boundary at roughly 20%, often referred to as the "twilight zone".

In the twilight zone one can no longer determine whether two aligned protein sequences are homologous or not, based only on the percent sequence identity after optimal alignment. From a plot of alignment length, *L*, versus percent sequence identity, p^I, a curve $p^I(L)$ can be defined such that most protein pairs which appear above the curve are homologous. Around the curve, unrelated pairs start to appear, and their number increases rapidly as one descends below the curve into the midnight zone. To determine the correct sequence alignment becomes increasingly difficult as the sequence identity drops below the twilight zone and into the midnight zone of sequence alignments. This indicates the need for a sequence search tool which is capable of recognizing similarities of proteins even at very low levels of sequence identity.

In order to overcome the difficulties of sequence-only alignments, we make use of the fact that the 3D-structures of homologous protein domains are more evolutionary conserved than their amino acid sequences. It is quite usual that the peptide chains of two domains with a very low sequence identity, clearly in the midnight zone, adopt almost identical conformations, which means that their main-chain atoms are superimposable with a low root mean square distance, r.m.s.d.

The inclusion of structural information is in many cases beneficial and has improved the ability to find remote relationships. Secondary structure information was used in addition to sequences to construct so called ssHMMs. Tertiary structure superimpositions were used to generate substitution matrices, to construct sequence profiles, and to build hidden Markov models.

Hidden Markov models are the most powerful of the profile methods, and have been used in a variety of ways in combination with structural information. In an attempt to locate SH2 domains in Janus kinases, Al-Lazikani et al. combined sequence alignments of close relatives with a multiple sequence alignment derived from a structural alignment of SH2 domains. From the resulting alignment they built a hidden Markov model which successfully identified SH2 domains. Their approach required manual intervention and was implemented for one domain family only.

Griffiths-Jones and Bateman compare family HMMs built from protein structure superimpositions that were obtained from the HOMSTRAD database with HMMs based on ClustalW and T-Coffee alignments of the same sequences. The sequences in their alignments cover almost the entire range of sequence identities, since no identity cutoff was applied. The authors conclude that even though the structures improve the alignments, they do not increase the ability of the HMMs to find sequence homologues.

In assessing sequence-based protocols that employ HMMs for superfamily recognition, Sillitoe et al. also include an approach that exploits multiple structural alignments from the CATH database when building the models. For each CATH superfamily they define structural subgroups of representative domains with less than 35% mutual sequence identity. Within each subgroup, the representative domains are used to make a structure-based multiple sequence alignment. Guided by the alignment, multiple sequence alignments of relatives of each representative are merged. The final alignment is converted to an HMM, called a 3D-HMM. The authors conclude that adding 3D-HMMs to their library of 1D-HMMs, built from single structural seeds, does not improve remote homologue recognition but positively affects the accuracy of sequence alignments for remote homologues.

Scheeff and Bourne compare HMMs based on structural alignments with sequence-only models. Their approach is based on the superfamily level of the Structural Classification of Proteins, SCOP, considering the first five classes only. As master sequences they select domains with at least 80 residues and a mutual BLAST E-value below 10^{-3}. For each master sequence they create a multiple sequence alignment in an iterative process, and a corresponding single-master HMM. Within each superfamily, they then compare the single-master HMMs to hidden Markov models that they derive from structurally aligned master sequences, extended with the sequence alignments to each master. They find that for structure prediction on the superfamily level, the HMMs built from structure linked alignments do not provide an overall improvement compared to sequence-only models, but that they are complementary at the edge to the twilight zone.

Similarly, Casbon and Saqi compare a single structure-based HMM that represents the entire superfamily, with multiple sequence-based HMMs, each representing an individual member of the superfamily. They base their analysis on the SCOP classification, and use pre-selected sequences with less than 10% mutual sequence identity from the ASTRAL compendium. Only superfamilies with at least five members in the 10% cutoff selection were considered. The structure-based HMM representing a superfamily is built from a structure-based alignment. The single domain HMMs within a superfamily are derived from multiple sequence alignments obtained after five iterations of PSI-BLAST. In their analysis of homologue detection, they use profile-profile methods instead of the common sequence-profile comparison. Casbon and Saqi find that, on the whole, multiple models perform better in detecting homologues, although single structure-based models display better alignment accuracy.

One conclusion that can be drawn from the work mentioned above is that the inclusion of structural information results in HMMs that perform as good as or better than HMMs without structure information. In particular, the structures improve the performance of the HMMs for detecting sequences with low sequence identities and show an improvement in coverage at low levels of false positives.

Except for Al-Lazikani et al. and Griffiths-Jones and Bateman, all studies were carried out on the superfamily level instead of the family level. However, the family level provides much more detailed domain specific information for accurate annotation. Even though the inclusion of many sequences might contribute to better statistics within the HMMs, large numbers of sequences might not be essential if one instead ensures that the HMMs are built from MSAs that contain a well balanced distribution of very diverse sequences within the same family.

Our approach is to exploit the fact that the 3D-structure of a protein is more conserved than its sequence and make use of structure-anchored hidden Markov models, saHMMs, to reliably assign structural family memberships. At the core of our approach lie multiple structure superimpositions of homologous domains belonging to the same family. In order to maximize the sequence diversity of the alignments, we only include domains whose mutual sequence identities fall below the HSSP-curve, $p^I(L,0)$. These domain sequences, called the saHMM-members, are collected into our "midnight ASTRAL set" and are used for multiple structure superimpositions.

Based purely on spatial criteria, structure-anchored multiple sequence alignments, saMSAs, are assembled and used to build saHMMs, each representing one SCOP family. We assume that the saMSAs provide a less biased indicator of the evolutionary variability at each aligned position compared to MSAs based on statistical methods. We show that only a few distantly related homologous sequences are sufficient to capture the essence of an entire domain family.

The main steps required to create the database of saHMMs are displayed in Figure 1. The database can be used in two ways: (i) A query sequence can be searched against the saHMMs in order to find which of the saHMMs gives the highest score, i.e. describes the sequence best, thus identifying the domain family the sequence most likely belongs to. In case the query sequence comprises more than one domain, one can expect one hit for each domain. (ii) If one is interested in a certain domain family, the saHMM describing this family can be used to search protein sequence databases or translations of newly sequenced genomes for hitherto unidentified members of the family. In either case, saHMMs are able to identify proteins as belonging to one specific family and not another within the same superfamily. A match provides the user not only with a family membership of the identified domain, and hence a hint to its function, but also with a probable 3D-structure.

Materials and Methods

Structural superimposition of domains

For all of structural superimpositions we use the software tool MUSTANG. The program is, in principle, able to superimpose any number of structures and can produce structure-anchored sequence alignments in msf-format, which is suitable for input to HMMER. MUSTANG proved to be best suited for our automated saHMM construction pipeline.

The midnight ASTRAL set

For the definition of homologous structural domains, we apply the SCOP classification on the family level. The SCOP database version 1.69 contains 70859 domains which are divided into 11 classes. We use only the seven true classes, which comprise 2845 domain families harbouring 67220 domains. Excluded are the "Not a true class" entries such as coiled-coil proteins, peptides, low resolution structures and designed proteins. The SCOP associated ASTRAL compendium provides Protein Data Bank, PDB-style coordinate files for individual domains.

The PDB, and, consequently, the SCOP and ASTRAL databases are highly redundant. In order to assure maximum sequence diversity within each family we include only sequences whose mutual sequence identities are equal to or less than the limiting curve $p^I(L,n)$ defined as:

$$p^I(L,n) = n + \begin{cases} 100 & \mathit{for}\ L \leq 11, \\ 480 \cdot L^{-0.32(1+e^{-L/1000})} & \mathit{for}\ 11 < L \leq 450, \\ 19.5 & \mathit{for}\ L > 450, \end{cases}$$

These selection criteria will ensure a wide evolutionary spread of the homologous representatives and avoid the bias for very similar sequences. The $p^I(L,n)$ curve is analogous to the HSSP-curve described by Rost. Here, $p^I(L,n)$ is the cutoff percentage of identical residues over an alignment length L, and n is the distance, in percent, from the curve. Two random sequences with a percent sequence identity above the original HSSP-curve, for which $n = 0$, are in the majority of the cases homologous.

A flowchart outlining the selection of representatives for the midnight ASTRAL set is depicted in Figure 2. The algorithm selects, for each family, only those domains that were determined by X-ray crystallography to a resolution of 3.6 Å or better and have mutual sequence identities equal to or less than $p^I(L,0)$.

Within each family we construct all-against-all pairwise structural superimpositions, in order to obtain pairwise structure-anchored sequence alignments from which we calculate percent sequence identities. If the sequence identity of a pair of superimposed domains falls above $p^I(L,0)$, we preliminary discard the domain with the worst resolution. If the resolutions, at which the two structures were determined, differ by less than 10% from their average, we choose the domain with the lower mean thermal factor, B-factor. In case of equal mean B-factors, one domain is chosen randomly. The mean B-factor, which is a measure for data quality, is calculated as the average of the B-factors for all Cα atoms in the domain.

After the first round of selection, all the preliminary discarded protein domains are again compared to all domains left, in order to assure that only domains with sequence identities above $p^I(L,0)$ are permanently discarded. The rationale behind this step is that in the process of removing domains, it is possible that a sequence A is removed due to high identity to sequence B. If B is later removed due to high identity to sequence C, it is conceivable that A and C have low sequence identity. Thus A has to be compared with C, and in case the identity is equal to or less than $p^I(L,0)$, both A and C must be kept. The selected domain sequences are taken as representatives for this particular family and are called *saHMM-members*. As a minimum requirement for building an saHMM, the domain family must be represented by at least two structures. We therefore exclude all families with only one representative from the midnight ASTRAL set.

Construction of saHMMs

In order to build structure-anchored hidden Markov models, we first construct a structure-anchored multiple sequence alignment of the saHMM-members within each family. The saMSAs are then used as input for HMMER version 2.2g with default parameters for `hmmbuild`. All saHMMs are calibrated using `hmmcalibrate` with default settings to obtain fitted E-values. In this way, we create one saHMM for each SCOP protein domain family represented in the midnight ASTRAL set.

Performance evaluation

From the 850 saHMMs based on $p^I(L,0)$, we removed 19 from further analysis, since their saHMM-members are the sole entries in SCOP, which means that there are no additional sequences in those families that can be used as queries. This leaves 831 saHMMs for evaluations. For analysis purposes, we constructed a subset of SCOP that contains only those sequences that belong to families with an saHMM. After excluding the saHMM-member sequences from this subset of SCOP, we obtained 40877 query sequences, called the *test-set*.

The performance at a given E-value threshold e can be evaluated with respect to the following two criteria: the *coverage*, which is expressed as the percentage of all sequences in the test-set that are matched with the correct saHMM with an E-value less than or equal to e, and the *accuracy*, which stands for the percentage of all correct hits with an E-value of at most e. Matches between a sequence and an saHMM from the same family are counted as correct hits, also called true positives (*tp*). All hits outside the family are considered as false positives (*fp*), even if they fall into the correct superfamily.

For all searches we use HMMER version 2.2g and, unless otherwise stated, the E-value cutoff is set to $e = 0.01$ for searches with sequences versus saHMMs, and $e = 0.1$ for searches with saHMMs versus test-set sequences. These E-value cutoffs proved to give accuracies of approximately 95%, which we consider as sufficient.

Determining errors per query and coverage

The number of errors per query, EPQ, is calculated as the total number of *fp* considering a certain E-value threshold *e*, divided by the total number of queries. In the case of sequence searches vs. saHMMs, the number of queries is equal to the number of sequences used for searches. In the case of searching saHMMs vs. sequences, the number of queries corresponds to the number of saHMMs, in other words, the number of families. The coverage is calculated as described in the previous section.

Construction of exclude-one-saHMMs

For the 387 SCOP families with at least three saHMM-members, we construct so called *exclude-one-saHMMs*, exo-saHMMs, by excluding one representative sequence at a time and building new saHMMs from the superimposition of the remaining domains. In this way, we obtain a collection of *n* exo-saHMMs for a family with *n* saHMM-members. We then test if each of the excluded sequences is able to find the corresponding exo-saHMM. Before searching with an excluded sequence, we exchange the full family saHMM with the exo-saHMM that lacks that sequence.

Similarly, we test if each of the exo-saHMMs is able to find the associated excluded sequence among all sequences in SCOP.

Histograms of performance per saHMM

We analyse the performance on a per saHMM basis by counting the number of families for which a given fraction of family members is matched to the correct saHMM and present the results in form of a histogram. In the histogram, we sum the number of families over ten percent coverage intervals, where the first bin contains all families in the range $0 \leq x < 10\%$, the second bin those in the range $10 \leq x < 20\%$, etc. The last bar is not binned, and contains all families with exactly 100% coverage. We carried out the analysis for three accuracy requirements: 99.5%, 90% and without requirement on accuracy.

Ability to find new members of a family

Those domain sequences that are present in SCOP 1.71 (released October 2006), but not in SCOP 1.69 (released July 2005), are used to search against the saHMMs, which are based on SCOP 1.69. In addition, we select for each domain family those query sequences that have a sequence identity equal to or less than $p^I(L,0)$ compared to the saHMM-members. These low identity sequences are used to search the saHMMs.

Comparing saHMMs to PSI-BLAST

All sequences in the midnight ASTRAL set are used, one at a time, as queries in PSI-BLAST searches. First, PSI-BLAST (blastpgp 2.2.13) is run for five iterations versus the NCBI nr-database (downloaded March 30, 2006). The resulting position specific scoring matrix, PSSM, one for each saHMM-member, is thereafter used to search SCOP version 1.69. Default parameter values are used throughout.

In order to be able to compare PSI-BLAST searches with searches of saHMMs vs. SCOP-sequences, the PSI-BLAST results obtained for each saHMM-member within the same family are pooled. Since the pooled PSSM matches can contain duplicates, we consider only non-redundant matches. In case of two or more hits to the same sequence, we keep the match with the lowest E-value.

In addition, we test the ability of PSI-BLAST to correctly assign a sequence to its family, even at low sequence identity, and compare the results to those obtained from searches with saHMM-members versus exo-saHMMs, described above. Here, a query sequence is counted as assigned to the correct family if it obtains a match to at least one other family saHMM-member. Matches to sequences outside the correct family are counted as false positives. We consider only hits to sequences within the midnight ASTRAL set.

Comparing saHMMs to Pfam HMMs

The classification of domains in Pfam is not identical to that of SCOP. Therefore, we have mapped Pfam (version 19.0, released Nov. 2005) onto SCOP 1.69. The relationships between corresponding families in the two databases are determined by finding the SCOP classification of PDB sequences that are part of Pfam-A seed alignments. For the comparison, we use as queries those sequences that are new in SCOP 1.71 and that belong to families with both an saHMM based on SCOP 1.69 and an Pfam_ls HMM, version 19.0.

Results and Discussion

The midnight ASTRAL set and corresponding saHMMs

Our midnight ASTRAL set contains 3129 low identity, non-redundant domains. The domains correspond to 850, out of 2845, SCOP “true class” families. Each family is represented by 2 to 38 domains, the saHMM-members, which are used to automatically construct one saHMM per family. The distribution of representative domains per saHMM is shown in Figure 3.

Performance evaluation of the saHMMs

The collection of saHMMs allows us to carry out two types of searches. Firstly, searching with sequences against the database of saHMMs, this allows us to find the domain content of a protein sequence. Secondly, using the saHMMs to search a collection of sequences or a sequence database, e.g., in order to scan a genome for members of a particular domain family. In the performance evaluations, both types of searches are benchmarked.

Below we demonstrate that the saHMMs are highly family specific. They can be used to assign an unknown domain to its correct family and clearly distinguish it from domains belonging to other families within the same superfamily. This task is not trivial and becomes particularly difficult if the unknown domain is distantly related to the rest of the domain sequences in the family.

The ability to identify family members

i) Searching with sequences against saHMMs

The sequences from the test-set are used for searches against the collection of saHMMs. Counting only the highest scoring hit for each search, and restricting the E-value to less than ten, we obtain a coverage and an accuracy of about 99% (see Table I). In addition, close to one third of the few false positive top hits are matches to saHMMs within the correct superfamily.

Table I also displays the results when we restrict the E-value cutoff to 0.01 and consider all hits, not just the top hit. The figures show both high coverage and that the hits obtained from a sequence search against saHMMs are highly family specific.

Using a set of saHMMs built at an even lower sequence identity cutoff, $p^{I}(L,-10)$, results in equally high specificity (98.5% accuracy), although the coverage only reaches 71.7%. This result shows that, even though the mutual sequence identities of the domains used to build the saHMMs are exceedingly low, the models correctly describe the essential characteristics of the family and are able to identify a large majority of the family members with high accuracy.

ii) Searching with saHMMs

For searches of saHMMs versus sequences we use an E-value cutoff of e = 0.1. At this E-value threshold, both the coverage and the accuracy of the saHMMs are 95% when searching for family members in the test-set (Table I). Closer analysis shows that the vast majority of the *fp* hits are matches within the correct superfamily. Only a small fraction of the saHMMs fails to recognize any family member in the test-set.

As shown in Figure 3, more than half of the saHMMs are built from only two saHMM-members. One would expect that their performance is inferior compared to the performance of the saHMMs constructed from more sequences. Surprisingly, the vast majority, 91%, of the saHMMs built from only two sequences achieve full coverage, and 97% have an accuracy of at least 99.5%.

The ability to find low sequence identity homologues

The way the sequences for the midnight ASTRAL set are selected implies that each sequence in the test-set has a pairwise sequence identity above $p^I(L,0)$ with respect to at least one sequence in the midnight ASTRAL set. In the following, we analyze the performance of the saHMMs with respect to sequences whose identity is equal to or less than $p^I(L,0)$ when compared to the saHMM-members, in other words, fall into the so called midnight zone.

In order to carry out the “search for a specific pin in a stack of needles”, we construct 2194 exclude-one-saHMMs, exo-saHMMs, for the domain families with at least three saHMM-members.

i) Low identity sequences vs. exo-saHMMs

The 2194 excluded sequences are used, one at a time, to query the collection of saHMMs, with one modification: for each of the query sequences we exchange the full family saHMM with the exo-saHMM that lacks that sequence. The search results show that 38.4% of the excluded sequences can be matched to the corresponding exo-saHMM (see Table I) and an additional 2.8% obtain hits to saHMMs belonging to the correct superfamily. Only one match falls outside the superfamily.

If we relax the E-value cutoff to 10 and consider only the top scoring hit per query, the coverage increases to 66.4% at the cost of reduced accuracy (see Table I). However, 82.5% of the top scoring matches are within the correct superfamily. The coverage values can be interpreted as the probability of assigning the correct family to a sequence with very low sequence identity to the saHMM-members.

ii) exo-saHMMs vs. low identity homologues

We evaluate the ability of each of the exo-saHMMs to find its missing sequence among the 2194 excluded sequences (see Table I). The results vary significantly for different families. For some families, all of the exo-saHMMs find their excluded sequence, while for other families none do. In summary, 29.5% of the exo-saHMMs are able to identify the missing sequence. This coverage can be interpreted as the probability of detecting a new family member with a sequence identity of at most $p^I(L,0)$ compared to each of the sequences used to build the saHMM. Taken together, the exo-saHMMs produce 826 hits of which 78.5% are within the correct domain family. As previosly, the vast majority of the false positive matches fall within the correct superfamily (see Table I).

The results of both types of searches show that the saHMMs are able to detect very low sequence identity homologues with high accuracy. The majority of the sequences for which we obtain a hit are matched to the correct family, and the matches outside the family are almost exclusively within the correct superfamily. This property demonstrates the usefulness of the saHMMs to assign the correct family to remote homologues, i.e. to find a specific pin in a stack of needles.

Performance distribution of the saHMMs

In the previous paragraphs, we focused on the overall performance of searches with sequences against saHMMs and vice versa. In this section, we analyze the performance on a per saHMM basis. One way of carrying out this analysis is to count the number of saHMMs that fulfil a certain performance requirement and present the resulting numbers in form of a histogram.

The results from searches with test-set sequences against saHMMs are summarized in Figure 4. The majority of the saHMMs show perfect performance as visualized by the right most red bar. For 635 families all family members could be matched to the correct saHMMs with an accuracy of at least 99.5%. This corresponds to 76.4% of the families evaluated.

The fact that the number of families for which we obtain full coverage increases only by 32 if we place no restrictions on accuracy, as shown by the right most dark blue bar in Figure 4, demonstrates the high quality of the saHMMs.

When investigating the ability to identify family members in a sequence database, we obtain similar results (data not shown). For example, 568 of the 831 saHMMs are able to identify all of their family members with at least 99.5% accuracy.

The results clearly demonstrate that the majority of our saHMMs are very accurate and family specific.

Ability to recognize new sequences

In order to assess the ability of the saHMMs to assign the correct domain families to as yet un-annotated sequences, we use the 4630 domain sequences that are present in SCOP version 1.71 but not in version 1.69 to search against the saHMMs. A summary of the results is presented in Table II. We find that 2761 of the sequences belong to families for which we have an saHMM in our collection, which is based on SCOP version 1.69. Among these sequences, 85.2% obtain a top score to the correct saHMM, with an E-value less than or equal to 0.01. The number of sequences obtaining correct top scores increases only marginally if we allow matches within the superfamily.

The sequences without a corresponding saHMM should not obtain any matches. In accordance with this, our search results show that only a small fraction of these orphan sequences obtain matches at all, which are exclusively to an saHMM from the correct superfamily.

Considering all sequences, 98.7% of the matches are to the correct family and all hits are within the proper superfamily.

In order to evaluate the ability to detect low sequence identity homologues, we select, for each domain family, those sequences that have a sequence identity equal to or less than $p^{I}(L,0)$, compared to the saHMM-members. This results in 458 low identity sequences belonging to families with an saHMM. Even though the sequence identity is very low, more than a quarter of the sequences match the correct saHMM with almost perfect accuracy of 99.2%. If we relax the E-value threshold to ten, then more than two thirds, 68.8%, of the low identity sequences obtain top scores to the correct saHMMs.

Performance of the saHMMs on the superfamily level

Of the few false positive hits obtained in our evaluations of the saHMMs, almost all matches fall within the correct superfamily. Considering this, we evaluate in which way the accuracy and coverage is affected on the superfamily level, when the results from the individual saHMMs are pooled within a superfamily. Our approach differs from the work of Gough *et al.* who used all individual SCOP superfamily sequences, with less than 95% mutual sequence identity, as seeds to construct one HMM from each seed and combine the search results within each superfamily.

The coverage increases from 95.0% to 96.3% if we, within each superfamily, pool the matches of a search with the saHMMs against the test-set. At the same time the accuracy improves from 94.9% to 99.7%.

If we use the test-set sequences to search against the saHMMs and pool the results within each superfamily, the coverage increases from 93.3% to 97.7%, and the accuracy from 94.6% to

99.9%. These values demonstrate that pooling the results on the superfamily level will improve the coverage and, somewhat more pronounced, the accuracy. Nevertheless, since the vast majority of the matches are correct on the family level, we decided not to combine the saHMMs into superfamilies and instead report the results on the family level. Our strategy has the advantage that the structural and functional information is more specific.

Comparison with other methods

Using saHMMs to annotate unknown human proteins

Public databases contain thousands of protein sequences that are labelled "unknown". In order to investigate the ability of the saHMMs to annotate "unknown" sequences, we searched the National Center for Biotechnology Information, NCBI, for human proteins labelled "unknown" and found 1986 such sequences (as of Nov. 2007). Of these, 232 proteins can be matched to at least one of our saHMMs, resulting in 530 annotated non-overlapping domains belonging to 102 different domain families (See Additional file 1 for a list of all matches). As before, the E-values were restricted to 0.01 and below. The classic Zinc-finger domain family (SCOP family g.37.1.1, sunid 57668) receives with 83 hits by far the most matches, which were distributed over 20 individual proteins. Each protein received between one and 15 classic Zn-finger domain hits, and, in some cases, hits to additional domain families. For 17 of the Zn-finger proteins, the NCBI annotation is incomplete in the sense that none (13 proteins, e.g., AAY14760) or not all of the Zn-finger domains (e.g., AAX93276, missed 5 of 7) are identified in the NCBI sequence entry.

Included in the list over families receiving many hits are such common domains as the EGF-type module with 68 hits and the LDL receptor-like module with 63 hits. Many domains identified by the saHMMs are involved in signalling, for example the protein kinase catalytic subunit with 17 hits, the SH3 domain obtaining 16 hits, the PDZ domain, 9 hits, and the SH2 domain with 5 hits. (A summary of all matches is provided as Additional file 2). These protein domains are often involved in cancer and other common human diseases. Through the structural information associated with the match, it might be possible to use the saHMM-members as template structures to build comparative models, thus providing a starting point for further computational and experimental analysis such as mutagenesis studies, identification of active sites and interaction surfaces, and possibly for drug design.

Comparing saHHMs to PSI-BLAST PSSMs

The saHMMs were compared to PSI-BLAST PSSMs using "Errors per Query (EPQ) versus Coverage" plots. The advantage of these graphs over Receiver Operating Characteristic, ROC, plots is that they communicate essentially the same information, while the EPQ vs. coverage plots better represent the high degrees of accuracy and the vast background of non-homologues encountered in sequence comparisons.

The aim of searching with an saHMM versus a sequence database is to identify as many family members as possible, with as few false positives as possible. In this type of search, an error per query corresponds to the number of false positives per domain family and not per single sequence. This means that an EPQ of one corresponds to one false positive per 53 sequences, since the families in our test-set harbour on average 53 sequences.

In Figure 5, we plot the EPQ vs. coverage for searches with saHMMs against sequences and E-values between zero to ten (solid curve). The dotted curve illustrates the results of PSI-BLAST PSSM searches against SCOP sequences, as described in the methods section.

The graphs show that the saHMMs are able to identify family members with few errors per query and high coverage. At an EPQ of 2.5, which corresponds to an E-value cutoff of 0.1 for the saHMMs, the coverage is about 95% whereas the coverage of PSI-BLAST is roughly 63%. In general, the EPQ values of PSI-BLAST PSSMs are larger than those of the set of saHMMs for all coverage values. The results show that at a given coverage, the saHMMs are able to accurately identify family members with clearly less errors per query compared to PSI-BLAST PSSMs. This demonstrates that a few diverse, structurally aligned sequences outperform the PSSMs built from a large number of sequences without any restrictions on diversity.

Comparing exo-saHMMs to PSI-BLAST PSSMs

In Figure 6, the EPQ values are plotted against the coverage for searches of low identity sequences vs. exo-saHMMs. The plot also contains the results obtained with PSI-BLAST PSSMs as described in the methods section. It should be noted that PSI-BLAST PSSMs have an advantage in this comparison since they are derived from searches in the NCBI's nr-database. Consequently, they can contain sequences whose mutual identity exceeds *$p^I(L,0)$*, compared to the query sequences. In this way, the PSSMs might contain bridging sequences, with a sequence identity above *$p^I(L,0)$* to both the query sequence and another saHMM-member within the same family, making it easier for the PSSMs to find these sequences.

For proper sequence annotations it is important to consider only reliable matches, i.e. with a low error rate. As can be seen in Figure 6, the exo-saHMMs have fewer errors per query than the PSSMs up to about 58% coverage, where the two curves cross at an EPQ value of 0.12. Below this value, the exo-saHMMs achieve a higher coverage, at a given EPQ, than the PSI-BLAST PSSMs. This demonstrates that the exo-saHMMs perform better at a low error rate, despite the inbuilt advantage of the PSSMs.

Performance of saHMMs compared to Pfam HMMs

In the following, we compare the performance of the saHMMs based on SCOP version 1.69 to the performance of the corresponding Pfam_ls HMMs, version 19.0. First, we select the 2597 sequences that belong to families with both an saHMM and an HMM in Pfam, from the 4630 sequences new in SCOP 1.71. When we screen these sequences against the HMMs, the correct family relationships are detected for 94% of the sequences using the saHMMs, and for 88% of the sequences using Pfam. In this comparison, we consider matches with E-values less than 10 and count only the top hits. It is of interest to note that 243 of the sequences with correct hits to saHMMs fail to obtain a match to the correct Pfam HMM. Of these 243 hits, 77 can be counted as matches within the midnight zone since they have a sequence identity of at most *$p^I(L,0)$* compared to the saHMM-members based on SCOP 1.69.

Conclusions

In a fully automated and straightforward approach, we constructed a collection of 850 structure-anchored hidden Markov models. Their main strength lies in the fact that they are built from multiple 3D-structure alignments protein domains. The structure comparisons provide structure-anchored sequence alignments even in the case of very low mutual sequence identities, without the need for string based alignment algorithms and scoring matrices. The choice of a proper multiple structure alignment method is crucial for the success of the saHMMs. After careful assessment, we decided to use the program MUSTANG .

Our approach focuses on the family level. We restrict the mutual identities of the representative sequences to *$p^I(L,0)$* or below, which guarantees a high sequence diversity among the saHMM-members at the same time as the sequence characteristics that define the entire family are preserved. We demonstrate that the saHMMs are able to identify, with high accuracy, sequences as belonging to the family they represent and are so specific that they can clearly distinguish them from members of other families, even within the same superfamily.

It might appear trivial to place a sequence into the correct family. However, considering a sequence which is distantly related to the other sequences in the family, it is much more difficult to place the unknown sequence into its correct family than into its superfamily.

The saHMMs identify newly sequenced family members as well as family members with very low sequence identity to the saHMM-members, also in cases when the saHMMs are built from only two representatives.

In an evaluation of the ability to recognize remote homologues, we find that the exo-saHMMs are better than PSI-BLAST PSSMs in recognizing low identity sequences at low error rates. We assume that the full saHMMs will perform at least as well in a real situation.

In a search with "unknown" human sequences against the saHMMs, we demonstrate that we are able to clearly identify domains for which there was no previous annotation.

Comparing corresponding saHMMs and Pfam HMMs, shows that the structure-anchored HMMs outperform Pfam in assigning the correct family membership to new sequences. In addition, the saHMMs are able to identify family relationships that are not recognized by Pfam, and vice versa. These examples show the potential of the saHMMs and that they represent an indispensable complement to existing annotation methods.

In summary, we are able to construct saHMMs for 30% of the seven true class families in SCOP, with which we cover 65% of the sequences. Without doubt, these numbers are bound to improve due to the exponential increase of deposited structures in the PDB. As new domains are added to the midnight ASTRAL set, we will be able to increase the number of saHMM-members in existing families and add saHMMs for new families to our hidden Markov model collection. The collection of saHMMs is the foundation of a publicly available server for "Family Identification with Structure-anchored HMMs", FISH, accessible from http://babel.ucmp.umu.se/fish/.

Acknowledgements

The authors thank Lixiao Wang and Marek Wilczynski for support with the FISH server and Åke Sandgren for assistance with parallel computing. Part of this research was conducted using the resources of the High Performance Computing Center North (HPC2N). B.K. acknowledges the partial support for this project by the Swedish Foundation for Strategic Research grant A3.02:13. U.H.S. acknowledges the partial support for this project by a grant from the Knowledge Foundation (KK-Stiftelsen).

References

1. Sander C, Schneider R. Database of homology-derived protein structures and the structural meaning of sequence alignment. Proteins 1991;9(1):56-68.
2. Rost B. Twilight zone of protein sequence alignments. Protein Eng 1999;12(2):85-94.
3. Wilson CA, Kreychman J, Gerstein M. Assessing annotation transfer for genomics: quantifying the relations between protein sequence, structure and function through traditional and probabilistic scores. J Mol Biol 2000;297(1):233-249.
4. Eddy SR. Hidden Markov models. Curr Opin Struct Biol 1996;6(3):361-365.
5. Park J, Karplus K, Barrett C, Hughey R, Haussler D, Hubbard T, Chothia C. Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. J Mol Biol 1998;284(4):1201-1210.
6. Doolittle RF. Of URFs and ORFs, A primer on how to analyze derived amino acid sequences: University Science Books; 1986. 103 p.
7. Mika S, Rost B. UniqueProt: Creating representative protein sequence sets. Nucleic Acids Res 2003;31(13):3789-3791.
8. Abagyan RA, Batalov S. Do aligned sequences share the same fold? J Mol Biol 1997;273(1):355-368.
9. Watson HC, Kendrew JC. The amino-acid sequence of sperm whale myoglobin. Comparison between the amino-acid sequences of sperm whale myoglobin and of human

hemoglobin. Nature 1961;190:670-672.

10. Zuckerkandl E, Pauling L. Evolving Genes and Proteins. J Theor Biol 1965;8 (2):357.
11. Rossmann MG, Argos P. A comparison of the heme binding pocket in globins and cytochrome b5. J Biol Chem 1975;250(18):7525-7532.
12. Flaherty KM, McKay DB, Kabsch W, Holmes KC. Similarity of the three-dimensional structures of actin and the ATPase fragment of a 70-kDa heat shock cognate protein. Proc Natl Acad Sci U S A 1991;88(11):5041-5045.
13. Brenner SE. A tour of structural genomics. Nat Rev Genet 2001;2(10):801-809.
14. Hargbo J, Elofsson A. Hidden Markov models that use predicted secondary structures for fold recognition. Proteins 1999;36(1):68-76.
15. Rice DW, Eisenberg D. A 3D-1D substitution matrix for protein fold recognition that includes predicted secondary structure of the sequence. J Mol Biol 1997;267(4):1026-1038.
16. Blake JD, Cohen FE. Pairwise sequence alignment below the twilight zone. J Mol Biol 2001;307(2):721-735.
17. Shi J, Blundell TL, Mizuguchi K. FUGUE: Sequence-structure Homology Recognition Using Environment-specific Substitution Tables and Structure-dependent Gap Penalties. Journal of Molecular Biology 2001;310:243-257.
18. Gnanasekaran TV, Peri S, Arockiasamy A, Krishnaswamy S. Profiles from structure based sequence alignment of porins can identify beta stranded integral membrane proteins. Bioinformatics 2000;16(9):839-842.
19. Kelley LA, MacCallum RM, Sternberg MJ. Enhanced genome annotation using structural profiles in the program 3D-PSSM. J Mol Biol 2000;299(2):499-520.
20. Al-Lazikani B, Sheinerman FB, Honig B. Combining multiple structure and sequence alignments to improve sequence detection and alignment: application to the SH2 domains of Janus kinases. Proc Natl Acad Sci U S A 2001;98(26):14796-14801.
21. Griffiths-Jones S, Bateman A. The use of structure information to increase alignment accuracy does not aid homologue detection with profile HMMs. Bioinformatics 2002;18(9):1243-1249.
22. Tångrot J, Kågstrom B, Sauer UH. Structure anchored HMMs (saHMMs) for sensitive sequence searches. Umeå: Report, UMINF-03.18, Dept. of computing science, Umeå University; 2003.
23. Sillitoe I, Dibley M, Bray J, Addou S, Orengo C. Assessing strategies for improved superfamily recognition. Protein Sci 2005;14(7):1800-1810.
24. Casbon JA, Saqi MA. On single and multiple models of protein families for the detection of remote sequence relationships. BMC Bioinformatics 2006;7:48.
25. Scheeff ED, Bourne PE. Application of protein structure alignments to iterated hidden Markov model protocols for structure prediction. BMC Bioinformatics 2006;7(1):410.
26. Mizuguchi K, Deane CM, Blundell TL, Overington JP. HOMSTRAD: a database of protein structure alignments for homologous families. Protein Sci 1998;7(11):2469-2471.
27. 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. The CATH Domain Structure Database and related resources Gene3D and DHS provide comprehensive domain family information for genome analysis. Nucleic Acids Res 2005;33(Database issue):D247-251.
28. Murzin AG, Brenner SE, Hubbard T, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247(4):536-540.
29. Chandonia JM, Hon G, Walker NS, Lo Conte L, Koehl P, Levitt M, Brenner SE. The ASTRAL Compendium in 2004. Nucleic Acids Res 2004;32(Database issue):D189-192.
30. Tångrot JE, Wang L, Kågström B, Sauer UH. Design, construction and use of the FISH server. Lecture Notes in Computer Science 2007;LNCS 4699:647–657.

31. Konagurthu AS, Whisstock JC, Stuckey PJ, Lesk AM. MUSTANG: a multiple structural alignment algorithm. Proteins 2006;64(3):559-574.
32. Eddy S. HMMER: profile HMMs for protein sequence analysis. http://hmmerwustledu/ 2003.
33. Brenner SE, Koehl P, Levitt M. The ASTRAL compendium for protein structure and sequence analysis. Nucleic Acids Res 2000;28(1):254-256.
34. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. The Protein Data Bank. Nucleic Acids Res 2000;28(1):235-242.
35. Hobohm U, Scharf M, Schneider R, Sander C. Selection of representative protein data sets. Protein Sci 1992;1(3):409-417.
36. Brenner SE, Chothia C, Hubbard TJ. Population statistics of protein structures: lessons from structural classifications. Curr Opin Struct Biol 1997;7(3):369-376.
37. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25(17):3389-3402.
38. Finn RD, Mistry J, Schuster-Bockler B, Griffiths-Jones S, Hollich V, Lassmann T, Moxon S, Marshall M, Khanna A, Durbin R, Eddy SR, Sonnhammer EL, Bateman A. Pfam: clans, web tools and services. Nucleic Acids Res 2006;34(Database issue):D247-251.
39. Gough J, Karplus K, Hughey R, Chothia C. Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure. J Mol Biol 2001;313(4):903-919.
40. Zweig MH, Campbell G. Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. Clin Chem 1993;39(4):561-577.
41. Gribskov M, Robinson NL. Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. Comput Chem 1996;20(1):25-33.
42. Brenner SE, Chothia C, Hubbard TJ. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. Proc Natl Acad Sci U S A 1998;95(11):6073-6078.
43. Tångrot J, Wang L, Kågström B, Sauer UH. FISH--family identification of sequence homologues using structure anchored hidden Markov models. Nucleic Acids Res 2006;34(Web Server issue):W10-14.

Figures

Figure 1 - Steps involved in constructing the collection of saHMMs.

For each SCOP family we select only homologous domains with low pairwise sequence identity. For these domains we generate a multiple structure superimposition. The resulting saMSA is then used as input for building an saHMM which become part of the saHMM database.

Figure 2 - Flowchart showing the construction of the midnight ASTRAL set.

For each family in SCOP we structurally superimpose pairs of family members using MUSTANG. If the resulting structure-derived sequence identity of a pair falls above $p^I(L,0)$, we preliminarily place the domain with the worse resolution into the "remove" set. In case of similar resolutions, the domain with the highest mean B-factor is put into the "remove" set. After the first round of selection, all the protein domains in the "remove" set are once more compared to the domains left. This will assure that only domains with sequence identities above $p^I(L,0)$ are permanently discarded.

Figure 3 - Number of saHMM-members per family.

A histogram showing the number of saHMMs built from 2, 3, 4, etc. selected family members. More than half of the saHMMs are constructed from two members, while only about five percent are based on more than 10 members. One saHMM, representing the Ig I set family, harbours the maximum of 38 saHMM-members.

Figure 4 - The number of saHMMs fulfilling certain performance requirements.

Results obtained when searching with the test-set versus the saHMMs using an E-value cutoff of 0.01. The number of families which reach certain levels of coverage are binned into 10% coverage intervals, and are shown for three levels of accuracy: 99.5% (red), 90% (cyan), and no requirement on accuracy (blue).

Figure 5 – saHMMs and PSI-BLAST PSSMs vs. test-set.

Single-logarithmic plot of EPQ vs. coverage for the results from searches with saHMMs (blue solid curve) and PSI-BLAST PSSMs (red dotted curve) versus test-set sequences. Here, the number of queries corresponds to the number of saHMMs, in other words, the number of families. The dotted horizontal line marks an EPQ value of 2.5, which corresponds to roughly 0.05 errors per sequence, since each family contains on average 53 members. The E-value cutoffs 10^n (with n = -6, -5,..., +1, from left to right) are marked on the blue solid curve.

Figure 6 – Low identity sequences vs. exo-saHMMs and PSI-BLAST PSSMs.

Single-logarithmic plot of EPQ vs. coverage for the results from searches with low identity sequences versus exo-saHMMs (blue solid curve) and PSI-BLAST PSSMs (red dotted curve). Note that the exo-saHMM curve lies below the PSI-BLAST curve for EPQ values less than 0.12. Below a coverage value of about 0.2 the curves are noisy and difficult to interpret due to the sparsity of data points.

Tables

Table I - Performance of the saHMMs and exo-saHMMs

	Nr of sequences or saHMMs	Accuracy (%)	Coverage (%)	fp within correct superfamily(*) (%)	hits outside correct superfamily (%)	sequences resp. saHMMs w/o hit (%)
test-set vs. saHMMs	40877	94.6	96.3	99.4	0.03	3.6
test-set vs. saHMMs, top hits	40877	99.2	98.5	32.3	0.5	0.7
saHMMs vs. test-set	831	94.9	95.0	99.5	0.03	0.6
excluded sequences vs. exo-saHMMs	2194	93.3	38.4	99.9	0.1	58.8
excluded sequences vs. exo-saHMMs, top hit	2194	76.3	66.4	22.3	18.4	12.9
exo-saHMMs vs. excluded sequences	2194	78.5	29.5	92.7	17.7	74.0

(*) Here we have calculated the percent of the false positive matches, fp, that are hits within the correct superfamily.

Table II – Performance on newly sequenced sequences

		Number of sequences	Match to correct family (%)	Match within superfamily (%)	No match at all (%)
no saHMM		1869	--	1.6	98.4
saHMM in saHMMdb	all sequences	2761	85.2	85.3	14.8
	low id. sequences	458	26.6	0.2	73.1

Additional files

Additional file 1 – saHMM matches listed by protein

The file contains the complete list of matches, with E-values less than or equal to 0.01, obtained from a search of human protein sequences labelled “unknown” vs. the collection of saHMMs.

Additional file 2 – Number of matches per saHMM

The file contains a list of saHMMs that obtain at least one hit in a search of "unknown" human protein sequences vs. the collection of saHMMs.

Figure 1.

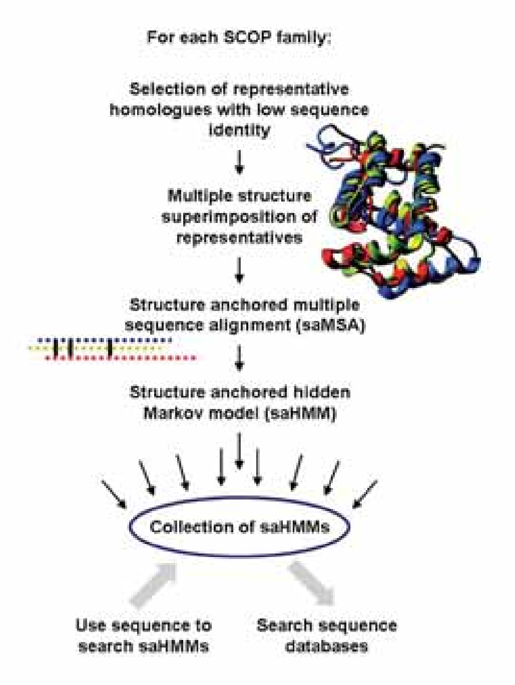

For each SCOP family:
Selection of representative homologues with low sequence identity
Multiple structure superimposition of representatives
Structure anchored multiple sequence alignment (saMSA)
Structure anchored hidden Markov model (saHMM)
Collection of saHMMs
Use sequence to search saHMMs
Search sequence databases

Figure 2.

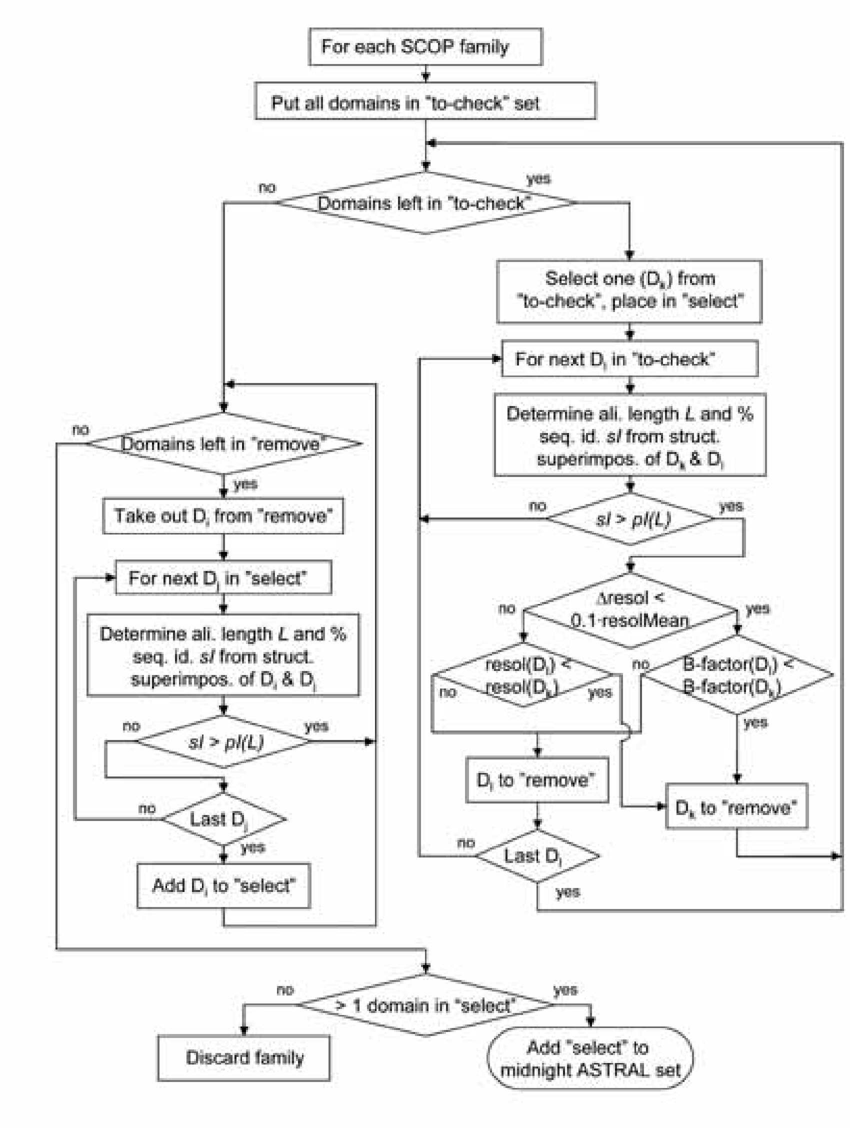

For each SCOP family
Put all domains in "to-check" set
no
yes
Domains left in "to-check"
Select one (D_k) from "to-check", place in "select"
For next D_i in "to-check"
Determine ali. length L and % seq. id. sI from struct. superimpos. of D_k & D_i
no
yes
$sI > pI(L)$
no
yes
Δresol < 0.1·resolMean
resol(D_i) < resol(D_k)
no
yes
no
B-factor(D_i) < B-factor(D_k)
yes
D_i to "remove"
D_k to "remove"
no
Last D_i
yes
no
Domains left in "remove"
yes
Take out D_i from "remove"
For next D_j in "select"
Determine ali. length L and % seq. id. sI from struct. superimpos. of D_i & D_j
no
yes
$sI > pI(L)$
no
Last D_j
yes
Add D_i to "select"
no
yes
> 1 domain in "select"
Discard family
Add "select" to midnight ASTRAL set

Figure 3.

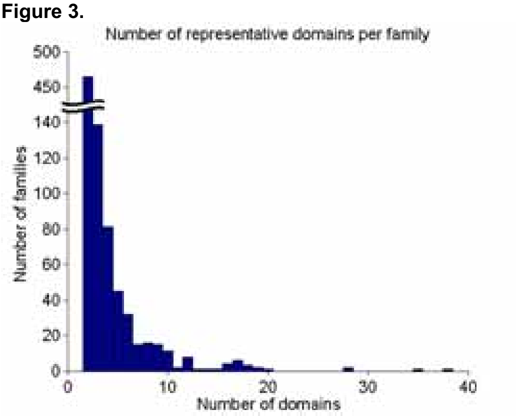


Figure 4.

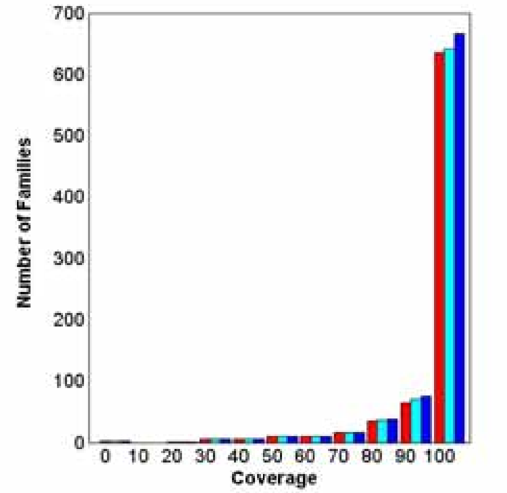


Figure 5.

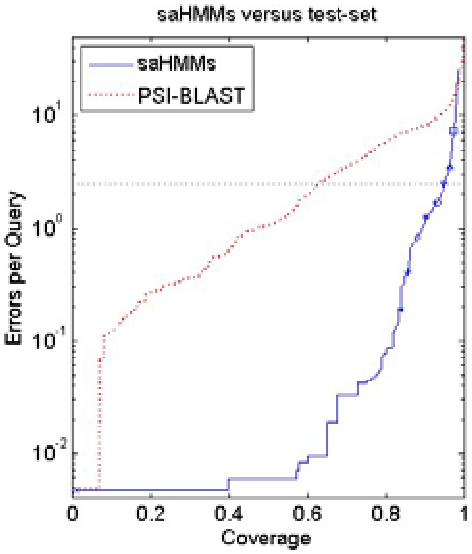
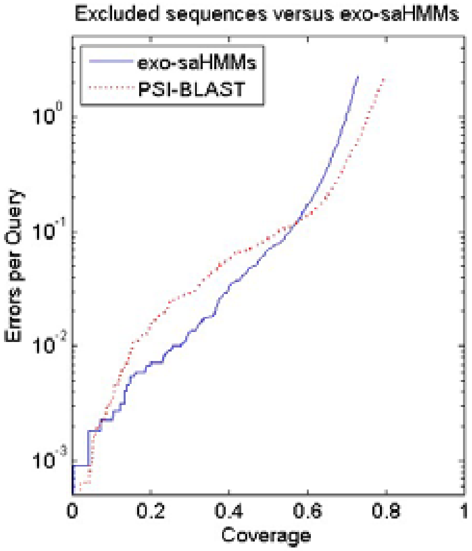


Figure 6.



II

Paper II

FISH - Family Identification of Sequence Homologues using Structure Anchored Hidden Markov Models*

Jeanette Tångrot[1,2], Lixiao Wang[1], Bo Kågström[2,3] and Uwe H. Sauer[1]

[1] *Umeå Centre for Molecular Pathogenesis,*
[2] *Department of Computing Science*
and
[3] *High Performance Computing Center North*
Umeå Universty
S-901 87 Umeå, Sweden
{jeanette,bokg}@cs.umu.se, {lixiao, uwe}@ucmp.umu.se

Abstract: The FISH server is highly accurate in identifying the family membership of domains in a query protein sequence, even in the case of very low sequence identities to known homologues. A performance test using SCOP sequences and an E-value cut-off of 0.1 showed that 99.3% of the top hits are to the correct family saHMM. Matches to a query sequence provide the user not only with an annotation of the identified domains and hence a hint to their function, but also with probable 2D and 3D structures, as well as with pairwise and multiple sequence alignments to homologues with low sequence identity. In addition, the FISH server allows users to upload and search their own protein sequence collection or to quarry public protein sequence data bases with individual saHMMs. The FISH server can be accessed at http://babel.ucmp.umu.se/fish/.

*By permission of **Nucleic Acids Research**, Oxford University Press, 2006

W10–W14 *Nucleic Acids Research, 2006, Vol. 34, Web Server issue*
doi:10.1093/nar/gkl330

FISH—family identification of sequence homologues using structure anchored hidden Markov models

Jeanette Tångrot[1,2], Lixiao Wang[1], Bo Kågström[2,3] and Uwe H. Sauer[1,]*

[1]Umeå Center for Molecular Pathogenesis, UCMP, [2]Department of Computing Science and
[3]High Performance Computing Center North, HPC2N, Umeå University, Umeå, Sweden

Received February 14, 2006; Revised March 1, 2006; Accepted April 14, 2006

ABSTRACT

The FISH server is highly accurate in identifying the family membership of domains in a query protein sequence, even in the case of very low sequence identities to known homologues. A performance test using SCOP sequences and an *E*-value cut-off of 0.1 showed that 99.3% of the top hits are to the correct family saHMM. Matches to a query sequence provide the user not only with an annotation of the identified domains and hence a hint to their function, but also with probable 2D and 3D structures, as well as with pairwise and multiple sequence alignments to homologues with low sequence identity. In addition, the FISH server allows users to upload and search their own protein sequence collection or to quarry public protein sequence data bases with individual saHMMs. The FISH server can be accessed at http://babel.ucmp.umu.se/fish/.

INTRODUCTION

The detection of homologous proteins with known function and well-determined three-dimensional (3D) structures is crucial for the correct characterization and annotation of newly sequenced proteins. Since proteins are modular and can harbour many domains, it is advisable to characterize the constituent domains rather than the protein as a whole. Existing internet resources, such as Pfam (1), Superfamily (2), SMART (3), CD search (4) and others, provide the user with versatile tools for domain identification. Nevertheless, the definition field of millions of database entries still contains remarks such as 'hypothetical', 'putative', 'unidentified' or 'function unknown'.

The FISH server can be used as a complement to existing annotation methods. One can compare a query sequence with all structure anchored hidden Markov models (saHMMs) and, in case of a match, assign family membership on the domain level for such sequences even in the case of low sequence identity.

Furthermore, it is important to discover those proteins in a database that harbour a certain domain, independent of sequence identity and annotation status. The FISH server provides such a tool, where a user can employ individual saHMMs for searching against a sequence database and obtain hits even if the sequence identity is 20% or less and falls below the so called 'twilight zone' curve, *pI* (5).

METHOD

Construction of structure anchored hidden Markov models

FISH, which stands for Family Identification with Structure anchored HMMs, is a server for the identification of sequence homologues on the basis of protein domains. At the heart of the server lies a collection of 982 saHMMs, each representing one SCOP (6) domain family (Tångrot, J., Kågström, B. and Sauer, U.H., manuscript in preparation). The saHMMs are built with HMMER 2.2g (7) from structure anchored multiple sequence alignments, saMSAs. The saMSAs are derived from multiple structure superimpositions of representative homologous domains. In order to maximize the sequence variability within each domain family, we superimposed only those domains whose mutual sequence identity falls below the 'twilight zone' curve, *pI* (5). The selected domains are hereafter called the saHMM-members. Their coordinate files were obtained from the SCOP version 1.69 associated ASTRAL compendium (8) and were superimposed with STAMP (9). Only high-quality X-ray crystal structures were used. Since at least two structures are needed for superimposition and because of the stringent sequence identity restrictions, our collection of saHMMs currently covers ~35% of SCOP families belonging to true classes. We expect this number to increase due to the exponential rate at which 3D structures become available.

*To whom correspondence should be addressed. Tel: +46 90 785 6784; Fax: +46 90 77 80 07; Email: uwe@ucmp.umu.se

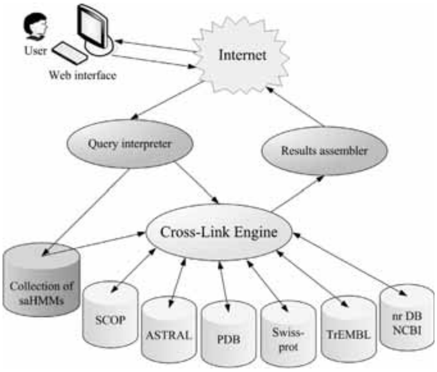


Figure 1. Schematic layout of the FISH server architecture. The user initializes a query via the web interface. The query is processed by the query interpreter, using the collection of saHMMs. The cross-link engine integrates information from the associated data bases [SCOP, ASTRAL, PDB, nr (NCBI), Swiss-Prot and TrEMBL] with the results of the query. The results assembler compiles the search results and presents them to the user via the web interface.

Brief description of the FISH server

The architecture of the FISH server is displayed in Figure 1. Flat file databases were imported into a relational data base (MySQL) and cross-linked. The MySQL database is implemented on a Linux platform. The user interface is written in Perl, PHP and JavaScript, and integrated with the Apache web server.

The user inputs a query via the web interface. The query interpreter processes the input, using the collection of saHMMs. The cross-link engine merges information from the associated databases with the results of the query. The results assembler presents the outcome of the search to the user via the web interface. The search results can be sent to the user by e-mail in the form of a www-link and are stored on the server for 24 h.

USE OF THE FISH SERVER

The organization of the FISH server input and results pages is schematically outlined in Figure 2 and described in the following.

Sequence vs. saHMM search

Using the FISH server for a sequence vs. saHMMs search is straightforward. The user is required to enter an amino acid sequence in FASTA or text format, or to upload a sequence file. The *E*-value cut-off is adjustable and determines the level of significance of the reported hits.

The FISH search results are presented in a hierarchical manner (see Figure 2). At the top of the results hierarchy is the ‘overview of results’ page (see Figure 3). It contains a table of all matches, sorted by ascending *E*-values up to the selected *E*-value cut-off. The lengths of the schematic arrows below the table correspond to the query sequence length. For each found domain, the position of the matching sequence interval is schematically marked by a coloured box. By following the links on the overview page the user obtains increasingly detailed information about each match.

In the table displayed in the ‘overview of results’ window, each saHMM identifier links to the SCOP lineage of that domain family as well as to a table listing the saHMM-members (Figure 2, left hand side, and Figure 4). Each entry in the saHMM-member field links to a saHMM-based pairwise sequence alignment of the query with that member and further to links providing coordinate information.

The chain identifier field links to a page with the sequence of the ASTRAL domain, followed by the sequence contained in the protein data bank file with the ASTRAL sequence interval marked in orange. This page also provides a link to the corresponding NCBI sequence entry.

The Coordinate icon in the table leads the user to an interactive Java window running Jmol version 10.00 (http://www.jmol.org) where the domain structure of the saHMM-member can be visualized. The user can rotate the structure and analyze it by zooming in on details or by applying a variety of colouring schemes and display options.

The coloured boxes on the sequence arrows in the ‘overview of results’ window lead the user to alignments of the query sequence with the saHMM consensus sequence. Links on this page lead the user to a sequence alignment of the query sequence with the saMSA used to build the saHMM (right hand side of Figure 2). The multiple sequence alignment can be viewed in different formats such as Stockholm, MSF and A2M.

It is also possible to view all pairwise sequence alignments of the query sequence with the individual saHMM-members. All alignments are anchored on the saHMM.

Using the SCOP sequences to test the performance of the server we found that in 99.3% of the cases the top hit matches the correct saHMM, choosing an *E*-value cut-off of 0.1. The matches obtained in a sequence vs. saHMM search provide the user with a classification on the SCOP family level and outline structurally defined, putative domain boundaries in the query sequence. This information can be used for sequence annotation, to design mutation sites, to identify soluble domains, to find structural templates for homology modelling and possibly for structure determination by molecular replacement.

Performance test on new sequences

In the following we assess the ability of the saHMMs to assign the correct domain family membership to newly sequenced proteins. For this purpose we used the 24 957 domain sequences that are contained in SCOP 1.69 (released July 2005) but not in SCOP 1.61 (released Nov. 2002), to quarry the collection of 682 saHMMs based on SCOP 1.61. Here and in the following two paragraphs we consider a hit only if it is the top match with an *E*-value equal to or better than 0.1.

Using the classification of SCOP 1.69 we find that 14 173 of the query sequences (57%) belong to domain families for which we have a saHMM based on SCOP 1.61. Ideally, all of these sequences should find a match to the correct family saHMM.

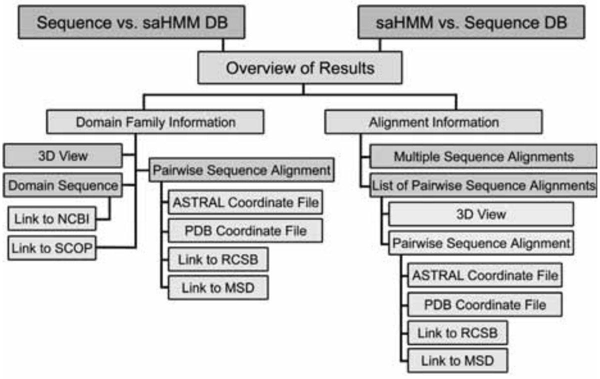
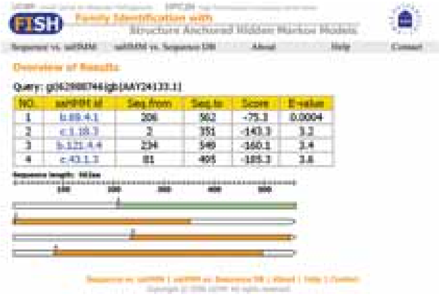


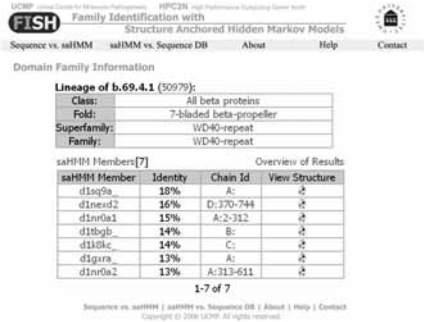
Figure 2. Organization of the FISH server input and result pages. The result pages are similar for a search of a query sequence versus the collection of saHMMs and for a search with a saHMM versus a sequence database. The information available can be roughly divided into domain family information (left branch) and alignment information (right branch). The domain family information includes SCOP classification, the sequences and 3D structures of the saHMM-members, and pairwise sequence alignments of the query to each member. The alignment information provides multiple and pairwise alignments of the query sequence to the consensus sequence extracted from the saHMM and the sequences used to build the saHMM. All alignments are anchored on the saHMM. Links to relevant data bases are provided.



Query: gi|62988746|gb|AAY24133.1|

NO.	saHMM id	Seq.from	Seq.to	Score	E-value
1	b.69.4.1	206	562	-75.3	0.0004
2	c.1.18.3	3	351	-143.3	3.2
3	b.121.4.4	234	549	-160.1	3.4
4	c.43.1.3	81	495	-185.3	3.6

Figure 3. Overview of results page. This page contains a table of all matches, and a graphical representation of the matches mapped onto a sequence arrow. The position of the matching sequence interval is marked by a colour coded box. Green corresponds to *E*-values <0.1, yellow to an *E*-value interval between 0.1 and 1.0 and orange to an *E*-value >1.0. By following the links on the overview page the user obtains more detailed information about each match, such as the SCOP lineage, pairwise and multiple sequence alignments, and 3D structures of domain members. Shown is a search carried out with AAY24133.1, a human protein labelled 'unknown'.


UCMP
HPC2N
FISH
Family Identification with
Structure Anchored Hidden Markov Models
Sequence vs. saHMM
saHMM vs. Sequence DB
About
Help
Contact
Domain Family Information

Lineage of b.69.4.1 (50979):

Class:	All beta proteins
Fold:	7-bladed beta-propeller
Superfamily:	WD40-repeat
Family:	WD40-repeat

saHMM Members[7] Overview of Results

saHMM Member	Identity	Chain Id	View Structure
d1sq9a_	18%	A:	
d1nexd2	16%	D:370-744	
d1nr0a1	15%	A:2-312	
d1tbgb_	14%	B:	
d1k8kc_	14%	C:	
d1gxra_	13%	A:	
d1nr0a2	13%	A:313-611	

1-7 of 7

Sequence vs. saHMM | saHMM vs. Sequence DB | About | Help | Contact
Copyright © 2006 UCMP. All rights reserved.

Figure 4. Domain family information page. The SCOP lineage of the domain family is shown, as well as a table listing the saHMM-members. Each saHMM-member links to a pairwise sequence alignment of the query with the member, anchored on the saHMM and to links with coordinate information. The chain entry shows the sequence of the saHMM-member. The domain structures of the saHMM-members can be visualized interactively by following the link under view structure.

Our results show, that 10 513 sequences (74%) are able to identify the correct saHMM as their top hit. This number increases to 10 737 sequences (76%) if we accept matches on the superfamily level as well. Of the 10 784 domain sequences for which we do not have a saHMM (as of version 1.61), 183 sequences (2%) found a match to a saHMM within the correct superfamily. No hit was obtained for 10 561 sequences (98%), which demonstrates that our saHMMs are very domain family specific.

The combined searches resulted in a total of 11 202 hits of which 10 513, i.e. 94% of all matches, were to the correct family saHMM. An additional 407 hits (4%) were correct on the superfamily level.

Comparing saHMMs with Pfam HMMs

To compare the performance of the FISH server with Pfam, we used saHMMs based on SCOP 1.61 and the corresponding Pfam_ls HMM release (version 7.8, released November

2002). Since the definition of a SCOP family differs from the Pfam definition, the relationships between SCOP and Pfam families were determined by finding the SCOP classification of PDB sequences that are part of the Pfam-A alignments. Of the 24 957 sequences new in SCOP 1.69 compared with version 1.61, a total of 11 592 sequences belong to families with both an HMM in Pfam and a saHMM, and are used as query sequences. In the following we consider only top hits with an *E*-value <10 as matches.

The correct family relationships were detected for 9574 of the sequences (83%) using the saHMMs and for 10 128 sequences (87%) using Pfam. It is of interest to note that 812 of the sequences with hits to the correct saHMM did not find the correct HMM in Pfam.

Detecting remote sequence homologues

We further selected, for each domain family, those sequences in the set of 11 592 query sequences that had a sequence identity below the 'twilight zone' curve compared with the saHMM-members based on SCOP 1.61. This left us with 3247 new low identity sequences, of which 2014 sequences (62%) obtained hits to the correct family saHMMs even though the sequence identity to the saHMM-members is very low. Interestingly, 79 of these relationships were not detected by Pfam, despite the possibility that some of the query sequences could have a sequence identity above *pI* to Pfam-A seed sequences.

saHMM searched vs. sequence database

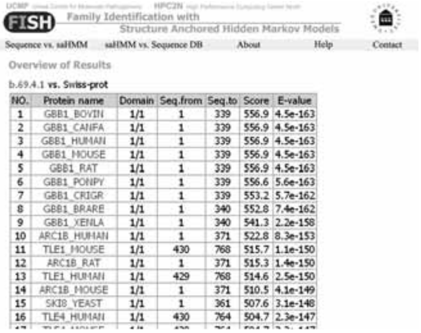
By choosing a saHMM that represents a particular SCOP domain family to search a sequence database, one can identify members of that domain family within protein sequences. In this way it is possible to identify previously un-annotated sequences on the domain family level, even in case of very low sequence identities.

The input page of the saHMM vs. sequence database search is divided into two parts. To the left is a section with several options for selecting a saHMM to use for the search, and to the right is the actual input section.

There are several ways of choosing the saHMM to search with. If one knows which SCOP domain family to use, and how to find it in the SCOP classification, the saHMM can easily be located by browsing the classification tree. Otherwise, the saHMM can be located using the free text search option. All SCOP domain families whose description matches the text search are listed. Those with a saHMM can be selected for searching.

Alternatively, the name of the saHMM can be written directly in the input field on the right. The user can also select which sequence database to search against and input an appropriate cut-off for the *E*-value.

The results are reported in the form of a table (see Figure 5), where the matches are sorted by *E*-value with the best hit listed first. Above the results table, the user can follow a link to information about the domain family as well as sequence and structural information about the domains used to build the saHMM.



Overview of Results

b.69.4.1 vs. Swiss-prot

NO.	Protein name	Domain	Seq.from	Seq.to	Score	E-value
1	GBB1_BOVIN	1/1	1	339	556.9	4.5e-163
2	GBB1_CANFA	1/1	1	339	556.9	4.5e-163
3	GBB1_HUMAN	1/1	1	339	556.9	4.5e-163
4	GBB1_MOUSE	1/1	1	339	556.9	4.5e-163
5	GBB1_RAT	1/1	1	339	556.9	4.5e-163
6	GBB1_PONPY	1/1	1	339	556.6	5.6e-163
7	GBB1_CRIGR	1/1	1	339	553.2	5.7e-162
8	GBB1_BRARE	1/1	1	340	552.8	7.4e-162
9	GBB1_XENLA	1/1	1	340	541.3	2.2e-158
10	ARC1B_HUMAN	1/1	1	371	522.8	8.3e-153
11	TLE1_MOUSE	1/1	430	768	515.7	1.1e-150
12	ARC1B_RAT	1/1	1	371	515.3	1.4e-150
13	TLE1_HUMAN	1/1	429	768	514.6	2.5e-150
14	ARC1B_MOUSE	1/1	1	371	510.5	4.1e-149
15	SKI8_YEAST	1/1	1	361	507.6	3.1e-148
16	TLE4_HUMAN	1/1	430	764	504.7	2.3e-147

Figure 5. saHMM vs. sequence database search. The results for the search with the saHMM representing the SCOP family b.69.4.1 (50979) are reported in the form of a table listing the matches sorted by *E*-value. Only part of the table is shown in the figure. Above the results table is a link to information about the domain family as well as sequence and structural information about the domains used to build the saHMM. Each protein name contains a link to the corresponding sequence entry, an alignment of the matching sequence to the saHMM consensus and the option to view both multiple and pairwise alignments anchored on the saHMM.

Each protein name in the results table is linked to the corresponding sequence entry, in which the matching sequence interval is marked in orange. An alignment of the matching sequence to the saHMM consensus is shown below the sequence, with the option to view both multiple and pairwise alignments anchored on the saHMM. In the pairwise alignments view, the sequence identity of the found match to each saHMM-member is displayed in a table. From there, links allow the user to view the structure of the members and to obtain coordinate information.

A search with a saHMM vs. SwissProt can take anything from 15 min up to ∼9 h. Searching TrEMBL, which is about 10 times larger, takes considerably longer. In order to minimize the waiting time for the user, we pre-calculated the searches of all 982 saHMMs vs. SwissProt, TrEMBL and the NCBI non-redundant database, nr, using an *E*-value cut-off of 100. Depending on the *E*-value choice of the user, the results are extracted and presented up to that value.

In addition, users can choose to upload and search their own protein sequence databases.

SUMMARY

The FISH server is a versatile tool with a dual function. On the one hand, the user can perform sensitive sequence searches versus a collection of saHMMs, which can provide matches even within the 'midnight zone' of sequence alignments. On the other hand, the user can choose one of the saHMMs to perform a search against a protein sequence data base. Since the saHMMs are based on structure anchored multiple sequence alignments, the alignment of the query to the saHMM-members can be used to draw conclusions about the probable secondary and tertiary structure of the query sequence.

A comparison of FISH saHMMs with Pfam HMMs shows that the methods are comparable in their ability to assign family memberships. Our findings also show that each collection of HMMs can assign family memberships to sequences that are missed by the other, thus complementing each other.

Further we demonstrate that for sequences with very low sequence identity to the saHMM-members a correct assignment was made for about 62% of the sequences. This demonstrate the ability to detect remote homologues on the domain family level.

ACKNOWLEDGEMENTS

The authors thank our colleagues at UCMP for critically scrutinizing the FISH server and for suggestions for improvements. Part of this research was conducted using the resources of the High Performance Computing Center North (HPC2N). U.H.S. acknowledges the partial support for this project by a grant from the Knowledge Foundation (KK-Stiftelsen) B.K. acknowledges the partial support for this project by the Swedish Foundation for Strategic Research grant A3.02:13. Funding to pay the Open Access publication charges for this article was provided by the Knowledge Foundation (KK-Stiftelsen).

Conflict of interest statement. Declared.

REFERENCES

1. 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. and Eddy,S.R. (2004) The Pfam protein families database. *Nucleic Acids Res*., **32**, D138–D141.
2. Madera,M., Vogel,C., Kummerfeld,S.K., Chothia,C. and Gough,J. (2004) The SUPERFAMILY database in 2004: additions and improvements. *Nucleic Acids Res*., **32**, D235–D239.
3. Letunic,I., Copley,R.R., Pils,B., Pinkert,S., Schultz,J. and Bork,P. (2006) SMART 5: domains in the context of genomes and networks. *Nucleic Acids Res*., **34**, D257–D260.
4. Marchler-Bauer,A. and Bryant,S.H. (2004) CD-Search: protein domain annotations on the fly. *Nucleic Acids Res*., **32**, W327–W331.
5. Rost,B. (1999) Twilight zone of protein sequence alignments. *Protein Eng*., **12**, 85–94.
6. Murzin,A.G., Brenner,S.E., Hubbard,T. and Chothia,C. (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol*., **247**, 536–540.
7. Eddy,S.R. (1998) Profile hidden Markov models. *Bioinformatics*, **14**, 755–763.
8. Chandonia,J.M., Hon,G., Walker,N.S., Lo Conte,L., Koehl,P., Levitt,M. and Brenner,S.E. (2004) The ASTRAL Compendium in 2004. *Nucleic Acids Res*., **32**, D189–D192.
9. Russell,R.B. and Barton,G.J. (1992) Multiple protein sequence alignment from tertiary structure comparison: assignment of global and residue confidence levels. *Proteins*, **14**, 309–323.

III

Paper III

Design, Construction and Use of the FISH Server [*]

Jeanette Tångrot[1,2], Lixiao Wang[1], Bo Kågström[2,3] and Uwe H. Sauer[1]

[1] Umeå Centre for Molecular Pathogenesis,
[2] Department of Computing Science
and
[3] High Performance Computing Center North
Umeå Universty
S-901 87 Umeå, Sweden
{jeanette,bokg}@cs.umu.se, {lixiao, uwe}@ucmp.umu.se

Abstract: At the core of the FISH (**F**amily **I**dentification with **S**tructure anchored **H**idden Markov models, saHMMs) server lies the midnight ASTRAL set. It is a collection of protein domains with low mutual sequence identity within homologous families, according to the structural classification of proteins, SCOP. Here, we evaluate two algorithms for creating the midnight ASTRAL set. The algorithm that limits the number of structural comparisons is about an order of magnitude faster than the all-against-all algorithm. We therefore choose the faster algorithm, although it produces slightly fewer domains in the set. We use the midnight ASTRAL set to construct the structure-anchored Hidden Markov Model data base, saHMM-db, where each saHMM represents one family. Sequence searches using saHMMs provide information about protein function, domain organization, the probable 2D and 3D structure, and can lead to the discovery of homologous domains in remotely related sequences.

The FISH server is accessible at `http://babel.ucmp.umu.se/fish/`.

[*] By permission of Springer-Verlag, Berlin © 2007 Springer-Verlag

Design, Construction and Use of the FISH Server

Jeanette Tångrot[1,2], Lixiao Wang[1], Bo Kågström[2,3], and Uwe H. Sauer[1]

[1] Umeå Centre for Molecular Pathogenesis
[2] Department of Computing Science
[3] High Performance Computing Center North (HPC2N)
Umeå University, S-901 87 Umeå, Sweden
jeanette@cs.umu.se, lixiao@ucmp.umu.se, bokg@cs.umu.se, uwe@ucmp.umu.se

Abstract. At the core of the FISH (**F**amily **I**dentification with **S**tructure anchored **H**idden Markov models, saHMMs) server lies the midnight ASTRAL set. It is a collection of protein domains with low mutual sequence identity within homologous families, according to the structural classification of proteins, SCOP. Here, we evaluate two algorithms for creating the midnight ASTRAL set. The algorithm that limits the number of structural comparisons is about an order of magnitude faster than the all-against-all algorithm. We therefore choose the faster algorithm, although it produces slightly fewer domains in the set. We use the midnight ASTRAL set to construct the structure-anchored Hidden Markov Model data base, saHMM-db, where each saHMM represents one family. Sequence searches using saHMMs provide information about protein function, domain organization, the probable 2D and 3D structure, and can lead to the discovery of homologous domains in remotely related sequences.
The FISH server is accessible at `http://babel.ucmp.umu.se/fish/`.

1 Introduction

Genome sequencing projects contribute to an exponential increase of available DNA and protein sequences in data bases. Millions of sequence entries contain remarks such as “hypothetical”, “unidentified”, or “unknown”. It is therefore crucial to develop accurate automated sequence annotation methods. For proper characterization of newly sequenced proteins it is important to associate them with homologous proteins of well characterized functions and possibly high quality three dimensional (3D) structures. Proteins are modular and can harbour many domains. Consequently, it is advisable to characterize the constituent domains rather than the protein as a whole. Existing resources, such as Pfam [1], Superfamily [7], SMART [6] and others, provide the user with versatile tools for domain identification. Common for these methods is that they use protein sequence alignments that include as many sequences as possible, even with high sequence identity of up to 95%, to construct hidden Markov models, HMMs. At the core of our approach lies a data base of structure-anchored hidden Markov models, saHMMs. In contrast to the other methods, we derive structure anchored multiple sequence alignments, saMSAs, exclusively from multiple structure superimpositions of protein domains within SCOP families [9]. Only spatial distance criteria are considered to find matching residues and to deduce the multiple sequence alignments from which the saHMMs are built. Great care is taken to ensure sequence diversity among the domains by including only such members with a mutual sequence identity below a certain cut-off value. We call the data set containing the low mutual sequence identity domains the "midnight ASTRAL set", since it was derived using the ASTRAL compendium [2]. We have made the saHMM data base, saHMM-db, publicly available through the FISH server, which has been introduced and briefly described earlier [13]. FISH, which stands for Family Identification with Structure-anchored HMMs, is a versatile server for the identification of domains in protein sequences. Here, we describe the algorithms behind the server in more detail, in particular the creation of the midnight ASTRAL set. In addition, we present a layout of the cross-linking of the underlying data bases and describe in more detail how to use the server.

2 The Midnight ASTRAL Set and Selection Algorithms

The midnight ASTRAL set is the non-redundant collection of representative domains used to construct the saHMMs. In order to maximize the sequence variability within each SCOP domain family [9], we included only domains with low mutual sequence identities, below the "twilight zone" curve, $p^I(L, 0)$ [10],[8]:

$$p^I(L,n) = n + \begin{cases} 100 & for \;\; L \leq 11, \\ 480 \cdot L^{-0.32 \cdot \left(1+e^{-L/1000}\right)} & for \;\; 11 < L \leq 450, \\ 19.5 & for \;\; L > 450. \end{cases} \qquad (1)$$

The function $p^I(L, 0)$ defines the limit of percent sequence identity for clearly homologous protein sequences, as a function of the alignment length L.

To construct the midnight ASTRAL set, representative domains must be selected for each of the 2845 SCOP families belonging to true classes. Individual families can harbour as few as one domain and as many as 1927 domains. We have evaluated two methods for selecting saHMM-members into the midnight ASTRAL set. Both methods are modified versions of the algorithms described by Hobohm *et al.* [4]. The algorithms select, for each SCOP family, only those domains that were determined by X-ray crystallography to a resolution of 3.6 Å or better, and have mutual sequence identities equal to or less than $p^I(L, 0)$.

Within each family we construct pairwise structural superimpositions in order to obtain the percent sequence identities. The coordinate files of the domains are obtained from the ASTRAL compendium [2] corresponding to SCOP version 1.69 [9]. We have evaluated several structure alignment programs, and found that, currently, MUSTANG [5] results in the best performing saHMMs (to be published elsewhere). In case the program fails to align two structures, the pair of domains is treated like a pair with too high sequence identity. As a minimum requirement for building an saHMM, the SCOP domain family must be represented by at least two structures. Therefore, all families with only one representative were excluded from the midnight ASTRAL set.

All computations were done in parallel, using up to 20 processors on the HPC2N Linux cluster Seth. The compute nodes on Seth are AMD Athlon MP2000+ with 1GB of memory per dual node, connected in a high-speed SCALI network.

2.1 Algorithm 1 for Selecting saHMM-Members

Algorithm 1 is designed to limit the number of structural comparisons. It works by removing one of the domains in a pair from further consideration, if the mutual sequence identity falls above $p^I(L, 0)$.

Outline of Algorithm 1

```
1. Collect all family members with < 3.6 Å resolution into to-be-checked set.
2. Take domain d1 from to-be-checked set, place in select set.
3. For each other domain d2 in to-be-checked set.
   (a) Pairwise structural alignment of d1 and d2 to determine sequence identity $sI$ and alignment length $L$.
   (b) If $sI > p^I(L, 0)$ then dToRemove = selectOne(d1,d2).
       i. place dToRemove in to-remove set.
       ii. if dToRemove = d1 repeat from 2.
4. Repeat from 2 until no more domains remain in to-be-checked set.
```

In order to retain the highest quality structures for constructing optimal structure superimpositions as the basis for the saHMMs, the algorithm selects the domain with the better resolution. In cases where the resolution values of the structures to be compared are too similar, i.e., they differ by less than 10% of their average, we exclude the domain with the higher mean thermal factor, B-factor. This rule applies in particular to domains extracted from the same PDB (Protein Data Bank) file. The mean B-factor reflects the data quality and is here calculated as the arithmetic mean of the B-factors for all C_α atoms within the domain. The function `selectOne` is used to select which domain to remove in case of high sequence identity.

Outline of function `selectOne`

```
1. Read in domains to compare: d1 and d2
2. if |resolution(d1) - resolution(d2)| < 0.1·mean(resolution(d1), resolution(d2))
   (a) if the mean B-factor for d1 is smaller than the mean B-factor of d2, then set dToRemove = d2
   (b) else set dToRemove = d1
3. else if resolution of d2 is poorer than that of d1, then set dToRemove = d2
4. else set dToRemove = d1
```

After the first round of selection, all the preliminary discarded protein domains stored in the to-remove set are again compared to all domains in the select set, in order to assure that only domains with sequence identities above $p^I(L, 0)$ are permanently discarded. The rationale behind this additional step is that in the process of removing domains, it is possible that a domain A is removed due to high sequence identity to domain B. If B is later removed due to high sequence identity to domain C, it could be that A and C have low mutual sequence identity. Thus A must be compared with C, and in case the identity is equal to or less than $p^I(L, 0)$ both A and C must be kept.

2.2 Algorithm 2 for Selecting saHMM-Members

We evaluated a second algorithm, called Algorithm 2, which is designed to maximize the number of representative domains. Using Algorithm 2, one first fills an $n \times n$ score matrix M based on all-against-all structural comparisons of all n members within a particular SCOP family. An entry M_{ij} is a measure of the level of sequence identity and the relative data quality of domains d_i and d_j, and is defined as:

$$M_{ij} = \begin{cases} 1 & \text{if} \quad i = j, \\ 0 & \text{if} \quad sI \le p^I(L, 0), \\ 1 + 1/n & \text{if} \quad d_j = \texttt{dToRemove}, \\ 1 - 1/n & \text{if} \quad d_i = \texttt{dToRemove}. \end{cases} \tag{2}$$

Which domain to remove in case of too great sequence identity is determined using the same procedure `selectOne` as described for Algorithm 1. To select representative domains using M, we remove in each step the domain similar to most other domains, until no more similarities can be detected. The domain d_k, corresponding to row index k in M, which is similar to most other domains is the one with the highest row sum:

$$k = \text{argmax}_i(\sum_j M_{ij}). \tag{3}$$

Removing the domain d_k from the set corresponds to setting elements $M_{ki} = 0$ and $M_{ik} = 0$ for all i, including the diagonal element M_{kk}. The process is finished when $\max_i(\sum_j M_{ij}) = 1$. The representative domains are those with 1 on the diagonal ($M_{yy} = 1$ for all representatives y). For reasons described in Algorithm 1, all removed domains are checked once more against all selected domains to make sure that no representatives were mistakenly discarded.

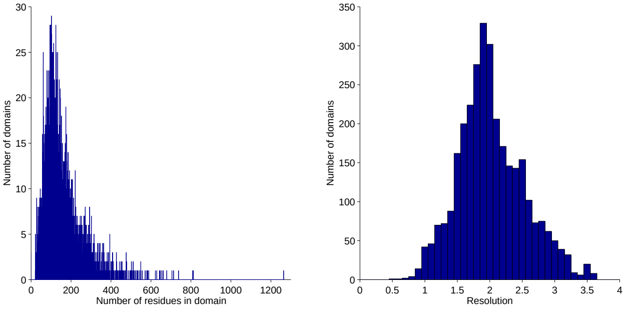
2.3 Comparing Algorithm 1 and Algorithm 2

Calculations using Algorithm 1 result in 3129 domains in the midnight ASTRAL set, representing 850 different SCOP domain families. These families cover 65% of the SCOP domains and correspond to 30% of the SCOP families belonging to true classes. Algorithm 2 gives 3293 domains in the midnight ASTRAL set, which represent 894 SCOP domain families. These families cover about 60% of SCOP domains and correspond to 31% of the true class SCOP families.

The advantage of Algorithm 2 is that it produces more saHMM-members for the midnight ASTRAL set. However, it is time expensive due to the all-against-all structural comparisons, which cause the problem to scale quadratically with the number of domains. It was not practical to use Algorithm 2 for the four very largest families, each harbouring more than 600 domains. Even so, the computing time used to select representative domains with Algorithm 2 exceeded the total time used by Algorithm 1 by an order of magnitude. We therefore decided against Algorithm 2, and will from now on use Algorithm 1 to select saHMM-members, even though Algorithm 1 results in a slightly reduced coverage of SCOP families.

2.4 Analysis of the Midnight ASTRAL Set

In Fig. 1(a) the distribution of lengths of domains within the midnight ASTRAL set selected with Algorithm 1 is displayed. The sharp peak shows that the most common sequence length of the saHMM-members is about 100 residues. The length varies from 21 amino acids for the shortest domain up to 1264 residues for the longest. In Fig. 1(b) the distribution of resolutions at which the structures of the domains were determined is displayed. The majority of the crystal structures from which the domains are extracted fall into the resolution range between 1.5 to 2.5Å. This assures a high confidence in the determined structures.



(a)

(b)

Fig. 1. Distribution of (a) sequence lengths and (b) resolutions among domains in the midnight ASTRAL set.

3 The saHMM Data Base

The construction of structure-anchored Hidden Markov Models, saHMMs, requires three major steps. First, the non-redundant midnight ASTRAL set must be generated as was described above. Then a multiple 3D superimposition of the peptide chains of these domains, called the saHMM-members, is constructed. By using only spatial criteria to compare their structures, it is possible to match those amino acids that are from different chains and in close spatial vicinity, into a structure anchored multiple sequence alignment (see also [12]). The final step involves building the saHMMs from the deduced structure-anchored multiple sequence alignment.

The coordinate files of the saHMM-members are obtained from the ASTRAL compendium corresponding to SCOP version 1.69. The domains are superimposed with MUSTANG [5] and the saHMMs are built using HMMER 2.2g [3].

We implemented several Perl programs in order to automate the process from raw SCOP family classification of domains, through the construction of the midnight ASTRAL set, to the creation and testing of the saHMMs. The programs perform tasks such as detecting and correcting inconsistencies between the notations used in SCOP and the ASTRAL coordinate files, standardizing the notation used in the coordinate files and parsing of results to convert output from one program to input for another.

3.1 Coverage of SCOP

Since at least two structures are needed for superimposition, and because of the stringent sequence identity restrictions, our collection of saHMMs currently includes 850 saHMMs, which cover about 30% of the 2845 SCOP families belonging to true classes and 65% of the 67210 domain sequences. We expect these numbers to improve due to the exponential increase of deposited 3D structures.

4 The FISH Server

4.1 Design of the FISH Server

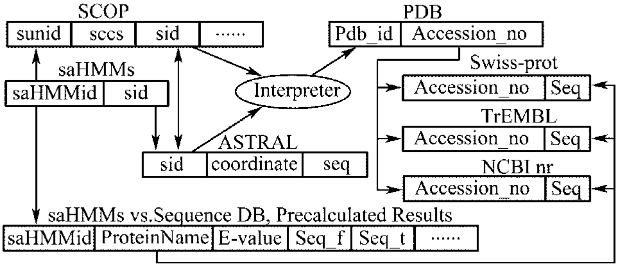


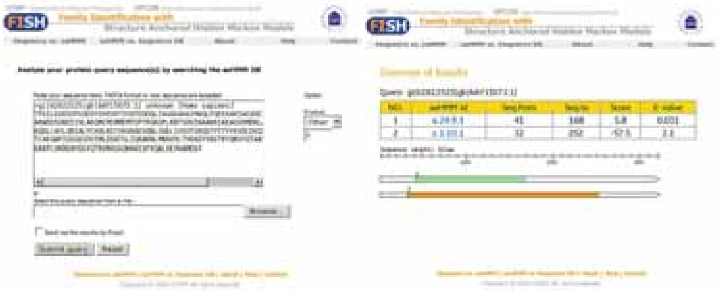
Fig. 2. Schematic view of the data base cross-linking used in the FISH server.

Flat file data bases were imported into a relational data base (MySQL implemented on a Linux platform) and cross-linked (Fig. 2). The user interface is written in Perl, PHP, and JavaScript and integrated with the Apache web server. The user inputs a query via the web interface. The query interpreter analyzes the input, using the collection of saHMMs. The cross-link engine merges information from the associated data bases with the results of the query. The results assembler presents the outcome of the search to the user via the web interface. The search results can also be sent to the user by e-mail in the form of a www-link and are stored on the server for 24 hours.

4.2 How to Use the FISH Server

Sequence Searches vs. the saHMM-db

Using the FISH server, a user can compare a query sequence with all models in the saHMM-db. Matches obtained in such a search provide the user with a classification on the SCOP family level and outline structurally defined, putative domain boundaries in the query sequence. This information is useful for sequence annotation, to design mutations, to identify soluble domains, to find structural templates for homology modelling and possibly for structure determination by molecular replacement.

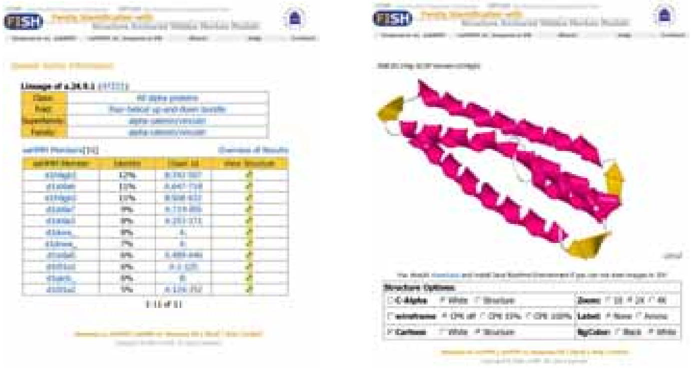


(a) (b)

Fig. 3. Sample (a) input and (b) results pages from a sequences vs saHMMs search.

Fig. 3(a) displays an example of the input page. The user enters one or more query sequences and can select an E-value cut-off for the results. The E-value of a hit is the expected number of false matches having at least the same score as the hit, and hence is a measure of the confidence one can have in the hit. The closer the E-value is to zero, the more the match can be trusted. In the 'overview of results' page (Fig. 3(b)) the list of matches is sorted in increasing order with respect to the E-value, up to the chosen cut-off. When selecting one entry from the list, the family specific information for that match is displayed (Fig. 4(a)). The top table provides information about the SCOP classification. It is followed by a table listing all saHMM-members of this family together with details about, for example, the percent sequence identity of the query sequence aligned to the member. For each saHMM-member, it is possible to view the structure of the selected domain in an interactive Java window, as shown in Fig. 4(b).

Below the list of matches in the 'overview of results' page (Fig.3(b)) is a horizontal bar graph representation of the query sequence, where matches are marked as coloured ranges. A light green range corresponds to an E-value of 0.1 or less, a yellow range to 0.1 ≤ E-value ≤ 1.0 and an orange range for E-values above 1.0. Each coloured range links to a pairwise alignment of the query sequence and the saHMM consensus. The user has the option to display a multiple sequence alignment of the query sequence and the saHMM-member sequences in different formats. In addition, it is possible to reach a list with pairwise comparisons of the query and each saHMM-member. All alignments are anchored on the saHMM.



(a) (b)

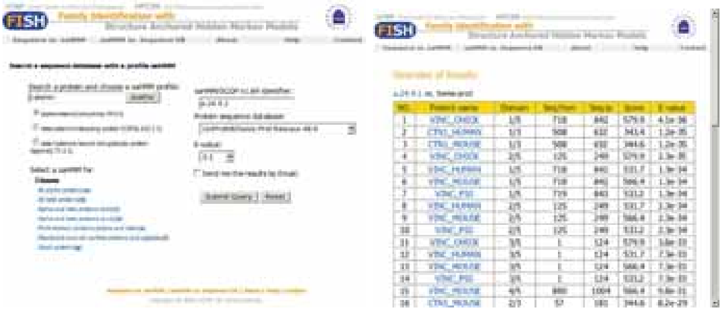
Fig. 4. Example pages displaying (a) the domain family information of the top hit from Fig. 3(b) and (b) the structure view of the domain with highest sequence identity compared to the query sequence.

saHMM Searches vs. a Sequence Database

Furthermore, the FISH server allows the user to employ individual saHMMs for searching against a sequence data base to find those proteins that harbour a certain domain, independent of sequence identity and annotation status. For this purpose, the user can choose a particular saHMM from a list of available models and specify against which data base to perform the search. Currently, the Swiss-Prot, TrEMBL and the non-redundant data base, nr, from NCBI are available for searching. In addition, a user has the option to upload his/her own sequence database, as long as its size does not exceed 2 MB. In this way it is possible to identify previously un-annotated sequences on the domain family level, even in case of very low sequence identities, below $p^I(L, 0)$. For each match, the user obtains the corresponding sequence entry, as well as pairwise and multiple sequence alignments of the matched sequence and the saHMM-members, anchored on the saHMM. Information about the domain family used for searching is also easily available.

A search with a single saHMM vs. SwissProt can take from 15 minutes up to about nine hours. Searching TrEMBL, which is about ten times larger, takes considerably longer. In order to minimize the time a user has to wait for the results, we pre-calculated the searches of all 850 saHMMs vs. SwissProt, TrEMBL and nr using an E-value cut-off of 100. Depending on the E-value choice of the user, the results are extracted and presented up to that value. The computations were done in parallel, by searching the databases with several saHMMs concurrently, using up to 20 processors on the HPC2N Linux cluster Seth.

Fig. 5 shows an example of (a) the input page and (b) the results page of a search with an saHMM versus a sequence database. In the example, SwissProt was used. The results of the search are represented in form of a list sorted by E-value up to the user-specified cut-off.


FISH
Family Identification with
Structure Anchored Hidden Markov Models
Search a sequence database with a profile saHMM
Submit Query
Reset
Overview of Results
a.24.9.1 vs. Swiss-prot

No.	Protein name	Domain	Seq from	Seq to	Score	E-value
1	VINC_CHICK	1/5	718	842	579.6	4.5e-36
2	CTN1_HUMAN	1/3	508	632	343.4	1.2e-35
3	CTN1_MOUSE	1/3	508	632	344.6	1.2e-35
4	VINC_CHICK	2/5	125	249	579.6	2.3e-35
5	VINC_HUMAN	1/5	718	842	531.7	1.3e-34
6	VINC_MOUSE	1/5	718	842	566.4	1.3e-34
7	VINC_PIG	1/5	719	843	533.2	1.3e-34
8	VINC_HUMAN	2/5	125	249	531.7	2.3e-34
9	VINC_MOUSE	2/5	125	249	566.4	2.3e-34
10	VINC_PIG	2/5	125	249	533.2	2.3e-34
11	VINC_CHICK	3/5	1	124	579.6	3.6e-33
12	VINC_HUMAN	3/5	1	124	531.7	7.3e-33
13	VINC_MOUSE	3/5	1	124	566.4	7.3e-33
14	VINC_PIG	3/5	1	124	533.2	7.3e-33
15	VINC_MOUSE	4/5	880	1004	566.4	9.6e-31
16	CTN1_MOUSE	2/3	57	181	344.6	8.2e-29

(a) (b)

Fig. 5. Example of (a) input and (b) results of a search with the catenin saHMM (a.24.9.1) vs SwissProt version 1.69. Only the top part of the results page is shown.

5 Conclusions

The foundation of the structure-anchored hidden Markov model method is the 3D superimposition of carefully chosen domains representing the SCOP domain family to be modelled. For the selection of the representative domains, called the saHMM-members, we evaluated two algorithms, Algorithm 1 and Algorithm 2. Even though the use of Algorithm 2 results in 164 more saHMM-members in the midnight ASTRAL set, which leads to 44 more saHMMs, we prefer Algorithm 1 since it is more than an order of magnitude faster and can handle even the largest families in a reasonable amount of time. The resulting saHMMs together constitute the saHMM-db, which covers 30% of the SCOP families and 65% of the domains belonging to true classes. So far, every new SCOP release has lead to new saHMMs and has increased the number of saHMM-members for many families. As the number of deposited structures grows, we anticipate that the saHMM-db will cover more of SCOP. In addition, we expect that new domain sequences will be added to families, which in turn increases the number of saHMM-members and improve saHMMs with only few saHMM-members. The saHMM-db is publicly available through the FISH server, which is a powerful and versatile tool with dual function. On the one hand, the user can perform sequence searches versus the saHMM-db, and possibly obtain matches even for remote homologues, within the "midnight zone" of sequence alignments. On the other hand, the user can choose one of the saHMMs to perform a search against a protein sequence data base. Since the saHMMs are based on structure anchored sequence alignments and the structures of all representatives are known, the alignment of a sequence to the saHMM-members can be used to draw conclusions about the secondary and tertiary structures of the sequence.

Acknowledgements We thank Åke Sandgren for assistance with parallel computing and Marek Wilczynski for support with the FISH server. Part of this research was conducted using the HPC2N cluster resources. B.K. acknowledges the partial support for this project by the Swedish Foundation for Strategic Research grant A3.02:13. U.H.S. acknowledges the partial support for this project by a grant from the Knowledge Foundation (KK-Stiftelsen).

References

1. Bateman, A., Coin, L., Durbin, R., Finn, R.D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E.L.L., Studholme, D.J., Yeats, C., Eddy, S.R.: The Pfam protein families database. *Nucleic Acids Research* **32** (2004) D138–D141
2. Chandonia, J.-M., Hon, G., Walker, N.S., Lo Conte, L., Koehl, P., Levitt, M., Brenner, S.E.: The ASTRAL Compendium in 2004. *Nucleic Acids Research* **32** (2004) D189–D192
3. Eddy, S. R.: Profile Hidden Markov Models. *Bioinformatics* **14** (1998) 755–763
4. Hobohm, U., Scharf, M., Schneider, R., Sander, C.: Selection of representative protein data sets. *Protein Science* **I** (1992) 409–417
5. Konagurthu, A. S., Whisstock, J. C., Stuckey, P. J., Lesk, A. M.: MUSTANG: A multiple structural alignment algorithm. *PROTEINS: Structure, Function, and Bioinformatics* **64** (2006) 559–574
6. Letunic, I., Copley, R.R., Pils, B., Pinkert, S., Schultz, J., Bork, P.: SMART 5: domains in the context of genomes and networks. *Nucleic Acids Research* **34** (2006) D257–D260
7. Madera, M., Vogel, C., Kummerfeld, S.K., Chothia, C., Gough, J.: The SUPERFAMILY database in 2004: additions and improvements. *Nucleic Acids Research* **32** (2004) D235–D239
8. Mika, S., Rost, B.: UniqueProt: creating representative protein sequence sets. *Nucleic Acids Research* **31** (2003) 3789-3791
9. Murzin, A. G., Brenner, S. E., Hubbard, T., Chothia, C.: SCOP: a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology* **247** (1995) 536–540
10. Rost, B.: Twilight zone of protein sequence alignments. *Protein Engineering* **12** (1999) 85–94
11. Russell, R. B., Barton, G. J.: Multiple Protein Sequence Alignment From Tertiary Structure Comparison: Assignment of Global and Residue Confidence Levels. *PROTEINS: Structure, Function, and Genetics* **14** (1992) 309–323
12. Tångrot, J.: The Use of Structural Information to Improve Biological Sequence Searches. *Lic. Thesis, UMINF-03.19.* Dept. of Comput. Sci., Umeå Univ. (2003)
13. Tångrot, J., Wang, L., Kågström, B., Sauer, U.H.: FISH – family identification of sequence homologues using structure anchored hidden Markov models. *Nucleic Acids Research* **34** (2006) W10–W14

IV

Paper IV

Combinatorial Selection Improves Hidden Markov Model Performance*

Jeanette Tångrot[1,2], Bo Kågström[2,3] and Uwe H. Sauer[1]

[1] *Umeå Centre for Molecular Pathogenesis,*
[2] *Department of Computing Science*
and
[3] *High Performance Computing Center North*
Umeå Universty
S-901 87 Umeå, Sweden
{jeanette,bokg}@cs.umu.se, uwe@ucmp.umu.se

Abstract:

Motivation Correct domain identification is a prerequisite for reliable protein sequence annotation as well as proper functional and structural association. Previously, we have shown that domain family specific structure-anchored hidden Markov models, saHMMs, are an accurate and sensitive way to identify domains. The family specific saHMMs are built from structure alignments of low sequence identity SCOP domain representatives, the saHMM-members. Here we investigate how the number of available domains and their selection affect the performance of the saHMMs.

Results We observe a wide spread of performance for saHMMs based on the minimum of two domains from the same SCOP family, and that the saHMMs built from all available saHMM-members do not necessarily perform best. By using a combinatorial selection approach, we are able to improve on the performance of HMMs in situations where only a limited number of sequences are available. For four representative SCOP families, we show that combinatorial selection can be used to dramatically increase the performance of the corresponding saHMMs. We further provide evidence that a few well selected sequences suffice to create saHMMs that capture the characteristic features of a domain family. The improvements will be implemented in the FISH server (http://babel.ucmp.umu.se/fish/).

*From UMINF 07.14, 2008

Combinatorial Selection Improves Hidden Markov Model Performance

Jeanette Tångrot[1,2], Bo Kågström[2,3] and Uwe H. Sauer[1]
[1]Umeå Centre for Molecular Pathogenesis
[2]Department of Computing Science
[3]High Performance Computing Center North
Umeå University, S-901 87 Umeå, Sweden

Abstract

Motivation

Correct domain identification is a prerequisite for reliable protein sequence annotation as well as proper functional and structural association. Previously, we have shown that domain family specific structure-anchored hidden Markov models, saHMMs, are an accurate and sensitive way to identify domains. The family specific saHMMs are built from structure alignments of low sequence identity SCOP domain representatives, which are called the saHMM-members. Here, we investigate how the number of available domains and their selection affect the performance of the saHMMs.

Results

We observe a wide spread of performance for saHMMs based on the minimum of two domains from the same SCOP family, and that the saHMMs built from all available saHMM-members do not necessarily perform best.

By using a combinatorial selection approach, we are able to improve on the performance of HMMs in situations where only a limited number of sequences are available.

For four representative SCOP families, we show that combinatorial selection can be used to dramatically increase the performance of the corresponding saHMMs. We further provide evidence that a few well selected sequences suffice to create saHMMs that capture the characteristic features of a domain family. The improvements will be implemented in the FISH server (http://babel.ucmp.umu.se/fish/).

Contact: Uwe H. Sauer, Umeå Centre for Molecular Pathogenesis, Umeå University, S-901 87 Umeå, Sweden, e-mail: uwe.sauer@ucmp.umu.se, tel: +46-(0)90-785 6784, fax: +46-(0)90-77 80 07.

Introduction

The addition of structural information to sequence recognition methods can improve their ability to associate sequences with the correct family or superfamily (Hargbo and Elofsson, 1999; Rice and Eisenberg, 1997). Many of these methods make use of structure-derived sequence alignments(Al-Lazikani, et al., 2001; Blake and Cohen, 2001; Kelley, et al., 2000; Scheeff and Bourne, 2006).

Previously, we have described a method based on structure-anchored hidden Markov models, saHMMs (Tångrot, et al. 2008). For each SCOP family, an saHMM is derived from a structural alignment of family members with low mutual sequence identities, the so called saHMM-members. We showed that a few domains, from the same SCOP family (Murzin, et al., 1995) but divergent in sequence, are sufficient to create saHMMs that are able to score family members with high coverage and accuracy. The collection of saHMMs currently holds 850 models with on average

95% coverage (Tångrot, et al., 2008; Tångrot, et al., 2006). The majority of the saHMMs obtain correct matches exclusively, resulting in an average false positive rate of 5%.

However, about 20% of the saHMMs achieve below 95% coverage, and for some, the coverage is less than 65%. In addition, we observe large variations in accuracy, due to a few saHMMs with large numbers of false positive hits.

We examine how the number and combination of saHMM-members used to build an saHMM affects its performance, and whether using all available saHMM-members results in the best performing saHMM. By applying a combinatorial selection procedure to identify the best performing saHMM, we are able to demonstrate that the performance of the saHMMs can be improved by selecting the optimal combination and number of saHMM-members.

In order to measure the performance of a method, one often uses quantities such as coverage (*tp*/(*tp* +*fn*)), and accuracy (*tp*/(*tp*+*fp*)). Here, *tp* is the number of true positives, i.e., the number of correct matches; *fn* is the number of false negatives, i.e., family members missed; and *fp* is the number of false positives, i.e., incorrect matches. In order to obtain a single value for measuring performance, the quantities coverage and accuracy can be combined by, for example, using the geometric mean or the harmonic mean, also called the F-measure (Lewis and Gale, 1994).

To select the optimal sequence combinations, in order to optimize the performance of the saHMMs, we introduce an alternative performance score, the FI-score, which is defined as (*tp*-*fp*)/*N* and takes into consideration both the coverage and the number of false positive hits for a given saHMM. We motivate the use of the FI-score since it penalizes the false positive matches more than other scores and identifies as top performers those saHMMs that have low false positive rates.

An alternative approach to increase the performance of HMMs is used by Pfam (Finn, et al., 2006). The Pfam HMMs are built from trusted seed alignments. In case the resulting HMM is not able to recognise all members of the family, some of the missed sequences are added to the seed until all members are found. However, the modifications of the seed alignment are performed manually.

Methods:

Construction of saHMMs

The construction of the saHMMs is described previously (Tångrot, et al., 2006). In short, a selection of representative domains, called saHMM-members, is made for each domain family, such that all saHMM-members have a sequence identity below a certain limit when compared to any other saHMM-member within the same family. The equation used for calculating the percent identity cutoff, $p^I(L,0)$, is the HSSP-curve as defined in Mika and Rost (Mika and Rost, 2003). Throughout, the saHMM-members are structurally aligned with MUSTANG (Konagurthu, et al., 2006), and an saHMM is built from the resulting structure-anchored multiple sequence alignment using HMMER 2.2g (Eddy, 2003; Eddy, 1998). The domain families are defined according to SCOP version 1.69 (Murzin, et al., 1995) and the coordinate files for the domains are obtained from the corresponding ASTRAL compendium (Chandonia, et al., 2004).

The FI-score

For evaluating the performance of the saHMMs, we introduce the Family Identification score, FI-score, defined as:

$$\text{FI-score} = (tp\text{-}fp)/N. \qquad \text{Equation 1}$$

The FI-score takes into account both the coverage and the accuracy. Here, *tp* is the number of true positives, i.e. the number of family members found with an E-value below a given cut-off, and *fp* is the number of false positives, i.e. the number of matches to an incorrect family with an E-value below the given cut-off. *N* is the total number of family members in the database. The FI-score can not exceed 1.0, however, if *fp* > *tp*, the score becomes negative. In this way, the FI-score penalizes large numbers of false positive matches much harder than for example the F-measure (Lewis and Gale, 1994). This is useful when aiming at high overall accuracy.

Combinatorial selection of saHMM-members

We investigate the effect of the number of saHMM-members used to build the saHMM on its ability to recognise family members. Of the *n* saHMM-members of a given domain family, we generate all possible combinations of *k* domains, where $2 \leq k \leq n$. Starting with $k = 2$, all possible combinations of two proteins are extracted and structurally aligned, and saHMMs are constructed from the resulting pairwise alignments. The procedure is repeated for combinations of $k = 3, 4, 5$, etc., saHMM-members, up to the complete set of *n* members. The total number of possible combinations, when choosing *k* domains out of *n* domains, is $m = n!/(k!(n-k)!)$. The value of *m* increases rapidly, and reaches a maximum for $k=n/2$, if *n* is even, or $k=(n\pm1)/2$, if *n* is odd. In order to reduce the computations, we calculate a set of 1000 random combinations in case *m* exceeds 1000. Each of the resulting saHMMs is used to query SCOP for family members, and the FI-score at an E-value cut-off of 0.1 is calculated and plotted. In this way, performance plots are generated for combinations of saHMM-members within the four selected representative families as well as the Ig V-set domain family.

Families studied

For closer investigation we selected the immunoglobulin V-set domain family (SCOP id: b.1.1.1, sunid: 48727). With 1691 domains, it is one of the largest families in SCOP and gives rise to $n = 28$ saHMM-members. The large number of family members, as well as saHMM-members, makes this family ideal for studying the effect of combinatorial selection as a means to improve the performance. For this family, we generate 1000 random combinations for each of $3 \leq k \leq 25$.

In addition to the V-set domains, we selected one example of a low performing saHMM from each of the four major SCOP classes, in order to investigate whether combinatorial selection can improve the performance. The four chosen low performing saHMMs find up to 60% of their family members and are built from $n \geq 8$ saHMM-members. As a representative for the all alpha class we select the Di-heme elbow motif family (SCOP id: a.138.1.3, sunid: 48711), whose saHMM is built from 8 saHMM-members and finds 30% (E=0.1) of the remaining 57 family members. For the all beta class, we select the family of E-set domains of sugar-utilizing enzymes (SCOP id: b.1.18.2, sunid: 81282) as the representative. The saHMM is built from 16 saHMM-members, and finds 56% of the 118 possible family members. For the alpha/beta class, the family of FAD/NAD-linked reductases, N-terminal and central domains (SCOP id: c.3.1.5, sunid: 51943) serves as our example. The saHMM, built from 15 saHMM-members, finds 55% of the 197 domains present in the family apart from the saHMM-members. The alpha+beta class is represented by the family of MHC antigen-recognition domains (SCOP id: d.19.1.1, sunid: 54453). The saHMM is in this case built from 10 saHMM-members, and finds 60% of the remaining 403 family members.

HMM-logos

All HMM-logos were created using the LogoMat-M server (http://www.sanger.ac.uk/cgi-bin/software/analysis/logomat-m.cgi) (Schuster-Bockler, et al., 2004).

Results and discussion

Limited combinatorial combinations of V-set domains

In order to examine how the performance of the saHMMs is affected by varying the number and combination of saHMM-members used, we focus our investigation on the immunoglobulin V-set domain family, one of the largest SCOP families. With a large number of saHMM-members, the V-set is especially suited for this study.

In Figure 1, we summarize the performance of V-set domain saHMMs based on the combinatorial selection of an increasing number of saHMM-members. The number of saHMM-members used in the saHMMs, *k*, is indicated along the x-axis, and the y-axis shows the FI-score. Hence, the first column in the plot shows the ability of saHMMs built from random combinations of two saHMM-members to accurately identify family members. The second column shows the performance of HMMs built from three sequences, etc.

In Figure 1, each mark represents a set of saHMMs built from the same number of saHMM-members and that give rise to FI-scores within the same interval. FI-scores below zero are binned into the lowest bin. The colour code extends from dark blue, indicating one saHMM, to dark red, indicating 50 or more saHMMs.

As *k*, the number of members used to build the saHMMs, increases, the average FI-score increases as well. The top FI-score, for a given *k*, increases as more members are added. At about half of the total number of saHMM-members included in the saHMM, the upper bound of the FI-score reaches a value of 0.985 after which it increases only marginally, whereas the lower bound FI-score continues to rise much more pronounced as more members are added.

When analyzing the V-set domain family results in more detail, we find that the saHMM built from all 28 saHMM-members has a FI-score of 0.989, with 1679 family members found and 6 false matches. One combination of 22 sequences has the highest FI-score, 0.992, with 1683 family members found and 5 false matches. However, another combination of 22 saHMM-members leads to an saHMM with a considerably lower FI-score of 0.711. For lower *k*, the spread in performance is even more pronounced. For example, the best combination of nine saHMM-members has a FI-score as high as 0.976, while the worst performing combination only achieves a very poor FI-score of 0.011.

These results indicate that the performance of the resulting saHMM strongly depends on the selected saHMM-members.

saHMMs built from two saHMM-members

For protein families with only a few structures determined, or in case the determined structures have similar sequences with mutual sequence identities exceeding $p^I(L,0)$, the family might have only two saHMM-members left after the selection procedure. This is the minimum number of members from which one can build an saHMM. Since about half the saHMMs in our database are built from only two saHMM-members, we examine the performances of the 378 V-set saHMMs built from alignments of two randomly selected representatives, in order to estimate the variation in performance.

As can be seen from the first column in Figure 1, the saHMMs show a large spread of FI-scores. Only five saHMMs have FI-scores higher than 0.9. The best combination of two sequences yields an saHMM able to find as many as 1572 family members, with no false hits, corresponding to a FI-score of 0.930. However, the large majority does not perform well. About 85 % of the saHMMs have a FI-score below 0.5.

These results indicate that it is not possible to know a priori how well an saHMM built from only two saHMM-members will perform. However, it should be noted that also saHMMs built from only two saHMM-members can perform remarkably well.

Comparing HMM-logos of V-set HMMs

HMM-logos (Schuster-Bockler, et al., 2004) visualize the amino acid probability distributions within HMMs. The height of an amino acid symbol and the width of its stack reflect the level of conservation of that residue. The lower the stack, the closer are the emission probabilities to the background distribution of the HMM. The expected number of inserted residues at each position is visualized by the width of the red-shaded stacks where the dark shaded part reflects the likelihood of inserting at least one amino acid.

We compare the logos of selected saHMMs built from two or more V-set sequences, as well as for the Pfam HMM representing the V-set domain family. An inspection of the probability distributions within the saHMM built from all 28 saHMM-members reveals a typical pattern of amino acids with large deviation from background probabilities, visible as tall capitals in Figure 2. They are characteristic for this particular domain family and are important for their fold and function. The figure shows one of the two conserved cysteins that are known to be crucial for the family (Gerstein and Altman, 1995). Comparing the logo of the best performing saHMM built from two members to the logo of the saHMM containing all saHMM-members, one observes that the crucial amino acids obtain high probabilities in both cases. However, the probabilities for less conserved amino acids are approaching the background in the case of the saHMM built from 28 sequences (Figures 2a and 2c).

The best performing saHMM, built from 22 saHMM-members, shows the same characteristic pattern as the saHMM built from all saHMM-members (Figures 2b and c). A similar pattern is also found in the logo generated from the Pfam HMM (FI-score 0.925) representing the V-set domain family (Figure 2d). The most pronounced difference between the Pfam HMM and the best saHMMs is the distribution of insertions and deletions.

From the HMM logos we observe that the saHMMs are comparable to the Pfam HMM of the V-set domain family in assigning high probabilities to key amino acids, despite considerably fewer sequences used for the saHMMs, 22 or 28 sequences, respectively, compared to the Pfam seed alignment, which is based on 121 sequences.

Combinatorial analysis of low performing saHMMs

For the four domain families representing the major SCOP classes, we investigate whether the saHMMs built from all saHMM-members are the best performing models, or if the number and combination of saHMM-members affects the performance of the saHMM.

Plotting the FI-score versus the number of members used to build the corresponding saHMM (Figure 3), we find a large variation in performance for saHMMs built from a given number of saHMM-members. This can be seen from the spread of the FI-score within each column. Furthermore, as more saHMM-members are added, both the highest and the lowest FI-score boundary increase for each consecutive column. However, the saHMM built from the maximum number of saHMM-members, *n*, does not obtain the highest FI-score. We observe a decrease of the top FI-scores as *k* approaches *n* and the number of possible combinations, *m*, decreases. This observation holds for all four domain families.

We conclude that within each domain family the quality of hidden Markov models does not necessarily improve with the number of saHMM-members used, and that it is possible to obtain saHMMs with higher FI-scores by selecting a subset of the saHMM-members. In order to identify the highest scoring saHMM for a given domain family we carry out a combinatorial selection analysis.

Using combinatorial selection to improve the saHMM-db

On average, the 850 saHMMs in our collection obtain a FI-score of 0.899 and attain 95% coverage. Since the four example families indicate that selecting a subset of the saHMM-members can result in an saHMM that performs better than the saHMM built from all saHMM-members, we investigate whether it is possible to improve the worst performing saHMMs in the collection by combinatorial selection. For each saHMM with at most 65% coverage, we examine whether the number and combination of saHMM-members improves the performance of the saHMM and identify the best performing saHMM(s) according to the FI-score at an E-value cut-off of $e = 0.1$.

We find that 38 of the saHMMs in our collection fall into this class of which 17 are built from at least three saHMM-members. The 17 saHMMs have an average FI-score of 0.298 and on average

42% coverage. After the optimization, the average coverage of the 17 families increases to 65% and the average FI-score to 0.649 (see Figure 4). However, three of the saHMMs could not be further improved by combinatorial selection.

For the remaining 14 families, the coverage for an individual family increased with between 2% and 59%. The FI-score for the worst performing family increased from -1.386 to 0.603 which is due both to a decrease in the large number of false positives and an increase in coverage.

There are 56 saHMMs built from at least three members in the coverage interval between 65% and 90%. Their average coverage and FI-scores are 81% and 0.777, respectively. Of these, 35 can be improved by selecting an optimal subset of members. After optimization, the average coverage increased to 86% and the average FI-score to 0.856 (Figure 4).

The overall effect on the database is less dramatic, with the average coverage increasing from 95% to 96%, and the average FI-score from 0.875 to 0.888. However, individual saHMMs can be improved significantly through combinatorial selection.

Conclusions

The combinatorial selection analysis demonstrates that the performance of saHMMs built from a certain number of saHMM-members shows a broad distribution. The performance tends to improve as the number of saHMM-members included in the saHMM increases.

However, examining the FI-scores of the best performing saHMMs shows that the increase levels off after a certain number of saHMM-members used in the models, and that the FI-score even decreases in some cases. This means that the best performing saHMM is not necessarily built from all saHMM-members and that there is room for improvement. The combinatorial selection process improves their average FI-score about two-fold for saHMMs with 65% or less coverage. A closer look reveals that the largest contribution comes from a decrease in the number of false positive hits, with a less dramatic improvement of the coverage.

The selection of the best performing saHMM for families with a coverage < 90%, improves the performance of the entire collection of saHMMs. The (few) individual low performing families are considerably improved in the process, which contributes to higher accuracy and coverage of the saHMMs provided by the FISH server.

We use HMM-logo plots of the V-set family to visualize the features of the top performing saHMMs built from 2, 22 and 28 saHMM-members, and the corresponding Pfam HMM. The comparisons show essentially the same pattern of high probability residues for the saHMMs as for the Pfam HMM, despite the lower number of sequences used for the saHMMs.

We have shown that the performance of individual saHMMs can be improved through combinatorial selection of saHMM-members, and that the maximum number of available members does not necessarily result in the best performance.

We hypothesize that the concept of combinatorial selection as applied to saHMMs is applicable to "standard" HMMs as well.

Acknowledgements

This research was conducted using the resources of High Performance Computing Center North (HPC2N)

References:

Al-Lazikani, B., Sheinerman, F.B. and Honig, B. (2001) Combining multiple structure and sequence alignments to improve sequence detection and alignment: application to the SH2 domains of Janus kinases, Proc Natl Acad Sci U S A, 98, 14796-14801.

Blake, J.D. and Cohen, F.E. (2001) Pairwise sequence alignment below the twilight zone, J Mol

Biol, 307, 721-735.

Chandonia, J.M., Hon, G., Walker, N.S., Lo Conte, L., Koehl, P., Levitt, M. and Brenner, S.E. (2004) The ASTRAL Compendium in 2004, Nucleic Acids Res, 32, D189-192.

Eddy, S. (2003) HMMER: profile HMMs for protein sequence analysis.

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

Finn, R.D., Mistry, J., Schuster-Bockler, B., Griffiths-Jones, S., Hollich, V., Lassmann, T., Moxon, S., Marshall, M., Khanna, A., Durbin, R., Eddy, S.R., Sonnhammer, E.L. and Bateman, A. (2006) Pfam: clans, web tools and services, Nucleic Acids Res, 34, D247-251.

Gerstein, M. and Altman, R.B. (1995) Average core structures and variability measures for protein families: application to the immunoglobulins, J Mol Biol, 251, 161-175.

Hargbo, J. and Elofsson, A. (1999) Hidden Markov models that use predicted secondary structures for fold recognition, Proteins, 36, 68-76.

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

Konagurthu, A.S., Whisstock, J.C., Stuckey, P.J. and Lesk, A.M. (2006) MUSTANG: A multiple structural alignment algorithm, Proteins, 64, 559-574.

Lewis, D.D. and Gale, W.A. (1994) A sequential algorithm for training text classifiers. Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, pages 3-12. Springer-Verlag New York, Inc., Dublin, Ireland.

Mika, S. and Rost, B. (2003) UniqueProt: Creating representative protein sequence sets, Nucleic Acids Res, 31, 3789-3791.

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

Rice, D.W. and Eisenberg, D. (1997) A 3D-1D substitution matrix for protein fold recognition that includes predicted secondary structure of the sequence, J Mol Biol, 267, 1026-1038.

Scheeff, E.D. and Bourne, P.E. (2006) Application of protein structure alignments to iterated hidden Markov model protocols for structure prediction, BMC Bioinformatics, 7, 410.

Schuster-Bockler, B., Schultz, J. and Rahmann, S. (2004) HMM Logos for visualization of protein families, BMC Bioinformatics, 5, 7.

Tångrot, J., Kågstrom, B. and Sauer, U.H. (2008) Structure-Anchored Hidden Markov Models for Accurate Domain Recognition in Protein Sequences, (manuscript submitted).

Tångrot, J., Wang, L., Kågström, B. and Sauer, U.H. (2006) FISH--family identification of sequence homologues using structure anchored hidden Markov models, Nucleic Acids Res, 34, W10-14.

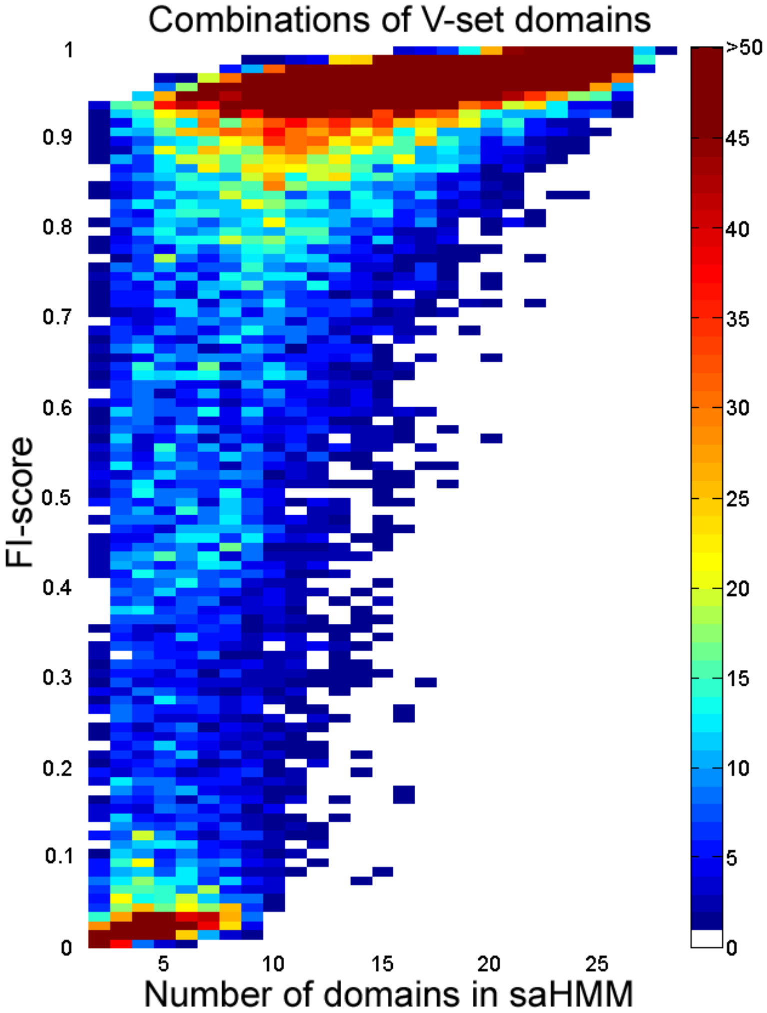


Figure 1. Performance distribution of saHMMs built from different combinations of saHMM-members within the V-set domain family. The y-axis shows the performance, measuerd as the FI-score. The number of saHMM-members used in the saHMMs is shown along the x-axis. Each mark in the plot represents one or more saHMMs, as illustrated by their colour.

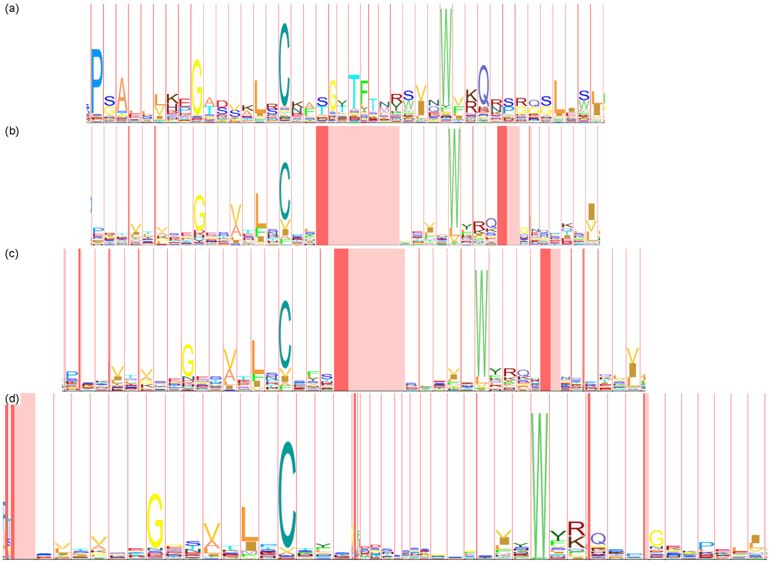


Figure 2. Comparison of HMM-logos created from different HMMs of the V-set domain family: (a) made from two saHMM-members, (b) made from 22, and (c) the full saHMM made from all 28 saHMM-members. For comparison, the HMM-logo generated from the Pfam V-set domain HMM is shown at the bottom (d). Only a small section of the HMM-logos is shown.

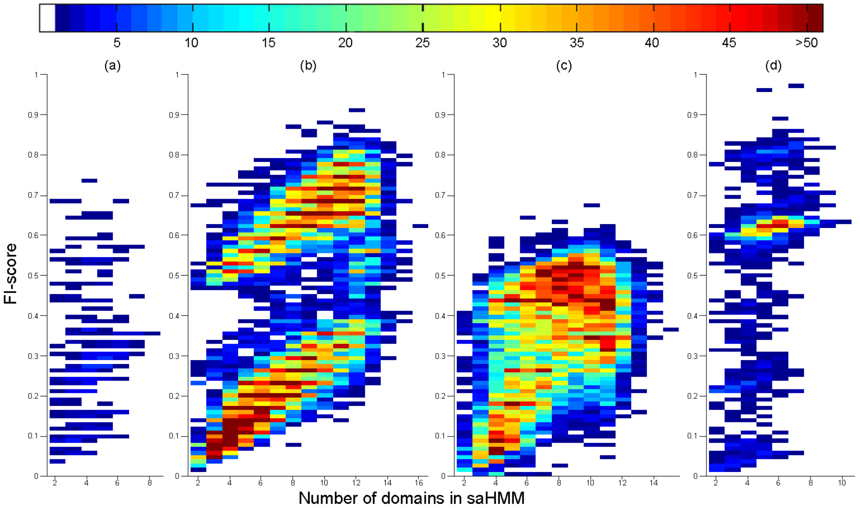


Figure 3. Performance distributions as in Figure 1 for (a) the Di-heme elbow motif family, (b) the family of E-set domains of sugar-utilizing enzymes, (c) the family of FAD/NAD-linked reductases, N-terminal and central domains, and (d) the family of MHC antigen-recognition domains.

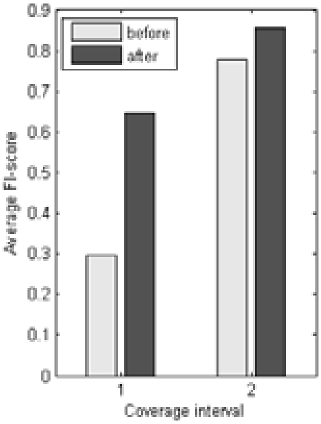


Figure 4. Improvement in FI-score due to the combinatorial selection. The results are shown for saHMMs divided into two groups; group 1 contains saHMMs with initial coverage of less than 65%, group 2 has between 65% and 90% coverage before optimization. The range between 90% and 100% is not included since about half the saHMMs in this range already reach 100% coverage and a large part of the remaining half contains only 2 sequences and therefore cannot be optimized

V

Paper V

Hidden Markov Models That Use Predicted Secondary Structures For Fold Recognition[*]

Jeanette Hargbo[†] and Arne Elofsson

Department of Biochemistry, Stockholm University, Stockholm, Sweden
jeanette@cs.umu.se, arne@sbc.su.se

Abstract: There are many proteins that share the same fold but have no clear sequence similarity. To predict the structure of these proteins, so called "protein fold recognition methods" have been developed. During the last few years, improvements of protein fold recognition methods have been achieved through the use of predicted secondary structures (Rice and Eisenberg, J Mol Biol 1997;267:1026-1038), as well as by using multiple sequence alignments in the form of hidden Markov models (HMM) (Karplus et al., Proteins Suppl 1997;1:134-139). To test the performance of different fold recognition methods, we have developed a rigorous benchmark where representatives for all proteins of known structure are matched against each other. Using this benchmark, we have compared the performance of automatically-created hidden Markov models with standard-sequence-search methods. Further,we combine the use of predicted secondary structures and multiple sequence alignments into a combined method that performs better than methods that do not use this combination of information. Using only single sequences, the correct fold of a protein was detected for 10% of the test cases in our benchmark. Including multiple sequence information increased this number to 16%, and when predicted secondary structure information was included as well, the fold was correctly identified in 20% of the cases. Moreover, if the correct secondary structure was used, 27% of the proteins could be correctly matched to a fold. For comparison, blast2, fasta, and ssearch identifies the fold correctly in 13-17% of the cases. Thus, standard pairwise sequence search methods perform almost as well as hidden Markov models in our benchmark. This is probably because the automatically-created multiple sequence alignments used in this study do not

[*] By permission of **Proteins: Structure, Function, and Genetics** © copyright 1999 Wiley-Liss, Inc., A Wiley Company

[†] Now Jeanette Tångrot

contain enough diversity and because the current generation of hidden Markov models do not perform very well when built from a few sequences.

Key words: protein structure; HMM; Scop; HSSP; threading; blast; fasta; ssearch; protein fold recognition

Hidden Markov Models That Use Predicted Secondary Structures For Fold Recognition

Jeanette Hargbo and Arne Elofsson*
Department of Biochemistry, Stockholm University, Stockholm, Sweden

ABSTRACT **There are many proteins that share the same fold but have no clear sequence similarity. To predict the structure of these proteins, so called "protein fold recognition methods" have been developed. During the last few years, improvements of protein fold recognition methods have been achieved through the use of predicted secondary structures (Rice and Eisenberg, J Mol Biol 1997;267:1026–1038), as well as by using multiple sequence alignments in the form of hidden Markov models (HMM) (Karplus et al., Proteins Suppl 1997;1:134–139). To test the performance of different fold recognition methods, we have developed a rigorous benchmark where representatives for all proteins of known structure are matched against each other. Using this benchmark, we have compared the performance of automatically-created hidden Markov models with standard-sequence-search methods. Further, we combine the use of predicted secondary structures and multiple sequence alignments into a combined method that performs better than methods that do not use this combination of information. Using only single sequences, the correct fold of a protein was detected for 10% of the test cases in our benchmark. Including multiple sequence information increased this number to 16%, and when predicted secondary structure information was included as well, the fold was correctly identified in 20% of the cases. Moreover, if the correct secondary structure was used, 27% of the proteins could be correctly matched to a fold. For comparison, blast2, fasta, and ssearch identifies the fold correctly in 13–17% of the cases. Thus, standard pairwise sequence search methods perform almost as well as hidden Markov models in our benchmark. This is probably because the automatically-created multiple sequence alignments used in this study do not contain enough diversity and because the current generation of hidden Markov models do not perform very well when built from a few sequences. Proteins 1999;36:68–76.** © **1999 Wiley-Liss, Inc.**

Key words: protein structure; HMM; Scop; HSSP; threading; blast; fasta; ssearch; protein fold recognition

INTRODUCTION

The most promising method for predicting the structure of a protein is to identify a protein with a known structure that shares the same fold. Traditionally, this has been done by identifying proteins that have similar sequences. However, of late, many examples of structures that have similar folds but no detectable sequence similarity have been found,. This has led to the development of methods to detect the fold of a probe sequence from a library of known target folds. These methods are often referred to as fold recognition methods.

Fold recognition methods can roughly be divided into three different types, based on the type of information that they use. Within each category there are many different implementations. The three types of methods are sequence-based methods,[1,2] structure-based methods,[3,4] and prediction-based methods.[5–9,10] In this study, we introduce a new method that combines multiple-sequence-alignment methods with predicted secondary structure information. We also compare the performance of hidden Markov models with standard sequence-based methods. All these comparisons are made with a more rigorous benchmark than those used in most earlier studies.

Sequence-based methods are the oldest methods for fold recognition.[11] It seems a bit surprising that sequence-based methods are able to detect a similar fold of proteins that show no sequence similarity, but the amino-acid sequence contains much information about the physical environment at each position in the sequence. Thus, even if there is no detectable sequence similarity between two proteins that have the same fold, the corresponding positions in the proteins will have similar properties. Moreover, there are many examples where there is no obvious sequence similarity, but where two proteins clearly are homologous. Of course, these targets might be detected with improved sequence-based methods. One way to increase the performance of sequence-based methods is to use information from a family of sequences, instead of from just one sequence. With the inclusion of multiple sequence alignment information and modern computational methods, such as hidden Markov models, sequence-based methods have proven to be successful in fold recognition.[2]

Abbreviations: Scop, a structural classification of proteins database; HMM, Hidden Markov Model; ssHMM, hidden Markov models that use secondary structure information; predHMM, hidden Markov models that use secondary structure information with predicted secondary structures.

*Correspondence to: Arne Elofsson, Department of Biochemistry, Stockholm University, 106 91 Stockholm, Sweden. E-mail: arne@biokemi.su.se

Received 16 September 1998; Accepted 1 March 1999

A hidden Markov model (HMM), or more correctly a profile-HMM, is a generalized version of a profile that is mathematically more consistent. A general description of HMMs (applied in speech recognition, where they were originally used) has been written by Rabiner and Juang.[12] In biology, HMMs have been used in many different areas, such as gene prediction,[13] membrane protein prediction,[14] and protein sequence comparisons.[1,2] One major difference between profile-HMMs and a profile is that in a profile the penalty for gaps or insertions are the same in every position of the alignment, even though some regions are more variable than others. Ideally, these regions should have a smaller penalty for gaps than more conserved areas. In the HMM, the penalties are position-dependent, and are learned from the training data.

An alternative type of information has been used in the structure-based fold recognition methods. These methods do not use sequence information to determine if two proteins have the same fold or not. Instead, they use an energy function that describes how well a probe sequence matches a target fold. The energy function is often obtained from a database of known protein structures, and can be used, for instance, to describe the environment of each residue[15] or the probability of finding two residues at a certain distance from each other.[3,4]

Proteins having a similar fold also have similar secondary structures, so that even though the amino acid sequences may have changed a great deal during evolution, the secondary structure will still be the same for related proteins belonging to the same fold. Today, the secondary structure can be predicted from the amino acid sequence with an accuracy of more than 70%.[16] Several approaches attempt to use this information, in addition to the amino acid sequence, to recognize the correct fold.[5,6,9] Fischer and Eisenberg[5] align a probe sequence to known folds and then calculate the probability of the protein having a certain fold. The score for an aligned amino acid normally depends on how likely it is to have that particular amino acid in that position in the fold, but Fischer and Eisenberg also take the predicted secondary structure into account, increasing the score if it fits the secondary structure of the fold and decreasing the score otherwise. The addition of the secondary structure information seems to help significantly in recognizing the correct fold, indicating that, even though the predicted secondary structure is not completely correct, it still contains a lot of useful information that could complement other information.

Usually, a HMM only uses the amino acid sequence when modeling a protein family, making very distant homologues difficult to recognize. The aim of this work is to create a HMM that uses the predicted secondary structure in addition to the primary sequence. By combining the information from both sequence and secondary structure, it should be possible to recognize even distant or non-homologous proteins that share a similar fold. The idea of using secondary structure predictions and multiple sequence information HMMs has been proposed earlier but not tested in this type of benchmark.[8,17] In addition, our implementation of this approach differs from earlier attempts.

MATERIALS AND METHODS

An Implementation of HMMs Using Secondary Structure Information

The program package HMMER, version 1.8.4,[18] was modified to include secondary structure information when building a hidden Markov model (HMM) of a protein family, as well as when matching an amino acid sequence to an HMM. The secondary structure HMMs (ssHMMs) are models of protein families based both on amino acid sequence and on secondary structure information.

Ordinary profile HMMs consist of a sequence of match states, analogous to positions in a multiple sequence alignment, and corresponding insert and delete states. To each insert and match state a probability distribution over all amino acids is associated, these distributions giving the probability of a certain amino acid, given that particular state. The parameters of the model are the probabilities for transitions between states and the amino acid probability distributions, and these are optimized so that all sequences belonging to the modeled family obtain high probabilities and all other sequences low. Thus a sequence $s = x_1 \ldots x_L$ following the path $q = q_0 \ldots q_{N+1}$ through model μ has the probability

$$P(s|q, \mu) = \prod_{i=1}^{N+1} T(q_i|q_{i-1}) \prod_{i=1}^{N} P(x_{l(i)}|q_i) \qquad (1)$$

where $T(q_i|q_{I-1})$ is the probability for a transition from state q_{I-1} to q_i and $l(I)$ is the index for amino acid x in the sequence in state q_i, $P(x_{l(I)}|q_i)$ is the probability of having amino acid $x_{l(I)}$ in state q_i, and N is the number of states in the path. The lower indexes represent the position in the path. The theory behind HMMs has been described in more detail in earlier work.[1,18,19] In comparison with sequence profiles, one of the major differences is that for each position there is a correct transition probability for each gap and insertion parameter.

The ssHMM has an extra distribution of probabilities for the secondary structures E, H, and L associated with each insert and match state. In each state, the model emits a probability for the amino acid, as before, but in addition to this it emits another probability for the secondary structure assigned to that position. In this way, the probability for the sequence is higher if the secondary structure is the same as in the modeled family. The total probability for a sequence $s = x_1 \ldots x_L$ having the secondary structure $ss = y_1 \ldots y_L$ given the path $q = q_0 \ldots q_{N+1}$ and model μ is now:

$$P(s,ss|q, \mu) = \prod_{i=1}^{N+1} T(q_i|q_{i-1}) \prod_{i=1}^{N} P(x_{l(i)}|q_i) \prod_{i=1}^{N} P(y_{m(i)}|q_i) \qquad (2)$$

where $y_{m(I)}$ is the secondary structure emitted in state q_i. The emission probabilities of the secondary structures are found in the same way as the amino acid emission probabilities when training the model. The combined HMM will be referred to as a secondary structure HMM (ssHMM). The modified HMMER program is available from http://www.biokemi.su.se/~arne/sshmm/

As the number of parameters in the model increases, additional information is needed to produce a useful model. To decrease the number of free parameters, the emission probabilities $P(x|i_k)$ for the insert states are set equal or to some background frequency. The problem with having too little information, i.e., too few training sequences, concerns fitting, i.e., a HMM created from this data will be able to recognize only proteins that are very closely related to the proteins used to create the HMM. In this situation, a prior distribution can be used, and the model is not allowed to specialize too much. However, a prior distribution assumes that any change from one amino acid to another is equally probable, which is not the case. [19] A standard HMM could be seen as building a sequence profile using an identity matrix, which certainly is not the most efficient matrix to use. The inclusion of substitution parameters into HMMER can be made through the use of a special prior distribution using a substitution matrix. The inclusion of substitution matrices are made when building the HMM by adding a partial count to all amino acid types when a certain amino acid is found in a position. This partial count is related to the probability of an amino acid having been replaced by another particular amino acid. In this study, we have used the Pam250 substitution matrix, which was included in the HMMER package. For the secondary structure counts, we were not able to create a prior distribution that significantly improved the performance. Therefore we chose not to use any. At the beginning of the training, all secondary structures are assumed to occur at equal probabilities. Thus, even if a position is found in only one secondary structure type, the other secondary structure types will also have a small probability of occurrence.

A library of ssHMMs was built from the sequences and secondary structures of a representative set of all proteins with a known structure. For a given protein, all related proteins in Swissprot were found through the HSSP database,[20] and the secondary structure was assumed to be the same for all proteins in a family. The multiple sequence alignment from HSSP, together with the secondary structure, was used to build a ssHMM, as described above. For comparison with the original HMM method, HMMs not using the secondary structure were also created, as were HMMs (and ssHMMs) using substitution matrices. These last will be referred to as HMM-pam and ssHMM-pam. Finally, another set of HMMs, ignoring multiple sequence alignments, were created. These will be referred to as HMM-single, ssHMM-single, etc. For a complete description of all HMMs built see Table I.

To match a protein against a library of HMMs, a query sequence is matched against all HMMs. We examined the four different alignment algorithms included in HMMER local, global, endsfree, and fragmentary matches. However, in all cases, the hmms program that uses a global alignment algorithm performed best, and only results using this algorithm were evaluated in this study. When a protein is matched against a ssHMM it is necessary to assume the secondary structure of the protein; this was done in two different ways. First, the correct secondary structure was used. Second, the secondary structure predicted by predator[21] was used. The tests using the predicted secondary structures are referred to as predHMM etc. (see Table I). The rather mediocre performance of 68% was probably due to the fact that 45% of the sequences in our database had 10 or fewer homologous sequences in HSSP. For comparison with the standard sequence search methods we have used blast2,[22] fasta,[23] and ssearch[23] on our benchmark. These methods were used with default parameters, and the scoring has been done by using the expectation-values.

TABLE I. Description of Information Used in Methods Studied[†]

Name	SS in HMM	Query True SS	Query Pred SS	Substitution matrix	MSA
HMM					X
predHMM	X		X		X
ssHMM	X	X			X
HMM-single					
predHMM-single	X		X		
ssHMM-single	X	X			
HMM-pam				X	X
predHMM-pam	X		X	X	X
ssHMM-pam	X	X		X	X
HMM-pam-single				X	
predHMM-pam-single	X		X	X	
ssHMM-pam-single	X	X		X	
blast2				X	
fasta				X	
ssearch				X	

[†] SS in HMM, secondary structure in the HMM; Query True SS, correct secondary structure in query sequence; Query Pred SS, predicted secondary structure in query; MSA, multiple sequence alignment.

Measuring the Performance

To compare the performance of different fold recognition methods, it is of great importance to use a large and well-crafted benchmark. Several recent studies[6,24,25] have shown that a useful benchmark can be created using Scop[26] as a standard for classifying proteins into families of similar fold or of evolutionary relationship. Scop is a database in which all known protein structures are classified into a hierarchical classification: class, fold, superfamily, and family. In this study we have focused on proteins that have the same fold but belong to different families, according to Scop. Two proteins that are classified into to the same fold have the same secondary structure elements in a similar topological arrangement, while two proteins that belong to the same family have a clear common evolutionary origin. Two proteins classified into the same fold but to different families might belong to the same superfamily or they might not.

We created a benchmark from the pdb40 dataset of Scop version 1.37. This dataset contains a subset of Scop where no proteins have more than 40% sequence identity to any other member of the dataset.[25] However, this dataset did not completely match the latest release of HSSP in that (1) HSSP was created from another subset of pdb and (2) the proteins in Scop are divided into domains, whereas proteins in HSSP are not. To overcome this problem, we matched each sequence in pdb40 to the HSSP database and replaced the sequence with the HSSP sequence if the match had a significance better than 1.e-5 using fasta, and if the alignment produced was of the same length as the original sequence. Using this procedure, 1,130 out of 1,272 sequences in pdb40 were retained. This procedure removed all Scop entries of the "non-proteins" class and many of the peptides, as they were not present in HSSP. For each of the 1,130 sequences, the multiple sequence alignment and the secondary structure were read from HSSP. On average, 26 sequences were included in a sequence family. However, many of these sequences were identical or almost identical to the original sequence. This dataset of sequences and multiple sequence alignments is available from http://www.biokemi.su.se/~arne/sshmm/

TABLE II. Description of the Benchmark

Data	Number of data points
Protein domains in pdb40	1,272
Protein domains both in HSSP and in pdb40	1,130
Protein domains with at least one true match (another domain from the same fold but from another family)	730
Number of pairwise comparisons	1,273,618
True matches (protein domains from the same fold but different families)	8,312
False matches (protein domains from different folds and families)	1,265,306
Number of different protein families	666
Number of different protein folds	359

In our benchmark, see Figure 1, all proteins were matched to the HMMs of all other proteins, and for each pair the folds and families (according to Scop) were recorded. As the family classification in Scop is a subclassification of a fold, two proteins can belong either to the same family, to two different families but to the same fold, or to two different folds. If the two proteins belong to the same family, we have eliminated them from further consideration, because this indicates that they are homologous and thereby not a good test of fold recognition methods. If the fold, but not the family, of the two entries is the same, the match was considered to be a *true* match, while if the two entries belong to different folds they were considered to be a *false* match. To create a good benchmark it is necessary to have a large and complete set of proteins; in our benchmark set there are 730 proteins that have at least one true match, i.e., there are 400 proteins in the database that do not have any true match. These 400 entries were retained, because they provided potentially important information about false matches. The total number of true hits is 8,312, and there are more than 1.2 million false hits in the benchmark (see Table II). The benchmark includes proteins from 359 different folds and 666 different families in Scop. We believe that this benchmark contains a significant fraction of all possible targets for fold-recognition.

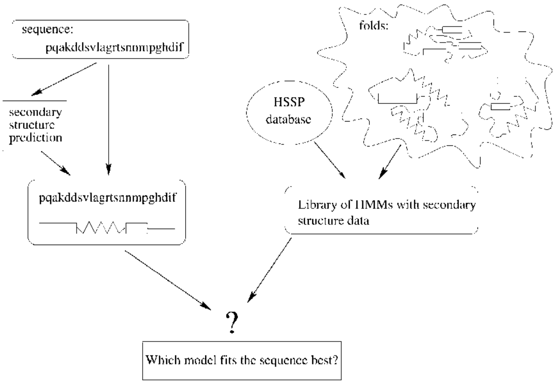

sequence:
pqakddsvlagrtsnnmpghdif
secondary structure prediction
pqakddsvlagrtsnnmpghdif
folds:
HSSP database
Library of HMMs with secondary structure data
?
Which model fits the sequence best?

Fig. 1. A schematic description of the ssHMMs. First a library of representative folds is created, second, all homologous sequences of these proteins are found. These multiple sequence alignments, together with the secondary structures of the representative proteins, are used to construct the library. For the probe sequence, a secondary structure prediction is performed. Finally, the sequence with the predicted secondary structure is probed against all folds in the fold library.

We have used two different criteria to analyze the performance of a fold recognition method on our benchmark. First, we simply examined at what rank the first true hit was found. This is a very intuitive measure, however, and it does not measure the reliablity of a match of a certain score. For some proteins there are several possible correct hits and with this measure the first match could be to any one of these proteins, while for others there is only a single match. Second, as a complementary measure, we have used specificity-sensitivity plots, or spec-sens plots, as in Rice and Eisenberg.[6] The main advantage of this method is that it describes the ability of a method to find all pairwise matches in the benchmark. The sensitivity is based on the model's ability to find all members of the same fold. In other words:

$$\mathrm{SENS}(score) = \mathrm{TP}(score)/(\mathrm{TP}(score) + \mathrm{TN}(score)) \qquad (3)$$

where TP(*score*) is the number of true hits that have a score above *score*, and TN(*score*) is the number of true hits with a score less than *score*. The specificity measures the probability that a pair of sequences with a score greater than a certain threshold really belong to the same fold. The specificity is defined as:

$$\mathrm{SPEC}(score) = \mathrm{TP}(score)/(\mathrm{TP}(score) + \mathrm{FP}(score)) \qquad (4)$$

where FP(*score*) is the number of false hits that have a score above *score* and TP is defined as above. The sensitivity is plotted as a function of specificity, each point in the plot corresponding to a certain score. One difference between our two measures is that the spec-sens curves represent a method's ability to recognize all proteins from the same fold (but from different families), while the simple counting method measures the ability of a method to identify any member of the same fold (but from another family).

RESULTS AND DISCUSSION

Every two years there is a community-wide effort, CASP, to analyze protein structure prediction methods by blind predictions, allowing predictors to "guess" the structure of soon-to-be solved protein structures.[27] At the second CASP process in 1996, five groups were selected for the best performance in the threading category. One of these groups used predicted secondary structures,[7] another group used hidden Markov models (HMM),[2] a third group used a hidden Markov model that only used secondary structure and matched a predicted secondary structure against this model.[8] The last two groups[4,28] used either human expert knowledge or a physical energy function in their threading studies. The success of using HMMs and the idea of using predicted secondary structures makes it a natural step to try to combine these two methods, as we have done in this study.

This study is based on matching all proteins in our test set against all other proteins of the test set. Each protein is classified as belonging to a protein family and as having a certain fold, according to Scop.[26] The Scop classification is hierarchical, i.e., a fold is a superset of one or several families, and thus two proteins might belong to the same fold but to different families. Two proteins from the same fold, but from different families, are not assumed to be homologous but still have a similar structure. A match between two proteins is ignored if the two proteins belong to the same family, it is considered as a true match if the proteins belong to different families but to the same fold, and it is considered to be a false match if the proteins belong to different folds. Using this benchmark, we have compared the performance of the newly developed ssHMMs, standard HMMs, and pairwise sequence comparisons methods.

Secondary Structure Increases the Performance of HMMs

Earlier studies showed that including predicted secondary structure sequence into single sequence-based search methods increased the performance significantly.[5,6,9] Therefore, we believed that the same would be true for hidden Markov models. In Figure 2 it can be seen that our assumption are apparently correct, as the sensitivity of a hidden Markov model is increased when the secondary structure is included. For instance, at a specificity of 5%, the sensitivity increases from 2% to 30% if the true

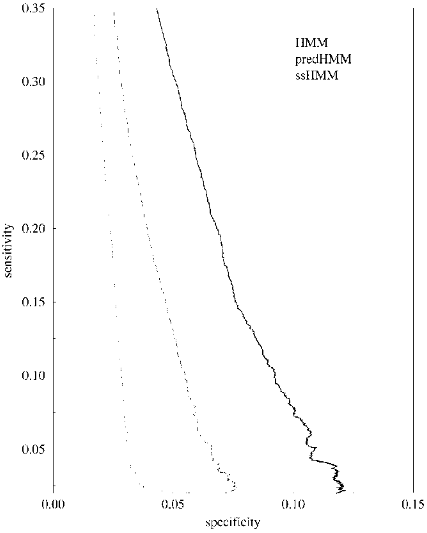


Fig. 2. A specificity versus sensitivity plot of HMM, predHMM, and ssHMM. It can be seen that the sensitivity increases when predicted or true secondary structure information is included.

TABLE III. Sensitivity of Methods at Specificity = 5% and 10%

Name	Spec = 5%	Spec = 10%
HMM	2%	1%
predHMM	13%	1%
ssHMM	30%	8%
HMM-single	0%	0%
predHMM-single	6%	0%
ssHMM-single	17%	0%
HMM-pam	11%	6%
predHMM-pam	11%	2%
ssHMM-pam	26%	7%
HMM-pam-single	8%	2%
predHMM-pam-single	17%	5%
ssHMM-pam-single	24%	11%
Blast2	3%	2%
Fasta	5%	3%
ssearch	13%	6%

TABLE IV. Fraction of Possible True Hits Placed at Ranks 1, 5, 10, and 25

Name	#1	#5	#10	#25
HMM	12%	24%	32%	45%
predHMM	19%	38%	47%	59%
ssHMM	30%	49%	59%	69%
HMM-single	4%	15%	24%	38%
predHMM-single	10%	29%	38%	51%
ssHMM-single	14%	34%	44%	56%
HMM-pam	16%	30%	40%	51%
predHMM-pam	20%	36%	45%	57%
ssHMM-pam	27%	48%	56%	67%
HMM-pam-single	10%	22%	31%	44%
predHMM-pam-single	17%	35%	44%	55%
ssHMM-pam-single	21%	39%	48%	60%
Blast2	17%	30%	37%	48%
Fasta	13%	25%	37%	43%
ssearch	17%	25%	30%	40%

secondary structure is used and to 13% if the predicted secondary structure is used (Table III). The fraction of the possible hits that were ranked in first place is increased as well, from 12% to 30% when using the secondary structure, and to 19% if the predicted secondary structure is used (Table IV). The increase in performance is similar to that reported for single sequence-based methods; for instance, Fischer and Eisenberg increased the fraction of hits found in first rank from 54% to 65% by using predicted secondary structures and the BLOSUM62 matrix.[29] In the study by Rice and Eisenberg, the sensitivity increased from approximately 15% to 30% when predicted secondary structures were used at 5% specificity.

It should also be noted that our benchmark seems significantly more difficult than the benchmark used by Fisher and Eisenberg, as they were able to detect 54% of the proteins in first place using sequence alignment methods, while we were able to detect only 17%. The difficulty of the benchmark used by Rice and Eisenberg seems to be similar to the difficulty of ours.

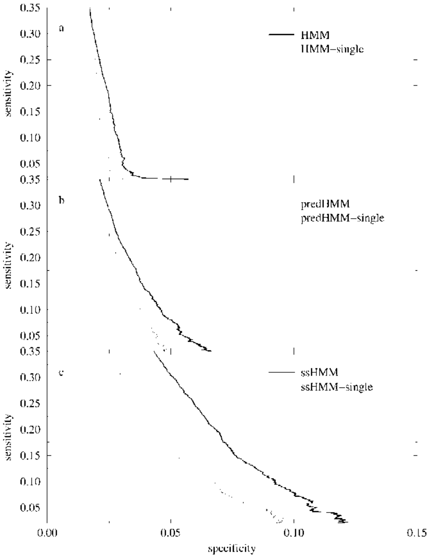


Fig. 3. When multiple sequence alignment is used (bold lines) the sensitivity of the hidden Markov models is increased, compared to using only single sequence alignments. In (**a**) standard HMMs are used, in (**b**) predHMMs, and in (**c**) ssHMMs.

Using Multiple Sequence Information Increases the Performance of HMMs

It has been assumed that using multiple sequences improves the performance of sequence-based search methods. However to our knowledge, there has been no studies showing that this is in fact true, using as complete benchmark as the one we have used here. Figure 3 shows that the sensitivity at a given specificity is increased for models built from multiple sequences compared to models built from just one sequence. This is most obvious for the ssHMMs, where at a specificity of 5%, the sensitivity increases from 17% to 30% when using multiple sequences to build the ssHMMs, compared to single sequences. A clear increase can also be seen for ordinary HMMs, and when using predicted secondary structures. The number of sequences placed at rank one is more than doubled when building models from multiple sequence alignments. They increase from 14% to 30% for the ssHMMs, from 10% to 19% using predHMMs, and from 4% to 12% for the ordinary HMMs (Table IV). It should be remembered that when using multiple sequence alignments we have used only automatically-created alignments from HSSP, and for many proteins these alignments do not contain enough

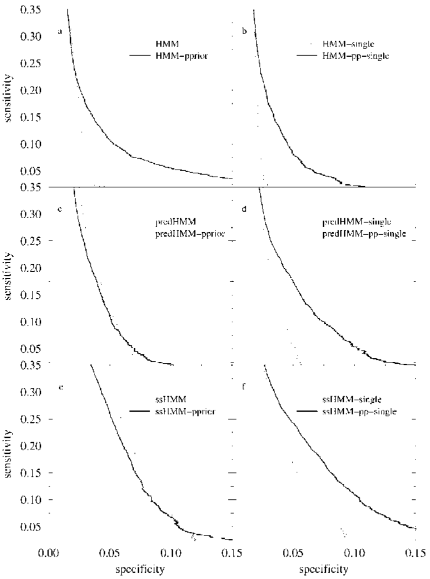


Fig. 4. The specificity is increased when a substitution matrix is used (bold lines). In (**b,d,f**) HMMs created from single sequences are used, while in (**a,c,e**) multiple sequence HMMs are used. In (a,b) standard HMMs are used, in (c,d) predHMMs, and in (e,f) ssHMMs.

diversity to perform as well as HMMs created from a more diverse set of sequences.

Using a Substitution Matrix Increases the Performance of HMMs

A standard hidden Markov model does not include any information about which substitutions are most likely, i.e., a substitution matrix is not used. If the protein family is large enough and diverse enough this should not be a problem. However, in our benchmark, we have many small families with low diversity. By including a substitution matrix we attempted to overcome this problem. As can be seen in Figure 4a,b, the use of a substitution matrix when building the models increased the sensitivity significantly. For hidden Markov models built from multiple sequence alignments, the sensitivity increases from 2% to 11%, at a specificity of 5%, when using the substitution matrix. When comparing Figures 4a and 3a, and Figures 4a and 4c, it can be seen that the use of a substitution matrix helps more than the use of multiple sequence alignments.

In Figure 4d,f, it can be seen that the ssHMMs and predHMMs built from single sequences have higher sensitivities when using a substitution matrix than when not.

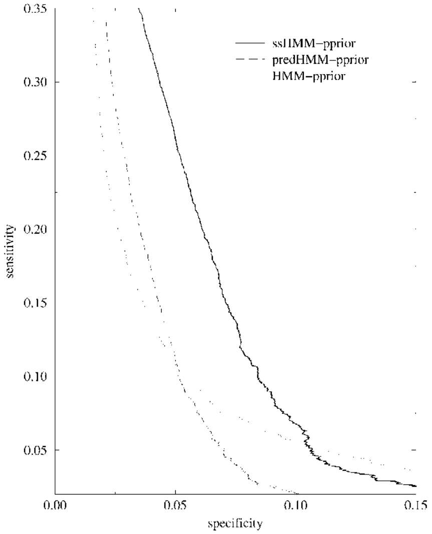


Fig. 5. A final comparison of HMMs that use multiple sequence information as well as substitution matrices. It can be noted that at higher specificities the sensitivity is lower for the ssHMMs and predHMMs than for the HMMs that do not use secondary structure information.

However, for the ssHMMs built from multiple sequence alignments, using a substitution matrix does not seem to improve the performance. On the contrary, the sensitivity decreases from 30% to 26% when the substitution matrix is added to the ssHMMs (Fig. 4e, Table III). This indicates that the prior distribution might not be optimized for the secondary structure HMMs. For these, another prior, where the secondary structure is included, could be used.

When Creating a Hidden Markov Model It Is Best to Use Multiple Sequence Alignments and Substitution Matrices

From the previous results it was concluded that the use of multiple sequence alignments and substitution matrices give the best results. A comparison between the HMM-pam methods with or without secondary structure information can be seen in Figure 5. At a low specificity (<5% for predHMM and <10% for ssHMM), the secondary structure HMMs have a higher sensitivity than the ordinary HMMs. For higher specificities, however, the ordinary HMMs have a higher sensitivity. One possible explanation of this is that the ssHMMs give very high scores to some false matches. When studying false matches with high scores for predHMM-pams, we found that there were a few families that caused a very large part of these false positives. The majority of these matches were between different families that all consisted of a various number of alpha helices. From this data, it seems plausible that the contribution from the secondary structure was ranked too high in comparison with the contribution from the sequence. The secondary-structure-based HMMs still place more correct sequences at high ranks than the ordinary HMMs (Table IV). For example, the number of sequences correctly ranked as number one is increased from 16% to 27% when adding the secondary structure, and to 20 % when using predicted structures.

In Tables III and IV and in Figure 5, a summary of all methods is shown. The ranks clearly support the conclusions that using multiple sequence alignment, predicted secondary structures, and a substitution matrix improves the performance of HMMs. For instance, when using a single sequence HMM, only 4% of the probe sequences recognize a correct target. This figure increases to 12% when using a multiple sequence alignment, and to 10% when using either a predicted secondary structure or a substitution matrix. When using a combination of all three methods, the number of probe sequences that recognize a correct target is increased further to 20%. The number of probes that recognize a correct target among the top 10 hits is increased from 24% to 31–38% when using multiple sequence alignments, predicted secondary structures, or substitution matrix, and to 45% when using all three.

The sensitivity shows a pattern similar to that of the ranks, although there are also some notable differences. First, it can be seen that predHMM-pam-single performs better than the HMMs that use multiple sequence alignments. This might indicate that the use of substitution matrices is not the optimal choice with the ssHMMs, as discussed above. Second, the standard HMMs that use substitution matrices perform better at higher specificity than the predHMMs. This might be due to the occurrence of a few false positives that have very high scores, as described above.

HMMs Perform as Well as but not Better Than Single-Sequence-Based Methods

The performances of all these methods were compared with the performance of single-sequence-based methods—fasta, blast, and ssearch. It could be assumed that the performance of ssearch should be similar to the performance of single sequence HMMs using a substitution matrix. However, ssearch performs better than HMM-single, as can be seen in Tables III and IV. Actually, all the single-sequence-based methods perform significantly better than HMM-pam-single and when it comes to ranks, they actually perform as well as standard HMMs. When studying the spec-sens curves it can be seen that the performance of blast and fasta are not superior to HMM-pam-single. However, ssearch still performs as well as standard (multiple sequence) HMM methods.

The reason why the multiple sequence information does not improve the performance further is probably due to the following. (1) In our benchmark, 45% of the HMMs are built from sequences with less than 10 sequences and HMMER is not optimized for small families. Furthermore, even in the case where there are several sequences they are often very similar, and thus still fail to provide the necessary diversity. (2) The gap penalties in a HMM are calculated individually for each position in the model. However, when an HMM is created from a family with low diversity, and thus few gaps, the gap penalties will not be optimal for recognizing a distant member of the family. (3) Blast, fasta, and ssearch use an extreme value distribution to fit the scores. This method has been included in HMMER-2.0, and consequently the performance has improved (data not shown). (4) When a hidden Markov model is created, it includes a process of optimizing the transition probabilities. Ideally, one should make several tries and create several hidden Markov models for a given sequence family and then use the one that performs best. However, this was not possible in this study, due to computational limitations. All these points show some of the limitations of the current generation of HMMs, but also indicate some easy methods to improve the performance of HMMs.

In fold recognition it is not enough to identify the correct fold of a protein, it is also necessary to make the correct alignment between the two proteins to obtain three-dimensional studies. In the alignments obtained for ssHMM and the other methods from our benchmark, however, most pairs in our benchmark contained proteins that were very distantly related, or not homologous at all, and these proteins are extremely difficult to align correctly. We were, unfortunately, not able to detect any significant improvement of the alignments using ssHMM (data not shown). In a future study we plan to create an alignment benchmark using a set of less difficult proteins to align and examine whether ssHMMs, or standard HMMs, are able to align proteins better than standard pairwise sequence methods.

Use of ssHMM in CASP3

The ssHMM method, together with other methods and manual judgment, were used for blind predictions in the CASP3 process.[27] Three successful fold predictions were made of CASP3 targets T0046, T0053, and T0071a. T0046 (gamma-adaptin, ear domain) is an IG-like fold, and several methods (ssHMM, standard HMMs, and threader[3]) consistently scored high for IG-like domains. For T0053 (CbiK protein), we mainly focussed on the threader results. Our best prediction was T0071 (Alpha adaptin ear domain), in which, using ssHMM, we were able to identify the first 125 residues as an IG-like fold. We were also able to produce a rather good alignment, with 21 out of 125 residues correctly aligned.

Summary

The program package HMMER was modified to allow the construction of hidden Markov models (HMMs) that use the secondary structure, in addition to the amino acid sequence, to model protein families. This was accomplished by adding a distribution over emission probabilities for secondary structures to each match and insert state in the model. It was shown that the resulting secondary structure HMMs perform better than the ordinary HMMs, with both the true and the predicted secondary structures used to recognize proteins having the same fold as the modeled sequences. We have also analyzed the performance of automatically-created HMMs, using a rigorous benchmark. It was shown that using a substitution matrix improved the performance of HMMs. Finally, it was shown that the automatically-created HMMs did not perform significantly better than single sequence based methods.

REFERENCES

1. Krogh A, Brown M, Mian I, Sjölander K, Haussler D. Hidden Markov models in computational biology: application to protein modeling. J Mol Biol 1994;235:1501–1531.
2. Karplus K, Sjölander K, Barrett C, et al. Predicting structures using hidden Markov models. Proteins Suppl 1997;1:134–139.
3. Jones D, Taylor W, Thornton J. A new approach to protein fold recognition. Nature 1992;358:86–89.
4. Flöckner H, Domingues F, Sippl M. Proteins folds from pair interactions: a blind test in fold recognition. Proteins Suppl 1997;1:129–133.
5. Fischer D, Eisenberg D. Protein fold recognition using sequence-derived predictions. Protein Sci 1996;5:947–955.
6. Rice D, Eisenberg D. A 3D–1D substitution matrix for protein fold recognition that includes predicted secondary structure of the sequence. J Mol Biol 1997;267:1026–1038.
7. Rice D, Fischer D, Weiss R, Eisenberg D. Fold assignments for amino acid sequences of the CASP2 experiment. Proteins Suppl 1997;1:113–122.
8. Di Francesco V, Geetha V, Garnier J, Munson P. Fold recognition using predicted secondary structure sequences and hidden Markov models of proteins folds. Proteins Suppl 1997;1:123–128.
9. Rost B, Schneider R, Sander C. Protein fold recognition by prediction-based threading. J Mol Biol 1997;270:471–480.
10. Elofsson A, Fischer D, Rice D, Le Grand SDE. A study of combined structure/sequence profiles. Fold Des 1996;1:451–461.
11. Dayhoff M, Barker W, Hunt L. Establishing homologies in protein sequences. Methods Enzymol 1983;91:254.
12. Rabiner L, Juang B. An introduction to hidden Markov models. Los Alamitos CA: IEEEE ASSP Magazine. Jan 4–15, 1986.
13. Krogh A. Two methods for improving performance of an HMM and their application for gene finding. ismb 1997;5:179–186.
14. Sonnhammer E, von Heijne G, Krogh A. A hidden Markov model for predicting transmembrane helices in protein sequences. ismb 1998;6:175–182.
15. Bowie JU, Lüthy R, Eisenberg D. A method to identify protein sequences that fold into a know three-dimensional structure. Science 1991;253:164–170.
16. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 1993;232:584–599.
17. Hubbard JT, Park J. Fold recognition and ab initio structure predictions using hidden Markov models and β-strand pair potentials. Proteins 1995;23:398–402.
18. Eddy SR. HMMER—hidden Markov model software. http://www.genome.wustl.edu/eddy/hmmer.html
19. Durbin R, Eddy S, Krogh A, Mitchison G. Biological sequence analysis. Cambridge, UK: Cambridge University Press; 1998.
20. Sander C, Schneider R. Database of homology derived protein structures and the structural meaning of sequence alignment. Proteins 1991;9:56–68.
21. Frishman D, Argos P. Seventy-five percent accuracy in protein secondary structure prediction. Proteins 1997;27:329–335.
22. Altschul S, Madden T, Schaffer A, et al. Gapped blast and ψ-blast: a new generation of protein database search programs. Nucleic Acids Res 1997;25:3389–3402.
23. Pearson W. Comparison of methods for searching protein sequence databases. Protein Sci 1995;4:1145–1160.
24. Abagyan R, Batalov S. Do aligned sequences share the same fold? J Mol Biol 1997;273:355–368.
25. Brenner S, Chothia C, Hubbard T. Assessing sequence comparison methods with reliable structurally identified evolutionary relationships. Proc Natl Acad Sci USA 1998;95:6073–6078.
26. Murzin AG, Breener SE, Hubbard T, Chothia C. Scop: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–540.
27. Moult J, Hubbard T, Bryant S, Fidelis K, Pedersen J. Critical assessment of methods of proteins structure predictions (CASP): round II. Proteins Suppl 1997;1:2–6.
28. Murzin A, Bateman A. Disant homology recognition using structural classification of proteins. Proteins Suppl 1997;1:105–112.
29. Henikoff S, Henikoff J. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci USA 1992;89:101915–10919.