Advanced Stochastic Protein Sequence Analysis

Thomas Plötz

Dipl.-Inform. Thomas Plötz
AG Angewandte Informatik
Technische Fakultät
Universität Bielefeld
email: tploetz@techfak.uni-bielefeld.de

Abdruck der genehmigten Dissertation zur Erlangung
des akademischen Grades Doktor der Ingenieurwissenschaften (Dr.-Ing.).
Der Technischen Fakultät der Universität Bielefeld
am 18.04.2005 vorgelegt von Thomas Plötz,
am 13.06.2005 verteidigt und genehmigt.

Gutachter:

PD Dr.-Ing. Gernot A. Fink, Universität Bielefeld
Dr. rer. nat. Karsten Quast, Boehringer Ingelheim Pharma GmbH und Co. KG
Prof. Dr. Jens Stoye, Universität Bielefeld

Prüfungsausschuss:

Prof. Dr. Robert Giegerich, Universität Bielefeld
PD Dr.-Ing. Gernot A. Fink, Universität Bielefeld
Dr. rer. nat. Karsten Quast, Boehringer Ingelheim Pharma GmbH und Co. KG
Prof. Dr. Jens Stoye, Universität Bielefeld
Dr.-Ing. Frank G. Zöllner, Universität Bielefeld

Gedruckt auf alterungsbeständigem Papier nach ISO 9706

Advanced Stochastic Protein Sequence Analysis

Dissertation zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)

der Technischen Fakultät der Universität Bielefeld
vorgelegt von

Thomas Plötz

Bielefeld – April 2005

Acknowledgements

I often compared the process of writing my PhD-thesis with a bicycle ride across an alpine pass: It is a long and hard ascend that requires a lot of endurance. Sometimes it hurts, but most of the time it is very exciting to climb the mountain and to reach different points of view which is the prerequisite for new thoughts and ideas. Similar to a ride across an alpine pass, writing my PhD-thesis would not have been possible without the support of a strong team which I would like to thank here.

First of all, I am very much obliged to my supervisor PD Dr.-Ing. Gernot A. Fink. Over the years, a very close collaboration with him has been established, including countless fruitful discussions with respect to all kinds of pattern recognition problems. His fresh and honest way always gives me the motivation to follow new ideas but also to at least double check them before arguing. It is difficult to imagine how my work would have been developed without the influence of his great experience and his numerous ideas. Thank you very much, Gernot, you actually made me stand here at the top of the alpine pass.

The research performed for this thesis was embedded in a cooperation with Boehringer Ingelheim and the Boehringer Ingelheim Pharma GmbH und Co. KG Genomics Group. I would like to thank the project partners, especially Dr. Andreas Weith, Dr. Karsten Quast, Dr. Andreas Köhler, and Ogsen Gabrielyan, for their enthusiastic support. I am very grateful to Dr. Quast who agreed to review this thesis.

Over the years Birgit Möller and I have become real friends, sharing similar ideas, supporting each other for individual goals and having fantastic discussions. Although Birgit wrote her PhD-thesis at the very same time as I, including her own hard ascend to her alpine pass, she always had an open mind for my problems. Her very exact proofreading including productive criticisms substantially helped me to improve the quality of this thesis.

My second proofreader was Erich Wehmeyer who checked the thesis for any language related traps and failures. I am very grateful for his native speaker expertise and his willingness to correct the work.

Furthermore, I would like to thank Prof. Dr.-Ing. Gerhard Sagerer, the leader of the Applied Computer Science group at the Bielefeld University, who always encouraged me to cut my own path even when it was much rockier than expected. The productive atmosphere within the working group including discussions, collaborations, chocolate and tea, gave the background for successfully finishing this thesis.

Finally, my wife Alexandra Schubert played an important role for the successfull completion of this thesis. I would like to thank her for her assistance, affection, and emotional support. Alex, without you, everything would count for nothing.

Contents

1 Introduction

Millennia ago, the ancient Egyptians used selected micro-organisms to produce cheese, wine and bread. Apparently, they were very experienced in food-making, because in Egypt one of the cradles of civilization could develop and the high quality catering certainly had a positive influence on this process. However, strictly speaking, they had no idea *why* their food became so tasty, giving them the power to build giant pyramids and to establish science and culture. It took ages until the reasons for it, the foundations of molecular biology, could be explained.[1]

In fact, not until 1866 the Augustinian monk Gregor Mendel developed the first general theory of heredity by means of the analysis of garden peas which represents the base for all further molecular biology research. Later on James Watson and Francis Crick discovered the double-helical structure of DNA in 1953 which determined the major breakthrough on the way to understanding the microbiological foundations of life [Wat53]. Between these principle breakthroughs lay almost hundred years and they were gained thousands of years after the Egyptians baked their tasty bread.

Due to the development of several revolutionary methodologies in the last decades, the speed of knowledge gain could be increased dramatically. Not before Fred Sanger and Walter Gilbert in 1977 independently invented powerful sequencing methods, nowadays' large-scale sequencing projects of complete organisms (e.g. the Human Genome Project [Lan01, Ven01]) became possible. In 1983 the polymerase chain reaction (PCR) was developed by Kary B. Mullis enabling the massive amplification of DNA to build vast amounts of identical copies which is a prerequisite for further analysis. Based on these more technological developments additionally enabling quantitative and not restricted exclusively to qualitative examinations, the focus of molecular biology could be shifted towards more complex questions such as the understanding of complete metabolic systems. Compared to the age of the ancient Egyptians, nowadays, due to the *principle* understanding of microbiological processes the secrets of e.g. tasty bread are known. Furthermore, insights into molecular biology even allow the development of synthetic drugs aiming at effective therapies against severe illnesses like cancer.

The analysis of genetic sequences plays a key role for modern molecular biology research. Once the genome of an organism is readily sequenced, the more difficult task of *understanding* the data, i.e. extracting knowledge from it, needs to be solved. First of all, genes must be localized within the DNA, followed by the prediction and classification of putative proteins. In order to reach higher-level knowledge about complex metabolisms, e.g. to develop therapies for healing illnesses, fundamental insights into the biochemistry of organisms, including the interactions between proteins are essential. Based on this data, pharmaceutical research is performed aiming at new (synthetic) drugs. Since huge amounts of data need to be analyzed, modern bioinformatics techniques play a key role in molecular biology.

[1] Good starting points for detailed readings about the life and food of ancient Egyptians are e.g. [Hel75, Red01].

In the last decade(s), bioinformatics has become an impressive success story. Compared to traditional research in molecular biology, i.e. explorations in the so-called wet labs, *in-silico* investigations are mostly cheaper and faster by some orders of magnitude. Here, the term in-silico stands for experiments using bioinformatics methods on computers. Thus, contrary to traditional research driven by individual cases (i.e. facts, organisms and compounds already known), broad systematic investigations in a high-throughput manner have become possible enabling more exhaustive explorations. Research in molecular biology is mostly based on some kind of pattern recognition, namely sequence comparison. Obviously, *computational* biology is predestinated for such tasks. The mapping of putative functions of various genes, predicted using bioinformatics methods, gave access to the understanding of at least parts of complex metabolic systems. Without such methods fundamental insights which are now widely accepted would not have been possible for years. Thus, the relevancy and the success of computational biology cannot be underestimated.

Encouraged by very promising research results, presently the so-called *post-genome* era has widely been proclaimed. Here, the focus of research lies on the analysis and understanding of complete biological systems, i.e. protein-protein interactions, or metabolic pathways. In fact, it is reasonable to analyze higher-level relationships between proteins in order to solve complex questions of molecular biology.

However, the post-genome era strongly depends on the results of the "preceding" genome era, which means the detection and classification of genes and proteins. Unfortunately, this sequence analysis problem is far from being solved. For example, the exact number of coded genes in the human genome is still not clear. In [Ven01] both a pessimistic gene number of 26 000 and a more optimistic figure of 40 000 is given. The Human Genome Consortium initially found evidence for approximately 30 000 transcripts and recently only 20 000 to 25 000 genes were supposed [IHG04]. Surprisingly, the sets of genes found by both groups interleave only to a percentage of approximately 21% [Hog01].

It is believed that only about 60 percent of the proteins encoded by human genes can be detected using present methodologies of molecular biology and bioinformatics. Reasons for this are manifold: the process of alternative splicing is still not completely understood and pseudo-genes exist which do not encode any actual proteins etc. Thus, protein prediction may already partially be doomed. Due to the complex three-dimensional folding mechanism, proteins with more or less similar biological functions exist, which are distantly related at the sequence level. These sequences are very difficult to find and even worse, their correct classification is currently almost impossible. So, if the classification of sequences already fails, the failure of the analysis of their interactions in metabolic pathways is almost preprogrammed.

The basic assumption of molecular biology is that similar functions of proteins are caused by similar structures (structure-function relationship). Classifying protein data simply follows this principle, whereas the biological functions themselves can be defined at various levels of granularity. The primary structure, i.e. the linear sequence of amino acids, which is obtained by the sequencing process as principally proposed by Sanger or Gilbert, mainly controls the three-dimensional structure of proteins. Once a new protein sequence is predicted, it is classified regarding its similarity to sequences whose functions are already known.

Traditionally, protein sequence analysis is performed using some kind of string comparison. The most obvious technique is Dynamic Programming, where two sequences are mutually aligned and alignment-costs are calculated serving as the base for classification. These techniques are suitable for closely related sequences, so-called close homologues, encoding basic biological functions. Unfortunately, the more abstract the biological functions of interest, the weaker the sequence-level similarities of proteins. However, these so-called remote homologues are much more interesting for molecular biologists than the functions encoded by closer homologue sequences.

For the classification of remote homologue sequences, probabilistic models of protein families are the methodology of choice. Based on various machine learning approaches, models for sequences sharing the same biological function are established and a more or less fuzzy evaluation is performed for classification. Although these models significantly outperform the traditional approaches already outlined, the general problem of remote homology classification is still not solved at all! Current probabilistic models suffer from several principle problems, thereby preventing further major breakthroughs in remote homology detection. As one example most of them require large sample sets for training *robust* models. Unfortunately, for most protein families of interest only very few sample sequences exist. As mentioned above, the functions of about 40 percent of the human proteins are still not clear. However, there is strong evidence that these proteins are in fact remote homologues. Consequently, in order to actually reach the post-genome era and to continue the success of modern molecular biology research, improved probabilistic models of protein families are badly needed. Therefore, substantial effort is dedicated to this field of research.

Focus

Formulating the analysis of protein sequences as a general pattern recognition problem, namely the treatment of signals evolving in time, the use of powerful probabilistic models became possible. For proteins, time is conceptually substituted by the location of amino acids in the sequences of interest. Such probabilistic models applied to bioinformatics tasks, originate from different application domains of pattern classification like automatic speech recognition or the classification of handwritten script. Consequently, the developments presented here represent a strict pattern recognition view on the bioinformatics problem of protein sequence analysis.

The goal of this thesis is the development of advanced stochastic models for protein families. Therefore, the currently most promising probabilistic modeling approach which is an enhancement of traditional pairwise sequence analysis techniques, namely Profile Hidden Markov Models (Profile HMMs), is analyzed. Based on the capabilities and drawbacks of Profile HMMs, enhancements for HMM based protein family modeling are developed. When applying advanced stochastic protein family models, substantial improvements for remote homology analysis tasks become possible serving as the base for obtaining further insights into biological processes. The results of improved remote homology analysis applying the new techniques can be used for e.g. pharmaceutical purposes during drug discovery.

The basic idea of enhanced protein family HMMs, is to adopt and to transfer techniques developed for alternative pattern recognition applications to the sequence analysis domain. In order to reach this, a more abstract view on biological sequence data as *signals* in their fundamental meaning is used. Based on these "protein signals", relevant features are extracted by applying various signal processing and general pattern recognition techniques. The resulting feature based protein sequence representation is the base for all further developments.

Contrary to current discrete Profile HMMs, semi-continuous feature based (SCFB) variants of protein family HMMs are developed. In combination with new techniques for both robust model estimation and evaluation, improved remote homology analysis becomes possible. In addition to SCFB Profile HMMs consisting of the same model architecture like state-of-the-art protein family HMMs, models with reduced complexity are developed. The basic motivation for this is the limitation of model parameters which need to be trained requiring substantial amounts of sample data. Once the complexity of protein family models is reduced while keeping (or even improving) their effectiveness for remote homology analysis, significantly less training sequences are sufficient for robust model estimation. Since the evaluation of feature based protein family models requires substantially higher computational effort, the focus of the developments is on efficient model evaluation techniques. Therefore, techniques known from alternative pattern recognition domains are adopted and transferred to bioinformatics tasks.

Organization of the Thesis

The thesis is principally divided into two parts. First, the state-of-the-art in sequence analysis is summarized. Here, the second chapter briefly reviews the foundations of modern protein analysis relevant for this thesis. Following this, the most important current sequence analysis techniques are discussed in chapter 3.

The second part of the thesis deals with the development of approaches for advanced stochastic protein family modeling. In chapter 4, first, the currently most successful probabilistic approach for remote homology analysis, namely modeling protein families using Profile HMMs, is quantitatively evaluated by means of a representative task. Based on the analysis of the capabilities of state-of-the-art techniques, in the second part of chapter 4 concepts for enhancements are presented. In chapter 5 advanced stochastic protein family models are described in detail. They were integrated into a prototypical HMM framework for remote homology detection – the GRAS^{2}P system.[2] By means of the GRAS^{2}P system, numerous experimental evaluations are performed. The presentation and discussion of their results is given in chapter 6.

The thesis is finished with a conclusion in chapter 7, where the key issues are reviewed. Furthermore, the practical application of techniques for advanced stochastic protein sequence analysis is summarized.

[2] GRAS^{2}P is an acronym for *G*enetic *R*elationships *A*nalysis based on *S*tatistical *S*equence *P*rofiles.

2 Principles of Modern Protein Analysis

The ancient Egyptians probably found their magic ingredients for making tasty cheese and bread, the micro-organisms responsible for fermentation, by chance. Nowadays, research activities related to molecular biology are well founded and more goal-oriented.

Basically, turning milk into cheese is an enzymatic reaction caused by certain bacteria. When the milk curdles, several proteins and their mutual interactions play an important role. Proteins are, however, also the reason for various diseases, no matter what organisms are actually attacked. Most of such illnesses are caused by malfunctions during synthesis of certain proteins. An immoderate increase of the number of certain proteins may lead to severe illnesses, e.g. cancer. On the other hand, the resulting lack of proteins if too few of them are generated may imply a similar dramatic effect. As an example of putative malfunctions in protein synthesis, diabetes is caused by missing the pancreas' capability of producing the protein *Insulin*.

The concept of protein-interactions can be generalized to any kind of metabolic processes. Consequently, the foundations of everyday life situations as well as of very complex tasks of molecular biology belong to the same base – proteins. Thus, research in molecular biology is always more or less related to them.

In this chapter the foundations of proteins and protein analysis in typical tasks of molecular biology are described. Here, the explanations are in no case exhaustive since this would go far beyond the scope of this work. Since the thesis is related to the improvement of probabilistic models for protein analysis, i.e. bioinformatics and general pattern recognition, only the relevant and absolutely necessary principles are summarized. For more detailed information regarding the fundamentals of molecular biology, the reader is referred to the numerous textbooks, monographs and special publications dealing with the topic from a more biological point of view. The argumentations given here, are mainly based on [Str91, Lew94, Bra98, Gon01, Mer03, Jon04].

First, in section 2.1 one of the most important principles of molecular biology is outlined – the central dogma. Here, the so-called information flow between the various levels of molecular biology is described, starting from DNA up to proteins. All further analysis is based on this foundation. The proteins themselves, as the result of a rather complicated process of gene expression which is unfortunately still not completely understood, are the fundamentals of life. Thus, the focus of this chapter lies on the description of proteins. In section 2.2 the biochemical properties are reviewed followed by a discussion of their meaning for metabolic processes. Protein analysis is motivated throughout the remaining parts of this chapter. In section 2.3 possible relationships between single proteins, i.e. the formation of families at various levels of abstraction, are described. Following this, the protein analysis scheme is summarized by means of the drug design task. This practical example of molecular biology processing will serve as a reference for further argumentation throughout the whole thesis.

2.1 The Central Dogma of Molecular Biology

The Augustinian monk Gregor Mendel established the concept of *genes* as basic units containing heredity information. Up to the year 1944 it was widely assumed that chromosomal proteins contain this genetic information. Based on Fred Griffith's work on *Pneumococcus* bacteria [Gri28], Oswald Avery, Colin MacLeod and Macly McCarty published their discovery that a nucleic acid of type deoxyribose plays an important role in heredity [Ave44]. The major knowledge they gained was that cleaned *Deoxyribonucleic Acid (DNA)* contains the genetic information. Following this, in 1953 James Watson and Francis Crick proposed a model for the double-helical structure of DNA [Wat53]. The detection of the importance of DNA as well as the correct description of its three-dimensional structure became the foundations of molecular biology.

DNA itself was discovered in 1869 by Johann Friedrich Miescher while isolating white blood cells. He (and others) found out that DNA represents large and rather simple molecules consisting of a sugar ring, a phosphate group and one of four nitrogenous bases: adenine (A), thymine (T), guanine (G), and cytosine (C). A fifth base was discovered, too, namely uracil (U) which is chemically similar to thymine. The chemical bonds linking together the nucleotides are always the same. Thus, the backbone of DNA is very regular and the "individuality" of each molecule is reasoned by the actual sequence of the bases A, T, C, and G. In figure 2.1 the biochemical composition of DNA is summarized.

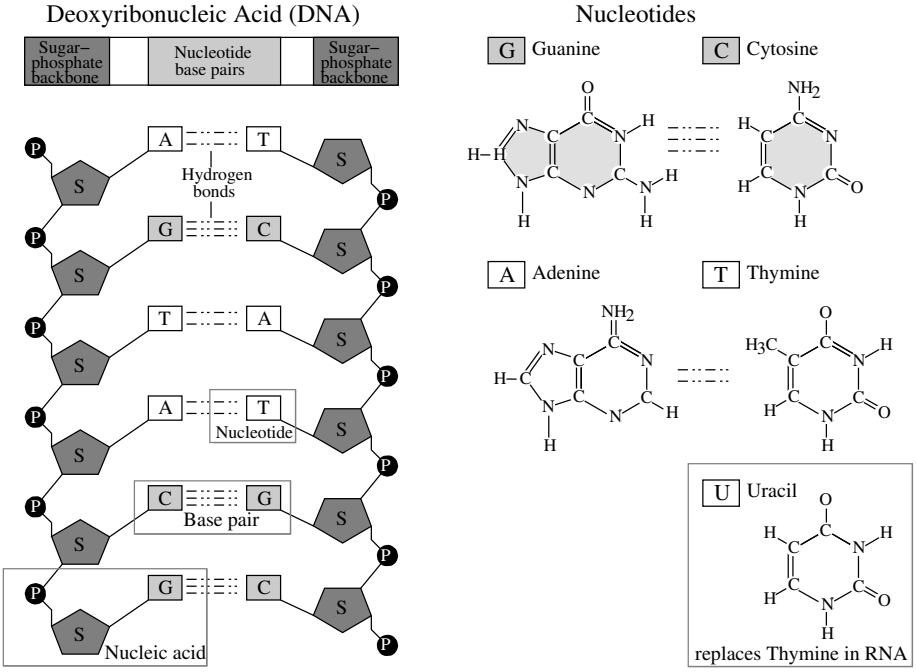


Figure 2.1: Illustration of the biochemical composition of DNA (courtesy of [Lej04]).

Generally, DNA contains all the information necessary for describing individual organisms of all living species. Here, genes play an important role since they are expressed as proteins, the fundamentals of life (which will be described in more detail in the succeeding

section). The genetic code defines the mapping of base triplets, so-called *codons*, to amino-acids forming the building blocks of proteins. So, the sequence of bases can be understood as a template for protein synthesis.

However, actually it is not DNA which is directly expressed but *ribonucleic acid (RNA)*. This template is generated by transcription in the form of *messenger RNA (mRNA)* – a working copy of the appropriate DNA fragment. Compared to the chemical composition of DNA, here, thymine is replaced by uracil. For prokaryotes, i.e. organisms consisting of cells not including a nucleus, this process of transcription is rather simple since their genes are always expressed in their complete length. Contrary to this, in organisms containing cells including a nucleus (eukaryotes), a difficult process of splicing, i.e. removal of non-coding parts, so-called *introns*, needs to be performed. For both kinds of organisms, the resulting mRNA is directly translated to proteins using the (redundant) genetic code, i.e. the mapping of base codons to amino acids. The principles of protein synthesis can be summarized as shown in figure 2.2.

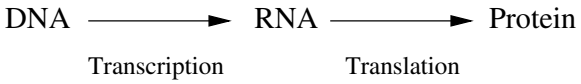


Figure 2.2: Principle of protein synthesis based on genetic information in DNA.

Depending on the actual species, genes are only one rather small fraction of DNA. As an example the human genome with a length of more than 3 000 megabases approximately contains only 30-40 000 genes. The redundant genetic code does not allow simple "back-transcription" since amino-acids are encoded by more than one codon. Genes and proteins are directly linked, since every protein is encoded by a gene. The inverse formulation of this principle does not hold for most higher-developed organisms. Compared to the moderate number of genes, the number of proteins synthesized from it is mostly considerably larger. Obviously, some genes code for multiple different proteins. Here the boundaries of introns are not fixed, resulting in different coding parts for the same gene in multiple expressions. In the literature this very complicated behavior is described as *alternative splicing*. Additionally, several genes exist which do not code for any proteins – so-called *pseudo-genes*. In summary, the information-flow in principle is an irreversible process. Based on these observations, Francis Crick formulated the *central dogma of molecular biology*:

> *"The central dogma states that once 'information' has passed into protein it cannot get out again. The transfer of information from nucleic acid to nucleic acid, or from nucleic acid to protein may be possible, but transfer from protein to protein, or from protein to nucleic acid, is impossible. Information means here the precise determination of sequence, either of bases in the nucleic acid or of amino acid residues in the protein."*
>
> Francis Crick, 1958
> (taken from [Lew94, p.161])

After complete genomes of various organisms could be sequenced (cf. the human genome [Lan01, Ven01]) one of the basic goals of molecular biology research is the actual deciphering of the data. In order to obtain fundamental insights in biochemical processes for

healing diseases etc. principle understanding of proteins as well as of protein synthesis is demanded. However, the central dogma of molecular biology severely constrains this discovery process.

2.2 Proteins: The Fundamentals of Life

Most components of cells in living organisms consist of only six different elements: hydrogen (H), carbon (C), nitrogen (N), oxygen (O), sulphur (S) and phosphor (P). Inside the cell they are linked and form molecules like water (H_2O) or phosphate (PO_4). Actually, most molecules have a really large size consisting of thousands of atoms. These *macro-molecules* are built up by lots of basic units. As very prominent examples of macro-molecules, polysaccharids like starch or cellulose represent long chains of sugar molecules.

2.2.1 Biochemical Composition

Basically, proteins are one of the most complicated kind of macro-molecules. Here, the basic units mentioned above are amino acids. Although recently further components were discovered (cf. e.g. [Böc91] for a review of selenocysteine and [Atk02] for the description of pyrrolysine), it is widely accepted, that only a limited number of 20 standard amino acids exists.[1] Every amino acid has a central carbon atom (C_α) to which a hydrogen atom (H), an amino group (NH_2) and a carboxyl group (COOH) are attached (cf. figure 2.3). The differences between the diverse amino acids are caused by the side chain (R) attached to the C_α. In fact, 20 different side chains are genetically specified where groups of three nucleotides, so-called codons, encode the biochemical composition of the side chain and thus the amino acid itself.

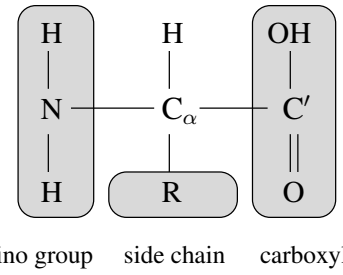


Figure 2.3: General structure of amino acids consisting of amino group (NH_2) and carboxyl group (COOH) as well as of the side chain (R), which determines the general differences among them.

Differences between the 20 amino acids are manifold. The side chains differ in their size, charge, hydrophobicity, chemical reactivity and shape. Whereas glycine has a rather simple side chain, namely a single hydrogen atom, e.g. phenylalanine contains a circular side chain of carbon atoms connected to additional hydrogens. All proteins of all species are based on this set of "building blocks". Carl Branden and John Tooze give an excellent

[1] In fact, the 21st and 22nd amino acids are very special cases rarely occurring due to posttranslational enzymatic modifications in negligible amounts of species. Thus, the general theory, till now, remains valid for the prevailing majority of organisms.

overview of the specialties of the different amino acids in [Bra98, p. 6f.]. In table 2.1 the names of the amino acids as well as their single-letter and their three-letter abbreviations are gathered. Furthermore, three common groups of amino acids are introduced, where the exact specification of the actual side chains is ambiguous. Further groups can be defined, depending on biochemical properties shared between their members.

Amino Acid	Three-Letter Code	Single-Letter Code
Alanine	Ala	A
Arginine	Arg	R
Asparagine	Asn	N
Aspartic acid	Asp	D
Cysteine	Cys	C
Glutamine	Gln	Q
Glutamic acid	Glu	E
Glycine	Gly	G
Histidine	His	H
Isoleucine	Ile	I
Leucine	Leu	L
Lysine	Lys	K
Methionine	Met	M
Phenylalanine	Phe	F
Proline	Pro	P
Serine	Ser	S
Threonine	Thr	T
Tryptophan	Trp	W
Tyrosine	Tyr	Y
Valine	Val	V
Either of D or N	Asx	B
Either of E or Q	Glx	Z
Undetermined	X	X

Table 2.1: Names and abbreviations of the 20 different amino acids occurring in proteins and the ambiguous groups B, Z and X.

In proteins the ends of two adjacent amino acids are joined by the formation of peptide bonds. Chemically this means, the carboxyl group of one amino acid condenses with the amino group of the next eliminating water. This process of bonding is repeated resulting in polypeptide or protein chains. The formation of a succession of such peptide bonds generates a “backbone”, from which the various side chains stick out. In figure 2.4 the creation of a peptide bond between two hypothetical amino acids is outlined. Both the carboxyl group of the molecule on the left-hand side, as well as the amino group on the right are broken. The freed atoms recombine to water and both amino acids are bonded by the peptide bond shown in the middle of the sketch.

The general structures of all proteins, namely the backbone chain of carbon and nitrogen atoms, are identical. The differences between proteins are *principally* caused by the sequence of the side chains R_n of the N amino acids involved. Due to its linear character-

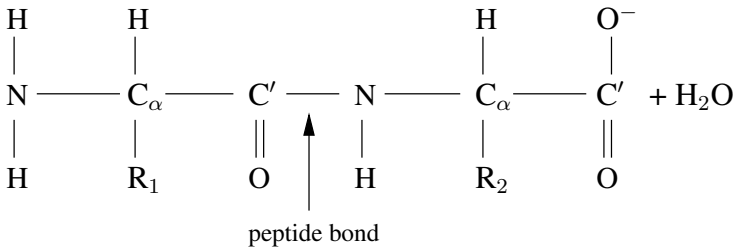


Figure 2.4: Creation of a peptide bond between two amino acids. The sequence of carbon and nitrogen atoms represents the backbone of the protein.

istics, the sequence is often called the *primary structure* of the protein.

Generally, proteins contain well defined three-dimensional structures. Actually, the function of a particular protein is defined by its structure i.e. the three-dimensional arrangement of the atoms. Within this conformation further sub-structures can be distinguished. The three-dimensional arrangements of sequences are strongly influenced by biochemical properties of the underlying amino acid rests, such as e.g. charge, hydrophobicity, or residue size.

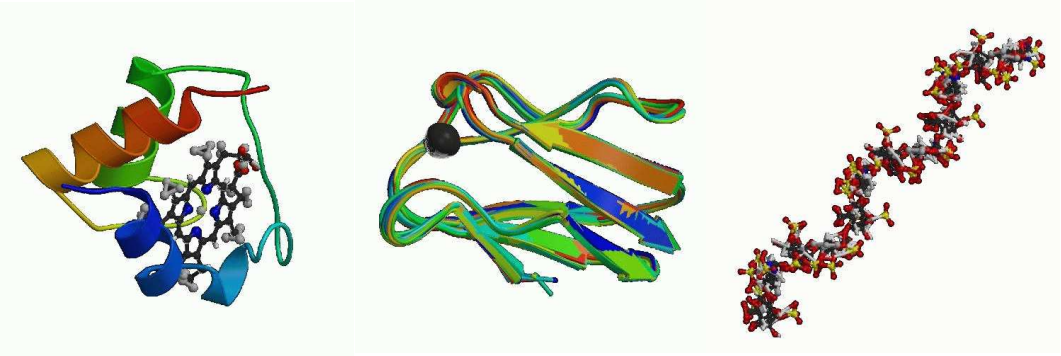
Dividing polypeptide chains into building blocks ranging from one C_α atom to the next C_α atom instead of using the peptide bond as delimiter as exemplarily shown in figure 2.4, is preferable for the description of structural properties of proteins.[2] Now, each C_α atom, except for the first and the last ones, belongs to two building blocks. All the atoms in such an unit are fixed in a plane with the bond angles and bond lengths nearly the same in all units in all proteins. By means of this alternative definition of peptide units, the side chains are not involved in the building blocks. The peptide units effectively represent rigid groups linked into a chain by covalent bonds at the C_α atoms. Thus, the only degrees of freedom they have are rotations around the bonds with angles ϕ and ψ. The local spatial arrangement of adjacent amino acids in regular steric conformations is called the *secondary structure* of proteins. Examples of such regular steric conformations are α-Helix, β-sheet or collagene-helix illustrated in figure 2.5.

Contrary to the secondary structure, the *tertiary structure* describes relationships between atoms (amino acid rests), that are further apart in the linear sequence. Obviously, the boundary between these two different structures is rather ambiguous.

For proteins consisting of more than a single polypeptide chain, a fourth kind of structure can be described – the *quaternary structure*. Here, higher-level building blocks are defined, namely the single polypeptide chains. The quaternary structure contains information regarding the spatial arrangement of such higher-level units including the description of their contact areas. In figure 2.6 the four standard structures are summarized.

In addition to the four kinds of protein structures outlined above, additional description levels were recently defined. In *super secondary structures* the aggregation of secondary structures is expressed serving as the transition between the secondary and the tertiary structure. Globally compact units are called *domains*. They have a special relevancy for

[2] Actually, the "natural" peptide bond remains valid since the alternative subdivision of polypeptide chains is used for *conceptual* argumentation only.



Cytochrome C–553 (Bacillus Pasteurii) | Plastocyanin (French Bean) | Disaccharide Repeating Units (Heparin)

Figure 2.5: Some regular steric conformations of proteins: α-helix, β-sheet and collagene-helix (from left to right; images taken from PDB [Ber02]).

higher-developed organisms because domains are often determined by exons, which are the coding parts of eucaryotic DNA. Usually domains represent the smallest units to which actual biological functions can be assigned.

Since the three-dimensional structure of proteins generally determines their function, the conformation is of immense importance. It is mainly determined by the sequence of amino acids, which was first proven by Christian Anfinsen [Anf73], subsequently becoming one of the principles of molecular biology.[3] Thus, for the majority of fundamental bioinformatics applications, namely the so-called sequence analysis techniques, the chains of amino acid symbols – the primary structures – serve as input data.

2.2.2 Biological Relevancy

As mentioned at the beginning of this chapter, proteins play a key role in almost all biological processes. In these premises, Lubert Stryer described the seven most important functions of proteins in his standard work [Str91, p. 15f.]. In order to prove the relevancy of proteins and to emphasize the demand for powerful protein analysis methods, these functions are briefly summarized.

Enzymatic catalyse: Most chemical reactions in biological systems, independent of the actual complexity of the reactions, are catalyzed by specific macro-molecules, so-called enzymes. *In vivo*, i.e. inside living organisms, the number of chemical reactions actually executed without these catalysts is almost negligible. In fact, the speed of chemical reactions is amplified by several orders of magnitude when enzymes are involved. The majority of currently known enzymes are proteins! Thus, it is actually proteins which control the chemical reactions in living organisms.

Transport and storage: Numerous small molecules and ions are transported by means of specific proteins. As a very prominent example, oxygen is transported within ery-

[3] In fact, Mr. Anfinsen received the Nobel prize for chemistry in 1972 for his epoch making work concerning the folding of ribonuclease based on the sequence of amino acids.

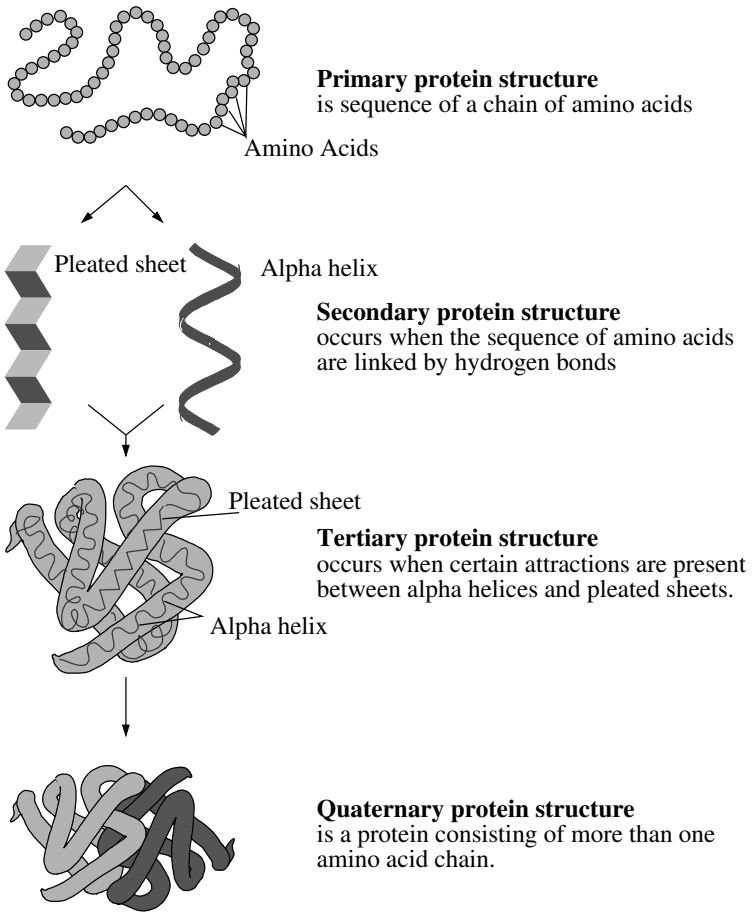


Figure 2.6: Summary of the three-dimensional arrangement of proteins: primary, secondary, tertiary and quaternary structures (courtesy of [Lej04]).

throcytes by hemoglobin and within muscle tissue by myoglobin. Both transporting substances are proteins and closely related.

Coordinated movement: Proteins are the essential elements of muscle cells. Both macroscopic and microscopic movements are based on contraction systems which are made of proteins.

Mechanical support: The high tensile strength of skin and bones is ensured by the protein collagen. Several illnesses like cellulitis, or even worse arteriosclerosis, are caused by lacking collagen proteins.

Immune defense: Antibodies represent highly specified proteins recognizing and destroying foreign substances like viruses or bacteria. Lacking such antibodies increases the probability of lethal virus attacks for almost any organism.

Creation and transmission of neural impulses: Receptor molecules, which are nothing else than proteins, transmit answers to specific stimuli. As an example, rhodopsin serves as the photo-optical receptor protein of the retina.

Control of growth and differentiation: The well controlled and temporary coordinated expression of genetic information is essential for the growth and differentiation of cells. Only a very small part of the genome is expressed. In higher-level organisms both growth and differentiation are controlled by growth factor proteins. If this mechanism fails, severe illnesses like cancer or diabetes may occur.

This non-exhaustive list of proteins' functions gives an overview of the immense relevancy proteins have in molecular biology. In fact, they are the fundamentals of life since life would not be possible without them. Nowadays, these fundamentals are investigated in basic research as well as in various application fields of molecular biology. Here, as very prominent examples both the pharmaceutical and the food industry need to be mentioned. Additionally, proteins are of major importance in other areas such as the development of new building materials based on synthetic adhesives [You99]. They may even play an important role for the next generation of computer architectures especially when considering that their (theoretical) storage capacity is enormous [Bir95, Gar99].

2.3 Protein Relationships

One foundation of molecular biology serving as the general definition for protein relationships can be summarized as follows:

Similar function of proteins is caused by similar structure.

The direct consequence of this principle is that the problem of protein analysis can be formulated as a classical pattern comparison task. Once the function of a single protein could be solved, e.g. in the traditional way in the wet lab, further knowledge can be obtained by finding proteins with similar structures. Usually, depending on the level of biological abstraction, proteins are clustered into so-called *protein families*, *superfamilies*, and *folds*.[4]

Generally the definition of protein similarity is based on the comparison of their three-dimensional structures. However, almost always primary structure data needs to be examined. As a result of complex folding processes spatially arranging the proteins, weak *sequence* similarities may occur although highly similar three-dimensional structures exist which dramatically complicates the sequence analysis task.

In this section a brief overview of the most common definitions of protein families and higher-level classification schemes is provided. Based on the *SCOP (Structural Classification Of Proteins)* classification [Mur95], a complete structural hierarchy is explained.

2.3.1 Protein Families

In section 2.2.1 the various levels of three-dimensional structures of proteins were described. Here, domains are introduced as globally compact units in the three-dimensional structure.

In fact, domains play a key role for the definition of relationships between proteins. When analyzing related proteins, it usually becomes obvious, that sequence similarity is not given

[4] Certainly, further subdivisions exist but since most definitions of abstraction levels are somewhat arbitrary, the argumentation in this thesis is restricted to the three most common levels.

for the complete length. Instead, regions containing strong similarities and sequence parts significantly diverging occur. The reason for this is the modular composition of proteins by means of domains whose exact definition is as follows (cf. [Mer03, p.12]):

> *A domain is the smallest unit of a protein containing a well defined structure which is spatially folded independently. Mostly, protein domains consist of 50-150 residues processing individual reactions whose interactions result in the overall function of the protein.*

Due to their fundamental biological meaning, domains are the criterion of choice for the definition of protein relationships. Note that domains are defined at the level of biological function which not necessarily coincides with sequential similarity. All sequences belonging to the same protein family contain the same domain. Especially proteins with smaller sequences contain only single domains. The actual name of the protein family is derived from the characteristic protein domain. The characterization of sequences belonging to a single protein family can be made in several ways, which will be described in detail in chapter 3.

Based on this definition of protein families, further higher-level relationships can be defined establishing a classification hierarchy with increasing level of abstraction for the definition of common biological functions. Actually, the borders between these levels are defined by means of the sequence identity percentages of the sequences belonging to the same units. To some extend, the definitions are rather arbitrary. “A residue identity of more than about 30% for clustering protein sequence pairs together into families is widely accepted in the literature” [Liu04]. Superfamilies are clusters of sequences sharing similar structures and evolutionary origins. In addition to this, groups can be defined by means of proteins having a common fold if their proteins consist of the same major secondary structures in similar arrangements.

Already in the 1970s it was postulated, that all proteins occurring naturally can be classified in certain families. The classification of protein sequences regarding their correct structural or functional family is of major importance for e.g. pharmaceutical research. In 1992 Cyrus Chothia supposed that the number of different families is rather limited [Cho92]. Concretely, he claimed, that only little more than 1 400 protein families exist. Although the actual number in the last few years shifted frequently in both directions ranging from 1 000 to 30 000 families and 400 to 10 000 folds etc., the basic assumption of an upper boundary for it remains valid.[5] So, depending on the level of biological abstraction the relationships between proteins can be formulated in different but limited number of ways.

2.3.2 Exemplary Hierarchical Classification

Throughout the years a large amount of protein sequences were obtained from various experimental sources. In order to allow molecular biologists a systematic exploration using these sequences, nowadays they are stored in central databases which are mostly publicly available. The most prominent, primary database is the *Brookhaven Protein Data Bank*

[5] Several statistical analysis concerning the theoretically exact number of protein families can be found e.g. in [Zha97, Ale98]. Due to the increase in the number of available protein sequences and thus the permanent change of the statistical base no *stable* number is presently accepted.

(PDB) which was established in 1971 at the Brookhaven National Laboratory, Long Island, New York, USA [Ber77]. Presently it contains more than 27 000 records describing the three-dimensional structures of macro-molecules.[6]

The PDB contains descriptions of protein structures without any classification regarding relationships. For this purpose, several additional databases exist, providing this classification based on various criteria. As an example, the goal of the SCOP database is the hierarchical classification of protein domains in terms of structural and evolutionary relationships [Mur95]. Here, the method used to construct the classification hierarchy is essentially the visual inspection and comparison of structures producing very accurate and useful results. The levels of abstraction within the classification hierarchy are defined as follows (cf. [Mur95]):

Family: Common evolutionary origins of protein domains are defined in two steps: first, all domains having residue identities of 30% and greater; second, protein domains with lower sequence identities but whose functions and structures are very similar.

Superfamily: Families whose members have low sequence identity percentages but whose structures and major functional features suggest that a common evolutionary origin is probable, are grouped in superfamilies.

Common fold: If members of superfamilies and families have same major secondary structures in the same arrangements with the same topological connections, they are defined as having a common fold.

Class: Different folds are grouped into classes. Most of the folds are assigned to one of five structural classes (based on the secondary structures which the sequences are composed of):

- All alpha (for domains whose structure is essentially formed by α-helices),
- All beta (the same as before but for β-sheets),
- Alpha and beta (for protein domains with α-helices and β-strands that are largely interspersed),
- Alpha plus beta (for those in which α-helices and β-strands are largely segregated), and
- Multi-domain (for those with domains of different folds and for which no homologues are known at present).

Presently, i.e. in the release 1.55, roughly 13 000 records of the PDB consisting of more than 30 000 domains are classified into about 600 folds, approximately 1 000 superfamilies and more than 1 500 families [Con02]. Together with the ever increasing number of PDB records and due to new research results, the size of the SCOP database steadily increases.

[6] For details regarding the PDB the reader is referred to the review article of Helen M. Berman and colleagues [Ber02].

2.4 Protein Analysis

Based on the foundations of molecular biology as basically described so far, nowadays, research activities are focused on gaining higher-level knowledge regarding the function and meaning of single proteins and their relationships. Traditionally, drug discovery is a very prominent branch of molecular biology research addressing protein analysis tasks.

The focus of this thesis is concentrated on the development of enhanced probabilistic models for remote homology analysis. Especially for pharmaceutical purposes, detecting new members of certain protein families is of major importance. Recently, the incorporation of bioinformatics techniques into certain parts of the drug discovery process initiated a general paradigm shift from experiments and studies inductively driven by the data in each segment of the value chain towards more deductive approaches. Here, instead of abstracting from already known facts about drugs, new knowledge is gained by means of broadband analysis of genome and protein data in a high-throughput manner.

In the following, the general drug discovery process is outlined with special focus on the incorporation of computational sequence analysis techniques. Throughout the remaining chapters of the thesis, this process will serve as *one* example for the application of techniques developed here. The argumentation is mainly adopted from the recent compilation of Alexander Hillisch and Rolf Hilgenfeld [Hil03].

2.4.1 The Drug Discovery Process

Drug development is an expensive and time-consuming process. A new drug today on average requires investments of $880 million and approximately 15 years of development, including the cost and time to discover potential biological targets, i.e. specific receptors identified to be modulated to alter their activity in some way for healing processes. Almost 75% of these costs is attributable to failure along the pharmaceutical value chain [Tol01]. More than half of the development time and thus the majority of investments is spent with clinical trials and the approval phase.[7]

Basically, molecular biology research in general and especially drug discovery can be understood as some kind of multi-stage “sifting-process”. Roughly speaking, given the universe of proteins, the number of candidates for a specific drug is reduced in a pipeline of cascaded techniques with increased complexity. At the end of this pipeline, hopefully, the desired substance is found and new drugs can be produced. Sorting out *early* substances not applicable is of major importance since higher-level techniques within the drug design pipeline are extremely complex and expensive. Generally, the drug discovery process can be divided into four main steps:

Target identification: “The identification of new and clinically relevant molecular targets for drug intervention is of outstanding importance to the discovery of innovative drugs” [Hil03, p.4]. Recently, it was estimated that present drug therapy is based on only about the tenth part of potential drug targets [Dre00]. Thus, a large potential for further developments presently remains unexploited. Traditionally, target identification is based on cellular and molecular biology. Bioinformatics techniques are

[7] For a detailed listing of the costs distribution regarding the specific phases of the drug development process cf. e.g. [Hil03, pp. 2ff] and the references therein.

applied in large scale to genomics and proteomics tasks in target identification. Here, novel drug targets are identified by systematically searching for paralogues of known drug targets, i.e. evolutionary related proteins that perform different but related functions. These new methods aim at discovering new genes or proteins and quantifying and analyzing differences in ill and healthy organisms. Since the genomes of complete organisms are available, it became more and more evident, that the complexity of biological systems lies at the level of proteins. It is at the protein-level that diseases become manifest and at which most drugs act [Hil03]. Thus, protein analysis techniques are extremely important for modern drug discovery.

Target validation/verification: Once a target has been identified, its relevancy in a disease process needs to be demonstrated. For this purpose, both, gain and loss of function studies are accomplished with so-called knock-out (loss of function) and knock-in (gain of function) animal models. Additionally, further proteomics approaches are applicable. Here, usually target hits obtained in the preceding step of target identification are verified by annotation of sequence sets with respect to targets already known.

Lead identification: Following to phases of exclusive treatment of target proteins, in the succeeding stages, actual compounds are sought, which interact with the target protein and modulate its activity. Two principal methods of compound identification are distinguished: random screening and rational design approaches. In high-throughput screening approaches, large numbers of compounds are tested for their ability to affect the activity of target proteins. Due to major progresses in molecular biotechnology, here a high degree of automation can be reached. Recently, alternative in silico or virtual screening becomes more and more common. Here, the docking processes of proteins are simulated using a computer and thus putative interactions relevant for pharmaceutical purposes can be investigated for lead identification.[8]

Lead optimization: In this final stage before actual (pre-)clinical tests and developments, several parameters regarding the biochemical composition of putative new drugs are optimized. This implies the chemical modification of small molecules and the subsequent pharmacological characterization. The goal of this very time-consuming and costly step is to obtain compounds with suitable pharmacologic properties to become a drug. Here, higher-level in vivo as well as in vitro and in silico techniques are applied to the drug candidates.

Following these steps, the remaining substances can be tested in pre-clinical and clinical environments (the fifth step of the general drug discovery process) which is extremely relevant to safety, and is thus closely monitored by several governmental organizations. Due to the enormous effort and costs required for the final stage, only the most promising candidates should enter this level. In figure 2.7 the drug discovery process is summarized by means of two triangles symbolizing the four main stages and the costs implied up to the final stage of (pre-)clinical development (fifth step at the top). The larger the progress of the process, the smaller the number of candidates remaining.

[8] Inbal Halperin and colleagues give an excellent overview of principles of protein docking used for computational lead identification, see [Hal02].

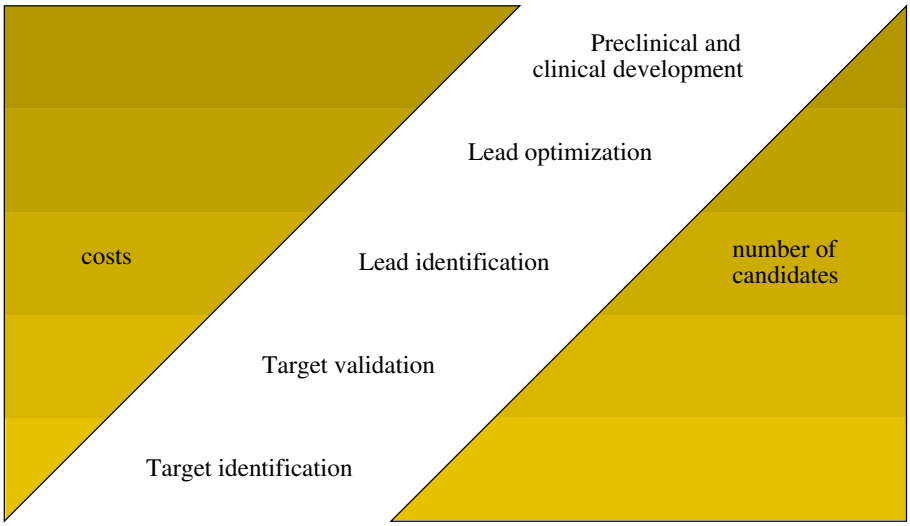


Figure 2.7: Sketch of the principle drug discovery process as a candidate elimination task: the larger the number of candidates in higher levels of drug detection, the higher the costs.

2.4.2 Protein Sequence Analysis

In the last few years, drug discovery has been strongly influenced by modern biotechnology and bioinformatics techniques. Due to forced automation, procedures within the phases of lead identification and optimization have been improved significantly, yielding better efficiency and thus accelerated developments.

In addition to this, the initial stages of the drug discovery process (specifically target identification and validation), can greatly benefit from bioinformatics and especially from sophisticated sequence analysis methods. This is reasoned by the fact that sequences of amino acids contain a huge amount of information. By means of modern information technology, comprehensive databases, and powerful computer (networks), this information can be exploited almost completely automatically. In [Str91, p. 60f.] the relevant information contained in protein sequences are summarized as follows:

- The comparison of a specific protein sequence with known sequences may uncover presently unknown family relationships. Thus, the function of the new protein can be predicted and deeper biological insights of higher-level biological systems can be obtained.

- Comparing the sequences of a single protein in various species gives hints regarding evolutionary pathways. By means of the proteins' differences across the species phylogenetic trees can be derived which are very useful for the analysis of species relationships.

- Due to the analysis of amino acid sequences, repeating sequence parts within proteins can be discovered. This is important for the analysis and understanding of evolutionary developments since numerous proteins developed from a single ancient gene by duplication and succeeding diversification.

- For the understanding of metabolic processes it is of major importance to know the location of the proteins and the timing of their expression. Sequences of amino acids contain signals designating the location of a protein and its processing.
- The analysis of sequence data provides the foundations for the synthesis of specific antibodies attacking specific proteins.

By applying modern bioinformatics technologies, the investments needed to develop drugs could be reduced by approximately $300 million and the time spent on developments could be cut by two years [Tol01]. Thus, in addition to the scientific progress due to additional gains in knowledge obtained by the more deductive approach of e.g. target identification and validation, the whole process can be cheapened and shortened. However, it must be mentioned that several potential obstacles like quality problems of the new targets or processing bottlenecks exist and need to be managed.

2.5 Summary

Proteins are the fundamentals of life and hence subject to a wide variety of research activities within molecular biology. Based on 20 different amino acids, numerous different proteins are synthesized by all living organisms enabling both basic biological functionalities and complex higher-level metabolic processes. Generally, the biological function of a protein is mediated by its three-dimensional structure which is mainly determined by the linear sequence of amino acids. One of the fundamental principles of molecular biology states that similar biological function of different proteins is reasoned by similar structure.

The majority of molecular biology research is based on this general principle which also justifies the formalization of protein relationships in so-called protein families, superfamilies, and folds. In order to uncover hidden evolutionary pathways, phylogenetic relationships between organisms and further coherencies, protein sequence analysis is of major importance. Here, unknown protein sequences are classified regarding their affiliation to certain protein families for determining their biological functions.

One example for the application of protein analysis techniques is the drug discovery process, which could benefit significantly from so-called in silico approaches, i.e. protein sequence analysis using bioinformatics methods. By means of computationally supported target identification and validation the time and money consuming process could be accelerated. Here, for therapeutically relevant protein families additional members are explored. Although promising results could be obtained, the problem of protein sequence classification is far from being solved. Especially for protein families containing sequences with weak residual similarities, the automatic prediction often fails. Thus, improved methods are solicited.

3 Computational Protein Sequence Analysis

> *"The probability that a functional protein would appear de novo by random association of amino acids is practically zero. In organisms as complex and integrated as those that were already living a long time ago, creation of entirely new nucleotide sequences could not be of any importance in the production of new information."*
>
> François Jacob [Jac77]

In the remarkable article of François Jacob about evolution and tinkering, the general conclusion was drawn that evolutionary processes in nature are in no way comparable to engineering approaches. Richard Durbin and co-workers very laconically summarized Jacob's argumentation at the beginning of their standard work on biological sequence processing: "Nature is a tinkerer and not an inventor" [Dur98, p.2].

In fact, it is this basic evolutionary paradigm that opens the research field of biological data analysis for the application of automatic computational sequence comparison techniques. Throughout the generations, by means of an extremely powerful mechanism of selection and duplication – the evolution – nowadays' fundamentals of life, i.e. proteins encoded by genetic sequences, emerged from common ancestors. Basically, the goal of almost all research activities dedicated to the field of protein analysis is to uncover the mutual relationships between proteins implied by evolutionary processes. At the level of primary structure data the natural tinkering process can possibly be reproduced by analyzing differences and similarities of particular protein sequences. Obviously, this is a task which is predestinated for automatic approaches.

Strictly speaking in terms of computer science, sequences, either DNA- or protein data, can be understood as strings of fixed lengths containing characters from a given lexicon or alphabet. For protein data, this inventory consists of the 23 single-letter codes of the 20 amino acids plus the ambiguous groups B, Z, and X (cf. table 2.1 on page 9 for details). Thus, the universe of proteins generally represents all words of a formal language. Unfortunately, the grammar of this language is not known. For this reason, analysis is performed by string comparison approaches. Traditionally, protein relationships are explored by mutually aligning sequences and calculating scores for the operations necessary to transform one sequence into another. These scores are used for the decision regarding putative relationships.

In this chapter the state-of-the-art of biological sequence comparison techniques is summarized. In the first part (section 3.1) fundamental direct sequence to sequence comparison techniques, so-called pairwise alignment methods, are reviewed. Since almost all approaches are based on Dynamic Programming and scoring techniques, this part begins with their general description. Especially for remote homology detection, pairwise sequence alignment is not always the methodology of choice. In order to capture highly diverging sequences belonging to a particular protein family of interest, these families are often explicitly modeled by various approaches. Thus, in the second part of the chapter (section 3.2),

sequence family analysis is described in detail. Enhanced probabilistic models for remote homology detection developed in this thesis are based on current stochastic approaches. Consequently, here the focus lies on probabilistic techniques, namely Hidden Markov Models. One basic goal of this thesis is the adoption of general pattern recognition techniques like signal processing methods to the bioinformatics task. There is hardly any literature related to this field of research. Although the problem of remote homology detection has still not been solved, this very promising branch of technology is almost completely neglected. In the final part of this chapter some of the rare sequence comparison approaches based on signal processing techniques are outlined.

In the last few years a huge amount of literature dedicated to protein sequence analysis has been published. When not explicitly referenced otherwise, the argumentations in this chapter are based on [Dur98, Sal98, Bal01, Mer03, Jon04, Mou04].

3.1 Pairwise Sequence Alignment

The process of natural tinkering can be observed for biological substances especially at the molecular level. Specific proteins evolutionary emerged by steady modifications and selections based on common parental sequences. It is a specialty of tinkering processes, that particular goals can be reached in multiple different ways. Speaking in terms of proteins' evolution, specific biological functions, i.e. three-dimensional structures, can be encoded by various alternative protein sequences. Generally, given two resembling sequences the probability of similar function and/or structure is rather high [Mer03, p.85].

By means of a comparison of spatial protein structures and the corresponding sequences, Chris Sander and Reinhard Schneider examined a threshold for the percentage of sequence similarity that implies common three-dimensional structures with high statistic significance. This threshold depends on the length of the sequences and exemplarily for sequences containing at least 80 residues, 30 percent identity is sufficient for the implication of structural similarity [San91].

Due to these principles, pairwise sequence comparison is of major importance for molecular biologists. If sequences are identical to some percentage, it implies strong evidence for structural and/or functional similarity. Typical applications of pairwise alignment techniques are the comparison of unknown sequences to a database of sequences already known and the classification regarding similarity, i.e. evolutionary distance, based on alignment scores. Such approaches require a clear definition of sequence similarity, which usually implies some metric d measuring the "distance" between two strings $\vec{s}_1$ and $\vec{s}_2$.

For the comparison of two arbitrary vectors, numerous different metrics were defined mathematically. The Minkowski distance as defined in equation 3.1, represents the most prominent *general* metric for arbitrary n-dimensional data vectors $\vec{x}$ and $\vec{y}$ parameterized by $k, k = 1 \ldots \infty$:

$$d_M(\vec{x}, \vec{y}) := \left(\sum_{i=1}^{n} |x_i - y_i|^k \right)^{\frac{1}{k}} . \tag{3.1}$$

Most notably, for $k = 1$ the city block (or Manhatten) distance is defined and $k = 2$ represents the well known Euclidean distance. In addition to such general metrics, specialized

distance measurements for strings exist. As one prominent example, the Hamming distance, which originated in information theory, counts the number of different characters in the two compared strings. Distances of that kind are very important, especially for information transfer.

Although well defined and commonly used in various application fields, metrics as defined above can only rarely be used for sequence comparison tasks. This is reasoned by the fact that they require strings of equal lengths. Due to evolutionary tinkering, in protein analysis tasks the comparison of sequences with identical lengths is usually the exception. Furthermore, sequences containing common sub-strings which are slightly shifted from one string to another will not be identified as similar by means of distances described so far (cf. figure 3.1). Contrary to this, in biological context such sequences are indeed similar! Thus, more flexible metrics were introduced processing both different string lengths and putative internal shifts.

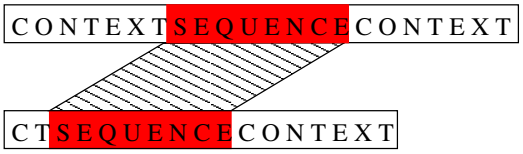


Figure 3.1: Illustration of a shift of identical sequence parts in strings containing different lengths. By means of traditional general metrics such as Minkowsky or Hamming distance the actual similarity between both strings (red boxes) wont be captured correctly.

Generally, for the comparison of biologically related strings, the edit- or Levenshtein-distance is widely used [Lev66]. Here, the minimal costs of insertions, deletions and substitutions required for transforming one string into another determine the distance between both sequences. Depending on the actual application, these edit-operations are individually scored. Compared to plain distance calculation for strings of equal length, here sequences are mutually *aligned* by wisely inserting, deleting, substituting, or matching characters and the scores for similarities are calculated as the “reciprocal” distances. For biological sequence analysis such scoring techniques are the methodology of choice. Throughout the years, a large variety of approaches for efficient and/or optimal alignment were proposed. The basic technique for optimally aligning sequences to each other is called *Dynamic Programming (DP)*.

3.1.1 Principles of Sequence Alignment

In the following the fundamentals of Dynamic Programming are briefly summarized with respect to the application of protein sequence analysis. First, the general scheme which produces scored alignments for two strings is introduced. Since the general meaning of sequence data to be aligned is rather important, several different scoring techniques, i.e. definitions of costs for the particular edit operations, exist. Mostly they are summarized in scoring matrices defining the substitution costs for exchanging symbols (amino acids). Thus, the principles of such scoring schemes are outlined. Although DP techniques guarantee optimal alignments scored depending on the substitution scheme actually used, they cannot provide the general decision whether two sequences are related. Even completely

unrelated sequences (e.g. random data) will be aligned optimally in the mathematical sense. The final classification result based on alignment scores needs to be extracted by means of the analysis of statistical significance of the score. Thus, in the third part of this section, the scoring process is briefly discussed in a more statistical manner.

Dynamic Programming

In his survey of the principles of Dynamic Programming, Sean Eddy gives an interesting explanation of the etymology of the term 'Dynamic Programming' [Edd04]. The technique itself was formalized in the early 1950s by Richard Bellman, who was working as a mathematician at RAND Corporation on optimal decision processes. He was searching for an impressive name that would shield his work from US Secretary of Defense Charles Wilson, who apparently was rather negatively minded concerning mathematical research. Since Bellmann's work involved time series and planning, he opted for 'dynamic' and 'programming' which initially had nothing to do with computers. He also liked the impossibility of using the term 'dynamic' in a pejorative sense. Bellman figured 'Dynamic Programming' was something not even a congressman could object to. For obvious reasons, the explanations regarding Dynamic Programming given in this thesis are focused on its applications to bioinformatics tasks implying string alignments. However, the original work of Richard Bellman is much more universal [Bel57].

The trivial solution of the alignment problem is the scoring of all possible alignments for two given strings and the selection by means of the optimal, that means the maximal score. Unfortunately, this is computationally not feasible, since there are approximately 2^{2N} different alignments for two sequences of length N [Edd04]. For sequences consisting of different numbers of residues the situation gets even worse. Thus, more sophisticated methods were developed. The general goal of all DP based techniques discussed here is the mutual alignment of two strings whereas partial results are calculated *just-in-time*, i.e. they are available at the time they are required for further calculations. Generally, each step of the algorithm reuses results of preceding steps enabling a recursive definition of DP.

The global alignment problem is divided into sub-problems lowering the complexity. For this reason, DP algorithms consist of four parts:

1. A recursive definition of the optimal score,

2. A Dynamic Programming matrix for storing optimal scores of sub-problems,

3. A bottom-up approach for filling the matrix by solving the smallest sub-problem first, and

4. A technique for traceback of the matrix to obtain the *global* optimal solution.

By means of a practical example in the following the DP algorithm will briefly be outlined. Therefore, two hypothetical protein sequences $\vec{s}_1 =$ ACDEF of length $N = 5$ and $\vec{s}_2 =$ GAHCDFE consisting of $M = 7$ residues are considered.

Recursive definition of the optimal score: The global alignment of both sequences $\vec{s}_1$ and $\vec{s}_2$ can end in three different ways: (i) the residues s_{1_N} and s_{2_M} are aligned to each other; (ii)+(iii) either final residue is aligned to a gap character whereas the end-residue of the remaining sequence was already aligned before. The optimal alignment will be the highest scoring of these three cases. Since the global alignment problem breaks into independently optimizable pieces, the solution of this sub-problem can be generalized recursively. As an example the scores of the three cases mentioned can be defined in terms of the optimal alignment of the preceding subsequences (prefixes). The score S of case (i) above is the score $S(s_{1_N}, s_{2_M})$ for aligning s_{1_N} and s_{2_M} plus the score $S(\vec{s}_{1_{1\ldots N-1}}, \vec{s}_{2_{1\ldots M-1}})$ for an optimal alignment of everything else up to this point. For the remaining cases (ii) and (iii) above, gap-penalties γ, i.e. negative scores penalizing the insertion of gaps, are added to the scores $S(\vec{s}_{1_{1\ldots N-1}}, \vec{s}_{2_{1\ldots M}})$ and $S(\vec{s}_{1_{1\ldots N}}, \vec{s}_{2_{1\ldots M-1}})$, respectively. Consequently, the recursive definition of the optimal alignment score can be formulated as follows:

$$S(s_{1_i}, s_{2_j}) = \max \begin{cases} S(s_{1_{i-1}}, s_{2_{j-1}}) + \sigma(s_{1_i}, s_{2_j}), \\ S(s_{1_{i-1}}, s_{2_j}) + \gamma, \\ S(s_{1_i}, s_{2_{j-1}}) + \gamma, \end{cases}$$

where $\sigma(s_{1_i}, s_{2_j})$ designates the score for aligning two residues s_{1_i} and s_{2_j}. For simplicity of argumentation *here* identical scores σ for all substitutions of any residues, and identical gap-penalties γ are assumed. In fact, the actual adjustment of these scores is rather crucial regarding the quality of the resulting alignment. Thus, substantial differences between various alignment algorithms are mostly based on sophisticated selections of these scores (see pages 28ff for details). The recursion is followed until trivial alignment problems with obvious solutions are reached (aligning nothing to nothing produces a score $S(\epsilon, \epsilon)$ of zero).

DP matrix for storing optimal scores of sub-problems: A principle problem for recursive algorithms is the very probable wastage of computational time due to multiple calculations of sub-problems occurring more than once. The deeper the recursion is moved into, the worse the situation will become. One obvious solution for such problems is to keep track of intermediate results. The fundamental difference between simple recursion and DP techniques is the use of a matrix for memorizing results, so that each sub-problem is solved only once. Structurally, the DP matrix consists of a tabular arrangement of the sequences to be aligned (cf. e.g. figure 3.2). Step-by-step every cell is filled with local alignment scores as described in the following.

Bottom-up approach matrix filling: Based on the recursive definition of the optimal alignment score, the boundary conditions for the leftmost column and the topmost row of the DP matrix are already known: $S(\epsilon, \epsilon) = 0;\ S(s_{1_i}, \epsilon) = \gamma;\ S(\epsilon, s_{2_j}) = \gamma$. For all trivial alignment cases the scores mentioned are used for filling the relevant parts of the matrix. Now, in a bottom-up way, from smallest problems to progressively bigger problems, the matrix is filled by evaluating the recursive definition of alignment scores. Since all results are stored inside the matrix, redundant calculations are avoided. For the global alignment of both strings at every step a traceback-pointer is kept designating the predecessor which leads to the locally optimal score of the appropriate step.

Traceback of the matrix to obtain the *global* optimal solution: The final *global* alignment of both strings considered can be obtained after the DP matrix has been filled completely. Starting in cell (N, M), i.e. at the lower-right corner of the matrix, determines the beginning of the traceback for uncovering the global alignment. The traceback follows the pointer yielding the maximal score for the next step. Note, that equal scores may occur. Here, the actual successor in the global alignment will be determined application dependent. The algorithm stops, when the traceback arrives at the upper-left corner of the matrix.

In figure 3.2 the optimum alignment of the exemplary sequences $\vec{s}_1 =$ ACDEF and $\vec{s}_2 =$ GAHCDFE is outlined. They are arranged in tabular form as described above and the DP matrix is filled using the recursive definition of the alignment score. Here an exemplary scoring system of +5 for a match, -2 for a mismatch and -6 for each insertion or deletion is applied. The red cells represent the final global optimal path for the string alignment which is shown below the matrix yielding the score of -1.

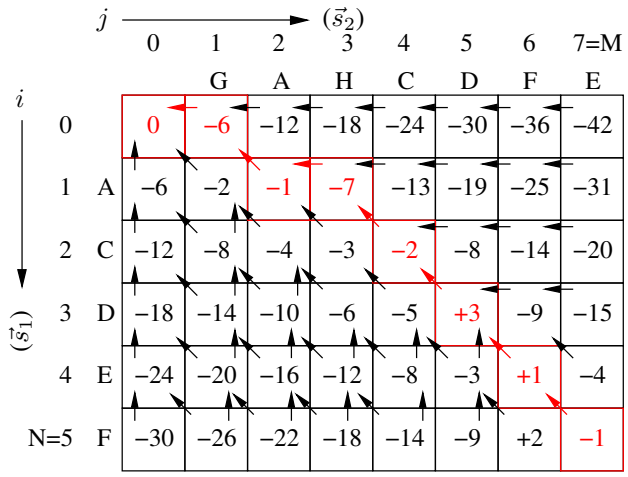


Figure 3.2: Global sequence alignment by applying the Dynamic Programming technique: By means of a recursive definition of the alignment score the DP matrix is filled and the global optimal path is extracted by traceback (red cells). The final alignment is shown below the matrix. Scoring system: match +5, mismatch -2, insertion/deletion -6.

According to the authors first using this dynamic programming approach for global alignments of protein sequences, this technique is mostly referred to as *Needleman-Wunsch algorithm* [Nee70, Got82]. Its algorithmic complexity can be approximated by $O(NM)$ for both memory and time requirements.

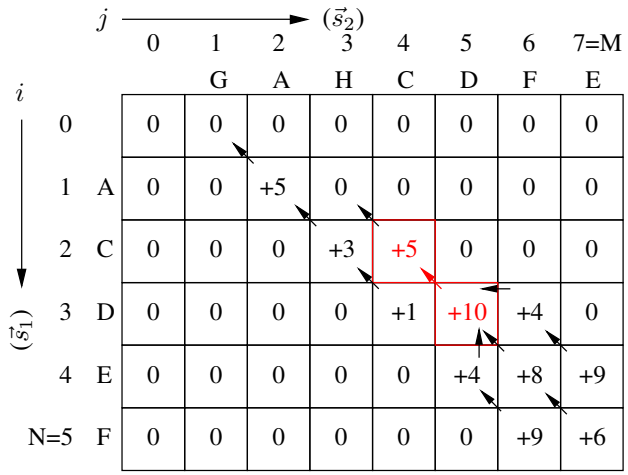
The previously described global alignment algorithm is widely used for protein classification tasks. Here, complete sequences are mutually aligned which is very useful for segmented data, i.e. sequences containing well defined start and end positions. Contrary to this, in molecular biology applications the alignment of parts of sequences is often demanded. Methods tackling the problem of finding partial matches are usually called local-alignment

techniques. The local version of Dynamic Programming techniques was developed in the early 1980s and is usually known as *Smith-Waterman* algorithm according to its inventors [Smi81, Got82]. The method is closely related to the algorithm described above. Basically, there are two major differences. First, a boundary of the locally optimal scores is introduced allowing $S(s_{1_i}, s_{2_j})$ to become zero if all other options have values less than 0:

$$S(s_{1_i}, s_{2_j}) = \max \begin{cases} 0, \\ S(s_{1_{i-1}}, s_{2_{j-1}}) + \sigma(s_{1_i}, s_{2_j}), \\ S(s_{1_{i-1}}, s_{2_j}) + \gamma, \\ S(s_{1_i}, s_{2_{j-1}}) + \gamma. \end{cases}$$

Reaching the boundary, i.e. the 0-case, implies the termination of a local alignment. Due to the zero option the initial filling of the DP-matrix is slightly changed. Instead of inserting the γ-values, these cells are filled with 0.

The second difference of local alignment methods compared to the general DP approach is that now an alignment can start anywhere in the matrix. Instead of taking the value in the bottom right corner for the best score, now the global maximum is searched over the whole matrix. The traceback ends if a cell containing 0 is met. Basically, multiple local maxima can occur and in fact one refinement of standard local alignment techniques is to process all high-scoring fragments of a pairwise alignment. However, the basic technique remains the same. Thus, all appropriate argumentation given here is directed to the principle procedure. In figure 3.3 the local alignment of the sequences given above is shown.



Optimum alignment score: +10

C	D
C	D
+5	+5

Figure 3.3: Local sequence alignment by applying the Smith-Waterman algorithm: The final alignment starts at the global maximum within the whole matrix and ends at the first cell containing a value of 0. Again, the optimal path is extracted by traceback (red cells). The final alignment is shown below the matrix. Scoring system: match +5, mismatch -2, insertion/deletion -6.

The alignment algorithms outlined here are the base for numerous refined approaches. Important enhancements are the possibility of finding multiple local matches or of finding overlapping matches. Since the focus of the thesis concentrates on probabilistic models of sequence families no exhaustive review of such refinements will be given here. Instead, the reader is referred to the tremendous amount of specialized publications addressing such refinements. A good starting point for this is the standard work of Richard Durbin and colleagues [Dur98]. However, due to their enormous importance for nowadays' molecular biology research, in section 3.1.2 two of the most prominent heuristic approaches for bounding the computational complexity of DP techniques will be presented. In the following, the parameterization of the general DP framework, i.e. the adjustment of the particular scores for edit operations, is explained.

Scoring Matrices

For simplicity of argumentation the basic sequence to sequence alignment techniques the previous sections dealt with were described using fixed scores for the substitution of residues.

When applying these sequence alignment techniques to practical applications of molecular biology, this simplification does not usually hold. Instead, the plain algorithms are used as some kind of general frameworks for sequence alignment. Depending on the actual application, i.e. the particular data inspected and the target domain of the investigations, these frameworks need to be configured wisely. Configuration here means, the adjustment of specified substitution scores for the edit operations involved in Dynamic Programming. The alignment procedures outlined above used identical scores (+5 for match and -2 for mismatch) for all arbitrary residue alignments. According to biological realities this adjustment is insufficient since it assumes the same probabilities for substituting any amino acid with any other. Obviously, this does not hold true for real applications. It is quite unrealistic to score the substitution of residues containing similar biochemical properties in the same way as the substitution of amino acids that completely differ. Thus, usually the substitution scores are adjusted in a per residue manner. These specific scores are arranged in tabular form in so-called substitution matrices. Recently, Shuichi Kawashima and Minoru Kanehisa compiled 71 mutation matrices [Kaw00].

According to the two main categories for sequence alignment applications, two general types of scoring matrices were created:

Reconstruction of evolutionary processes: By means of scores contained in matrices addressing the reconstruction of evolutionary processes mutation rates are represented. Usually, they are constructed based on the analysis of sequences and their reconstructed ancestors.

Comparison of protein domains: Here, the scores are based on the composition of protein domains at hand or close relatives of them. Usually, these entries are calculated by means of substitution frequencies which can either be measured in wet lab experiments or via information theoretical approaches.

Throughout the years, a large variety of specific matrices were created for both categories. In figure 3.4 the most prominent matrices for both types are shown. These matrices are sym-

	C	G	P	S	A	T	D	E	N	Q	H	K	R	V	M	I	L	F	Y	W
C	12																			
G	-3	5																		
P	-3	-1	6																	
S	0	1	1	1																
A	-2	1	1	1	2															
T	-1	0	0	1	1	3														
D	-5	1	-1	0	0	0	4													
E	-5	0	-1	0	0	0	3	4												
N	-4	0	-1	1	0	0	2	1	2											
Q	-5	-1	0	-1	0	-1	2	2	1	4										
H	-3	-2	0	-1	-1	-1	1	1	2	3	6									
K	-5	-2	-1	0	-1	0	0	0	1	1	0	5								
R	-4	-3	0	0	-2	-1	-1	-1	0	1	2	3	6							
V	-2	-1	-1	-1	0	0	-2	-2	-2	-2	-2	-2	-2	4						
M	-5	-3	-2	-2	-1	-1	-3	-2	0	-1	-2	0	0	2	6					
I	-2	-3	-2	-1	-1	0	-2	-2	-2	-2	-2	-2	-2	4	2	5				
L	-6	-4	-3	-3	-2	-2	-4	-3	-3	-2	-2	-3	-3	2	4	2	6			
F	-4	-5	-5	-3	-4	-3	-6	-5	-4	-5	-2	-5	-4	-1	0	1	2	9		
Y	0	-5	-5	-3	-3	-3	-4	-4	-2	-4	0	-4	-5	-2	-2	-1	-1	7	10	
W	-8	-7	-6	-2	-6	-5	-7	-7	-4	-5	-3	-3	-2	-6	-4	-5	-2	0	0	17

PAM250

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4																			
R	-1	5																		
N	-2	0	6																	
D	-2	-2	1	6																
C	0	-3	-3	-3	9															
Q	-1	1	0	0	-3	5														
E	-1	0	0	2	-4	2	5													
G	0	-2	0	-1	-3	-2	-2	6												
H	-2	0	1	-1	-3	0	0	-2	8											
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

BLOSUM62

Figure 3.4: Two major scoring-matrices: PAM250 (left) and BLOSUM62 (right). The amino acids of PAM250 are ordered according to their biochemical properties – related amino acids are adjacent.

metric, implying identical scores for both directions of residue substitutions. In addition to this, occasionally non-symmetric matrices are used (cf. e.g. [Lin01]). On the left-hand side the widely used PAM250 matrix [Day78] is shown for evolutionary motivated scoring matrices. It is one member of the family of PAM matrices. Generally, PAM is an acronym for "point accepted mutations" or "percent accepted mutations" and designates a measure for evolutionary divergence (distance) between two amino acid sequences. The general definition of PAM is as follows (cf. [Mer03, p.112]):

> "Two sequences $\vec{s}_1$ and $\vec{s}_2$ differ by one PAM unit, if $\vec{s}_2$ developed from $\vec{s}_1$ due to a sequence of accepted point mutations whereas per 100 residues one point mutation occurred on average."

PAMn matrices are created by comparing protein sequences diverging by n PAM units. The substitution scores are adjusted according to the expected frequencies of the appropriate residue change which is measured in various experiments. In figure 3.5 the global alignment of the two exemplary sequences given above using the PAM250 substitution matrix is illustrated. Obviously, the alignment has slightly changed at its end resulting in a global score of +3.

At the right-hand side of figure 3.4, the BLOSUM62 (block substitution) matrix as one prominent member of the second general category of scoring matrices is shown [Hen92]. Such matrices are created by counting the substitution frequencies of all amino acids when analyzing sequences of related protein domains. For the creation of the BLOSUM62 matrix Steven and Jorja Henikoff analyzed blocks (conserved parts of protein families extracted from multiple alignments without gaps) contained in the BLOCKS database [Hen99, Hen00]. By means of a heuristic procedure of block analysis, the scores are extracted using the following definition [Mer03, p.114]:

"Let $f(a_i)$ be the frequency of a residue a_i occurring at all positions within blocks of the BLOCKS database. Let $f(a_i, a_j)$ be the frequency of the column-wise occurrences of pairs a_i, a_j. Then the score S_{a_i,a_j} can be defined as:

$$S_{a_i,a_j} = \frac{f(a_i, a_j)}{f(a_i)f(a_j)}."$$

A common refinement of the extraction process is the restriction of the analysis to sequences with similarities less than or equal to a certain percentage. For the BLOSUM62 matrix this implies that all sequences analyzed for estimating the substitution scores must not have similarities larger than 62 percent.

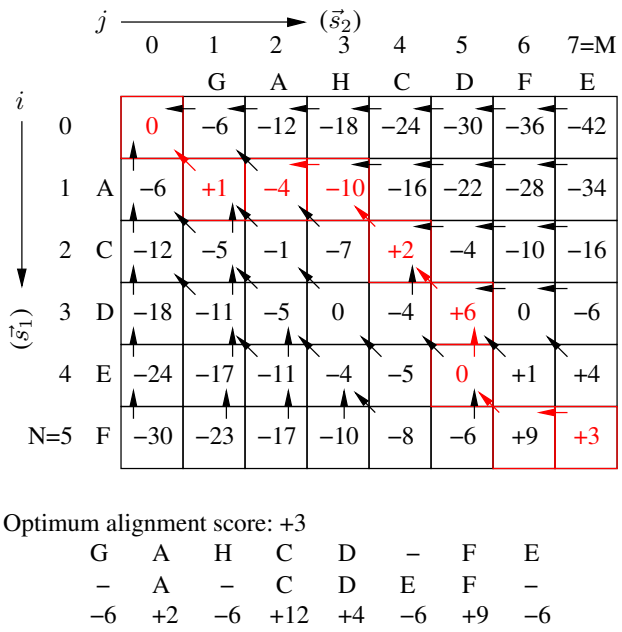


Figure 3.5: Global sequence alignment by applying the Dynamic Programming technique using the PAM250 scoring matrix and -6 for deletion/insertion.

In correspondence with the biological relevance, substitution matrices were studied in detail in recent years. From a more theoretical point of view, the underlying statistics of scoring schemes is well defined and formulated in terms of information theory (for details cf. e.g. [Alt91, Hen96b]).

Gap penalties

In addition to the scoring of substituting and matching residues of two sequences to be aligned, the general Dynamic Programming approach contains penalties for deleting and inserting symbols – so-called gap penalties γ. In the explanations and examples given so far, fixed penalties were assumed. For the alignment of the two artificial protein sequences $\vec{s}_1 =$ ACDEF and $\vec{s}_2 =$ GAHCDFE, the gap penalties were adjusted to a value of -6. Due to the trivial alignment task all gaps had the length of one. Principally, gaps span arbitrary

numbers of residues within a particular alignment. Thus, the mathematically correct definition of gap penalties is a cost function $\gamma(g)$ of the gap length g. Note that in order to keep consistency, the gap penalties for single insertions/deletions will still be called γ and the correct definition of the cost function depending on the gap length g will similarly be defined as $\gamma(g)$.

Applying constant gap penalties as in the examples given above implies a linear cost function for gaps of length g:

$$\gamma(g) = g\gamma.$$

Usually, i.e. for alignments of sequences containing several hundreds of residues, an alternative definition of the gap costs as an affine function is used:

$$\gamma(g) = \gamma - (g-1)e,$$

where γ is called the gap-open penalty and e is called the gap-extension penalty. Depending on the expectations for the characteristics of gaps in an alignment, both penalties are set individually taking into account probabilities for gaps and their lengths. Usually, e is set to something less than the gap-open penalty γ, resulting in smaller costs for long insertions and deletions than when using linear penalty functions. The reason for this is that the "biological" probability for single long gaps in protein sequence alignments is higher than for many small gaps.

Usually, gap penalties γ are adjusted independent of the residues the gap contains. Additionally, specialized residue specific penalties can be used if certain expectations about gap characteristics are available (e.g. due to prior knowledge).

Alignment Scores

Both sequences used so far for explaining pairwise alignment approaches do not originate from any real protein – they are artificial examples. Nevertheless, all algorithms delivered solutions to the general alignment problem by revealing the optimal sequences of edit operations resulting in alignment scores. In the next step, by means of these scores a classification decision needs to be taken which is biologically reasonable giving evidence for homology.

Log-odd Scores: Formally, the biologically driven decisions regarding homology via alignment scores can be performed using a well defined statistical framework by comparing two hypotheses. By means of the null hypothesis two sequences are assumed to be non-homologue which implies a random alignment R. The probability of a random alignment of two sequences $\vec{s}_1$ and $\vec{s}_2$ is just the product of the probabilities $p(\cdot)$ for the occurrences of each amino acid s_{1_i} and s_{2_i}:

$$P(\vec{s}_1, \vec{s}_2|R) = \prod_{i=1}^{N} p(s_{1_i}) \prod_{i=1}^{N} p(s_{2_i}). \tag{3.2}$$

Contrary to this, in the match hypothesis, aligned pairs of residues occur with a joint probability $p(s_{1_i}, s_{2_i})$ which can be understood as the probability that both residues s_{1_i} and s_{2_i}

were derived independently from their common (unknown) ancestor. The probability of this match alignment M can be formulated as follows:

$$P(\vec{s}_1, \vec{s}_2|M) = \prod_{i=1}^{N} p(s_{1_i}, s_{2_i}). \tag{3.3}$$

The homology decision can now be based on the ratio of both likelihoods which is called the *odds ratio*:

$$\frac{P(\vec{s}_1, \vec{s}_2|M)}{P(\vec{s}_1, \vec{s}_2|R)} = \frac{\prod_{i=1}^{N} p(s_{1_i}, s_{2_i})}{\prod_{i=1}^{N} p(s_{1_i}) \prod_{i=1}^{N} p(s_{2_i})} = \prod_{i=1}^{N} \frac{p(s_{1_i}, s_{2_i})}{p(s_{1_i})p(s_{2_i})}.$$

In order to obtain an additive scoring scheme (which is easier to handle) the logarithm is taken resulting in the *log-odds ratio* $S(\vec{s}_1, \vec{s}_2)$, a sum of individual scores $S(s_{1_i}, s_{2_i})$:

$$S(\vec{s}_1, \vec{s}_2) = \sum_{i=i}^{N} S(s_{1_i}, s_{2_i}) \tag{3.4}$$

$$S(s_{1_i}, s_{2_i}) = \log \frac{p(s_{1_i}, s_{2_i})}{p(s_{1_i})p(s_{2_i})}. \tag{3.5}$$

In fact, scoring schemes like BLOSUM or PAM are usually based on log-odds ratios where biological expertise is considered for the adjustments of particular probabilities of the amino acids.

Extended Bayesian Alignment Scores: The general idea of log-odds scoring as defined in equations 3.4 and 3.5 is the comparison of two hypotheses which generally means the comparison of two models: the random model R and the match model M. Basically, the posterior probability $P(M|\vec{s}_1, \vec{s}_2)$ that the match model M is correct, is required for the assessment of sequential relationships. By means of Bayes' rule, the prior probabilities $P(M)$ and $P(R) = 1 - P(M)$, and equations 3.4 and 3.5, this posterior probability can be derived as follows [Dur98, p.36 f]:

$$\begin{aligned} P(M|\vec{s}_1, \vec{s}_2) &= \frac{P(\vec{s}_1, \vec{s}_2|M)P(M)}{P(\vec{s}_1, \vec{s}_2)} \\ &= \frac{P(\vec{s}_1, \vec{s}_2|M)P(M)}{P(\vec{s}_1, \vec{s}_2|M)P(M) + P(\vec{s}_1, \vec{s}_2|R)P(R)} \\ &= \frac{\frac{P(\vec{s}_1, \vec{s}_2|M)P(M)}{P(\vec{s}_1, \vec{s}_2|R)P(R)}}{1 + \frac{P(\vec{s}_1, \vec{s}_2|M)P(M)}{P(\vec{s}_1, \vec{s}_2|R)P(R)}}. \end{aligned}$$

When using the following definitions:

$$S' = S + \log\left(\frac{P(M)}{P(R)}\right) \quad \text{where} \quad S = \log\left(\frac{P(\vec{s}_1, \vec{s}_2|M)}{P(\vec{s}_1, \vec{s}_2|R)}\right), \tag{3.6}$$

then the posterior probability that the match model is correct, results in

$$P(M|\vec{s}_1, \vec{s}_2) = \sigma(S'), \quad \sigma(x) = \frac{e^x}{1+e^x}.$$

Equation 3.6 implies adding the prior log-odds ratio, $\log\left(\frac{P(M)}{P(R)}\right)$, to the general alignment score S, which corresponds to multiplying the likelihood ratio by the prior odds ratio. If all expressions used are converted into real probabilities (which is difficult for the scoring scheme itself), the resulting value can principally be compared to 0, indicating whether the sequences are related. Especially when searching a database, i.e. analyzing a large number of alignments for significant matches, the prior odds ratio becomes important. Here, care needs to be taken, since the larger the number of sequences compared, the larger the probability of significant hits (falsely) obtained by chance. Thus, one solution is the comparison of $P(M|\vec{s}_1, \vec{s}_2)$ with $\log(N)$ (instead of 0), where N is the number of sequences in the database.

Statistics of Alignment Scores: Considering significance of alignment scores can also be performed using a classical statistical framework. The scores delivered by an alignment approach depend on several parameters. Among others these are the length of the sequences and the characteristics of the scoring scheme itself. As stated above, for database searches the number of sequences analyzed is important for the distribution of the scores. Thus, it is reasonable to determine an expectation value E for the number of alignments actually scored with S. Depending on the scoring scheme used which implies specific values for the two constants κ and λ, this expectation value for two sequences of length N and M (where for database searches usually $N \ll M$) can be described as follows (E-value):

$$E(S) = \kappa M N e^{-\lambda S}. \tag{3.7}$$

The number of alignments producing a score $\geq S$ is determined by a Poisson distribution:

$$P(S \geq S') = 1 - e^{-E(S')}.$$

Thus, the distribution of the alignment scores can be described by an extreme value distribution (EVD):

$$F(x) = 1 - e^{e^{\frac{x-\mu}{\beta}}}.$$

If the probability of the number of sequences randomly producing the same or a larger score than the actually observed score is small, then the observation is considered significant. As an example for database search applications, an E-value of 9 is interpreted as follows: It is expected that for a database of a given size M, nine alignments will be randomly scored with the same alignment score. Thus, the smaller the E-value is, the more significant the alignment score will be.

Generally the description of the scores' distribution by means of an EVD is analytically proved only for local alignments without gaps [Kar90, Alt96], but in various simulations strong evidence for the validity of the theory for gapped local and global alignments could be given. This has been only a brief overview of alignment statistics relevant for this thesis. A more detailed explanation is given by David Mount in [Mou04].

3.1.2 Heuristic Approximations

Basic Dynamic Programming techniques as described so far guarantee the optimum alignment of two sequences containing N and M residues, respectively, either globally (Needleman-Wunsch) or locally (Smith-Waterman). The price for this is rather high since their general computational complexity is quadratical, namely $O(NM)$. For practical applications, where homologues of sequences containing chains of several hundreds of amino acids are searched for in databases consisting of thousands of protein sequences, complete search using algorithms with quadratic complexity is mostly impossible.

In these premises, heuristic techniques were developed which significantly reduce the actual computational effort while searching by approximating the basic DP algorithms. The basic goal of such methods is the efficient analysis of large amounts of sequence data while keeping the accuracy compared to exact algorithms as high as possible. Strategies for tackling the problem of high computational effort are mostly based on two basic approaches:

Indexing: Scores for matching short substrings of length $k, k \ll N, M$ (so-called k-*tuples*; for protein sequence analysis k is usually set to $O(2)$) are pre-calculated and stored in an indexed database. Due to the limited number of k-tuples (for protein sequences containing 20 different amino acids there are 20^k possible k-tuples) the calculation as well as the database-storing and -retrieval is efficiently possible. When two sequences are compared, they are first subdivided into k-tuples and the database is searched for entries containing these k-tuples in the correct order. Due to indexing, the alignment scores of these k-tuples are already calculated and in the best case the alignment problem with computational complexity of $O(NM)$ can be scaled down to a simple database querying problem whose complexity is significantly less. The fallback case of ordinary (thus computationally more expensive) alignment needs to be considered for those database entries which are not captured by the scoring index, e.g. due to inconsistencies reasoned by outdated index structures. Since present databases grow rapidly and index calculation is rather slow, this case is not that exceptional and most implementations of alignment systems are optimized for smart fallback alignments.

Pruning: Comparing a query sequence with entries of a database generally implies the limitation to alignments of sequences that are "sufficiently" similar. Thus, the calculation of the (local) alignment can be stopped (and thus database screening will be accelerated,) if the local scores drop below a threshold. In practice this implies a narrowing of the DP matrix area. In this case the complete calculation would result in an alignment score which is not significant. Using this pruning technique, the computational effort can be severely limited.

Usually, both approaches described are used in a combined manner. Two prominent examples of heuristic alignment techniques are FASTA and BLAST which are briefly outlined in the following sections.

FASTA

In 1988 William Pearson and David Lipman presented a program suite for improved biological sequence comparison [Pea88]. These tools (FASTA, FASTP, and LFASTA) can be used for analyzing both protein and DNA sequences. In the following the basic version of the heuristic alignment method for protein sequences – FASTA – is explained.

Generally, FASTA is an approximation of the Smith-Waterman algorithm for global sequence alignments. Based on a four stage approach, local high scoring alignments are created, starting from exact short sub-sequence matches, through maximal scoring ungapped extensions, up to the final identification of gapped alignments most likely to be homologous to the query sequence. Note that in the original publication of the FASTA algorithm [Pea88], the single steps were not explicitly named. Rainer Merkl and Stephan Waack introduced comprehensible names in their description of the algorithm (cf. [Mer03, pp. 123ff]), which are adopted for clarity in the explanations given here.

1. Basically, the major speedup of the alignment calculation process is gained in the first stage of the FASTA approach – *hashing*. Here, for all k-tuples of the query sequence starting at position i within the input, the starting positions j of k-tuples within particular database entries *exactly* matching are searched. These pairs (i, j) are called *hot-spots* and are very efficiently obtained by hashing techniques. In order to retrieve the ten best diagonal sequences within a (hypothetical) DP matrix the relative positions of all hot-spots are evaluated using a simple scoring scheme. The actual limitation to the first *ten* diagonals is part of the FASTA heuristic. The diagonals extracted now contain mutually supporting word matches *without* gaps serving as seeds for further processing.

2. In the second step – *scoring 1* – the ten diagonal sequences scored maximally are processed further. Within the diagonal sequences optimum local ungapped alignments are obtained using a PAM or BLOSUM scoring scheme. Here, exact word matches from the first step are extended possibly joining several seed matches. The alignments produced here are called *initial regions*.

3. In the third step – *scoring 2* – gapped alignments of joined initial regions (based on the ungapped initial regions obtained in the second step of FASTA) are tried to be created, allowing for gap costs. The resulting alignment scores ordered from one to n are called *initn*.

4. In the final phase of FASTA – *alignment* – the highest scoring candidate matches in a database search are realigned by means of the conventional Smith-Waterman algorithm. Here, the evaluation of the DP matrix is restricted to a small corridor around the initial region of step two scored with *init1* producing the final alignment and the appropriate score.

The principles of the FASTA approach can easily be summarized graphically. In figure 3.6 the four phases of the algorithm are illustrated.

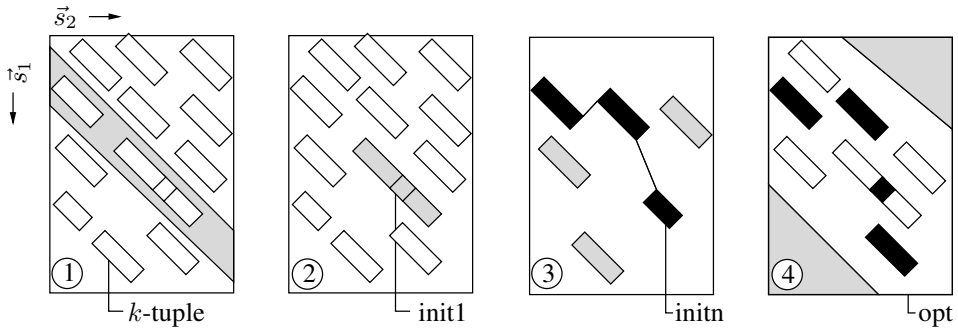


Figure 3.6: The four phases of the FASTA algorithm (adopted from [Mer03, p.125]): 1) Determination of positions of identical sub-strings (k-tuples) and scores for diagonals; 2) Determination of locally best scoring diagonals – best: init1; 3) Merging of locally optimal alignments – score: initn; 4) Final sequence alignment in small corridor of DP matrix around init1 – score: opt.

BLAST

The second important technique for approximation of the Smith-Waterman algorithm for local sequence alignments is the *Basic Local Alignment Search Tool – BLAST* of Stephen Altschul and colleagues [Alt90]. In fact, in the last decade BLAST has become the major tool for sequence analysis and most experimental evaluations in molecular biology research these days start with a BLAST run on one of the large sequence databases.[1]

The basic idea of this heuristic approximation is the extension of high-scoring matches of short sub-strings of the query sequence. Similar to the initial step of FASTA described in the previous section, BLAST starts with the localization of short sub-sequences contained in both the query sequence and the database sequences which produce significant scores. Such pairs are called *segment-pairs* or *hits*. Based on these hits, locally optimal pairs of sequences are searched containing one hit – so-called *High-Scoring Segment-Pairs (HSP)*. The boundaries of HSPs are determined in such a way that extensions or shortenings of the string would decrease the score.

Retrieving high-scoring alignments of a query sequence from a database is performed in a multi-stage approach. Initially all sub-strings consisting of w residues ($w \ll N, M$) are extracted from the query sequence. Based on these so-called w*-mers* all further steps are performed with respect to all database entries. As in the description of the FASTA algorithm, the names of the particular BLAST steps are not part of the original publication but adopted from [Mer03, p.129f].

- In the first step of BLAST – *localization of hits* – the database entry is inspected for high-scoring matches of all w-mers of the query sequence. Note, that BLAST does not require *exact* matches in the first stage, only significant scores.

- Based on the hits extracted in the first phase, in the second stage of BLAST – *identification of HSPs* – pairs of hits located on the same diagonal of a (hypothetical)

[1] Due to the overwhelming success of the tool and the resulting importance of BLAST alignments, according to the common speech of molecular biologists even the original etymology of the word 'blast' ($\rightarrow$ blow up with explosive [Swa86]) seems to be enhanced towards 'perform an alignment of the query sequence against a database' ...

Smith-Waterman matrix with a spatial context shorter than A are obtained. This distance can be measured by analyzing the differences on positions of the first symbols of two w-mers. Both hits are extended to an HSP and if the score of an HSP exceeds a threshold S_g an *extension with gaps* is initiated.

- Starting from a residue pair (so-called *seed*) the alignment is extended in both directions by means of standard DP. Here, only those cells of the DP matrix are considered where the calculated score is higher than the current maximum score minus a threshold X_g. Compared to the FASTA algorithm, the matrix area evaluated by BLAST is dynamically adjusted.

- In the final *output* stage, the resulting alignment containing gaps is returned if the calculated score (E-value) is below the threshold given.

The BLAST algorithm is likewise best summarized in a graphical manner which is shown in figure 3.7.

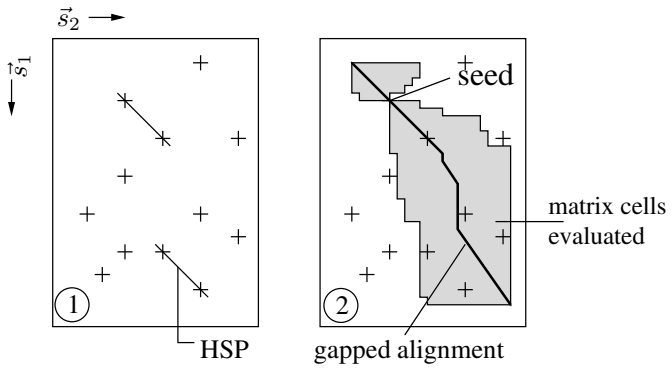


Figure 3.7: Illustration of the BLAST algorithm (adopted from [Mer03, p.130]): 1) Localization of hits (marked by '+') and extension to High-Scoring Segment-Pairs (HSP) if the distance of two hits on the same diagonal is $< A$; 2) Calculation of a gapped alignment (conjunction of HSPs across gaps) for HSP containing a score $> S_g$. The starting point for the alignment is the pair of residues designated by *seed*. Only those cells of the DP matrix are considered whose scores differ from the maximum by no more than X_g (area shaded grey in the right sketch).

3.2 Analysis of Sequence Families

By means of pairwise alignment techniques as described in the previous sections, relationships between *two* sequences can be analyzed. Although widely used (BLAST), such techniques represent only the fundamentals of general sequence analysis. Especially for biologically related sequences containing highly diverging primary structures, pairwise comparisons are often probable to fail. This means that no significant scores are obtained when using classical pairwise alignment techniques, although the particular sequences share, for example, a common ancestor.

As stated in section 2.3.1, protein sequences can be grouped into families, super-families, folds etc., with respect to certain higher-level relationships. The correct classification of a

query sequence with respect to its family affiliation[2] is of major importance for molecular biologists. Additionally, protein families contain substantially more information than single sequences since properties relevant for the *whole* group of family members are contained. Although a single member sequence might contain such properties, the probabilistic base for its detection is rather small when comparing the query sequence to single members. Due to statistical noise such weak features are usually hidden and by means of pairwise techniques they cannot be recovered. The incorporation of multiple sequences sharing such properties into the analysis process can amplify them or alleviate the statistical noise, respectively. Thus, the global analysis of complete families or the incorporation of family related information into the alignment process is promising, especially for remote homology detection tasks often performed in e.g. drug discovery.

One method for respecting family based relationships and properties is the extension of pairwise alignment techniques towards the creation of *multiple sequence alignments (MSA)*. Here, the general problem is the definition of a final alignment score for aligning the query sequence to all members of a particular family. The solution is the calculation of the sum of all scores obtained by pairwise alignments of the query sequence to all member sequences [Car88]. Unfortunately, the computational complexity for k sequences of length N is enormous making the plain technique feasible only for small numbers of sequences. Thus, numerous refinements and (heuristic) optimizations were proposed throughout the years. The most promising and widely used techniques are based on iterative, progressive determinations of the MSA where starting from an optimal global alignment of a single pair of sequences, edit operations (insertions of gaps etc.) are performed for establishing the final MSA. Prominent examples of MSA algorithms are CLUSTAL W [Tho94], DIALIGN [Mor99], and T-Coffee [Not00] which are exhaustively compared and evaluated in the work of Timo Lassmann and Erik Sonnhammer [Las02].

In the following sections an alternative and more promising approach for the analysis of sequence families is reviewed. The essentials of protein families are explicitly modeled using probabilistic approaches. Once robust family models are determined, query sequences are aligned to these models instead of to single sequences. Afterwards, the resulting alignment score is used for classification. Compared to pairwise sequence comparison techniques the alignment of query sequences to explicit stochastic models of protein families incorporates family information as for the creation of MSAs but generalized in a well defined statistical framework with much lower computational complexity. Every query sequence is aligned only once to every family model and not once to every sequence belonging to a particular protein family (as in the MSA case).

Generally, such techniques are the base for the developments performed for this thesis. Starting from position specific scoring schemes like Profiles and related refinements of standard alignment approaches, stochastic models for protein families are explained. The focus is concentrated on the technique which is currently most successfull, namely modeling protein families using Profile Hidden Markov Models.

[2]For simplicity of argumentation in the following the term 'family' is used as a generalized description of sequence relationships instead of detailed discrimination between protein families, super-families etc.

3.2.1 Profile Analysis

The analysis of multiple sequence alignments is principally very instructive for the understanding of the characteristics of sequences belonging to a particular protein family. Usually, sequences of interest are mutually aligned and the resulting MSA is analyzed by visual inspection according to the consensus, i.e. the columns of the MSA containing high-scoring conserved sequence parts. The automatic analysis of multiple sequence alignments can be performed using some kind of threshold based score analysis, i.e. the higher the scores of a contiguous part of the alignment, the higher the probability for conserved and thus putatively biologically relevant regions.

The basic problem for pairwise sequence analysis techniques when comparing distantly related data is that alignment scores will get lost within statistical noise. In order to sharpen the distinction between hits and misses of sequence database searches Michael Gribskov and colleagues proposed the concept of *Profile* analysis [Gri87]. Contrary to pairwise analysis, the information obtained from multiple alignments are integrated into the Dynamic Programming approach. Thus, pairwise techniques and MSA approaches are combined into one framework allowing more sensitive database search.

The basic idea of Profile Analysis is the creation of *position specific* scoring matrices. Here, contrary to traditional pairwise sequence comparison using one global scoring matrix as described in section 3.1.1, specific scores and gap penalties depending on the actual position within the alignment of the sequences of interest are applied. The position specific scores are obtained by analyzing the local residue frequencies of an underlying MSA. They are summarized in the so-called Profile by directly assigning the scores to the appropriate column resulting in a $N \times 20$ dimensional matrix, where N is the length of the MSA. The position specific scores themselves are extracted from the relative frequencies of all residues at a particular column of the MSA. After creating the Profile, it can be used for more sensitive sequence comparison when performing Profile based Dynamic Programming. In figure 3.8 the Profile for an exemplary multiple alignment of artificial protein sequences is illustrated.

Obviously, due to the very small number of sequences involved in the exemplary multiple alignment, the Profile is rather sparsely filled, i.e. most entries are set to zero. A relative frequency of zero implies the hard decision that the appropriate amino acid at the specified position will never be expected to occur which is rather critical. Although the problem is alleviated in real applications when significantly more sequences are mutually aligned (statistically the matrix contains less zero entries) it still remains. In order to avoid hard zero probabilities, some kind of pseudo-count methods are usually applied, thereby adjusting the probabilities of any unseen residues according to prior knowledge.

Generally, Profiles are not limited to complete MSAs. Instead, local Profiles for strongly conserved regions of MSAs are often created. The general procedure remains the same, though.

Iterative Pairwise Sequence Analysis

In section 3.1.2 two heuristic approximations for the general Dynamic Programming approach were outlined. For molecular biologists the most important tool for fast pairwise

	G ⋮ – G	A ⋮ A –	H ⋮ – H	C ⋮ C C	D ⋮ D D	– ⋮ E A	F ⋮ F E	E ⋮ – –
A	0.0	0.3	0.03	0.0	0.0	0.3	0.0	0.0
R	0.15	0.05	0.0	0.0	0.0	0.0	0.0	0.0
N	0.05	0.0	0.02	0.0	0.0	0.05	0.0	0.05
D	0.05	0.0	0.04	0.0	0.7	0.0	0.0	0.05
C	0.0	0.0	0.04	1.0	0.0	0.05	0.0	0.0
Q	0.1	0.05	0.0	0.0	0.0	0.0	0.0	0.1
E	0.0	0.0	0.1	0.0	0.0	0.2	0.33	0.0
G	0.3	0.3	0.2	0.0	0.0	0.0	0.0	0.3
H	0.05	0.0	0.2	0.0	0.0	0.2	0.0	0.2
I	0.0	0.0	0.03	0.0	0.0	0.0	0.0	0.0
L	0.0	0.0	0.03	0.0	0.0	0.1	0.0	0.1
K	0.2	0.033	0.0	0.0	0.0	0.0	0.0	0.0
M	0.0	0.033	0.05	0.0	0.0	0.0	0.0	0.0
F	0.0	0.033	0.05	0.0	0.0	0.02	0.66	0.02
P	0.0	0.0	0.0	0.0	0.0	0.02	0.0	0.01
S	0.0	0.1	0.1	0.0	0.0	0.0	0.0	0.01
T	0.0	0.0	0.04	0.0	0.05	0.02	0.0	0.01
W	0.0	0.05	0.0	0.0	0.0	0.0	0.0	0.05
Y	0.0	0.0	0.05	0.0	0.1	0.03	0.0	0.0
V	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1

multiple sequence alignment

Profile
(vectors containing relative frequencies
of amino acids at a particular column)

Figure 3.8: Profile for an exemplary multiple alignment of artificial protein sequences. For every column the relative frequencies are summarized in a substitution vector used for obtaining the resulting position specific scoring matrix (non-zero elements are marked red).

sequence comparison is the BLAST program. As stated above, if protein sequences which are to be aligned contain only minor similarities, the sensitivity of traditional pairwise DP approaches decreases significantly.

Based on the Profile approach described above one refinement of the general BLAST algorithm exists which increases its sensitivity for database searches regarding remote homologies: *Position-Specific-Iterated (PSI) BLAST* [Alt97]. Basically, it is a direct implementation of the Profile approach integrated within the powerful BLAST framework. A position-specific scoring matrix, the Profile, is automatically constructed from the output of a BLAST run. Afterwards, a modified BLAST operates on such a matrix in the place of a simple query as before. These two basic steps are multiply iterated, thereby continuously refining the database search. PSI-BLAST creates global Profiles, i.e. position specific scoring matrices for the whole sequence (and thus the complete MSA) of interest. The determination of database search hits which will be included into the MSA for creating the Profile is rather heuristical: All sequences which alignments to the query sequence score better than a predefined threshold represent candidates which are filtered further according to sequence similarity. All hits identical to the query sequence are eliminated and only one sequence is included from all candidates containing more than 98 percent similarity. Additionally, pairwise alignments are only performed between the candidates and the query sequence. Residues corresponding to gaps within the query sequence are not included. For zero frequencies of amino acids in particular columns of the Profile, pseudo-counts are calculated.

Step by step, by means of a slightly modified pairwise DP technique, implicitly a statistical representation of the protein family the query sequence belongs to is established. Using this, sensitive database searches can be performed. The consequential enhancement of such

techniques is the *explicit* modeling of protein families. In fact, explicit stochastic models of the essentials of sequence families are currently the methodology of choice for remote homology detection. Due to enhanced consideration of residualwise variation within the query sequence, broader search can be performed which is more effective than traditional pairwise comparison. In the remaining parts of this thesis the most promising approaches for explicit protein family modeling are introduced and enhanced.

3.2.2 Profile Hidden Markov Models

Based on the Profile approach of Michael Gribskov as described in the previous section, several techniques for probabilistic modeling of protein families were developed. Especially *Profile Hidden Markov Models (Profile HMMs)*, first introduced to the bioinformatics community by David Haussler and colleagues in 1993 [Hau93] and first generalized by Anders Krogh and coworkers in 1994 [Kro94a], and Pierre Baldi et al. [Bal94], play a key role in probabilistic sequence analysis. The comparison of highly diverging but related sequences using Profile HMMs for modeling the particular protein family is superior compared to traditional approaches like MSA analysis etc. Following this, large databases of sequence families were created using Profile HMMs as modeling base, e.g. Pfam containing HMM profiles for protein domains [Son98]. Furthermore, in recent years numerous bioinformatics applications were realized using (general) Hidden Markov Models. Probably as one of the first, in 1989 Gary Churchill used such stochastic models for heterogenous DNA sequences [Chu89] and later for the analysis of genome structure [Chu92]. One of the main applications for HMMs in bioinformatics is their use for gene finding (cf. e.g. [Kro94b, Bal95, Hen96a, Kul96, Bur97, Kro97, Luk98, Ped03]).

The origins of Hidden Markov Models, as one representative of graphical (Bayes) models, are general pattern recognition applications. Here, especially automatic speech recognition (cf. e.g. [Hua01] for a comprehensive overview) or the analysis of handwritten script (cf. e.g. [Wie03]) needs to be mentioned. Generally, HMMs are applicable to all signal data evolving in time. Substituting time dependency with position or location dependency, Hidden Markov Models can be used for the analysis of sequence data where the particular positions of residues are artificially interpreted as signal values at a given time-step.

The detailed explanation of (Profile) HMMs in the succeeding sections is organized as follows: First their formal definition is presented by means of general, discrete models. Furthermore, common (semi-)continuous enhancements for non-bioinformatics applications of Hidden Markov Models like automatic recognition of spoken language or handwritten script are discussed. Following this, the most important algorithms for HMM training and evaluation are presented. After their formal introduction, the use of HMMs for bioinformatics purposes is explained in detail – Profile HMMs as stochastic protein family models.

The general description of the theory of Hidden Markov Models is based on the monographs of Ernst Günter Schukat-Talamazzini [Sch95] and Gernot A. Fink [Fin03]. For the argumentation regarding the application of HMMs to probabilistic modeling of protein sequence families the standard works of Richard Durbin and colleagues [Dur98] as well as Pierre Baldi and Søren Brunak [Bal01] are used. In addition, numerous special publications, reviews, tutorials etc. have been published throughout the years. Generally, they are summarized and captured by the books given above. Exceptional cases will be marked explicitly.

Formal Definition of Hidden Markov Models

Principally, a Hidden Markov Model can be described as a generating finite automaton containing a fixed set of states, probabilistic state transitions and state-dependent emissions. The emissions can be either discrete symbols (of a finite inventory) or vectors of continuous observations.

Speaking more formally, a Hidden Markov Model represents a two-stage stochastic process. Here, the first stage describes a discrete, stationary, causal, random process by means of a sequence

$$\vec{s} = (s_1, s_2, \ldots, s_T)$$

of discrete random variables s_t whose domain is a finite set of states $\mathbf{S}$:

$$\mathbf{S} = \{S_1, S_2, \ldots, S_N\}.$$

The parameter t can be interpreted either as time-steps for signals evolving in time or as the positions of particular residues within sequences for the analysis of biological data. This stochastic process fulfills the so-called Markov property, i.e. the probabilistic selection of a particular state s_i is dependent only on a limited number of preceding states. For a first-order Markov process, this "memory" is limited to the immediate predecessor:

$$P(s_t|s_1 \ldots s_{t-1}) = P(s_t|s_{t-1}).$$

In combination with the stationary character of the process, i.e. its independence from absolute values of t, it is called a *homogeneous Markov chain*. Generally, besides first-order HMMs, higher-order HMMs can be defined, too. Since no *efficient* algorithms for both parameter training and model evaluation exist, and such models usually require enormous numbers of training samples for robust estimation, their practical relevancy is rather small. They will not be dealt with in the thesis at hand. Note that higher-order HMMs can principally be mapped to first-order models.

Due to the limitation to first-order Markov processes, the probabilistic state transitions $P(s_t = S_j|s_{t-1} = S_i)$ can be summarized in a quadratic $N \times N$ dimensional transition matrix $\mathbf{A}$:

$$\mathbf{A} = [a_{ij}] = [P(s_t = S_j|s_{t-1} = S_i)], \tag{3.8}$$

where

$$\forall i, j: \quad a_{ij} \geq 0 \qquad \text{and} \qquad \forall i: \quad \sum_{j=1}^{N} a_{ij} = 1$$

is fulfilled. The initialization of the Markov chain is described using the vector $\vec{\pi}$ of start probabilities:

$$\vec{\pi} = [\pi_i] = [P(s_1 = S_i)], \quad \text{where} \quad \sum_{i=1}^{N} \pi_i = 1. \tag{3.9}$$

It is the characteristic of HMMs that the state sequence produced by the first stage of the stochastic process cannot be observed. Instead, in the second stage, depending on the state actually selected so-called emissions are produced probabilistically. Since only these

emissions are observable while hiding the internal state sequence, the complete two-stage stochastic process is called *Hidden* Markov Model.

The elements of the sequence of emissions

$$\vec{o} = (o_1, o_2, \ldots, o_T)$$

can originate either from a finite, and discrete set of symbols

$$\mathbf{O} = \{O_1, O_2, \ldots, O_D\},$$

or they can be represented by vectors $\vec{o}_t \in \mathbb{R}^D$ of a D-dimensional vector space. These emissions are produced in dependence on the particular state selected in the first stage of the stochastic process according to the probability

$$P(o_t|o_1 \ldots o_{t-1}, s_1 \ldots s_t) = P(o_t|s_t).$$

Hidden Markov Models emitting discrete symbols are called *discrete HMMs* and similarly to the transition parameters their emission probabilities can be summarized in a $N \times D$ dimensional matrix:

$$\mathbf{B} = [b_{jk}] = [P(o_t = O_k|s_t = S_j)], \quad 1 \leq j \leq N, 1 \leq k \leq D \tag{3.10}$$

where

$$\forall j, k: \quad b_{jk} \geq 0 \qquad \text{and} \quad \forall j: \quad \sum_{k=1}^{N} b_{jk} = 1.$$

In the latter case of continuous emissions, so-called *continuous HMMs*, N-dimensional density vectors are defined:

$$\vec{B} = [b_j(\vec{o})] = [p(\vec{o}_t = \vec{o}|s_t = S_j)], \quad 1 \leq j \leq N, \vec{o} \in \mathbb{R}^D \tag{3.11}$$

where

$$\forall j: \quad b_j(\vec{o}) \geq 0 \qquad \text{and} \qquad \forall j: \quad \int_{\mathbb{R}^D} b_j(\vec{o}) d\vec{o} = 1.$$

By means of the definitions given above, a Hidden Markov Model λ is completely defined by

$$\lambda = (\vec{\pi}, \mathbf{A}, \mathbf{B}) \tag{3.12}$$

for discrete emissions and by

$$\lambda = (\vec{\pi}, \mathbf{A}, \vec{B}) \tag{3.13}$$

for continuous emissions. It describes a two-stage stochastic process where first in a homogeneous Markov chain initialized by the start probabilities $\vec{\pi}$ a state s_t is probabilistically selected only depending on its immediate predecessor. However, this state cannot be observed. Instead, depending on the actual state, in the second stage of the stochastic process an emission (either a discrete symbol or a continuous vector) is probabilistically generated. In figure 3.9 the general definition of Hidden Markov Models is illustrated by means of a discrete model containing three states.

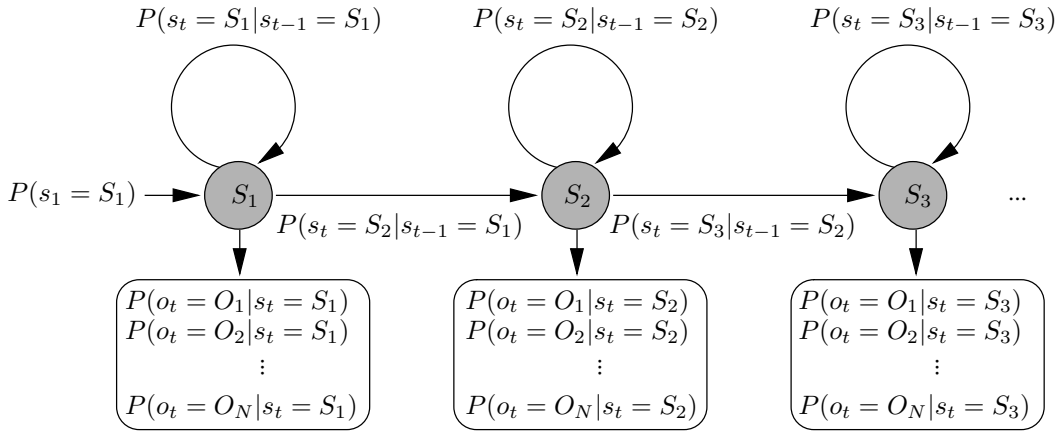


Figure 3.9: Definition of an exemplary discrete Hidden Markov Model containing three states and discrete emissions.

Modeling Aspects

Usually, for general explanations of Hidden Markov Models some standard "toy examples" are used. Richard Durbin and coworkers illustrate HMMs by means of the "occasionally dishonest casino" where based on the observations of a loaded die the most probable sequence of the two internal states 'fair' or 'loaded' is recovered [Dur98, p.54f]. Alternatively, Lawrence Rabiner created an example of weather observations (in terms of temperature values) which is analyzed for uncovering the sequence of "internal" weather states ('fine', 'sunny', 'rainy' etc.) [Rab89].

Actually, due to the two-stage architecture of HMMs, they are predominated for the classification of arbitrary data sequences with varying lengths. This is especially the case for signals evolving in time like real world speech signals or trajectories of handwritten script. Additionally, amino acid sequences belonging to certain protein families fulfill the same "constraints" for successful applicability. Due to high variability in both sequence length and actual residual composition, Hidden Markov Models are well suited for the analysis of biological data.

Although the theory of HMMs is well formulated and established, for practical applications several fundamental decisions regarding modeling need to be made, thereby significantly influencing the effectiveness of HMMs for their particular use. Among others the most important decisions are the choice of the actual model topology, and how the emissions are modeled. In the following, both modeling aspects are discussed.

Model Topology: Hidden Markov Models are one example for machine learning approaches used for pattern recognition tasks. Independent of the actual modeling subject, in a separate training step, parameters of the stochastic model are estimated. In the succeeding recognition stage these models are evaluated for the classification of new data. Depending on the quality of the trained model (which especially means its generalization ability to unknown data), the general classification problem can be solved satisfactorily.

One important decision for the design of HMMs for pattern recognition tasks is the selec-

tion of the model topology, i.e. the actual shape of the matrix $\mathbf{A}$ of state transition probabilities (cf. equation 3.8). In figure 3.10 some very common model architectures are illustrated.

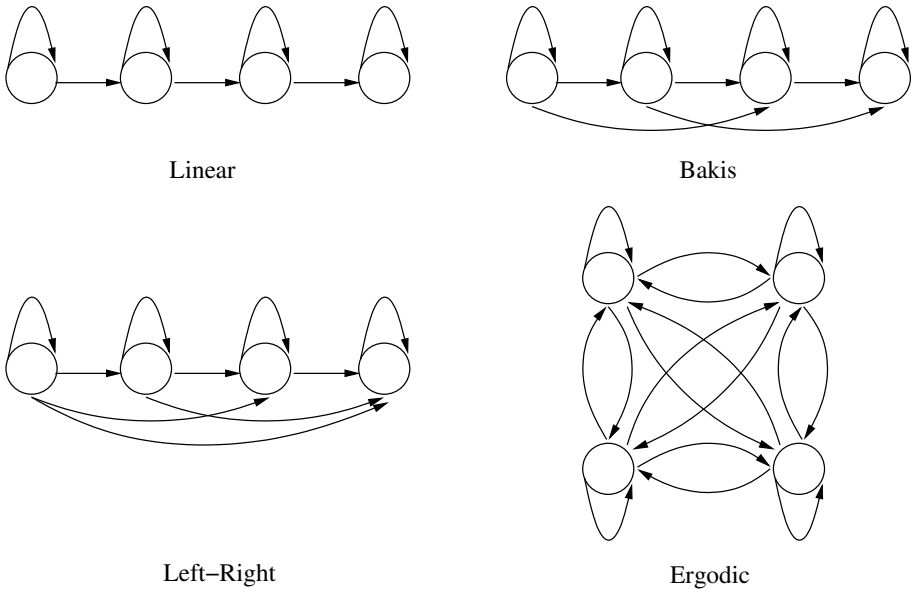


Figure 3.10: Common HMM model architectures for general pattern recognition applications differently restricting possible state transitions: Linear, Bakis, Left-Right, Ergodic (most restrictive → most flexible).

Most flexible modeling can be reached when using a fully occupied matrix corresponding to a so-called *ergodic* model architecture. Here, every state s_i is connected to every state $s_j, \forall i, j$ of the model. The major drawback of ergodic HMMs is the enormous amount of parameters. Since all parameters need to be trained using representative sample data, large amounts of training examples are required which is quite unrealistic for the majority of applications. Furthermore, the computational effort for both training and evaluation of ergodic models is considerable.

Most processes to be modeled by Hidden Markov Models contain some kind of structural information allowing for restrictions of the free transitions within the model architecture, namely the predefinition of transition probabilities to zero. The three model topologies shown in figure 3.10 besides the ergodic topology represent the most common choices for general pattern recognition applications in various non-bioinformatics domains:

- Linear models contain non-zero transition probabilities in $\mathbf{A}$ only at the diagonal and the first secondary diagonal adjacent to the right ($s_i \rightarrow \{s_i, s_{i+1}\}$), i.e. direct transitions are possible only to the particular state itself, and to its immediate neighbor.

- Bakis models additionally define non-zero transition probabilities at the second secondary diagonal adjacent to the right of the diagonal of $\mathbf{A}$ ($s_i \rightarrow \{s_i, s_{i+1}, s_{i+2}\}$), i.e. the abovementioned linear models are extended by transitions skipping a state's immediate neighbor.

- Left-Right models allow transitions to all neighbors to the right of a particular state ($\forall j \geq i : s_i \rightarrow s_j$, which corresponds to an upper triangular form of $\mathbf{A}$).

Generally, the decision for a particular model architecture usually represents a trade-off between flexibility and feasibility and needs to be taken wisely.

Especially for protein sequence analysis using HMMs, the choice of the actual model topology is rather crucial. Here, the majority of approaches is based on rather complicated model architectures which are described later in this section (cf. pages 54ff).

Type of Emissions: According to the formal definition of Hidden Markov Models, in the second stage of the stochastic process emissions are generated depending on the state selected in the preceding stage (cf. equations 3.10 and 3.11, respectively). In the simplest case, such emissions are represented by discrete symbols of a finite inventory. Especially for biological sequence analysis, presently, discrete HMMs are the methodology of choice which *seems* obvious since the input data is apparently of discrete nature (sequences of 20 discrete amino acids).

In early applications of HMMs for signal based pattern recognition tasks, discrete models were the technology of choice, too. However, modeling the emissions' distribution with discrete symbols requires (for continuous signals) a preprocessing step of mapping the data to their representatives, i.e. vector quantization. Due to distortion of the signal space which is implied by such a discretization step, major information is lost at a very early stage of signal processing. Usually this fundamental modification of the data which is putatively erroneous cannot be corrected in later steps of modeling. Thus, discrete modeling is rather critical.

In order to avoid such negative quantization effects, in continuous HMMs the emissions' distributions $b_i(\vec{o})$ are directly used. The data-driven modeling of the continuous emission space requires density functions which are parameterizable. Due to their moderate number of parameters and thus their easy mathematical treatment, usually, weighted sums of sufficient amounts of K multivariate normal densities $\mathcal{N}$ (mixture densities of Gaussians) are the methodology of choice for the representation of the emission data.[3] So, arbitrary density functions can be approximated:

$$b_j(\vec{o}) = \sum_{k=1}^{K_j} c_{jk} g_{jk}(\vec{o}) = \sum_{k=1}^{K_j} c_{jk} \mathcal{N}(\vec{o} | \vec{\mu}_{jk}, \mathbf{C}_{jk}) \tag{3.14}$$

where

$$\forall j: \quad \sum_{k=1}^{K_j} c_{jk} = 1 \qquad \text{and} \qquad \forall j,k: \quad c_{jk} \geq 0.$$

The parameters $\vec{\mu}_{jk}$ and $\mathbf{C}_{jk}$ represent the mean vector and the covariance matrix of the appropriate Gaussian. During a preceding training step the mixture components are usually approximated by applying a standard vector quantization method (like k-means [Mac67] or LBG [Lin80]) to a representative and sufficient sample set. In summary, HMMs containing a mixture density based representation of emissions are in fact *three-stage* stochastic processes:

[3] Note that the terms 'mixture' and 'mixture component' are used synonymously, whereas 'mixture density' explicitly designates a weighted sum of Gaussians.

1. For every t (time-step, position etc.), a state $s_t \in \mathbf{S}$ is selected probabilistically.
2. Depending on the actual state s_t, a mixture component $\mathcal{N}(\vec{\mu}_{jk}, \mathbf{C}_{jk})$ is selected.
3. According to the mixture selected, an emission symbol $\vec{o}_t$ is generated.

Actually, the quantization of the continuous signals is delayed up to the third stage of the stochastic process. Thus, contrary to discrete models, the complete modeling process is based on undistorted data. Due to their excellent approximation capabilities and the delayed quantization of the input data, continuous HMMs are usually superior to their discrete counterparts. Due to their outstanding performance, one of the fundamental enhancements of HMMs for protein sequences developed in this thesis is based on the substitution of state-of-the-art discrete models by continuous HMMs as described here.

In the case of continuous modeling of the emissions, a large number of parameters needs to be estimated, namely (for every state s_j) the mean vectors $\vec{\mu}_{jk}$ and covariance matrices $\mathbf{C}_{jk}$ for K_j mixture components. This requires large and representative sample sets which are often not available. Thus, it is more advantageous to use a common set of mixture components for all states. In the literature this transition from discrete to continuous modeling is referred to as *semi-continuous* HMMs developed by Xuedong Huang and colleagues [Hua89]:

$$b_j(\vec{o}) = \sum_{k=1}^{K} c_{jk} g_k(\vec{o}) = \sum_{k=1}^{K} c_{jk} \mathcal{N}(\vec{o}|\vec{\mu}_k, \mathbf{C}_k), \tag{3.15}$$

where

$$\forall j: \quad \sum_{k=1}^{K} c_{jk} = 1 \qquad \text{and} \qquad \forall j,k: \quad c_{jk} \geq 0.$$

Semi-continuous HMMs are mostly interpreted as discrete HMMs containing an integrated "soft" vector quantization where the mixture coefficients c_{jk} represent the emission probabilities of the discrete model which are weighted by means of the density values $g_k(\vec{o})$. In figure 3.11 the non-discrete emission modeling is summarized by means of a comparison between continuous (left) and semi-continuous (right) HMMs.

Algorithms

The theory of Hidden Markov Models has been well investigated. It is the existence of efficient and powerful algorithms for both training and evaluating the models which makes HMMs so attractive for various pattern classification tasks. In the following the most important algorithms are presented. In order to motivate the particular algorithms, first the actual application of HMMs for classification tasks needs to be discussed. For bioinformatics purposes, this includes the general determination of protein classes for single protein sequences as well as the detection of new members of protein families within sequence databases.

The base for classification using HMMs is Bayes' rule. If every pattern class ω_k is represented by a separate HMM λ_k, its posterior probability is defined as follows:

$$P(\lambda_k|\vec{o}) = \frac{P(\vec{o}|\lambda_k)P(\lambda_k)}{P(\vec{o})}. \tag{3.16}$$

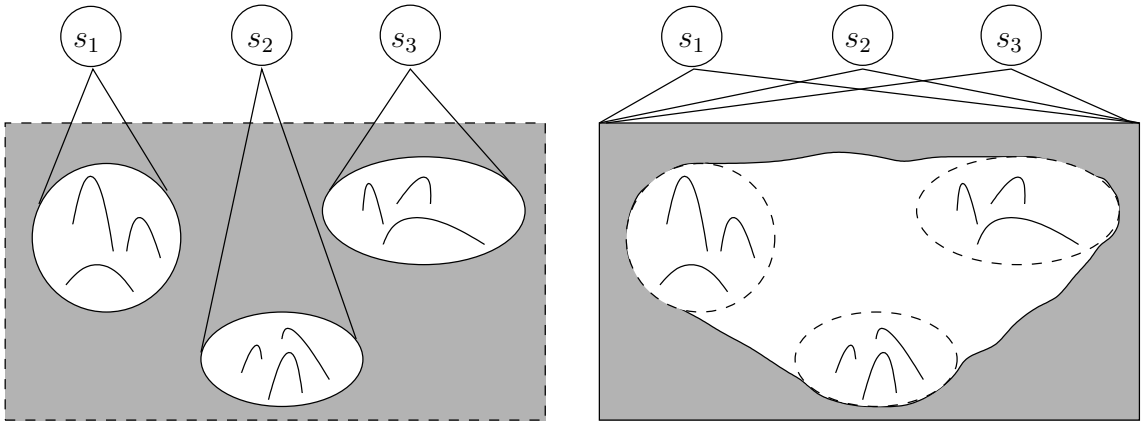


Figure 3.11: Comparison of non-discrete HMMs: Continuous emission modeling using state-specific mixture densities (left) vs. semi-continuous modeling sharing a common set of mixture components (right) – adopted from [Sch95, p.144].

The final decision regarding ω_k is taken according to the HMM λ_k delivering the maximum posterior probability $P(\lambda_k|\vec{o})$. Since the denominator of equation 3.16 is not important for maximization, it is actually ignored resulting in the following decision rule:

$$\lambda^* = \underset{\lambda_k}{\operatorname{argmax}}\, P(\vec{o}|\lambda_k)P(\lambda_k). \tag{3.17}$$

If no information about the prior probability $P(\lambda_k)$ is available, a uniform distribution is assumed. Thus, the final classification is exclusively dependent on the *generation probability* $P(\vec{o}|\lambda_k)$ which implies a Maximum-Likelihood classifier.

Based on this definition of the classification problem, in the following the particular algorithms for model evaluation and training are explained. The discussion is structured according to the argumentation of the three fundamental problems of Hidden Markov Models which is very common in the literature:

Evaluation: Based on the classification rule given above, here, the general probability of an HMM for generating the sequence of observation symbols is addressed.

Decoding: Here, the internal state sequence selected during generation of the emissions is tried to be uncovered.

Training: Finally, the estimation of optimal model parameters for the description of patterns assigned to a particular class is discussed.

Estimation of the Generation Probability – The Evaluation Problem: If HMMs are applied to pure classification tasks, i.e. every pattern class is modeled using a specialized HMM and the general probability of belonging to a particular pattern class ω_k needs to be estimated for query sequences, the so-called generation probability is determined. Here, the actual state sequence selected during generation is not important and thus not considered.

The trivial solution estimates the generation probability in a "brute-force" manner. Here, $P(\vec{o}|\lambda)$ is calculated by means of a summation over all possibilities of creating the sequence of observations $\vec{o}$ while selecting a particular state path $\vec{s}$:

$$\begin{aligned} P(\vec{o}|\lambda) &= \sum_{\vec{s}} P(\vec{o}, \vec{s}\,|\lambda) \\ &= \sum_{\vec{s}} P(\vec{o}|\vec{s}\,\lambda) P(\vec{s}\,|\lambda) \\ &= \sum_{\vec{s}} \prod_{t=1}^{T} a_{s_{t-1},s_t} b_{s_t}(o_t), \end{aligned} \tag{3.18}$$

where $a_{0i} = \pi_i$. Because of the computational complexity of $O(TN^T)$ this procedure is not feasible for most practical applications.

Due to the limited memory of first-order Markov processes, an efficient algorithm for the estimation of the generation probability with linear complexity (according to the length of the observation sequence) can be formulated by means of classic Dynamic Programming as described earlier. Therefore, specialized auxiliary variables can be defined, namely so-called *Forward-variables*

$$\alpha_t(j) = P(\vec{o}_1, \ldots, \vec{o}_t, s_t = S_j|\lambda), \tag{3.19}$$

capturing the probability that the partial sequence $\vec{o}_1, \ldots, \vec{o}_t$ could be observed when selecting the appropriate state $s_t = S_j$, and *Backward-variables*

$$\beta_t(j) = P(\vec{o}_{t+1}, \ldots, \vec{o}_T|s_t = S_j, \lambda), \tag{3.20}$$

analogously capturing the probability that the sequence $\vec{o}_{t+1}, \ldots, \vec{o}_T$ will be observed starting from $t+1$ if the current state $s_t = S_j$. Thus, the overall generation probability can be defined as follows:

$$\forall t: \quad P(\vec{o}|\lambda) = \sum_{i=1}^{N} \alpha_t(i)\beta_t(i), \tag{3.21}$$

where N designates the number of states. Since every state is only dependent on its immediate predecessor, the definition of the auxiliary variables can be given recursively resulting in two equivalent algorithms efficiently evaluating Hidden Markov Models by estimating $P(\vec{o}|\lambda)$. Both algorithms, the *Forward-*, and the *Backward-algorithm*, are summarized in figure 3.12.

Uncovering the most probable state path – Viterbi Decoding: Hidden Markov Models are the methodology of choice for pattern recognition tasks dealing with data containing substantial variance. Here, basically some kind of "real" internal states are assumed to produce the data observed. In the last section, the estimation of the general probability of an HMM generating the sequence observed was discussed. The (hypothetical) "origin" of the processed data, i.e. the internal HMM state sequence, was completely neglected.

In addition to this, for numerous applications the internal state sequence actually selected for generating the observed data is of major importance. The internal structure of the data

1. Initialization:	
$\alpha_1(i) = \pi_i b_i(\vec{o}_1)$	$\beta_T(j) = 1$
2. Recursion:	
$\forall t : t = 1, \ldots, T-1$	$\forall t : t = T-1, \ldots, 1$
$\alpha_{t+1}(j) = \sum_{i=1}^{N} \{\alpha_t(i) a_{ij}\} b_j(\vec{o}_{t+1})$	$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(\vec{o}_{t+1}) \beta_{t+1}(j)$
3. Termination:	
$P(\vec{o}\|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$	$P(\vec{o}\|\lambda) = \sum_{j=1}^{N} \pi_j b_j(\vec{o}_1) \beta_1(j)$

Figure 3.12: Forward- (left), and Backward-algorithm (right) for efficiently estimating the generation probability $P(\vec{o}|\lambda)$ of HMMs; both algorithms are equivalent and differ only in the actual direction of the recursion.

generation process can give valuable hints for the overall classification task. HMMs became so attractive because, due to uncovering their hidden state sequence for unknown data both the classification- and the segmentation- task can be solved in one *integrated* step. Especially for continuous speech recognition, and for database search using protein family HMMs, this feature is very important because classifying unsegmented data usually results in some kind of a "chicken and egg dilemma": If the correct segmentation of the data, i.e. the correct determination of start- and end-points of units to be classified (words, genes, proteins etc.), is not available, the classification process itself is very probable to fail. Unfortunately, for correct segmentation the classification problem needs to be solved before. The internal state sequence of HMMs represents the actual segmentation of the data.

Due to statistic modeling, only the *most probable* internal state sequence can be obtained. Because of the discrete set of states, the search space for decoding has a graph structure. Uncovering the most probable path through this graph implies uncovering the most probable internal state sequence which is referred to as *decoding task*. Formally, the goal of the decoding task is the determination of the state sequence $\vec{s}^*$ maximizing the posterior probability

$$P(\vec{s}|\vec{o}, \lambda) = \frac{P(\vec{o}, \vec{s}|\lambda)}{P(\vec{o}|\lambda)}$$

for a given HMM λ and an observation sequence $\vec{o}$. Again, the maximization is independent of the denominator. Thus, the most probable path is defined as:

$$P(\vec{o}, \vec{s}^*|\lambda) = \max_{\vec{s} \in S^T} P(\vec{o}, \vec{s}|\lambda) = P^*(\vec{o}|\lambda). \tag{3.22}$$

Generally, the decoding task can be solved efficiently by means of Dynamic Programming techniques. The principle procedure is similar to the calculation of forward probabilities (cf. figure 3.12). Contrary to this, the summation is replaced by maximization. Thus, the forward probability $\alpha_t(j)$ is replaced by

$$\delta_t(j) = \max_{s_1 \ldots s_{t-1}} \{P(\vec{o}_1, \ldots, \vec{o}_t, s_1 \ldots s_t|\lambda) | s_t = S_j\} \tag{3.23}$$

designating the maximum probability for generating the partial observation sequence $\vec{o}_1, \ldots, \vec{o}_t$ while selecting the most probable partial path and $s_t = S_j$. Simultaneously, a traceback matrix $\boldsymbol{\psi} = [\psi_t(j)]$ is created for the extraction of the most probable global state sequence which can be performed *after* the last observation. For classification purposes the probability $P^*(\vec{o}|\lambda) = P(\vec{o}, \vec{s}^*|\lambda)$ for generating the observation sequence $\vec{o}$ while selecting the most probable state sequence $\vec{s}^*$ is used analogously to the generation probability as defined before (cf. figure 3.12). However, both values usually differ because (partial) paths which are alternative to the Viterbi path might also contribute to the final generation probability but they are considered in the Forward-, and Backward algorithms only. Since these differences are only small and the fraction of the most probable path significantly dominates the general generation probability, in practical applications the classification accuracy does not suffer from these minor differences.

Usually, this efficient procedure for uncovering the most probable internal state sequence is referred to as *Viterbi-algorithm* according to its inventor [Vit67]. In figure 3.13 the complete algorithm is summarized.

1. Initialization: $t = 1, \quad \forall j = 1, \ldots, N:$

$$\delta_1(j) = \pi_j b_j(\vec{o}_1) \qquad \psi_1(j) = 0$$

2. Recursion: $\forall t = 1, \ldots, T-1, \quad j = 1, \ldots, N:$

$$\delta_{t+1}(j) = \max_i\{\delta_t(i)a_{ij}\}b_j(\vec{o}_{t+1}) \qquad \psi_{t+1}(j) = \operatorname*{argmax}_i\{\delta_t(i)a_{ij}\}$$

3. Termination:

$$P^*(\vec{o}|\lambda) = P(\vec{o}, \vec{s}^*|\lambda) = \max_i\ \delta_T(i) \qquad \vec{s}_T^* = \operatorname*{argmax}_i\ \delta_T(i)$$

4. Traceback: $\forall t = T-1, \ldots, 1:$

$$\vec{s}_t^* = \psi_{t+1}(\vec{s}_{t+1}^*)$$

Figure 3.13: Viterbi-algorithm for efficiently estimating the most probable internal state sequence of a Hidden Markov Model for generating the observation sequence. During recursion a traceback matrix is created which is evaluated for the actual extraction of the most probable path after the final observation.

Parameter Estimation – Baum-Welch Training: In the previous sections, two general methods for evaluating HMMs were discussed. Both of them implicitly assumed well established models optimized for the best possible representation of the statistical properties of data belonging to particular classes ω (words, protein families etc.). Since no analytical solution for optimally adjusting the parameters of an HMM is known according to the pattern class it represents given a set of sample data, these parameters are usually estimated by means of some kind of guided training procedures. The guidance of the parameter estimation process is mostly focused on the choice of the proper model topology (cf. figure 3.10) according to the assumed internal structure of the data examined. Based on this expert-made selection of the model architecture, which might be optimized in succeeding training steps,

the model parameters, namely the transition and the emission probabilities, are estimated iteratively.

In the following, the most prominent training procedure is briefly outlined. In addition to this, the application of alternative machine learning approaches to the parameter estimation task are reported in the literature. Among others, simulated annealing techniques for gradient descent approaches [Edd95a], and combinations of vector quantization and decoding techniques resulting in the so-called *segmental k-means* training [Jua90] were proposed.

Formally the task of parameter estimation can be defined as follows: Given an observation sequence $\vec{o}$, the model λ^* needs to be determined maximizing the generation probability $P(\vec{o}|\lambda^*)$:

$$P(\vec{o}|\lambda^*) = \max_{\lambda} \sum_{\vec{s} \in S^T} P(\vec{o}, \vec{s}\,|\lambda). \tag{3.24}$$

The final HMM parameters maximizing the generation probability as defined above are estimated iteratively starting from a suitable initial model λ_0. The optimization progress of all such iterative training approaches is monotone:

$$P(\vec{o}|\lambda') \geq P(\vec{o}|\lambda).$$

The actual convergence of the generation probability towards its maximum is strongly dependent on the initialization step. Thus, random initialization is usually unsuitable and mostly relative frequencies are counted on the training set serving as starting points for the training procedure. Normally, the difference of generation probabilities $\Delta P = P(\vec{o}|\lambda') - P(\vec{o}|\lambda)$ estimated using the HMMs λ and λ' of two succeeding training steps is used as stop criterion. If $\Delta P < \epsilon$ for a sufficient threshold ϵ it is very probable that the generation probability has reached its maximum. In order to keep clarity of argumentation, for the following explanations the treatment of a single observation sequence $\vec{o}$ is assumed. However, for "real world" applications usually a larger set of sequences is exploited where the parameters are estimated by averaging over the complete sample set.

The most common training procedure is the *Baum-Welch algorithm* generally representing a variant of the *Expectation-Maximization (EM) algorithm* [Dem77] which is a method for Maximum-Likelihood parameter estimation of stochastic processes with hidden variables. Here, the optimization criterion is the general generation probability of HMMs $P(\vec{o}|\lambda)$ as defined in terms of Forward-, and Backward variables $\alpha_t(j)$, and $\beta_t(j)$ in equation 3.21. The basic idea can be summarized in two steps:

1. The statistical parameters of a given HMM λ are replaced by improved estimations $\vec{\pi}', \mathbf{A}', \mathbf{B}'$ which are obtained by applying the most recent model to the training set and counting the relative frequencies.

2. For the modified model λ' the generation probability $P(\vec{o}|\lambda')$ is estimated and due to evaluation of the difference $\Delta P = P(\vec{o}|\lambda') - P(\vec{o}|\lambda)$ the decision for continuing or stopping the iteration is made.[4]

[4] Generally, both the generation probability $P(\vec{o}|\lambda)$ obtained from the Forward- or Backward-algorithm, or the state sequence specific generation probability $P^*(\vec{o}|\lambda)$ as generated by the Viterbi-algorithm can be used equivalently.

For efficiency, the Forward and Backward variables (cf. equations 3.19 and 3.20) are used. First the posterior probability $\gamma_t(i,j)$ of a transition from S_i to S_j for a given t

$$\begin{aligned}\gamma_t(i,j) = P(s_t = S_i, s_{t+1} = S_j|\vec{o},\lambda) &= \frac{P(s_t = S_i, s_{t+1} = S_j, \vec{o}|\lambda)}{P(\vec{o}|\lambda)} \\ = \frac{\alpha_t(i)a_{ij}b_j(\vec{o}_{t+1})\beta_{t+1}(j)}{P(\vec{o}|\lambda)} &= \frac{\alpha_t(i)a_{ij}b_j(\vec{o}_{t+1})\beta_{t+1}(j)}{\sum_{j=1}^{N}\alpha_T(j)},\end{aligned} \tag{3.25}$$

and the posterior probability of selecting a state S_i for a given t

$$\gamma_t(i) = P(s_t = S_i|\vec{o},\lambda) = \sum_{j=1}^{N}\gamma_t(i,j) \tag{3.26}$$

are defined. By means of these auxiliary variables, the HMM parameters can be estimated based on their preceding values (and the actual observations). The resulting definition of the start probabilities is as follows:

$$\pi_i' = P(s_1 = S_i) = \gamma_1(i) = \frac{\alpha_1(i)\beta_1(i)}{\sum_j \alpha_T(j)}, \tag{3.27}$$

whereas the transition probabilities can be estimated in the following way:

$$a_{ij}' = \frac{\sum_{t=1}^{T-1} P(s_t = S_i, s_{t+1} = S_j|\vec{o},\lambda)}{\sum_{t=1}^{T-1} P(s_t = S_i|\vec{o},\lambda)} = \frac{\sum_{t=1}^{T-1}\gamma_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}. \tag{3.28}$$

For discrete Hidden Markov Models, the emission parameters are estimated by:

$$b_{jk}' = \frac{\sum_{t=1}^{T} P(s_t = S_j, o_t = O_k|\vec{o},\lambda)}{\sum_{t=1}^{T} P(s_t = S_j|\vec{o},\lambda)} = \frac{\sum_{t:o_t=O_k} P(s_t = S_j|\vec{o},\lambda)}{\sum_{t=1}^{T} P(s_t = S_j|\vec{o},\lambda)} = \frac{\sum_{t:o_t=O_k}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}. \tag{3.29}$$

Contrary to this, for continuous HMMs, where the emissions are modeled by mixture densities, both mean vectors $\vec{\mu}_{jk}$ and covariance matrices $\mathbf{C}_{jk}$ need to be estimated. As previously mentioned (cf. page 47), the weights of the mixture components can be interpreted as the output of a discrete HMM. Thus, estimations for improved weights c_{jk} can be given in a similar form as described above. Therefore, the probability $\xi_t(j,k)$ for selecting the k-th mixture component ($M_t = k$) for a given t and state S_j for the creation of a particular emission o_t is defined as

$$\xi_t(j,k) = P(s_t = S_j, M_t = k|\vec{o},\lambda) = \frac{\sum_{i=1}^{N}\alpha_{t-1}(i)a_{ij}c_{jk}g_{jk}(o_t)\beta_t(j)}{P(\vec{o}|\lambda)}. \tag{3.30}$$

By means of this further auxiliary variable, the mixture weights can be estimated as follows:

$$c'_{jk} = \frac{\sum_{t=1}^{T} P(s_t = S_j, M_t = k|\vec{o}, \lambda)}{\sum_{t=1}^{T} P(s_t = S_j|\vec{o}, \lambda)} = \frac{\sum_{t=1}^{T} \xi_t(j,k)}{\sum_{t=1}^{T} \gamma_t(j)}. \tag{3.31}$$

For the estimation of both mean vectors and covariance matrices, the observations $\vec{o}$ are taken into account with probability $\xi_t(j,k)$:

$$\vec{\mu}'_{jk} = \frac{\sum_{t=1}^{T} \xi_t(j,k)\vec{o}_t}{\sum_{t=1}^{T} \xi_t(j,k)} \tag{3.32}$$

$$\mathbf{C}'_{jk} = \frac{\sum_{t=1}^{T} \xi_t(j,k)(\vec{o}_t - \vec{\mu}'_{jk})(\vec{o}_t - \vec{\mu}'_{jk})^T}{\sum_{t=1}^{T} \xi_t(j,k)}. \tag{3.33}$$

For the estimation of the emission parameters for semi-continuous modeling the $\xi_t(j,k)$ need to be replaced by their marginal distributions

$$\xi_t(k) = \sum_{j=1}^{N} \xi_t(j,k).$$

Profile HMMs for Sequence Families

In the last sections the general theory of Hidden Markov Models, including their mathematical definition and the most important algorithms for both estimating and evaluating them, was summarized. Originally, HMMs were proposed for processing (one-dimensional) signals evolving in time. A large variety of applications within the research field of pattern recognition based on "natural" signals can be realized using HMMs for probabilistic modeling of the appropriate "essentials" of specific pattern classes. Certainly the most prominent example is the application of HMMs to the task of automatic speech recognition.

Generally, HMMs are the methodology of choice for the classification of observation sequences with varying lengths containing moderate changes in their actual composition. This is reasoned by the fact that most sequential data can be interpreted as "signals evolving in time" due to minor abstraction from "time". Although protein sequences are obviously *not* evolving in time, their functional dependency on the residues' positions is comparable to the temporal relations of natural signals as previously mentioned. The most common application concept is the modeling of certain protein families using Hidden Markov Models. Here, all sequences belonging to a particular family (corresponding to the general pattern class ω) are interpreted as observations generated from a common origin. Due to the evolutionary variance, simple template matching approaches fail for sequence comparison. Contrary to this, protein family specific HMMs cover the complete statistical variance, i.e. evolutionary

divergences, as MSAs in one global model. If sufficient training samples are available for the creation of Profile HMMs, i.e. if the model parameters can be established statistically robust, Profile HMMs for protein families are superior in their classification accuracy.

Basically, Profile HMMs represent an enhancement of the Profile concept discussed earlier. Their conceptual linkage can be described as follows. Besides the probabilistic generation of residues at every column of an MSA, here, further structural information regarding the protein family covered by the MSA is incorporated in a stochastic model. According to its appropriate predecessor, a column of an MSA is probabilistically "activated", i.e. the probability distributions of the emission symbols assigned to every column of an MSA are evaluated depending on the actual column and its predecessor.

In the following, several practical aspects of Profile HMM applications for protein sequence comparison tasks are discussed. First, their actual use for sequence analysis is explained before certain modeling aspects relevant for successful applicability of these models are outlined. By means of these explanations it will become clear how Profile HMMs are used for statistically modeling the essentials of protein families and how they are used for the classification (alignment) of unknown sequences to these models. The fundamental difference to pairwise sequence comparison is the alignment of query sequences to models instead of to single sequences.

Practical use of Profile HMMs: In order to illustrate the general concepts of applying Profile HMMs for protein sequence comparison, their use for the initially mentioned example task of drug discovery, namely target identification, is utilized (cf. section 2.4.1). Generally, two main categories of applications within the task of sequence analysis directly corresponding to the application concepts of Profile HMMs can be distinguished:

1. Target search (screening): The major impact of the paradigm shift in molecular biology research towards the application of computational methods is the change from very specific investigations focused on previous expertise to broader analysis of complete organisms. In the early stages of the drug design pipeline as illustrated in figure 2.7 on page 18, complete genomes are scanned for putative targets. Usually, this process is called *screening* and performed with respect to distinct protein families of interest (e.g. being therapeutical relevant) in a high-throughput manner, i.e. highly automated for large databases.

Speaking more technically with respect to the current context, Profile HMMs are created using initial sample sequences of the particular protein family obtained elsewhere and afterwards applied to the target search process. Here, the complete database of sequences (e.g. all proteins corresponding to the genome of interest) is separately aligned to the model and the sequences producing significant scores are interpreted as search hits belonging to the protein family the Profile HMM represents. Within the formalism of Hidden Markov Models, the scores are calculated by estimating the generation probability $P(\vec{o}|\lambda)$ (Forward-algorithm) optionally regarding the most probable state sequence $P(\vec{o}, \vec{s}^*|\lambda)$ (Viterbi-algorithm). Applying the Viterbi-algorithm additionally reveals structural details of the sequences analyzed, e.g. the correct segmentation of protein domains.

The probability of a target hit is derived from the scores created by the appropriate Profile HMMs. As usual in pairwise sequence comparison, log-odd scores are calculated comparing the likelihoods for the target model and some kind of background (or random) model. By means of statistical tests as described in section 3.1.1 the significance for a target hit or miss can be estimated. The actual selection of the background model is rather crucial and the quality of database search results using Profile HMMs strongly depends on it. Several strategies exist for the actual choice of the background model as well as for regularizations of the Profile HMMs themselves. They will be discussed later in this chapter on page 59f.

2. Sequence classification (annotation): The second concept of Profile HMM application is the classification of sequences regarding *multiple* protein families. This task directly corresponds to classical pattern recognition applications. As in automatic speech recognition a limited set of classes (i.e. protein families) exists, and a complete database needs to be annotated regarding this inventory.

Contrary to the screening task described above, here family models are used for scoring in a competitive way (putatively enhanced by a general or background model capturing all sequences not belonging to any protein family covered by the set of Profile HMMs). Usually, within the drug discovery process, sequence classification as described here is performed for target validation.

The scoring process itself is similar to the one described for target search procedures. Sequences are aligned to *all* Profile HMMs and the most likely model determined by the best score (Viterbi- or Forward-probability) determines the actual decision regarding putative family affiliation. Contrary to e.g. speech recognition applications, the classification of sequences regarding protein family memberships is presently performed sequentially, i.e. every query sequence is aligned serially to every Profile HMM.

Emission type: The fundamental question to be answered for determining the function of a protein is how its three-dimensional structure is organized. Unfortunately, so far the general problem could not be solved. However, an accurate alignment of unknown sequences to sequences whose functions are already known provides a wealth of information for further analysis. Thus, most sequence analysis approaches are based on raw amino-acid data, i.e. string processing.

Profile HMMs can be interpreted as stochastic derivatives of MSAs. In these premises, at present discrete Hidden Markov Models are used exclusively for modeling protein families. In fact, to the author's knowledge there is no literature available addressing alternative emission modeling. The discrete symbol set of Profile HMMs consists of the 20 standard amino acids as described in section 2.2.1. Thus, every state of a Profile HMM contains discrete probability distributions for these symbols.

Modeling: So far the application concepts and emission details of Profile HMMs were discussed which gave an overview of the principles of model evaluation for protein sequence analysis. The existence of proper models stochastically capturing the essentials of the particular protein families was implicitly assumed.

However, before query sequences can be aligned to protein family models, these Profile HMMs need to be established, which is a rather crucial process because the quality of the models directly influences the quality of the alignments and thus the classification performance itself. As stated above, Profile HMMs are a generalization of the Profile concept. Profile HMMs stochastically represent multiple sequence alignments. Thus, the design of the model architecture strongly follows the concepts of MSA.

The conserved parts of an MSA (consensus), i.e. those columns containing less gaps than a given threshold ϑ (usually ϑ is set to 50 or 75 percent) are taken as the base of a Profile HMM. In fact, every column of the consensus string belonging to a particular protein family corresponds to an ordinary HMM state including its (state-specific) probability distribution of the emission symbols. Initially, the emission distribution is obtained by counting the frequencies of the amino acids at the particular column of the MSA. The consensus based states are directly connected in a linear chain. Since such a chain represents the conserved parts of a sequence family that most members share and thus during alignment match in terms of DP, these states are called *Match* states. Usually, the number of Match states determines the length of the complete Profile HMM.

The consensus of an MSA and thus the chain of Match states only represents sequence parts common to the majority of family members. In order to achieve more flexibility, i.e. to generate high alignment scores for sequences belonging to a particular family but not sharing all consensus parts like remote homologues, special states are included into the Profile HMM architecture. For every Match state two specialized states are added. Together, these three states are subsumed in so-called nodes containing high connectivity between states of the same node and between different nodes for flexibility. For obvious reasons this kind of model topology (which is the base for almost all present Profile HMM approaches) is called the *three-state architecture*.

First, insertions during the alignment of a query sequence to a Profile HMM need to be managed. This is performed using *Insert* states. Since residues are actually added during insertions, Insert states are in principal of the same type as Match states. However, due to the possibility of inserting sequence parts of generally arbitrary length, these states contain self-transitions. By means of the probability of a self-transition the length of the insertion is stochastically adjusted.

Second, deletions might occur during alignment. Consequently, in the Profile HMM architecture separate *Delete* states are incorporated for every node. Contrary to Match and Insert states, these states are “silent”, i.e. no emissions are generated.

The connectivity of Profile HMMs is almost comparable to fully connected Left-Right topologies (cf. figure 3.10). This is reasoned by the fact that highly diverging sequences need to be captured by the protein family models. Contrary to e.g. speech recognition applications where small semantic units are modeled, Profile HMMs are mostly very large often consisting of several hundreds states. Due to the high connectivity rate (especially reasoned by the Delete states allowing transitions from every state to all its successors) the computational effort for model evaluation is rather high.

The general model architecture of Profile HMMs consisting of three different kinds of states is shown in figure 3.14. For every column in the consensus, a node consisting of a Match (squares), an Insert (diamonds), and a Delete (circles) state is drawn. Exemplary for Match states the emission probability distributions are shown below the states. Principally,

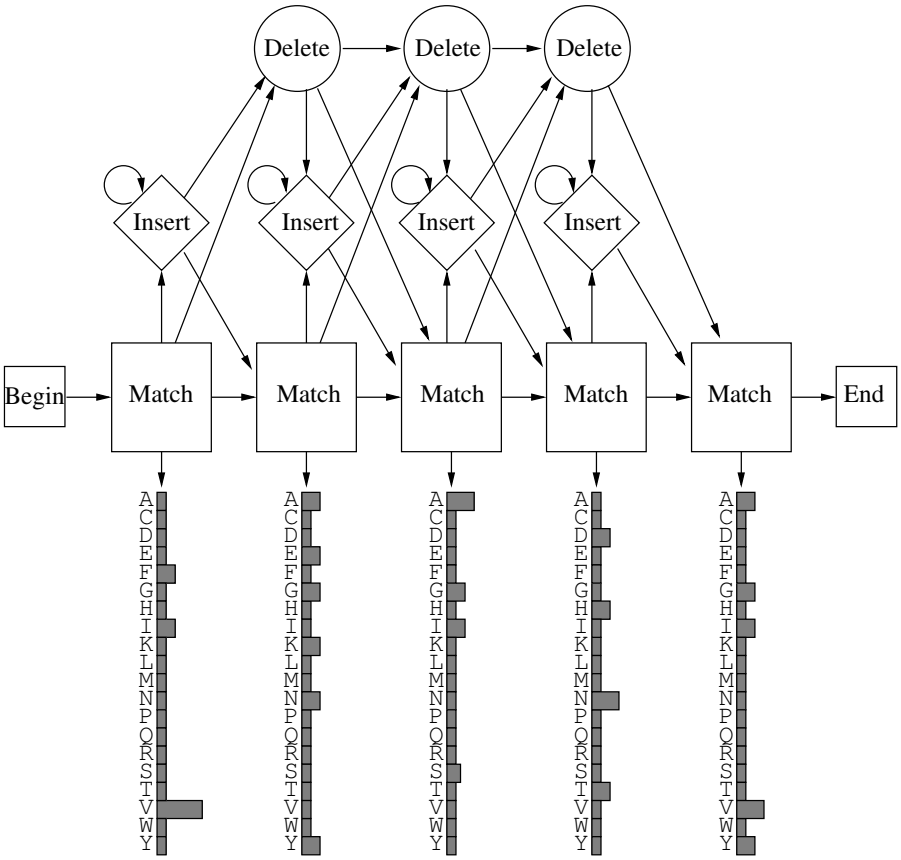


Figure 3.14: Common Profile HMM architecture based on three different kinds of states: Match (squares) for conserved columns of an MSA, Insert (diamonds) for insertions into, and Delete (circles) for deletions from consensus sequence. Delete states are silent whereas Match and Insert states produce emission symbols according to state-specific probability distributions (omitted for better readability for Insert states). For consistency to the standard literature (cf. e.g. [Dur98]), the silent Begin and End states are shown as (small) squares although non-emitting states are represented by circles.

Insert states also contain such distributions but for better readability they are omitted *here*. The rather complex connectivity of the model architecture is illustrated by arrows each representing transition probabilities. The silent Begin and End states are just for simplicity of the model management (all alignments principally start in Begin and end in End).

In addition to the principle three-state architecture, several refinements of the model topology were proposed. Generally, all of them are enhancements of the basic architecture usually introducing special states and transitions allowing local alignments of the submodel to parts of larger sequences. The most notable enhancement is the so-called Plan7 architecture, which incorporates several “garbage” states capturing all observations not explicitly modeled by the Profile HMM. In figure 3.15 this architecture is illustrated.

Most Profile HMMs are established by exploiting the result of a preceding multiple sequence alignment. Here, the model initialization is straightforward due to MSA analysis. Furthermore, modeling approaches not requiring preceding MSAs are described in the literature [Kro94a]. Based on several heuristics, the model length and thus the final architecture

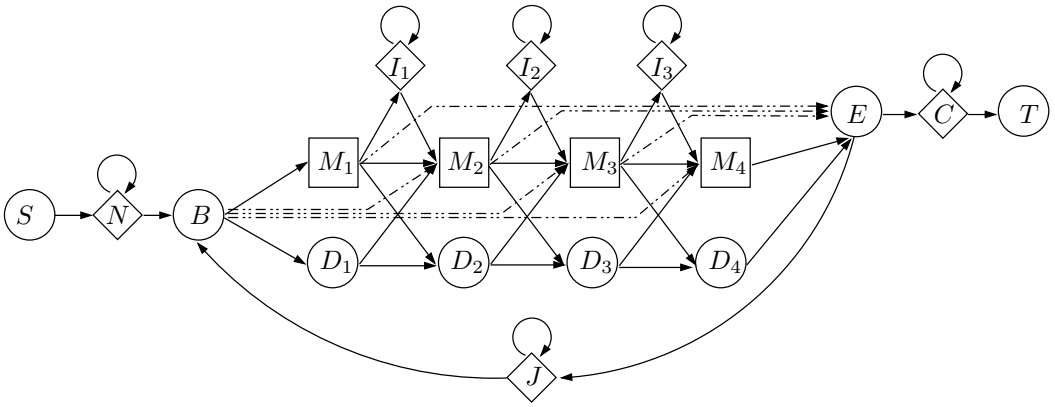


Figure 3.15: Refinement of the three-state Profile HMM architecture: Plan7. In the middle of the sketch the general Profile HMM architecture is drawn whereas at the borders flanking states and transitions for arbitrary alignment positions within larger sequences and "garbage" collection are shown.

is automatically learned in an iterative process by dynamically inserting and deleting nodes as necessary. The process is initialized by taking the first sequence of the training set as the (trivial) initial model. All following sequences are aligned to this model and depending on the actual use of the Insert and Delete states belonging to a certain node, this node is removed or a new one is created. The procedure converges rather quickly resulting in powerful Profile HMMs.

Model regularization: When applying Profile HMMs, query sequences are aligned to stochastic models established using machine learning approaches. It is the characteristic of such techniques to learn the models' parameters from representative training sets. On the one hand this procedure is very smart because models are created data-driven, i.e. without major external expertise regarding the data. In fact, this is the major reason for the superior performance of sequence analysis approaches using probabilistic models – if sufficient numbers of suitable training samples are available. Unfortunately, it is this advantage which turns into a major drawback if not enough training material is available. Without any further restrictions all model parameters, i.e. transition and emission probabilities, to which no training samples were assigned are fixed to zero which implies extremely poor generalization abilities for the resulting Profile HMM. Due to the complicated model architecture of Profile HMMs incorporating vast amounts of states, weak or even no estimates are not the exceptional case.

In order to overcome this so-called *sparse data problem*, several approaches were proposed *regularizing* Profile models. Most of them are focused on the incorporation of prior knowledge into the probabilistic framework of Hidden Markov Models by means of some kind of *pseudocounts*, i.e. adding artificial counts for the appropriate parameters. Generally, the problem exists for both transition, and emission probabilities, but the emission probabilities are more important for the actual scoring.

Kevin Karplus already compared several different regularizers for estimating distributions of amino acids in 1995 [Kar95]. The simplest way of avoiding zero probabilities is to add small, fixed, positive zero-offsets to each count of amino acids which is usually called

Laplacian's rule. Although this method is very simple, and thus seems attractive, the results are rather poor because really small probabilities are systematically overestimated. The suggested refinement of adjusting the offset values more specifically, e.g. by evaluating scoring matrices in dependency of the model parameters and the data observed, is only an option for special cases because again a lot of example data is required.

At present, the most promising method for regularizing Profile HMMs is the modeling of prior knowledge using mixtures of Dirichlet distributions [Bro93, Sjö96]. For certain alignment environments specific Dirichlet distributions are defined and the evaluation of the mixture of these Dirichlets delivers suitable pseudocounts for smoothing the amino acid probability distribution.

Since Dirichlet distributions $\mathcal{D}$ are the simplest form of parametric, multinomial discrete distributions they are a "natural choice" [Dur98, p.302] for probability parameters to use as prior distributions (cf. figure 3.16 for a more detailed motivation). A Dirichlet distribution is the distribution over the set of all probability vectors $\vec{p}$ (i.e., $p_i \geq 0$ and $\sum_i p_i = 1$). For proteins $p_i = P(\text{amino acid } i)$, and $K = 20$:

$$\mathcal{D}(\vec{p}|\vec{\alpha}) = \frac{\prod_{i=1}^{K} p_i^{\alpha_i - 1}}{Z}.$$

Here, $\vec{\alpha} = (\alpha_1, \ldots, \alpha_K)^T$ with $\alpha_i > 0$ are constant parameters specifying a single distribution. Z is a normalizing factor which can be expressed in terms of the gamma function. Further mathematical details regarding the definition of Dirichlet distributions and their statistical properties, which are beyond the scope of this thesis, are explained in e.g. [Joh72] and Dan Geiger discusses several theoretical aspects regarding Dirichlet distributions within general machine learning approaches in [Gei96].

For regularizing amino acid distributions, several sets of $\vec{\alpha}^e$ are defined usually corresponding to different types of alignment environments e. By means of these (mostly manual) choices of $\vec{\alpha}$ the prior knowledge for avoiding zero probabilities is incorporated. Given a count c_{ja} for a state j and an amino acid a the likelihood for a particular prior distribution $\mathcal{D}_e$ is estimated based on how well it fits the observed data. Both estimations are combined according to the posterior probabilities yielding the estimate of the emission parameter to be regularized [Dur98, p.117]:

$$b_j(a) = \sum_e P(e|\vec{c}_j) \frac{c_{ja} + \alpha_a^e}{\sum_{a'} (c_{ja'} + \alpha_{a'}^e)}$$

where $P(e|\vec{c}_j)$ are the posterior mixture coefficients calculated by Bayes' rule:

$$P(e|\vec{c}_j) = \frac{p_e P(\vec{c}_j|e)}{\sum_{e'} p'_e P(\vec{c}_j|e')}$$

and p_e are the prior probabilities of each mixture component $\mathcal{D}_e$. $P(\vec{c}_j|e)$ is the probability of the data according to the e-th Dirichlet mixture:

$$P(\vec{c}_j|e) = \frac{\left(\sum_a c_{ja}\right)! \Gamma\left(\sum_a c_{ja} + \alpha_a^e\right) \Gamma\left(\sum_a \alpha_a^e\right)}{\prod_a c_{ja}! \prod_a \Gamma(c_{ja} + \alpha_a^e) \prod_a \Gamma(\alpha_a^e)}$$

with $\Gamma(x)$ representing the standard gamma function.

Dirichlet mixtures can be adjusted very finely thus generating high quality multinomial distributions and it could be shown that good Profile HMMs can be created with small training sets [Sjö96].

Background model: The major difficulty in screening applications is the distinction between matches and mismatches for alignments of query sequences to a particular Profile HMM. As already introduced for the general pairwise sequence analysis approach (cf. page 31f.), the raw alignment scores are substituted by their likelihood ratio according to a proper null model, i.e. a background model which captures the random hypothesis.

Generally, the choice of a suitable background model is an issue for Profile HMMs, too. Alignment scores, which are here generation probabilities of Profile HMMs, are strongly dependent on the length of the query sequences. Thus, besides generally distinguishing model matches from random alignments, some kind of length normalization needs to be provided. Once a proper background model is found, log-odd ratios can be used for simple threshold based discrimination.

Several choices of null models were proposed for Profile HMMs. Technically, the generation probabilities are divided by the background distribution of the observation symbols as defined by the particular null model R (cf. section 3.1.1). Thus the final log-odd scores $S(\vec{o}, \lambda)$ for aligning a sequence of observations $\vec{o}$ of length T to a model λ are principally defined as follows:

$$S(\vec{o}, \lambda) = \log \frac{P(\vec{o}|\lambda)}{P(\vec{o}|R)} = \log \frac{P(\vec{o}|\lambda)}{\prod_{t=1}^{T} P(o_t)}.$$

The simplest choice of R is a uniform background distribution of all possible observation symbols, i.e. $P(o_t) = 1/K$ where K represents the number of different possible observations (20 when using plain amino acid symbols). Unfortunately, this simple choice of a background model did not prove very robust for practical applications because, independent of the actual emissions, a uniform background distribution is a rather artificial assumption.

Christian Barrett and colleagues summarized a variety of further null models in [Bar97]. Besides the flat distribution, where each amino acid is assumed equally likely as described above, they investigated the following choices of null models based on real experiments with *Globins*, *EF-hands*, and *Ferredoxins*:

- The background distribution of amino acids over all proteins,
- The amino acid frequencies of the appropriate training set,
- The average amino acid frequencies of the training set limited to the match states,
- The normalized geometric mean of match state probabilities of the amino acids,
- The amino acid frequencies of the scored sequence, and
- A complex null model based on the trained model's transitions multiplied with the geometric mean of its match states.

The most common procedure for estimating a statistical model M, i.e. its parameter vector $\vec{\theta}$, using a given sample set D, is the so-called *Maximum Likelihood (ML)* estimation. Given the data vectors of D the corresponding optimal parameter sets is obtained by maximizing the likelihood function $P(D|\vec{\theta}) = P(\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T|\vec{\theta})$ (for details regarding the actual maximization process cf. e.g. [Fin03, p.49ff]):

$$\vec{\theta}^* = \underset{\vec{\theta}}{\operatorname{argmax}}\, P(D|\vec{\theta}).$$

Especially when estimating the parameters for a multinomial distribution (like the distribution of the 20 standard amino acids) the actual ML estimation mostly implies counting relative frequencies. Here, the basic problem is the treatment of so-called zero frequencies, i.e. which values to assign to components of $\vec{\theta}$ where no corresponding data could be observed.

The simplest solution is to "trust" the data observed and assign zero to the appropriate parameters. However, it is rather unrealistic to assume hard zero probabilities only because the appropriate samples did not occur within the sample set used for model estimation. Instead, mostly certain prior knowledge regarding the model parameters exists which could be incorporated into the model estimation process.

One possible solution is the incorporation of prior knowledge using Bayes' rule:

$$P(\vec{\theta}|D) = \frac{P(D|\vec{\theta})P(\vec{\theta})}{P(D)} = \frac{P(D|\vec{\theta})P(\vec{\theta})}{\int_{\vec{\theta}} P(D|\vec{\theta})P(\vec{\theta})},$$

which implies a posterior estimation of the model parameters given D and some kind of prior knowledge regarding $\vec{\theta}$. The basic question to be answered here is, how to model this prior knowledge? The most convenient solution is the use of prior distributions which are generally similar to the distributions to be modeled – *conjugate distributions*. The distribution $P(\vec{\theta})$ (the prior knowledge) is called conjugate distribution for $P(D|\vec{\theta})$, if $P(\vec{\theta}|D)$ (the posterior distribution) and $P(\vec{\theta})$ are of the same kind, i.e. if they belong to the same family of distributions. This means, if a likelihood function is given and the appropriate conjugate distribution is known, the corresponding posterior distribution is known to be of the same kind. Thus, closed forms can directly be given and calculated which extremely simplifies the general problem of prior knowledge incorporation.

In fact, Dirichlet distributions are conjugates for multinomial distributions. Thus, if prior knowledge is modeled using a Dirichlet distribution with parameter vector $\vec{\alpha}$, the desired posterior $P(D|\vec{\theta})$ is a Dirichlet, too. It can easily be shown that the parameters of the posterior distribution are now $\vec{\alpha}' = \vec{\alpha} + \vec{\theta}$ which implies very easy model estimation.

Figure 3.16: General motivation for using Dirichlet distributions for model parameterization.

They summed-up with the conclusion that family specific null models seem to perform better than global models. Especially for remote homology detection the complex null model can improve the discrimination abilities of Profile HMMs. In a consecutive analysis, they found out that "... when scoring databases with an HMM built for [...] families, sequences that are compositionally biased towards [more rarely seen residues like cysteine] tend to receive inflated scores and become false positives [i.e. false classifications]" [Kar98]. Consequently, they propose to use the scores of the alignment of the reversed sequence as null probabilities because the reversed sequence has the same length and composition of the sequence itself which eliminates the bias mentioned above.

Motif based Models: By means of Profile HMMs usually complete protein families or superfamilies mostly based on domains are modeled. The more abstract the functional relationships of the member sequences are, the weaker are the similarities of them at the sequence level. Compared to traditional (pairwise) sequence comparison techniques the classification performance of Profile HMMs is still acceptable for such highly diverging sequences. This is reasoned by the probabilistic modeling of the essentials of the appropriate protein families.

However, in order to capture these essentials of the sequence families, rather complex model architectures allowing as much flexibility as possible are required (cf. figures 3.14 and 3.15, respectively). The price for this flexibility is really high because the parameters for $O(3N)$ states need to be estimated, where N designates the length of the consensus sequence. Generally, for robust modeling this implies large amounts of training samples. Even the use of sophisticated regularization techniques as described earlier does not solve the problem in general, it only alleviates the symptoms.

The sparse data problem for modeling robust Profile HMMs for protein families is generally well-known and principally the same as for modeling robust Hidden Markov Models for automatic speech recognition. Here, the apparent modeling base would be to establish HMMs for every word of a given lexicon. Unfortunately, especially for complex languages containing large numbers of different words, the training material is sufficient only in a few cases. Thus, almost all state-of-the-art systems for automatic speech recognition are based on HMMs for much smaller units – so-called triphones, i.e. phonemes including their surrounding (generalized) neighbor phonemes (cf. e.g. [Hua01, pp. 430ff]).

The conceptual equivalent of triphones in protein sequence analysis applications is the so-called *motif*. A motif is the smallest contiguous, conserved unit of e.g. a multiple alignment of related sequences. Several motif finding approaches were proposed in the literature (cf. e.g [Hud99] for an overview) but so far only one method was described which directly exploits the results of a motif finding algorithm for establishing HMMs.

William Grundy and colleagues developed both the *Multiple Expectation Maximization for Motif Elicitation (MEME)* algorithm for discovering motifs shared by a set of sequences using the EM-algorithm [Bai95] and the *Meta-MEME* system which establishes HMMs for the motifs obtained using MEME [Gru97]. By means of the *Motif Alignment & Search Tool (MAST)* these motif based HMMs are used for protein sequence analysis [Bai98]. Since the motifs found by MEME are highly conserved throughout most of the sequences analyzed in both length and composition, the resulting model architecture is less complex than the general Profile HMM topology. The motif based models include the chain of Match

states which is already known from Profile HMMs with transition probabilities of 1.0 between them and single Insert states at the end of each motif model allowing for residues not captured by the models. In figure 3.17 the resulting model architecture is illustrated with respect to the original Profile HMM topology whose additional states and transitions, which are skipped by the Meta-MEME models, are drawn in grey.

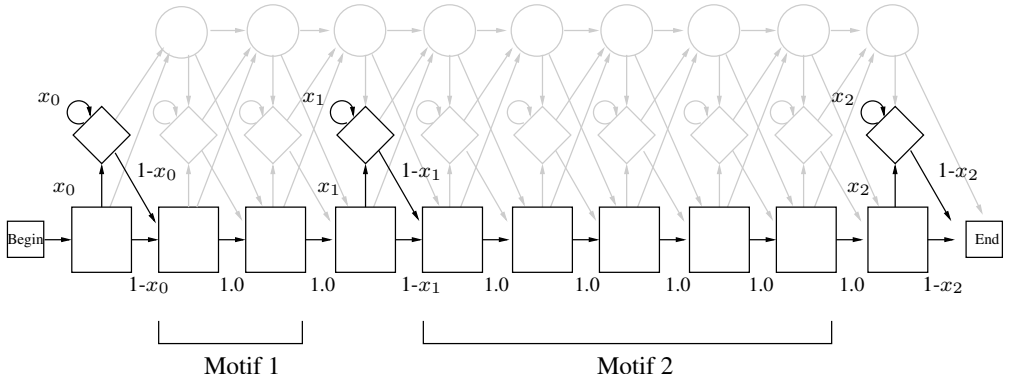


Figure 3.17: HMM topology as defined in the Meta-MEME system: Compared to the standard three-state Profile HMM architecture (grey *and* dark parts) only the darker states and transitions are defined (adopted from [Gru97]).

According to the authors of the Meta-MEME system, the concatenation of motif based HMMs instead of using conventional Profile HMMs leads to superior classification results, especially for smaller training sets. In [Gru97] they present significant improvements for the classification of *4Fe-4S ferredoxins*. However, the only motif based modeling approach using HMMs contains several heuristics and weak points which seem to prevent the good result mentioned above from generalization. Especially the isolated motif search and modeling procedures seem to be critical since the (very simple) model architecture is not optimized depending on the specifics of the appropriate protein families. Furthermore, the motif finding approach needs to be parameterized using substantial prior knowledge as the number of occurrences of single motifs and the range of their lengths etc. In summary, the present principle modeling of smaller parts is promising but several constraints let doubts arise regarding the *general* improvement of remote homology detection.

Summary: Hidden Markov Models are presently the methodology of choice for the classification of signals evolving in time containing both length and content variance. Especially for the automatic recognition of spoken language or handwritten script, HMMs are the dominating concept. The analyzed data is generally interpreted as the result of a two-stage stochastic process represented by a generating finite automaton. In the first step one state out of a finite set is probabilistically selected dependent on its immediate predecessor. By evaluating state specific probability distributions the emission (symbol) is generated.

In the last decade, a variant of Hidden Markov Models relevant for protein sequence analysis tasks has been established – Profile HMMs. Based on a generalization of the Profile approach, which assigns position-specific scoring matrices to a multiple sequence alignment, stochastic models for protein families of interest are established. The classification of a query sequence is performed by aligning it to the appropriate Profile HMM.

Presently, two major toolkits for protein sequence analysis using Profile HMMs exist: SAM [Hug96] and HMMER [Edd01]. Profile HMMs created by both systems are generally based on the classical three-state model architecture consisting of Match, Insert, and Delete states. Traditional pairwise sequence alignment techniques are outperformed by Profile HMMs for remote homology detection if sufficient numbers of training samples are available. However, the general problem of remote homology detection is also not solved at all by means of the most powerful Profile HMMs.

3.2.3 Further Probabilistic Modeling Approaches

Compared to conventional pairwise sequence analysis approaches as described in section 3.1, probabilistic approaches are currently the methodology of choice especially for remote homology detection tasks. Their superior classification performance is reasoned by the fact that by means of stochastic models more "fuzzy" search procedures become possible since uncertainty is *explicitly* integrated into the classification framework which is advantageous for highly diverging but related sequence data.

In the last sections, the most promising probabilistic models of protein sequence families were discussed – Profile Hidden Markov Models. Actually this thesis is directed to the analysis and improvement of HMM based sequence analysis approaches for remote homology detection. However, in addition to this powerful stochastic modeling technique, several alternative approaches for probabilistic sequence analysis were developed in the machine learning community.

In the following, a selection of alternative machine learning based sequence comparison approaches is presented. Here, the focus is on a general overview of applications, thus there is neither any claim for completeness nor for exhaustive explanations of all details. Instead, the interested reader is referred to the excellent monograph of Pierre Baldi and Søren Brunak [Bal01]. Although the selection presented here is certainly more or less subjective, it covers the most promising state-of-the-art alternatives to Profile HMMs.

Neural Networks

The application of artificial neural networks to the task of biological sequence comparison has a fairly long history. Beginning with applying the perceptron to the prediction of ribosome binding sites based on amino acid sequence input in 1982 [Sto82], up to present structure prediction approaches utilizing sophisticated model architectures (cf. e.g. [Wu00] for a review of current techniques and applications), they are the most prominent alternative of probabilistic models for sequence families to Profile HMMs.

Generally, neural networks can be described as parallel computational models consisting of densely interconnected adaptive processing elements called neurons. They can also be viewed as one broad class of parameterized graphical models representing a directed graph where the connection between two neurons i and j is weighted by w_{ij}. Usually, the basic units of neural networks are organized in multiple hierarchical layered architectures consisting of specific input and output layers whose neurons are visible representing the interface of the network to the outer world, and internal layers containing hidden neurons. Artificial neural networks are applied to various classification tasks as a general function approxima-

tion technique. Therefore, data is presented to the input layer and the classification result can be extracted from the membership probabilities of the N output neurons representing N classes. The membership probabilities are implicitly calculated by the neural network by means of the level of activation of all neurons. The activity of a single neuron i depends on its signal input which originates from the output of its connected neighbor neurons j, which are amplified or alleviated by the weight of the connections w_{ij}. The actual level of activation y_i is determined by the transfer function f_i of the particular neuron resulting in the following activation rule:

$$y_i = f_i(h_i) \quad \text{where} \quad h_i = \sum_{j \neq i} w_{ij} y_j.$$

Usually, the transfer function is of non-linear, sigmoid type. Common examples are the logistic function, or $\tanh$, or $\arctan$. For specialization regarding certain classification tasks of signal data, the weights between neurons are adjusted in a training step. Among other possibilities, this is mostly realized by minimizing an error function for the given training samples. Within the Bayesian framework (cf. HMMs) this corresponds to the usual procedure of model fitting and parameter estimation. Especially the existence of powerful training techniques like the widely used Backpropagation algorithm (cf. e.g. [Zel97]) is a good argument for their broad acceptance for probabilistic classification tasks.

The most important, and critical decision which needs to be taken when artificial neural networks are applied to a particular pattern classification task is the choice of a suitable representation of the data analyzed. Usually, in a preprocessing step related regions of the signal analyzed are subsumed in so-called receptive fields which are presented to the network. For the comparison of sequences with varying lengths, adjacent residues within a local context window are usually subsumed in the abovementioned preprocessing step. The length of the window is highly application specific and usually subsequent windows overlap significantly. Since only numerical values are processed during function approximation, all input data needs to be encoded in a proper numerical representation. For sequence analysis tasks, several approaches were developed:

- Amino acids are represented by 20-dimensional orthogonal vectors, which means e.g. $(1, 0, 0, \ldots, 0)^T$ for Alanine, $(0, 1, 0, 0, \ldots, 0)^T$ for Cysteine etc.
- Amino acids are grouped according to some biochemical criterion (hydrophobicity, polarity etc.) resulting in usually less than 20 new symbolic representations which are encoded using orthogonal vectors as described above.
- Based on the local context windows analyzed, some higher order statistics like the relative frequencies of certain n-mers, i.e. the term frequencies of n-gram terms, are treated as input data (cf. e.g. [Wan01]).
- Amino acids are directly mapped to numerical values by means of encoding schemes measuring certain biochemical properties based on practical expertise.

According to the facts discussed above, the typical procedure for sequence analysis using neural networks can be summarized as following:

1. Amino acid sequences are mapped to a proper numerical representation. In addition to this, certain abstraction is included since averaged values of local context windows are used instead of raw sequence data. As a general design issue, the output representation, i.e. how to obtain the classification result, needs to be defined.

2. Depending on the actual application a suitable model architecture is chosen for the desired neural network. Usually, the general characteristics of the network are fixed here (e.g. time-delay neural networks for explicit consideration of the position dependency of the residues vs. "simple" feed-forward networks) and the number of layers and neurons is determined (or learned).

3. Once the model architecture, and the input and output representations are determined, the model is trained using representative sample sets. The training method used mostly depends on the actual model type.

4. After the model building and training stages, the neural network can be used for the classification of unknown sequences regarding the protein families which the model captures. Therefore, all query sequences are converted into the numerical representation (chosen in 1) and the data is presented to the model. The activity levels of the output neurons determine the actual classification decision.

Artificial neural networks are very powerful especially for highly diverging but related sequences. Generally, by means of more or less complex model architectures almost arbitrary functions can be approximated which is advantageous for remote homology detection. The major drawback preventing general applicability to sequence analysis tasks is their substantial demand for training samples. Additionally, rather complicated model types with complex topologies are required for processing unsegmented data. In figure 3.18 the sequence analysis approach using neural networks is illustrated.

Stochastic Grammars

As described in chapter 2, protein sequences consist of residues originating from a fixed set of amino acids. They are represented as strings and sequence analysis is mostly based on some kind of string comparison. Speaking more formally, protein sequences of a particular protein family are words over the alphabet of amino acids. All *valid* words of the protein family are summarized in the "language" of protein sequences which is defined by its grammar – a compact set of rules for generating it. If the grammar of the language was known, the problem of sequence analysis would be solved because the generation rules only need to be evaluated for the query sequences regarding acceptance to the language, i.e. the protein family of interest. Unfortunately, except for trivial cases the grammar of protein families cannot be specified ab initio which complicates the sequence analysis task as already discussed.

In the last few years a generalization of the Hidden Markov Formalism has become popular for sequence classification tasks: *stochastic (context free) grammars (SCFGs)*. This is especially the case for RNA analysis. Here, the basic idea is to derive probabilistic rules for generating the language of words, i.e. RNA belonging to a certain class, or members of a particular protein family. Once the rules are obtained, they can be evaluated for query

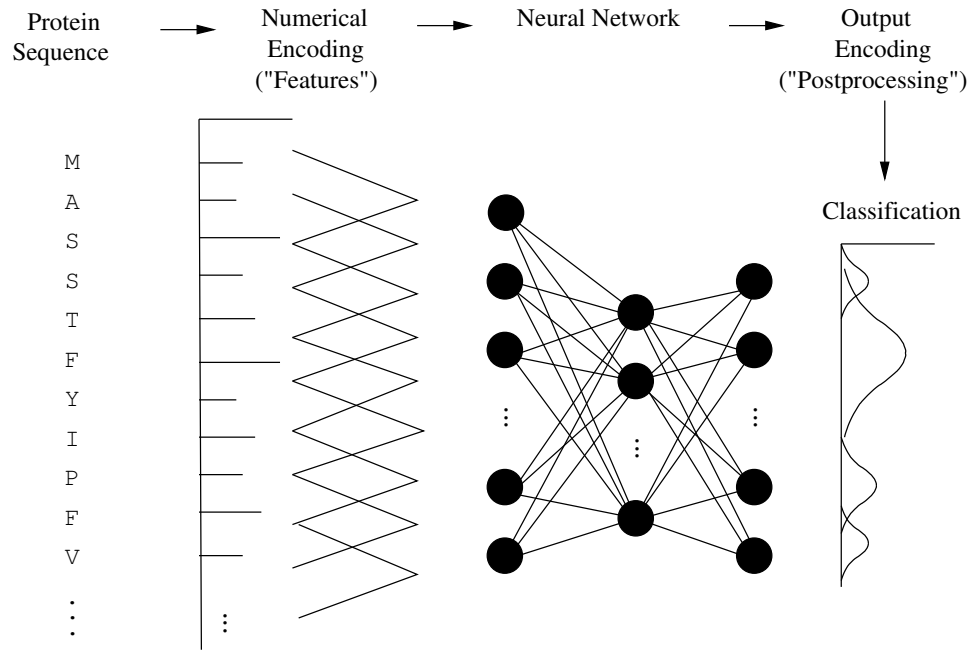


Figure 3.18: Illustration of the sequence classification approach using artificial neural networks: Based on the raw sequence data some kind of feature representation is extracted which is presented to the neural network whose output neuron activities are used for the final classification decision (from left to right).

sequences delivering a probability for accepting them as words of the language modeled. In summary, stochastic grammars are obtained by superimposing a probability structure on the set of production rules, e.g. (cf. [Bal01, p.282]):

$$\alpha \to \beta : \qquad P(\alpha \to \beta) \quad \text{with} \quad \sum_{\beta} P(\alpha \to \beta) = 1.$$

Thus, stochastic grammars are characterized by a set of parameters and can be interpreted as probabilistic generative models for the appropriate corresponding languages. Usually, the set of rules needs to be specified by experts which is a major drawback of the general method. Besides this, there are approaches where one tries to derive the set of rules from multiple alignments by creating and analyzing the assigned parse tree.

Due to their context free character, SCFGs are less restrictive than HMMs, theoretically allowing more flexible sequence comparison. This seems attractive, especially for remote homology detection, but the price for this flexibility is rather high. In order to obtain robust SCFGs for protein families, large datasets are required for deriving the generation rules. Although the same regularization mechanisms as for HMMs can be applied in principal, the problem has not been solved so far. In [Bal01] further limitations such as computational complexity are reported which currently prevent general applicability of SCFGs for *protein* sequence analysis.

Support Vector Machines

The majority of state-of-the-art sequence analysis methods including the probabilistic approaches described so far, are based on the examination and modeling of sequence *simi-*

larities. As an example Profile HMMs are optimized towards the detection of remote but still somehow similar sequences affiliated to a particular protein family. As previously mentioned (cf. pages 61ff), the discrimination between target hits and misses is of major importance in order to keep the number of false predictions as low as possible. This is especially true for screening applications like those performed for target identification in drug discovery tasks. Usually this problem is tackled by explicitly analyzing the ratio of scores obtained due to evaluation of the appropriate target model and a sophisticated background model. Consequently, this implies nothing else than explicitly analyzing the *dissimilarities* between two models.

Recently, in the machine learning community certain approaches have been developed addressing optimized discrimination between classes. Certainly the most prominent example of such techniques are the so-called *Support Vector Machines (SVMs)* introduced by Vladimir Vapnik. The monograph of Nello Christiani and John Shawe-Taylor on this topic is the basis for the following argumentation [Chr00]. For problems of linearly separating patterns of arbitrary dimensionality originating from two different classes[5], the specific hyperplane which optimally discriminates both classes including maximum generalization to unknown data is searched. An optimal hyperplane is defined as the linear decision function with maximal margin between the vectors of the two classes (cf. figure 3.19) and it was observed that to construct such optimal hyperplanes, only a small amount of the training data needs to be taken into account, the so-called *Support Vectors* which determine the margin [Cor95]. Due to their mathematical simplicity, linear discrimination planes are favorable.

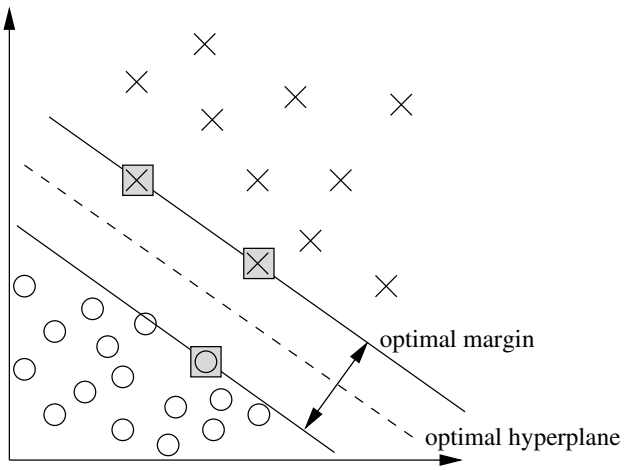


Figure 3.19: Example of a linearly separable problem in a two-dimensional space – the discriminating hyperplane is defined by means of the grey shaded support vectors (adopted from [Cor95]).

Unfortunately, most classification problems are not linearly separable when processing the appropriate data in its actual dimensionality, i.e. within the original data space. However, by transferring the data into a higher-dimensional space, usually the classification problem can be solved by means of the desired discriminating hyperplane. In fact, this is the basic idea of SVM based classification approaches.

[5]Generally, every n-class problem can easily be reduced to $n-1$ two-class problems. Thus, the theoretical considerations are usually restricted to two-class problems.

For a classification problem which cannot be solved by linear discrimination in the original data space but should be solved by means of the SVM technique, it first needs to be investigated how to obtain a proper higher-dimensional data space which allows the discrimination using a hyperplane. When Φ is assumed as a nonlinear transformation which maps data vectors $\vec{x} \in \mathbb{R}^n$ into a higher-dimensional space $\mathbb{R}^N$ with $N \gg n$, where the training samples are linearly separable, then the discrimination function is defined as follows:

$$f(\vec{x}) = \vec{w}^T \Phi(\vec{x}) + b.$$

Here $\vec{w}$ and b designates the parameters for the desired hyperplane. Due to the linear separability in the target space $\mathbb{R}^N$, $\vec{w}$ can be expressed by a linear combination of the transformed sample vectors which corresponds to the Perceptron learning algorithm (cf. [Zel97]):

$$f(\vec{x}) = \vec{w}^T \Phi(\vec{x}) + b = \sum_{i=1}^{I} \alpha_i y_i \Phi(\vec{x}_i)^T \Phi(\vec{x}) + b, \tag{3.34}$$

with α_i denoting the number of false classifications during training and $y_i \in \{-1, 1\}$ designating the classification result for the two-class problem. The size of the sample set used for obtaining the discrimination function is represented by I.

The basic practical problem with equation 3.34 is the exploding computational effort when enlarging the dimensionality of the data. Thus, usually the straightforward solution of direct transformation of the data cannot be applied. However, there are transformations Φ, whose dot product $\Phi(\vec{x})^T \Phi(\vec{z})$ can be expressed as function k of the dot product of both $\vec{x}$ and $\vec{z}$ in the original space, i.e.:

$$\Phi(\vec{x})^T \Phi(\vec{z}) = k(\vec{x}^T \vec{z}).$$

This implies that the dot product of two transformed vectors, which is a prerequisite for the actual classification according to equation 3.34, can be calculated *without* the computationally expensive transformation. Such functions $k(\vec{x}, \vec{z})$ are called *kernels* and some examples are:

- Polynomial kernels: $k(\vec{x}, \vec{z}) = (\vec{x}^T \vec{z} + c)^d,\ c \in \mathbb{R}, d \in \mathbb{N}$,
- Gaussian kernels or radial base functions – RBFs: $k(\vec{x}, \vec{z}) = \exp(-\frac{(\vec{x}-\vec{z})^T(\vec{x}-\vec{z})}{\sigma^2})$, or
- Sigmoidal kernels: $k(\vec{x}, \vec{z}) = \tanh(a\vec{x}^T \vec{z} - \Theta)$ for some a and Θ.

By means of such kernels, discriminating functions in higher-dimensional spaces can be defined without explicit calculation of the transformation Φ:

$$f(\vec{x}) = \vec{w}^T \Phi(\vec{x}) + b = \sum_{i=1}^{I} \alpha_i y_i \Phi(\vec{x}_i)^T \Phi(\vec{x}) + b = \sum_{i=1}^{I} \alpha_i y_i k(\vec{x}_i, \vec{x}) + b. \tag{3.35}$$

In order to obtain complete classification systems for real problems, several practical problems need to be solved after the formulation of the general problem (including the principle approach for a computational feasible solution). The discrimination function given via

formula 3.35 is optimized regarding the *training* samples. For effective use of SVMs, the discrimination power needs to be optimized for unknown data which implies the maximization of the so-called generalization ability of SVMs. Therefore, the generalization failure needs to be minimized which is usually performed due to maximization of the so-called *hard margin* $\gamma = \min_{1 \leq i \leq I} \gamma_i$ where $\gamma_i = \frac{\bar{\gamma}_i}{||\vec{w}||}$ denotes the geometric distance of an annotated training sample $(\vec{x}_i, y_i)$ regarding the discrimination function f and $\bar{\gamma}_i = y_i(\vec{w}^T \vec{x}_i + b)$ determines the according functional distance. For complicated problems where even the training samples cannot be separated linearly, the maximization of the hard margin γ is usually performed with respect to the minimum of a so-called *slack-vector* $\vec{\xi}$ whose components define *soft* margins for every training sample. The maximization itself is often performed by exploiting the Kuhn-Tucker theorem for a Lagrange optimization approach and a complete classification system can be obtained by means of the *Sequential Minimal Optimization (SMO)* technique. For details which are not in the focus of this thesis the interested reader is referred to e.g. [Cor95, Chr00].

According to the arguments given at the beginning of this section, Support Vector Machines are applied to the sequence analysis problem especially for screening applications where homologues of single sequences or complete families are searched within larger amounts of unknown data. Generally, two variants of applying SVMs are reported in the bioinformatics literature:

Direct sequence data processing: Sequences are mapped to some kind of numerical representation (cf. section 3.3.1 for an overview of the most common techniques) and SVMs are trained for every target class. As usual for SVM applications the optimization strategy is directed towards the maximum discrimination between sequences originating from the appropriate protein family of interest and all others.

As one example, Christina Leslie and colleagues in several publications used the so-called n-mer feature space representation where protein sequences are mapped to the corresponding n-spectrum which is the set of all n-length subsequences that the appropriate protein sequence contains.[6] Based on this representation the general SVM framework is used for remote homology detection by applying various kernel functions. Generally, the choice or design of proper kernel functions is the crucial part of SVM based sequence analysis techniques. According to the sequence data, usually some kind of string kernels are used, e.g. in [Les02] a simple spectrum kernel is applied which represents the dot product of two n-mer spectrum vectors. As an enhancement, in [Les04] the mismatch kernel was developed which smoothes the n-mer spectra by allowing at most m mismatches for contributing to a particular spectrum coefficient.

Post-processing of alignment scores: As an alternative to the direct processing of sequence data within the discriminative SVM framework, several researchers utilize the actual scores generated by a preceding conventional similarity based alignment step. Tommi Jaakkola and colleagues in [Jaa98, Jaa99] apply the Fisher kernel to alignment scores obtained by Profile HMM evaluation of protein sequences. Consequently, the generative approach of Profile HMM alignment delivers the features (i.e.

[6] In fact the so-called spectrum is actually a histogram of n-mer usage.

the scores) which are "postprocessed" by the discriminative SVM technique. Similar to this, the approach of Li Liao and William Noble is based on the scores obtained by a pairwise alignment technique like the standard Smith-Waterman algorithm and a succeeding application of SVMs [Lia02]. The authors argue, that their technique is significantly faster than the SVM Fisher method mentioned before.

The general idea of applying discriminative methods to the problem of homology detection is straightforward because it is a generalization of the widely used log-odd scoring approach. Support Vector Machines are a very powerful framework for discrimination tasks which recently emerged from the machine learning community. In several applications it could be shown that the proper application of SVMs for sequence analysis tasks, either directly used for sequence data or by postprocessing conventional alignment scores, can improve the classification accuracy. So far, the price for this enhancement is still rather high since larger training sets are required for *robust* modeling and the overall computational effort is not negligible, despite efficient kernel functions.

In addition to SVM based techniques, the idea of discriminative analysis can be generalized to optimized training techniques for Hidden Markov Models (cf. [Hua01, pp. 150ff] for a general overview and [Dur98, p.67f.], and [Edd95b, Mam96] for Profile HMM specific explanations). The major drawback of such techniques is again the larger number of training samples required for robust model estimation. This prevents them from general applicability.

3.3 Signal Processing based Sequence Comparison

For several applications of molecular biology research like target identification or validation within drug discovery tasks (cf. section 2.4), the overall situation regarding suitable training data is rather difficult. Since the applications are situated almost at the beginning of the general processing pipeline, primary structure data is the only source of information for detecting members of protein families which are in some way interesting for the researcher. This data is the direct result of the preceding sequencing operations and the result of sequence comparison directly influences all the succeeding steps of molecular biology processing. In the last sections, a broad overview of the state-of-the-art in analysis techniques based on amino acid sequence data was given. Most methods described, no matter whether conventional pairwise alignments or machine learning based probabilistic models of sequence families, are based on string processing.

Generally, symbolic information like strings, e.g. protein sequences, represents discrete data. All approaches for processing this data are limited to discrete techniques which are certainly powerful but the big arsenal of general signal processing methods cannot be applied at all or only in a very limited way. So far, discrete techniques were the obvious choice for sequence data because discrete data (sequences consisting of residues originating from a fixed inventory) was processed. The only exceptions were alternative representations for sequence classification using neural networks or support vector machines.

Throughout the years very effective techniques were developed for the analysis of natural, real-valued signals in a wide range of applications. Very prominent examples are various transformations like the well-known Fourier transformation. Since the problem of detecting

remote homologue protein sequences could not be solved *in general* in the last few years, new approaches to the task are demanded. One idea of the thesis is to open the most effective probabilistic models for protein families, Profile Hidden Markov Models, for the field of general signal processing.

To the authors knowledge, all present Profile HMM approaches reported in the literature are based on raw sequence, i.e. discrete, data. Furthermore, there are only few publications available regarding signal processing based protein sequence classification – the promising signal based analysis is currently mostly neglected by most researchers. In the following sections these rare approaches are summarized. Here, both techniques for protein classification and the related task of DNA analysis are considered. This is reasoned by the fact that these techniques can usually be generalized to the analysis of protein sequences (which will be discussed in the appropriate sections). First, in section 3.3.1 alternative sequence representations which enable the use of signal processing techniques are discussed. Following this, in section 3.3.2 the most promising current classification approaches are outlined.

3.3.1 Alternative Representations of Protein Sequences

Almost the whole set of powerful signal processing techniques was developed for the analysis of natural signals, i.e. real-valued functions. Thus, they cannot be applied to protein sequences without any modifications of the data. The trivial numerical representation of protein sequences would be to assign (arbitrary) numbers to every amino acid resulting in discrete but real-valued "signals". One choice of assignment could be: A = 1, R = 2, N = 3, ..., V = 20. The major drawback of this trivial assignment scheme is the incorporation of an artificial and certainly completely wrong relation. There is no reason for a *dramatically* larger distance between Alanine (A) and Valine (V) compared to the distance between Alanine (A) and Asparagine (N). Consequently, numerical encoding schemes must not incorporate distortions of the relationships between amino acids.

The signal based representations of protein sequences reported in the literature can be divided into two categories:

1. Based on theoretical considerations of the discrete symbol set, sequences are encoded into signals preserving the inter-symbol distances by means of various vector representations or statistical properties (cf. "Vector Space Derived Encoding Schemes").

2. For the second encoding method, the actual biochemical properties of amino acids are considered. Real-valued signals are obtained by means of direct mappings of protein sequences' residues to numerical values representing some biochemical property (cf. "Encoding based on Biochemical Properties").

In the following, both categories of sequence encoding including their most important representatives are discussed.

Vector Space Derived Encoding Schemes

The first category of encoding schemes was already introduced in the discussion of neural network based sequence classification approaches (cf. page 66). Here, the simplest correct,

i.e. distance conserving, encoding method was described as the creation of 20-dimensional orthogonal vectors. Generally, symbolic data is mapped to some N-dimensional vector-space. For the case of orthogonal base vectors described above, N designates the number of different symbols, i.e. for protein sequences $N = 20$.

Besides the simple base vectors, certain alternative definitions are imaginable. Here, the most important point is the conservation of distances between the appropriate symbols which does not necessarily imply equal distances. Depending on prior knowledge about the mutual pairwise relations between the appropriate amino acids (e.g. depending on biochemical properties – see next section) the distances can be adjusted individually. As one example for a vector space of DNA data, Dimitris Anastassiou in [Ana00] and [Ana01] defined the base vectors as the complex conjugate pairs $T = A^*$ and $G = C^*$, e.g.:

$$A = 1 + i, \quad T = 1 - i, \quad C = -1 - i, \quad G = -1 + i. \tag{3.36}$$

By means of these definitions and the genetic code, i.e. the well defined mapping of nucleotide codons to amino acids, numerical values can also be assigned to amino acids using the following procedure. Generally, the protein coding process can be modeled as a FIR digital filter[7], in which the input $x[n]$ is the numerical nucleotide sequence, and the output $y[n]$ represents the possible numerical amino acid sequence:

$$y[n] = h[0]x[n] + h[1]x[n-1] + h[2]x[n-2].$$

If $h[0] = 1, h[1] = 1/2$, and $h[2] = 1/4$ and $x[n]$ is defined by the parameters in equation 3.36, then $y[n]$ can only take one out of 64 possible values. Thus, the entire genetic code consisting of 64 codons which encode the 20 amino acids (or STOP) can be drawn on the complex plane as shown in figure 3.20. Here, exemplary Methionine (labeled Met) corresponds to the complex number $(1 + i) + 0.5(1 - i) + 0.25(-1 + i) = 1.17 + 0.88i$ [Ana01]. Note that this encoding scheme is partially redundant since some of the amino acids are multiply defined. Compared to the 20-dimensional vector space of the simple orthogonal base vector approach, here two-dimensional representations of the 20 amino acids are sufficient.

As an alternative to the complex number representation of amino acids, Paul Cristea proposed a tetrahedral representation of the genetic code and thus the amino acids [Cri01]. He argued that the classic cartesian representation of the genetic code depending on the actual nucleotides at the first, second and third position in codons which determines the actual amino acid, does not correctly reflect the natural structure of the genetic code. He developed optimal symbolic-to-digital mappings for nucleotides as well as for amino acids which are apparently suitable for the comparison of whole genomes (cf. [Cri03]).

Besides the encoding schemes described above which are all more or less related to some kind of vector space analysis, further encoding schemes based on theoretical considerations were proposed in the literature. As one example Gerhard Kauer and Helmut Blöcker in [Kau03] adopted an early approach of Kenneth Breslauer and colleagues regarding enthalpy based signal representation of DNA sequences [Bre86]. Here, the enthalpies of residues

[7] FIR is the acronym for *Finite Impulse Response* which designates a filter whose impulse response is finite in length. For details regarding general digital signal processing including filtering, the standard work of Alan Oppenheim and Ronald Schafer [Opp89] is a good starting point.

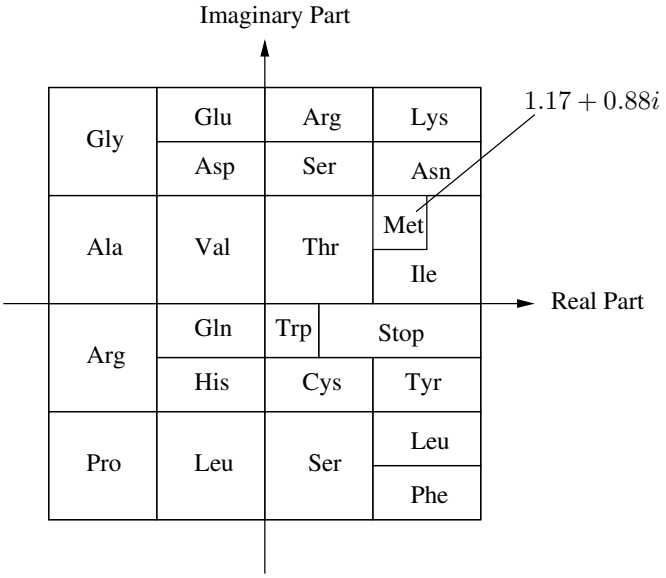


Figure 3.20: Numerical amino acid representation using complex plane; the exemplarily marked point designates the actual numerical representation of Methionine (adopted from [Ana01]).

with their respective neighboring nucleotides are taken as signal values. Such enthalpy data can be obtained from the literature and generally the mapping approach can be generalized to sequences of amino acids, too.[8]

Encoding based on Biochemical Properties

Natural signals evolving in time usually consist of measurements of some physical values captured at well defined, discrete time-steps. As one example, acoustic signals as processed in automatic speech recognition applications represent the progression of the sound pressure during uttering. Other examples of natural signals are the progression of air pressure or of temperature values which are important for weather forecast applications. All such signals have one characteristics in common: they represent the progression of some kind of physical, i.e. natural properties.

In these premises, for the second category of signal representation approaches for protein sequences, biochemical properties of the sequences' residues are used for obtaining the required numerical representations. This becomes obvious since the biological function of proteins is implied by their biochemical properties. By means of the so-called "wet-lab" analysis, i.e. by actual biochemical investigations of the biological matter of proteins, biochemical properties like hydrophobicity, charging, pH-value etc. are in fact measured. Of course, the sum of biochemical properties corresponds to the appropriate amino acids but the symbol data "shields" details due to abstraction. Throughout the years many researchers analyzed the biochemical properties of amino acids in great detail. The results of these experimental evaluations are usually reported in so-called amino acid indices serving as

[8] Enthalpy (symbolized H, also called heat content) is a thermodynamic quantity which represents the sum of the internal energy U of matter (here the particular residue) and the product of its volume V multiplied by the pressure P: $H = U + PV$ [Her89].

a mapping scheme for amino acids to numerical values. Shuichi Kawashima and Minoru Kanehisa in [Kaw00] compiled a large amount of such amino acid indices which are the base for most of the encoding schemes based on biochemical properties.

As one example the *Electron Ion Interaction Potential (EIIP)* is used for obtaining numerical representations of protein sequences. Originally proposed by physicists in the 1970s (cf. [Vel72]), Irena Cosic and colleagues developed various approaches for the classification of DNA as well as protein sequences based on this mapping [Cos97, De 00, De 02]. The EIIP describes the average energy of all valence electrons in a particular amino acid. Every amino acid or nucleotide, irrespective of its actual position in a sequence, can be represented by a unique number which is summarized for both nucleotides and amino acids in table 3.1.

Nucleotide	EIIP	Amino Acid	EIIP
A	0.1260	Ala (A)	0.0373
G	0.0806	Arg (R)	0.0959
T	0.1335	Asn (N)	0.0036
C	0.1340	Asp (D)	0.1263
U	0.0289	Cys (C)	0.0829
		Gln (Q)	0.0761
		Glu (E)	0.0058
		Gly (G)	0.0050
		His (H)	0.0242
		Ile (I)	0.0000
		Leu (L)	0.0000
		Lys (K)	0.0371
		Met (M)	0.0823
		Phe (F)	0.0946
		Pro (P)	0.0198
		Ser (S)	0.0929
		Thr (T)	0.0941
		Trp (W)	0.0548
		Tyr (Y)	0.0516
		Val (V)	0.0057

Table 3.1: The Electron Ion Interaction Potential (EIIP) values for nucleotides and amino acids (cf. [Cos97, p.13]).

The process of obtaining a signal based representation when utilizing biochemical properties can be summarized as follows.

1. Depending on the actual application of sequence analysis a proper amino acid index is selected. This step is rather crucial and expert knowledge is required. Very common choices are the EIIP values explained above or hydrophobicity indices (cf. e.g. [Man97, Mur02, Qiu03]).

2. All residues of all sequences (training-, query- and comparison data) are directly mapped to numerical representations using the amino acid index selected in the first step.

3.3.2 Signal Processing Methods for Classification

The motivation for applying signal processing techniques to the task of sequence analysis is given by the assumption, that highly diverging but related data contains regularities which are important for the actual biological meaning of the sequences but cannot be considered when raw amino acid data is processed. Furthermore such characteristics can be hidden e.g. due to (unknown) noise disturbing the hypothetical "protein generation process". Note that for this hypothetical process of protein generation so far no complete actual biological meaning is formulated. It is rather a question of theoretical considerations within the (artificial) framework of signal based representations of protein sequences.

After the description of signal based representations of protein sequences which are reported in the literature, in the following, protein sequence analysis techniques actually based on signal processing approaches are outlined. Among the rare publications dedicated to this topic, the majority looks at spectral analysis of protein sequences. Thus, first the general approach of spectral analysis is presented. Following this, actual applications for sequence comparison are briefly described.

General Spectral Analysis

In order to gain direct access to signal characteristics, in natural and engineering sciences the spectral analysis has been established as a powerful tool. Here, frequency information of a particular signal is directly accessible. The spectral representation, i.e. the direct description of frequencies and their contribution within the original signal, is usually obtained by transforming the signal of interest using some kind of function transformation technique. As the most prominent example, by means of the well-known Fourier transformation a signal is expressed as a weighted sum of sine and cosine terms. Besides this fundamental technique, throughout the years several refined methods were developed. Recently, a very flexible framework for obtaining spectral representations was developed, namely the Wavelet transformation.[9]

By means of this spectral representation, which is fully equivalent to the original signal representation, several signal analysis techniques can be applied much easier. As one example, the removal of (hypothetical) noise in a strictly signal theoretical meaning can be performed by simple filtering which corresponds to a spectral multiplication. Since the essentials of the particular family are usually of major interest for protein sequence analysis, here, lowpass filters can be applied. In addition, explicit structural filters can be used if some general clues about the protein family of interest is available a priori [Ana01, Kau03].

Certainly the most relevant spectral analysis technique is the detection of specialties of a particular protein family by spectral analysis. Such specialties which might distinguish one family from another can be obtained by e.g. peak detection. Peaks in the spectrogram represent some kind of characteristic frequency for certain biological function [Kri04]. As one example, in [De 02] peaks within certain scales of the multi-resolution analysis obtained by applying the Wavelet transform are explicitly interpreted as biological functions of the underlying protein like the oxygen-carrying function. Furthermore, several approaches are

[9] For mathematical and technical details of this powerful transformation see appendix A.

reported, where spectral analysis based peak detection using the Wavelet transformation is used for the prediction of structural properties [Mur02, Qiu03].

Finally, as one very interesting application of signal processing techniques, in [Arn96] the fractal scaling and organization properties of DNA sequences are analyzed using the Wavelet transformation, i.e. spectral analysis of DNA signals. Although explicitly dedicated to genomic data, this fascinating approach might be used for protein data, too.

The Resonant Recognition Model (RRM)

By means of the signal representation based on the EIIP mapping described in the previous section, Irena Cosic and coworkers developed the so-called *Resonant Recognition Model (RRM)* where protein classification is based on the analysis of cross-spectra [Cos97]. Several publications regarding this approach exist, all describing slight variations of the basic method for different applications and interpretations of the relations between RRM and two-dimensional as well as three-dimensional protein structures [De 00, De 01, Pir01, De 02].

Basically, within the RRM, pairwise sequence comparison is performed in the following way:

1. Both sequences involved in comparison ($\vec{s}_1$ and $\vec{s}_2$ each consisting of amino acid data) are mapped to the EIIP based signal representation ($\vec{x}_1$ and $\vec{x}_2$ each representing numerical values), i.e.:
$$\vec{s}_k \rightarrow \vec{x}_k, \quad k = \{1, 2\}.$$
If the sequences differ in length, the shorter one is extended by zero-padding.

2. For both sequences the spectral representation is obtained by applying (discrete) Fourier transformation (DFT):
$$X_k(n) = \sum_m x_k(m) e^{\frac{-2\pi i n m}{N}}, \quad n = 1, 2, \ldots, \frac{N}{2}, \; k = \{1, 2\},$$
where N designates the length of the appropriate sequence and $X_k(n)$ represents the n-th Fourier coefficient for the k-th protein sequence. Since the distances between adjacent residues of protein sequences are assumed equal (3.8 Å) and, therefore, set to $d = 1$, the resolution of the spectra is given by $1/N$. Because of their advantageous mathematical properties, recently the Fourier transform was substituted by the Wavelet transform (cf. appendix A).

3. In order to extract common spectral characteristics from sequences sharing the same or similar biological functions, cross-spectra are calculated using the following definition:
$$S_{1,2}(n) = X_1(n) X_2(n)^*, \quad n = 1, 2, \ldots, \frac{N}{2},$$
where $X_1(\cdot)$ are the DFT coefficients of the first sequence $\vec{s}_1$ in its numerical representation $\vec{x}_1$ and $X_2(\cdot)^*$ are complex conjugate DFT coefficients for the second sequence analogously.

4. Peak frequencies in the amplitude cross-spectral function define common frequency components of the two sequences analyzed. Thus, protein sequence classification and remote homology detection is performed based on peak detection within the cross-spectrum of two sequences.

The procedure described above can be generalized to the analysis of common frequency groups, i.e. biological functions, for a group of $K > 2$ protein sequences. Therefore, the definition of the cross-spectrum is extended towards a multiple cross-spectrum:

$$|M_{\{1,2,\ldots,K\}}(n)| = |X_1(n)| \cdot |X_2(n)| \cdot \ldots \cdot |X_K(n)|, \quad n = 1, 2, \ldots, \frac{N}{2}.$$

Peak frequencies in a multiple cross-spectrum denote common frequency components for all sequences analyzed. In figure 3.21 the procedure is summarized for two exemplary sequences of type *Fibroplast Growth Factor Protein*.

For sequence classification significant peaks within the cross-spectrum need to be determined. Therefore, in the original publications the authors proposed to utilize the measurement of the signal-to-noise ratio (SNR). For each peak the SNR is calculated as the ratio between signal intensity at the particular peak frequency and the mean value over the whole spectrum. Here, the basic assumption is that the prominent common frequency components manifested by the peak (if existing) describe the fundamental signal which is potentially disturbed by some kind of noise (all other frequency components). If the SNR is larger than some suitable threshold, the peak is significant and common biological functions can be assumed. Apparently, a SNR of at least 20 can be considered as significant.

By means of the RRM protein classification can be performed based on the following criteria [Cos97, p.18]:

1. One peak only exists for a group of protein sequences sharing the same biological function.

2. No significant peak exists for biologically unrelated protein sequences.

3. Peak frequencies are different for different biological functions.

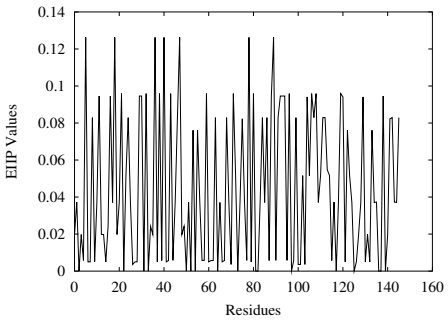
Especially the last point is important for real applications like functional mapping for a complete genome. Unfortunately, the differentiation between different biological functions is rather complicated since extremely small numerical differences need to be considered. Due to this difficulty and several more technical problems with the spectral analysis like the spectrum distorting padding of sequences with different lengths, the RRM based approach does not seem to be robust enough for remote homology detection at a broader scale although it is claimed otherwise.

```
PALPEDGGSGAFPPGHFKDPKRLYCKNGGFF
LRIHPDGRVDGVREKSDPHIKLQLQAEERGV
VSIKGVCANRYLAMKEDGRLLASKCVTDECF
FFERLESNNYNTYRSRKYSSWYVALKRTGQY
KLGPKTGPGQKAILFLPMSAKS
```

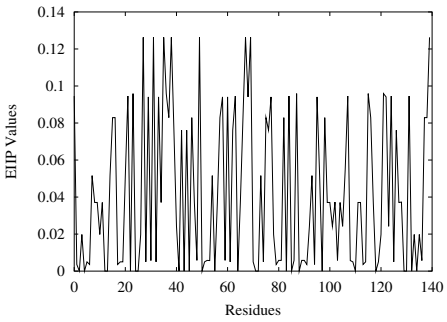
(a) Sequence 1 – Basic Bovine

```
FNLPLGNYKKPKLLYCSNGGYFLRILPDGTV
DGTKDRSDQHIQLQLCAEDIGEVYIKSTETG
QFLAMDTDGLLYGSQTPNEECLFLERLEENH
YNTYISKKHAEKHWFVGLKKNGRSKLGPRTH
FGQKAILFLPLPVSSD
```

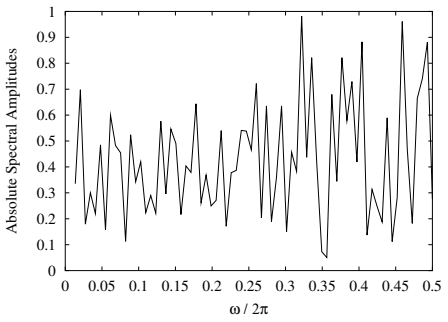
(b) Sequence 2 – Acidic Bovine



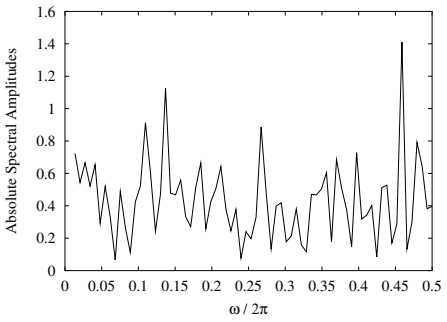
(c) EIIP representation of Sequence 1



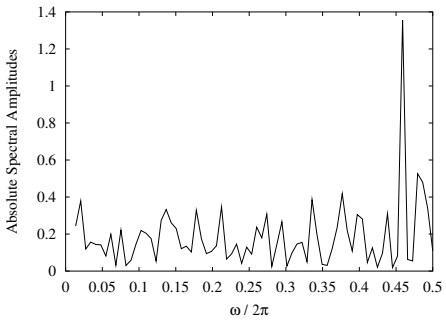
(d) EIIP representation of Sequence 2



(e) Power Spectrum of Sequence 1



(f) Power Spectrum of Sequence 2



(g) Cross-Power-Spectrum of both Sequences

Figure 3.21: The Resonant Recognition Model (RRM) procedure for two exemplary protein sequences: a+b) Amino acid representation of both sequences; c+d) graphical representation of the corresponding EIIP signals of both sequences; e+f) Power spectra of both signals; g) resulting cross-power-spectrum with prominent peak at 0.46 denoting common biological function (adopted from [Cos97, p.17], examples from [Vai04]).

3.4 Summary

Research performed in the field of molecular biology is generally dedicated to uncovering the biological functions of proteins. This is reasoned by the immense importance that proteins play in the metabolism of every organism. According to the fundamental theory of evolutionary development, on the molecular level specializations of proteins regarding certain biological functions are reflected by changes in their three-dimensional structure. Principally, the spatial folding of proteins and thus their biological function is caused by their primary structures. Due to this paradigm of similar function caused by similar amino acid sequence, vast amounts of basic molecular biology research can be performed in-silico by computational sequence analysis. Since the thesis is focused on the improvement of remote homology detection methods, in this chapter the state-of-the-art in general sequence analysis was summarized.

Due to the fact that protein data is usually given as sequences of amino acids, the majority of techniques is based on some kind of string processing. Traditionally, pairwise alignment approaches based on Dynamic Programming (DP) techniques are applied to the sequence classification problem. Here, powerful locally (Needleman-Wunsch) as well as globally (Smith-Waterman) optimizing algorithms were developed throughout the years. Unfortunately, especially for remote homology detection tasks, i.e. the analysis of highly diverging data, the performance of basic DP based techniques is not sufficient.

Thus, probabilistic approaches for explicitly modeling protein families recently became popular. One reason for their popularity is the fact that by means of these stochastic models less restrictive, i.e. some kind of “fuzzy” sequence classification can be performed. This is advantageous for the analysis of diverging but related data. Several techniques were developed within the bioinformatics community to address probabilistic sequence alignments. Examples of these are the use of Neural Networks, or Support Vector Machines. The approach which is currently most promising is based on the statistical modeling of protein families using Profile Hidden Markov Models which was described in detail in this chapter.

Besides the predominant string processing based techniques some approaches are documented in the literature which try to apply signal processing based techniques to the problem of sequence analysis. One major idea developed in this thesis is investigating alternatives to string processing. Therefore, the last section of this chapter was dedicated to related approaches. Although very powerful general signal processing based techniques exist, their relevancy for protein sequence analysis is rather small at present.

In this chapter a detailed qualitative analysis of existing sequence classification techniques with special focus on probabilistic techniques was given. In the following chapter a quantitative exploration of the currently most promising approach will be outlined. The effectiveness of Profile Hidden Markov models is investigated, based on a representative remote homology detection task whose results are the base for formulating the requirements for improved techniques: *Advanced Stochastic Protein Sequence Analysis*.

4 Concepts for Improved HMM Based Sequence Analysis

The basic motivation for almost all research activities within the field of molecular biology is to understand the fundamentals of biological processes at the level of protein synthesis and mutual interactions. By means of such insights, further knowledge gain about complex metabolic processes is addressed in order to e.g. develop drugs and therapies against severe illnesses like cancer or Parkinson's disease.

The general evolutionary concept states that similar biological functions are caused by similar three-dimensional structures. Thus amino acid sequences are a rich pool of information, and in the last few years a general paradigm shift in protein analysis was performed. Instead of (manually) generalizing expertise on specific proteins that was previously obtained, now broad screening techniques are applied allowing more general investigations on complete genomes by means of computational methods.

As introduced in section 2.4.1, the overall procedure of drug discovery can be interpreted as some kind of multi-stage "sifting-process". Based on the analysis of the universe of proteins, sequences not being of pharmaceutical interest are neglected for further complex analysis. This stage of early separation is rather crucial for the general success of drug discovery. If on the one hand too many actually improper candidate sequences remain after the target identification step, all following more complex stages of the drug discovery pipeline (cf. figure 2.7 on page 18) will become more expensive and time consuming. On the other hand, if protein sequences actually suitable for the desired pharmaceutical task are skipped, the complete drug discovery process will probably fail. Thus, the quality of computational methods applied to target identification and validation is of extreme importance.

In the previous chapter, the state-of-the-art in computational protein sequence comparison was summarized. Especially for the analysis of highly diverging but related data currently probabilistic models, namely Profile HMMs, of protein families are the most promising technique. However, despite improved sensitivity as well as specificity compared to applying pairwise sequence analysis techniques, the general problem is still very challenging. New approaches for probabilistic computational protein analysis are demanded for further improvements in protein classification.

This thesis is directed to the development of enhanced sequence analysis techniques which are based on Hidden Markov Models. In this chapter, new concepts for HMM based modeling of protein families are presented. Preceding this, in section 4.1 an exhaustive quantitative assessment of the capabilities of current Profile HMM based protein family models is presented. The analysis is performed for a typical task of remote homology detection for certain superfamilies. Concepts developed in this chapter are the basis for enhanced protein family Hidden Markov Models which will be described in detail in the next chapter.

4.1 Assessment of Current Methodologies' Capabilities

Profile Hidden Markov Models are applied to the task of protein sequence analysis as stochastic models of particular protein families. The general description of their theory and application concepts were presented in detail in section 3.2.2. A qualitative assessment of their capabilities for actual processing of biological data was given there, too.

In order to obtain a more detailed assessment of the principle capabilities of current Profile HMMs and as a baseline reference for further comparisons, in the following quantitative investigations regarding a typical task of remote homology detection at the superfamily level are presented. First, the datasets used for the task are completely specified. Following this, the evaluation results which are obtained by using common state-of-the-art software are presented in section 4.1.2. Currently two important frameworks for Profile HMM based sequence analysis exist:

SAM: The *Sequence Alignment and Modeling System (SAM)* was developed by the Computational Biology Group of David Haussler at the University of California in Santa Cruz, USA [Hug96]. As one specialty of SAM, unaligned sequences can be used for the establishment of models for protein families.

HMMER: The second major Profile HMM framework is basically the result of research activities performed by Sean Eddy and colleagues at the School of Medicine at the Washington University in St. Louis, USA [Edd01]. Profile HMMs are established by applying the standard training algorithms to multiply aligned training sequences. Using HMMER, libraries of Profile HMMs for protein families were created allowing broad database screening using probabilistic models of protein families [Bat00].

Both frameworks are rather similar regarding their underlying concepts of Profile HMM implementation and application. The common three-state architecture (cf. figure 3.14 on page 58) is generally used with refinements in HMMER where the Plan7 architecture (cf. figure 3.15 on page 59) is applied. Generally, raw sequence data is processed and log-odd scoring using specialized null models is the base for classification as well as detection tasks. Due to the general similarity of both frameworks, in the following the baseline results are obtained by using one of both packages, namely SAM. Informal experiments delivered only minor quantitative differences in evaluation results for both packages. Thus, the limitation to SAM experiments for the quantitative evaluation of the general method is arbitrary but justified.

4.1.1 Task: Homology Detection at the Superfamily Level

In order to evaluate the capabilities of current Profile HMM based approaches for protein sequence analysis and to compare the enhanced concepts developed in this thesis to them, the following task is defined:

> *For a given set of superfamilies, Hidden Markov Models will be established using reasonable amounts of training data. By means of these models, classification tasks representative for target validation as well as detection tasks*

representative for target identification within the process of general drug discovery are evaluated. The general focus will be put on classification accuracy for the first case as well as on sensitivity and specificity for the latter.

Here, classification means the assignment of the family affiliation to sequences which are known to originate from a closed set of protein families. This application is important especially for target validation tasks where sequences are annotated regarding certain models. Additionally, the general discrimination power of family models can be assessed since only the Profile HMMs themselves but no background models are involved in general. Thus, the results of this kind of evaluations give a first clue about the effectiveness of the appropriate methods.

In the second use case, homologue sequences of a particular protein family are searched in a general database of protein sequences which is typical for target identification tasks: For a family which is therapeutically relevant, sequences belonging to it are searched by comparison of query data to the probabilistic model of the appropriate family. By means of a threshold based analysis of the log-odd alignment scores the decision regarding affiliation to the target family is taken.

Datasets

For an objective judgment of the capabilities of certain sequence analysis techniques some kind of standardized datasets would be the optimal base for benchmarking. In alternative pattern recognition domains such datasets are very common and new methods are almost always assessed by comparison to state-of-the-art techniques based on this data. Unfortunately, for the bioinformatics domain the situation is rather different. There are hardly any standardized datasets available which are generally used within the community for the abovementioned objective judgment of sequence analysis techniques. Often new developments are explicitly directed to some specialized biological problem and the datasets used for evaluation are gathered correspondingly. Thus, the datasets for training as well as for the evaluation of both state-of-the-art probabilistic sequence analysis techniques and advanced stochastic protein family models developed in this thesis were defined by the author. The basic criterion for data selection was to maximally respect objectivity, i.e. the remote homology data used is as unbiased as possible regarding certain biological specialties.

According to Rainer Spang and colleagues, who cited in [Spa02] the "chicken and egg" problem for evaluating the effectiveness of annotation and search methods which was formulated by Steven Brenner and coworkers in [Bre98], it is rather difficult to assess the power of Profile HMMs by means of unknown data. The actual family affiliation of the data processed needs to be known in advance. Thus, existing sequence annotations obtained from one of the major public databases is used for the analysis of current approaches as well as for the comparison to the new techniques developed. Care needs to be taken for the actual selection of the reference database. If the database was created using automatic clustering techniques, the annotation will certainly not be completely error-free. An accurate annotation of the complete database implies the preceding successful solution of the computational sequence analysis problem, which is in these days unrealistic. However, when comparing the results of Profile HMM based predictions to such databases, the reference annotation created by an alternative classifier will be accepted as optimal. The

prediction results can only be compared to the potentially erroneous reference annotation obtained using alternative automatic classification approaches. Basically, false predictions need to be further examined with biological expertise since it is not clear whether the reference annotation was wrong or the new prediction. Unfortunately, many current databases were created automatically. Thus, they can hardly be used for serious assessments.

In order to properly simulate the actual situation for remote homology detection, the analysis needs to cover sequence data which is highly divergent. Many databases contain large amounts of redundant data or sequences being almost identical. Such close homologues can be processed by means of standard pairwise techniques as described in section 3.1. Probabilistic approaches address remote homologues, i.e. highly diverging but related sequences.

In these premises, the SUPERFAMILY hierarchy [Gou01] of the SCOP database [Mur95] is used for the assessments at the level of maximally 95% sequence similarity. Here, sequences belonging to a distinct superfamily must not have similarity values above 95%. This means, even data having sequence identities of only a few percent may belong to these superfamilies. Since the structural classification of the protein sequences contained in SCOP was performed manually by massively exploiting well-founded biological expertise, the quality of the labeling of the data is extraordinarily good (cf. the description of SCOP in section 2.3.2). Thus, it is predestinated for the general assessment of the capabilities of certain sequence analysis techniques.

Due to the complex model architecture of current Profile HMMs including the rather large number of parameters to be trained, the minimum number of training sequences was set to 44. This minimum number was chosen according to the analysis of the average length of sequences belonging to superfamilies and a rule of thumb for the number of examples per model parameter to be trained. Together with at least 22 sequences for the test case, 16 superfamilies containing at least 66 sequences each were selected for the evaluation. The sub-division of the sequences into training and test sets was performed fixed, i.e. no further leave-N out tests etc. are considered. In the following, the resulting corpus of training and test data is called SCOPSUPER95_66.

In table 4.1 a general overview of the corpus is given whereas in figure 4.1 the distribution of the similarity percentages over the datasets is illustrated by means of a histogram capturing all superfamilies included.

Inspecting the general corpus overview it can be seen that the sequences vary substantially in length for both training and test sets. This is the usual case for proteins subsumed in superfamilies which contain data with low sequence identity percentages but whose structures and major functional features suggest a probable common evolutionary origin (according to the superfamily definition of SCOP, cf. section 2.3.2).

In fact, the sequence similarities, which are limited to maximally 95 percent, are rather uniformly distributed all over the whole range with a small preference to the first third of the histogram for the appropriate training sets as well as for the assigned test sets. The bins of the similarity range histogram are defined via the SUPERFAMILY hierarchy each capturing five percent ranges with one exception in the first bin (0-10%).

SCOP Id	SCOP Superfamily Name	# Samples		Length (Mean/Std.-Derivation)	
		Training	Test	Training	Test
a.1.1	Globin-like	60	30	150.3 (13.6)	151.6 (11.1)
a.3.1	Cytochrome c	44	22	102.6 (24.1)	118.4 (32.6)
a.39.1	EF-hand	49	25	138.1 (48.0)	122.0 (39.3)
a.4.5	"Winged helix" DNA-binding domain	49	25	93.8 (26.6)	92.9 (23.1)
b.1.1	Immunoglobulin	207	104	108.9 (15.3)	106.7 (12.3)
b.10.1	Viral coat and capsid proteins	64	32	278.0 (92.9)	262.1 (85.2)
b.29.1	Concanavalin A-like lectins/glucanases	52	27	221.2 (51.2)	220.8 (72.9)
b.40.4	Nucleic acid-binding proteins	47	24	113.1 (36.6)	111.5 (47.2)
b.47.1	Trypsin-like serine proteases	55	28	231.4 (29.5)	226.0 (30.1)
b.6.1	Cupredoxins	50	26	143.9 (34.6)	139.0 (31.5)
c.1.8	(Trans)glycosidases	62	31	376.5 (76.4)	397.8 (84.0)
c.2.1	NAD(P)-binding Rossmann-fold domains	102	51	204.3 (58.9)	211.5 (75.1)
c.3.1	FAD/NAD(P)-binding domain	45	23	226.1 (93.3)	223.3 (86.3)
c.37.1	P-loop containing nucleotide triphosphate hydrolases	127	64	259.3 (120.4)	253.4 (85.6)
c.47.1	Thioredoxin-like	56	28	111.6 (38.2)	105.6 (35.3)
c.69.1	Alpha/Beta-Hydrolases	51	26	350.1 (103.7)	323.7 (25.0)
	Total:	1120	566		

Table 4.1: Overview of the SCOPSUPER95_66 corpus created for the assessment of current Profile HMM capabilities as well as for the comparison of the effectiveness of the methods developed in this thesis. For every superfamily the alpha-numerical SCOP Id as well as its real name as defined in the database is given. In the last row the total numbers of samples are summarized.

4.1.2 Capabilities of State-of-the-Art Approaches

The experimental evaluation of state-of-the-art techniques for both tasks defined for superfamily based remote homologue sequence analysis, classification and detection, was performed using the SAM package with default parameters. According to the suggestions of the authors, for all superfamilies of the SCOPSUPER95_66 corpus, Profile HMMs were estimated by applying the `hmmbuild` program with `-randseed 0` to the unaligned training sequences. During the training iterations, SAM created Profile HMMs of reasonable lengths including specialized Dirichlet mixture based model regularization. The emission probabilities of all Insert states were set to the geometric mean of the family specific amino acid frequencies of the match states as obtained during the final training step. Again, according to the suggestions of the SAM authors, all test sequences were aligned to the models using `hmmscore` with `-sw 2` which implies local alignments comparable to the standard Smith-Waterman approach (cf. section 3.1.1). Using the default setting, the alignment itself is performed using the Forward-algorithm as described in section 3.2.2 on page 48f. The alignment scores were determined as log-odd scores. As proposed by Kevin Karplus in

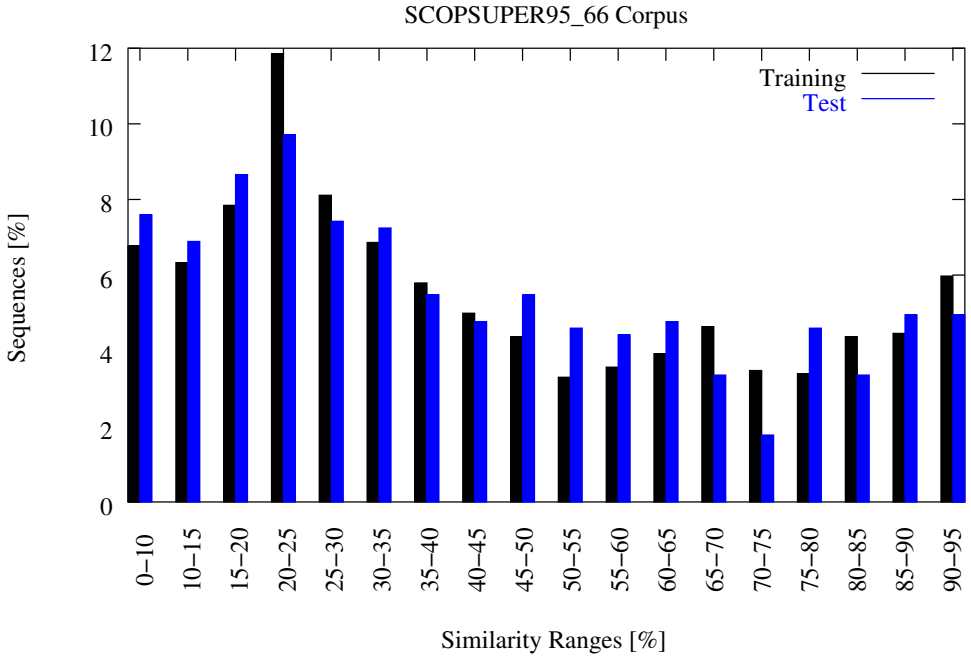


Figure 4.1: Histogram of similarity ranges for the SCOPSUPER95_66 corpus averaged over all 16 superfamilies involved (black: Training / blue: Test) illustrating the almost uniform distribution of similarities all over the whole range with one exception at 20-25%. Note that the lower limit of the identity ranges as defined by SUPERFAMILY is 10%. Thus, the first bin covers a broader range of 10% compared to the 5% bins otherwise.

[Kar98], the scores are calculated with respect to scores obtained by aligning the reverse sequence. Within the SAM package this null model is called *reversed null model* (the general motivation for this choice of null model was explained in section 3.2.2 on page 61). In order to determine the statistical significance of alignment scores for discriminating between target hits and misses during homology detection, extreme value distributions are evaluated as described in section 3.1.1.[1] For both tasks all models are evaluated independently for all appropriate test sequences which is compared to general pattern recognition applications rather unusual at least for the classification task.

Classification Accuracy

In general pattern recognition applications the capabilities of certain approaches for classification tasks are usually measured by means of the *classification accuracy* $\mathcal{C}$, and more common by its inverse the *classification error* $\mathcal{E}$. They are defined as the ratio of correct (for $\mathcal{C}$) / false (for $\mathcal{E}$) classifications and the overall number of decisions. Usually, the numerator of this ratio is further subdivided into the number of substitutions, deletions, and insertions. For protein sequence classification, i.e. global string alignment as defined for Dynamic Programming techniques, this subdivision is not important because usually com-

[1] In fact SAM's E-values are derived from log-odd scores S using the following formula: $E(S) = \frac{N}{1+\exp(-\lambda S)}$, where N denotes the database size (SCOPSUPER95_66: 566), and λ is a scaling parameter which is 1 when using natural logarithms [Hug96, p.91].

plete sequences are analyzed exclusively, i.e. without concatenations which could include insertions or deletions in the above meaning. Thus, both measurements are defined as follows:

$$\mathcal{C} = \frac{N_{\text{correct classifications}}}{N_{\text{overall decisions}}}, \qquad \mathcal{E} = \frac{N_{\text{false classifications}}}{N_{\text{overall decisions}}} = 1 - \mathcal{C}. \tag{4.1}$$

When applying state-of-the-art discrete Profile HMMs as created by SAM, for the SCOPSUPER95_66 corpus the classification accuracy is 67.1 percent. This corresponds to a classification error of 32.9 percent. For these values, the particular size of the test set, and a selected level of confidence of 95%, statistically significant changes are obtained when the percentages differ by more than 3.9 percent, i.e. the confidence interval is $[\pm 3.9\%]$ (cf. figure 4.2). To re-emphasize: For this representative task of remote homology classification almost one third of the decisions regarding the appropriate superfamily affiliation are wrong when using the currently most powerful sequence analysis techniques, namely Profile HMMs. This is problematic especially for target validation applications within the drug discovery pipeline.

In table 4.2 the results for the classification task are summarized.

Measure	Results [%]
Classification Accuracy $\mathcal{C}$	67.1
Classification Error $\mathcal{E}$	32.9
95% confidence interval	$\pm$ 3.9

Table 4.2: Summary of the classification results for the SCOPSUPER95_66 corpus when applying discrete Profile HMMs (SAM).

Following the assessment of current methodologies' capabilities for remote homology classification in the next section their actual effectiveness for detection tasks is evaluated.

Detection Performance

Usually target detection is performed by analyzing alignment scores regarding some threshold which discriminates between target hits and target misses. As discussed for pairwise alignment techniques (cf. section 3.1 on pages 31ff), the significance of scores is usually judged by means of so-called E-values denoting the probability of randomly occurred false predictions for particular scores. Especially for remote homology detection, the major difficulty is the actual determination of a suitable threshold for the target or non-target decision. For the comparison of different techniques, often *Receiver Operator Characteristics – ROC* curves are used (cf. e.g. [Mou04, pp. 192ff]). Here, the number of false positive predictions is illustrated as a function of the corresponding number of false negative predictions. The threshold selected for discrimination between target hit and miss, which is usually with respect to e.g. E-values, is implicitly given as a particular point within the ROC curve. By means of certain criteria, e.g. the costs for false positive predictions vs. the costs for false negative predictions, the optimum threshold can be determined by analyzing ROC plots. Generally, the closer the ROC curve is located to the lower and left borders of the diagram, the better is the detection performance of the underlying approach.

Classification error rates which are determined using a finite test set are, strictly speaking, only estimates for the general capabilities of a recognition system. In order to obtain correct rates for the sample space of the underlying statistical process, theoretically, an infinite number of experiments needs to be performed. Instead, for the error probabilities $\hat{\mathcal{E}}$, estimated using the finite test set, so-called *confidence intervals* $[\mathcal{E}_l, \mathcal{E}_u]$ are defined. Such an interval defines a range containing the actual probability $\mathcal{E}$ with a given statistical evidence, the so-called *level of confidence*.

A classification experiment can be interpreted as *Bernoulli*-process where only two events may occur: A (correctly recognized) and $\bar{A}$ (not or falsely recognized). Thus, the confidence interval for a given probability $\mathcal{E}$ needs to be determined by evaluating the Binomial distribution. According to the Moivre-Laplace theorem (local limit theorem), $\mathcal{E}$ can be interpreted as asymptotically normally distributed:

$$\frac{\hat{\mathcal{E}} - \mathcal{E}}{\sqrt{\frac{\mathcal{E}(1-\mathcal{E})}{N}}} \approx \mathcal{N}(0,1).$$

When performing N experiments, the lower and the upper boundaries of the confidence interval are defined as follows:

$$\begin{aligned}
\mathcal{E}_l &= \frac{N}{N+z^2}\left(\hat{\mathcal{E}} + \frac{z^2}{2N} - z\sqrt{\frac{\hat{\mathcal{E}}(1-\hat{\mathcal{E}})}{N} + \frac{z^2}{4N^2}}\right) \\
\mathcal{E}_u &= \frac{N}{N+z^2}\left(\hat{\mathcal{E}} + \frac{z^2}{2N} + z\sqrt{\frac{\hat{\mathcal{E}}(1-\hat{\mathcal{E}})}{N} + \frac{z^2}{4N^2}}\right).
\end{aligned}$$

Here, z designates the $(1-\frac{\alpha}{2})$-quantile of the standard normal distribution, whose value is documented in tables. Usually, for classification experiments, as performed in this thesis, a level of confidence of $1-\alpha = 95\%$ is used which means that the actual classification error rate $\mathcal{E}$ is within the given confidence range with a probability of 95 percent.

Based on confidence intervals, additionally, changes in classification error rates, caused by e.g. changes in the parameters of the underlying classification system, can be evaluated regarding their statistical significance. If the changed classification error rate is outside the confidence interval, these changes can be interpreted as statistically significant. Otherwise, they were most likely caused by chance.

Figure 4.2: General estimation of confidence intervals for probabilities, i.e. classification error or accuracy rates (cf. e.g. [Bro91, Sch96]). The derivation given is adopted from [Wie03].

For the assessment of the detection performance of discrete Profile HMMs, the scores obtained by aligning the sequences of the complete SUPERFAMILY database to the ap-

propriate superfamily models were analyzed. The complete SUPERFAMILY hierarchy of SCOP sequences with a maximum of 95% residue-level similarity contains approximately 8 000 entries. The number of false predictions were determined by means of the E-values generated by SAM. For a general overview about the detection performance of discrete Profile HMMs, in figure 4.3 the results of all alignments of SCOP sequences to all 16 superfamilies are summarized resulting in a single ROC curve. Since detection decisions are based on absolute scores, which are analyzed regarding their statistical significance, the number of false predictions can usually be limited to some reasonable number. Thus, a working area corresponding to such an exemplary limitation is separately shown as gray shaded rectangle.

It can be seen that the generally critical performance as measured for the classification task is problematic for the detection task as well. The number of false positive predictions remains rather high for reasonable numbers of false negatives. This is problematic especially for drug discovery applications because of the implied increased costs for useless further investigations within the drug design pipeline. Additionally, the number of false positives cannot be reduced until the number of false negative predictions increases significantly, which is again problematic for drug discovery because suitable candidate sequences might be missed.

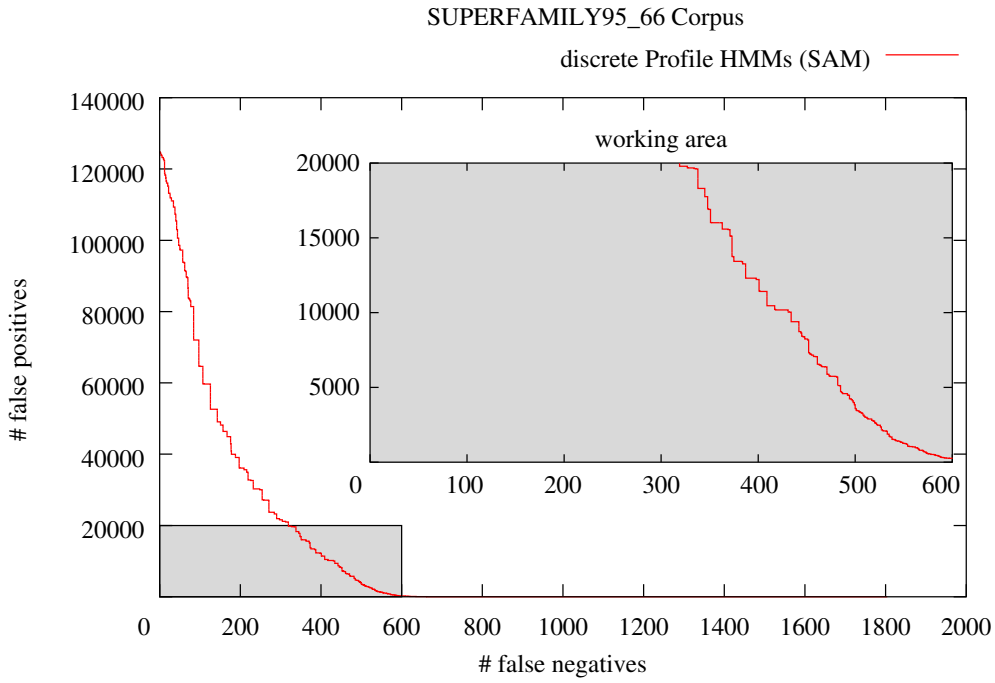


Figure 4.3: ROC curve illustrating the combined results for remote homology detection based on the SCOPSUPER95_66 corpus using discrete Profile HMMs (SAM). The grey shaded rectangle highlights the detection performance for the biologically most relevant working area when limiting the number of false predictions reasonably.

For further family specific analysis of the detection performance, the detection results are inspected individually for every superfamily contained in the SCOPSUPER95_66 corpus.

Generally, the results are comparable for all superfamilies (cf. figure 4.4, for clarity the presentation of the ROC curves is split into two diagrams). Some exceptions exist, either performing better than average (a.39.1, b.6.1, b.47.1) or worse (b.1.1, b.40.4, c.2.1). The analysis of these exceptional cases regarding the sequence similarities of the appropriate training and test sets as well as regarding the appropriate number of training samples does not show any specialties. In figure 4.5 the histograms of sequence similarities for the exceptional superfamilies mentioned above are shown individually illustrating the generally uniform distribution of the sequence similarities for the specific superfamilies. Thus, the reason for the non average performance seems to be intrinsic.

In order to judge the effectiveness of certain target detection methods for practical applications within e.g. pharmaceutical tasks, usually some characteristic values are extracted from the appropriate ROC curves. When fixing the number of false predictions maximally allowed (e.g. false positives), the corresponding number of false predictions (here false negatives) is a good measure. Usually, the percentage of false predictions is set to 5%. In table 4.3 these characteristic values are summarized illustrating the still rather weak performance of current discrete Profile HMMs for remote homology detection tasks.

False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
26.1	57.6

Table 4.3: Characteristic values for SCOPSUPER95_66 detection experiments: At fixed working points of the ROC curve allowing 5% false predictions, the numbers of corresponding false predictions are given. It can be seen that improvements are demanded since rather large percentages of false predictions occur.

4.2 Improving the Quality of HMM Based Sequence Analysis

After the detailed assessment of the general capabilities of the currently most promising technique for probabilistic modeling of protein families whose results were presented in the previous section, in the remaining parts of this thesis, new concepts for the improvements of the basic method will be presented. Obviously, such improvements are needed because even when using the most sophisticated sequence analysis techniques, namely Profile HMMs, the classification as well as the detection performance for a typical sequence analysis task are far from satisfying.

Generally, three basic issues relevant for the successful application of new powerful probabilistic models capturing the essentials of protein models for remote homology analysis can be formulated:

1. According to the assessment of the capabilities of state-of-the-art Profile HMMs for both classification and detection tasks, general improvements for both application types are demanded (*General Performance Improvement*).

2. Since one of the major applications of remote homology analysis using Profile HMMs is the detection of new members for therapeutically relevant protein families within

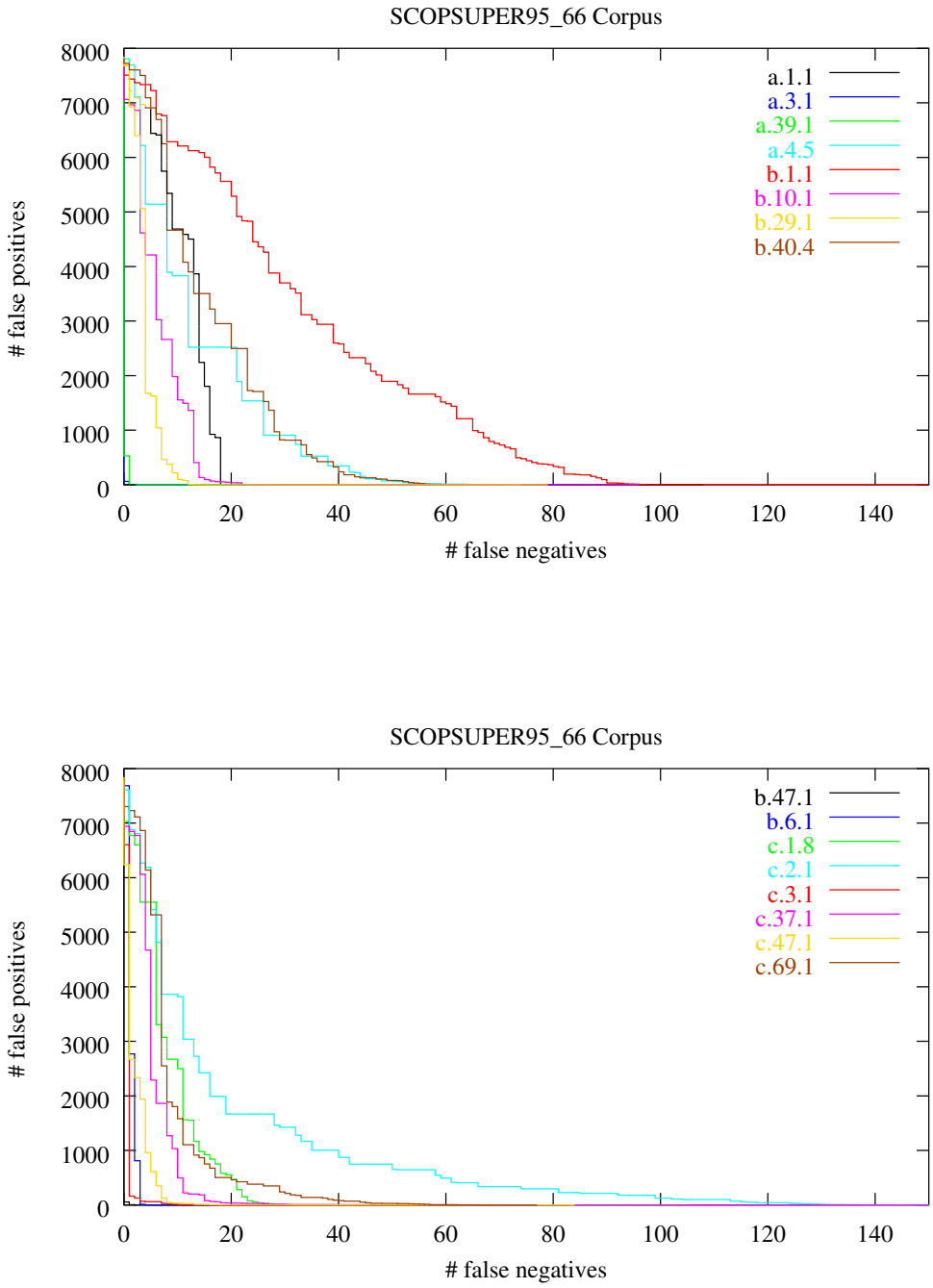
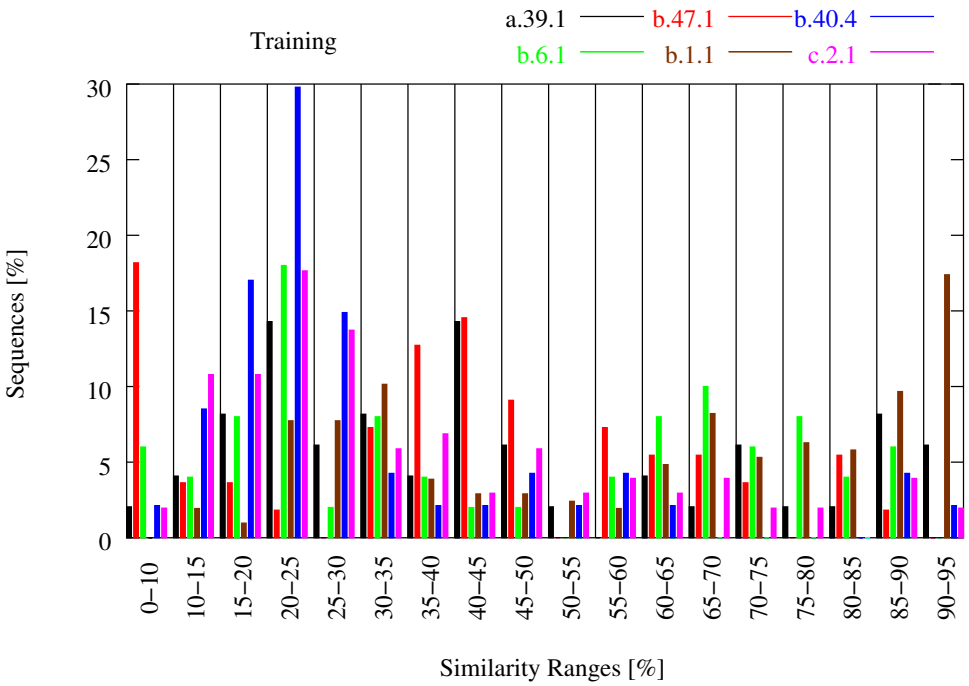
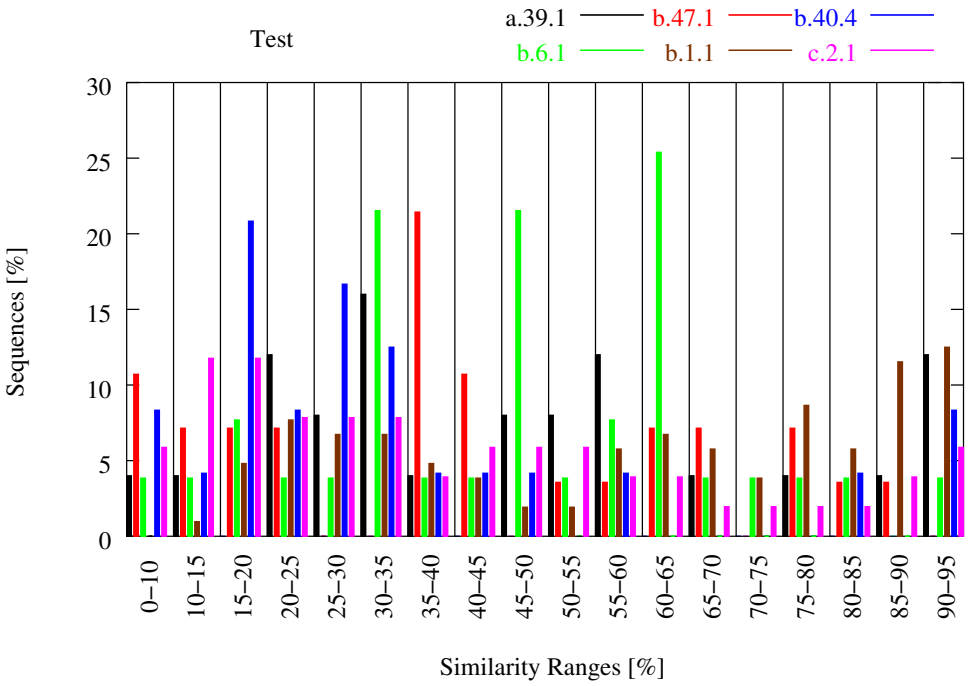


Figure 4.4: Results for remote homology detection based on the SCOPSUPER95_66 corpus using discrete Profile HMMs (SAM) shown individually for all 16 superfamilies by means of ROC curves. For clarity the illustration is split into two diagrams.



(a) Histograms for training sets



(b) Histograms for test sets

Figure 4.5: Individual histograms (training/test) of similarity ranges for selected superfamilies with non average detection performance, either significantly better (a.39.1, b.6.1, b.47.1) or worse than average (b.1.1, b.40.4, c.2.1).

drug discovery tasks, the number of sample sequences available for robust model estimation is usually limited. Thus, enhanced probabilistic protein family HMMs need to address this so-called *Sparse Data Problem*.

3. Nowadays, remote homology detection is usually performed by exhaustive screening of large sequence databases regarding target families of interest. Since the amount of data to be analyzed is already huge and still constantly growing, efficient evaluation of protein family HMMs is required (*Efficient Model Evaluation*).

In order to address these issues for improved protein family models, in the following concepts for advanced stochastic protein family modeling using Hidden Markov Models will be presented. First, the focus will be concentrated on a general description of the concepts. After the general presentation, detailed explanations of the new approaches addressing the three issues mentioned above will be given in chapter 5.

4.2.1 Semi-Continuous Feature Based Modeling

The general principle of molecular biology which justifies sequence comparison based research activities is that the sequence of amino acids mainly determines the three-dimensional structure and thus the function of proteins. Based on this, sophisticated string processing techniques were developed allowing the prediction of the biological function of proteins whose primary structure was sequenced by aligning them to sequences whose functions were already known. The majority of protein classification techniques is based on direct sequence to sequence or sequence to family alignment of raw amino acid data.

However, especially for protein families whose members share only minor sequence similarities, the classification and detection of new family members is still a challenging problem. Obviously (cf. the results of the assessment in the previous section), even the most sophisticated string processing techniques are not suitable enough for those *remote* homologue sequences. Since for several applications, like target identification at the beginning of the drug discovery pipeline, *principally* no further information regarding the proteins of interest are usable (e.g. their secondary structure), improved treatment of the data actually available seems to be the only option for better remote homology detection.

The three-dimensional structure of proteins which, according to the foundations of molecular biology (cf. chapter 2), determines their function, is reasoned by a complex folding process. Unfortunately, so far this process of spatially arranging the proteins' atoms is not completely understood. However, the certain angles of the backbone of a protein are determined by the local biochemical properties of the underlying sequence of amino acids. Thus, the actual three-dimensional occurrence of a protein is dependent on properties like hydrophobicity or electric charge of amino acids in their local context. Certainly, the biochemical properties are well summarized by the 20 standard amino acids but the number of such properties which are obtained throughout the years in countless wet-lab investigations is much higher than 20. Thus, exclusively analyzing amino acids and furthermore neglecting their local neighborhood seems rather critical.

One central concept for enhanced probabilistic protein family HMMs addressed by this thesis is the explicit consideration of biochemical properties as mentioned above for sequence analysis. Therefore, signal-like representations of biological data will be developed

which are richer than the standard representation using sequences consisting of the 20 standard amino acids. By means of this numerical data, features will be extracted which are relevant for family affiliations of protein data. Here, the huge arsenal of powerful signal-processing, and pattern recognition techniques can be applied. Such methods, which are very common for tasks of general pattern analysis, are currently not applicable for state-of-the-art Profile HMMs because these models are based on discrete symbolic data.

When Hidden Markov Models are applied to tasks where feature vectors originating from a principally continuous feature space are processed, discrete models are not suitable. The reason for this is the quantization error which principally exists when mapping continuous data to a symbolic representation. Instead, continuous modeling approaches as described in section 3.2.2 (pp. 46ff) directly integrate a mixture density representation of the continuous feature space. The mixture components are evaluated when processing feature vectors delivering continuous emissions. Generally, continuous Hidden Markov Models are the methodology of choice for HMM based pattern recognition tasks where feature vectors are processed. Thus, in addition to the richer sequence representation of biological data, *continuous* Profile HMMs will be developed. These models contain the general three-state topology (cf. figure 3.14 on page 58) but their emissions are based on a mixture density representation of the continuous space of the new feature vectors. Since the amount of family specific training data is usually rather small for e.g. target identification in drug discovery tasks, the developments will be focussed on a variant of continuous HMMs, namely *semi-continuous* Profile HMMs. By means of such models especially small training sets are efficiently exploited which makes them attractive for remote homology analysis.

The estimation of semi-continuous HMMs can principally be divided into two parts, namely the estimation of the feature space representation and the actual model optimization. In fact this separation is the basic advantage of semi-continuous modeling which can be exploited for robust estimation of protein family models using small family specific sample sets. The mixture density representation can thereby be obtained using general feature data, i.e. protein data which are not specifically assigned to the particular target family. In order to focus the resulting general feature space representation to a particular target family, mixture density adaptation can be performed. Various techniques were proposed for general feature space adaptation, e.g. MAP-, or MLLR-adaptation. In this thesis, semi-continuous protein family models are obtained using a mixture density based feature space representation which is estimated using general protein data. This considerable amount of sequences originates from one of the public sequence databases. Since the statistical base for general mixture density estimation is substantially larger than the amount of actual target family specific training data, the protein feature space can suitably be represented. The specialties of a target family of interest are respected by family specific weights of the mixture components which can be trained using little training data. Furthermore, this protein feature space representation is focused on a particular target family by applying mixture density adaptation techniques.

According to the three basic issues relevant for the successful application of new probabilistic models as defined in the previous section, the approach of feature based (semi-) continuous modeling principally addresses the general performance improvement and partially the sparse data problem.

4.2.2 Model Architectures with Reduced Complexity

The analysis of the technical procedure performed for the representative experiments of superfamily based remote homology analysis as described in section 4.1 reveals one principle problem of current Profile HMMs. For best covering the sequences belonging to a particular protein family very complex models are used. For robustly establishing such models rather large amounts of sample data are required. Even with the most sophisticated regularization techniques like Dirichlet mixture modeling of prior knowledge regarding amino acid distributions, the models' performance for remote homology detection is not satisfying.

Sequences summarized in protein families which are modeled by Profile HMMs are usually rather long. This is also the case when modeling the smallest *functional* protein unit – the protein domain. Thus, consensus strings consisting of hundred and more residues are very common (cf. the average lengths of the sequences belonging to the superfamilies of the SCOPSUPER95_66 corpus shown in table 4.1 which are reasonable indicators for the lengths of the consensus strings). Usually, the complete protein family is modeled using a single Profile HMM. According to the consensus' length, these models contain large amounts of states (e.g. the *Immunoglobulin* – b.1.1 – model contains 300 states).

For best flexibility that is necessary especially for the analysis of related sequences which can be highly divergent in both length and residue constitution, the general three-state Profile HMM architecture is very common. Every node of such a model represents a single column in the consensus string and contains three specialized states: Match, Insert, Delete. According to the general paradigm of sequence analysis using Dynamic Programming techniques this is straightforward. However, in order to reach the necessary flexibility when processing long sequences, principally large amounts of HMM parameters need to be trained which implies substantial numbers of example data.

In order to alleviate the sparse data problem, protein family models containing model architectures with reduced complexity are investigated with respect to their general performance for successfully classifying protein family affiliations of sequence data. Due to the newly developed feature based sequence representation the complex three-state model topology will become needless which implies new chances for reducing the overall number of states necessary for complete protein family modeling using HMMs.

Compared to general pattern recognition applications using HMMs, modeling large and diverging parts in a single model is rather uncommon. The analogy for speech recognition tasks would be to establish word models using single HMMs. Especially for languages containing a huge inventory of words this level of modeling is critical because the enormous amount of suitable training data required for robust model estimation is rarely available. Thus, usually substantially smaller parts of words are captured by HMMs (triphones representing a phoneme in its local neighborhood). Based on such so-called sub-word units (cf. e.g. [Fin03, pp. 152ff]), complex models for the description of complete words and compounds are created by concatenation of the basic parts. Principally, such modeling approaches are the methodology of choice for the majority of current general HMM based pattern recognition applications.

In order to tackle the sparse data problem defined in the previous section, modeling approaches based on smaller protein units are investigated in this thesis. Generally, these units are comparable to motifs (cf. section 3.2.2 on page 63 for motif based HMMs for protein

sequences). However, they will not be estimated on the raw amino acid data but using the feature based sequence representation developed in this thesis. In a completely unsupervised and data-driven approach relevant parts of protein families are determined and modeled using less complex and tighter models. These low-level and feature based building blocks of the protein family will be called *Sub-Protein Units (SPU)*. Compared to standard protein family modeling using (global) Profile HMMs, in the SPU based approach only the absolute essentials relevant for (i.e. conserved within) the sequences belonging to the particular protein family are considered. These essentials are captured by models with reduced complexity containing significantly less parameters. Due to the reduced number of parameters, the amount of training data required for robust model estimation can be reduced, too. For complete protein family modeling, SPUs will be appropriately concatenated.

Both approaches for estimating robust probabilistic protein family models containing significantly less parameters, i.e. the global feature based modeling technique and the latter SPU based approach, will be compared to the standard discrete Profile HMM modeling as well as to the semi-continuous feature based Profile HMMs and among each other. Finally, the decision for the best suitable variant will be discussed.

4.2.3 Accelerating the Model Evaluation

In the "Introduction for the impatient" of the manual for the *Wise*-tools, which is one commonly used framework for general sequence analysis using HMMs, the currently widespread opinion of the research community regarding the efficiency of model evaluation is characterized by the following (hypothetical) dialog between a Wise-user and the developers [Bir01]:

> *"[Question:] It goes far too slow*
> *[Answer:] Well . . . I have always had the philosophy that if it took you over a month to sequence a gene, then 4 hours in a computer is not an issue."*

Most current HMM based approaches for sequence analysis were developed with exclusive respect to the general method, i.e. neglecting the efficiency of the actual model evaluation. Contrary to the argumentation of the developers of Wise as cited above, "4 hours in a computer" is indeed an issue. As discussed in chapter 1, modern molecular biology is strongly influenced by the paradigm shift performed due to computational sequence analysis. The gain in biological knowledge obtained by broad screening of large databases for particular protein families became possible by powerful *and* efficient techniques like BLAST. Thus, when applying more sensitive and powerful techniques like Profile HMMs, efficiency is important, too. Some of the present techniques are performed on specialized, and distributed hardware solutions (e.g. HMMER was ported to the massive parallel PARACEL© GeneMatcher™ architecture). Although by means of such "brute-force" accelerations HMM based sequence analysis can be performed on large databases, the principle problem of inefficient model evaluation remains. Increasing the computational power for faster evaluation only treats the symptoms. Especially, when more complex procedures for emission estimations like feature based approaches are applied addressing the improvement of the classification accuracy for remote homologue sequences, *algorithmic* accelerations of the model evaluation are required.

Inspired by general pattern recognition applications of Hidden Markov Models, in this thesis concepts for accelerating the evaluation of protein family models are adopted and transferred to the bioinformatics domain. Following the argumentation given above, the focus of such acceleration techniques concentrates on algorithmic changes within the evaluation process. For annotation tasks (e.g. for drug target validation), currently multiple Profile HMMs are evaluated sequentially, i.e. every query sequence is aligned serially to every Profile HMM considered and the classification decision is determined by alignment score comparison. Such tasks are generally comparable to automatic speech recognition applications where signal parts are classified with respect to a fixed (large) inventory of words. Especially *online* speech recognition is not possible when performing sequential model evaluation combined with the posterior decision as for the bioinformatics case. Instead, all models are evaluated in parallel which makes the overall process much faster when using sophisticated model combinations by combined state spaces *and* pruning techniques. In this thesis, protein family models are similarly evaluated in parallel and certain pruning techniques are applied allowing for fast model evaluation.

Furthermore, effective pruning techniques are applied which significantly limit the computational effort for model evaluation on average. The basic motivation for such pruning techniques is the observation that in complex Hidden Markov Models (like Profile HMMs) very different local properties of the data are captured. However, when evaluating the models using the Viterbi- or the Forward- (Backward-) algorithm, large amounts of possible paths through the state space are analyzed. Paths covering certain local characteristics are very probable when observing such data but paths representing alternative local characteristics are very improbable to match this data. However, although they hardly contribute to the global solution in the general evaluation scheme, they are also considered. Pruning such "irrelevant" paths accelerates the model evaluation significantly. In fact, when combining multiple models into a common state space for parallel protein family HMM evaluation, and applying pruning techniques, significant parts of complete models can be skipped for evaluation.

Efficient model evaluation techniques generally address the third basic issue relevant for the successful application of enhanced probabilistic models for protein sequence analysis (cf. page 92). In this thesis it is considered for all parts of the developments.

Overview of the Concepts

The basic concepts for general improvements of HMM based sequence analysis which are developed in this thesis are graphically summarized in figure 4.6. The fundamental approach for improved classification performance of protein family HMMs, namely the feature based sequence representation, is illustrated in the upper frame. For all considered sequences of amino acids, relevant feature vectors based on biochemical properties of the sequences are extracted. The resulting high-dimensional and principally continuous feature space is represented using a mixture density (from left to right).

Based on the new feature representation, semi-continuous Profile HMMs are developed which is shown in the next frame (second from top). The model topology of Profile HMMs is kept fixed while substituting the discrete emissions with semi-continuous values obtained using the mixture density representation of the feature space and the feature vectors extracted from the appropriate protein sequences.

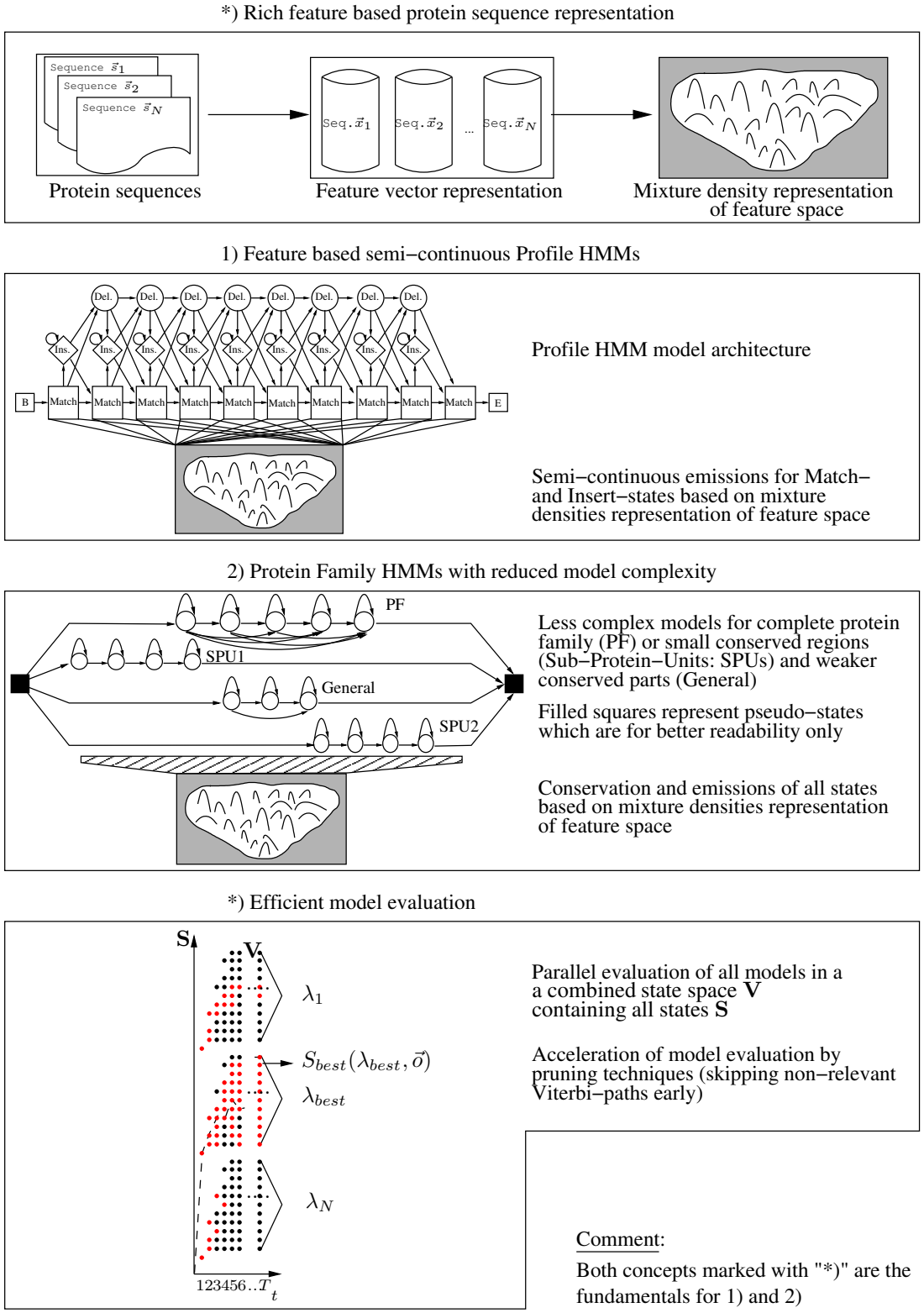


Figure 4.6: Concepts for improved HMM based sequence analysis developed in this thesis (see text for explanation).

In the third frame models with reduced complexity either for the whole protein family of interest (PF) or for Sub-Protein Units (SPU1 and SPU2) and weaker conserved parts of the protein family (General) are shown. All models are based on, and only possible due to the new feature representation. Generally, variants of a certain protein family or parts of it are possible which are evaluated in parallel. The particular models are, therefore, conceptually combined using so-called pseudo-states (filled squares). These pseudo-states gather the transitions from any variant to any other.

The lower frame illustrates the overall acceleration of the model evaluation using a combined state space and general pruning techniques. The states of three exemplary models are combined in the common state space and during evaluation irrelevant paths (black circles) are skipped for further analysis. Only the states marked red will actually be evaluated.

4.3 Summary

In the previous sections the performance of state-of-the-art discrete Profile HMMs for superfamily modeling was evaluated. It could be seen that remote homology analysis is still a challenging task. The classification error of almost 33 percent is problematic for target validation and the detection performance of the models is insufficient, too. The focus of the assessment was on the evaluation of the general method. Iterative model estimation techniques (cf. SAM's target98 approach in [Kar98]) are based on this approach. Thus, if the performance of general Profile HMMs could be improved, iterative techniques will benefit from the new methods, too.

Following the assessment of the capabilities of state-of-the-art Profile HMM techniques, concepts for improved HMM based sequence analysis were presented. Three basic issues relevant for such improvements are addressed by these concepts:

1. General Performance Improvement,
2. Sparse Data Problem, and
3. Efficient Model Evaluation.

Based on a richer feature representation of the protein sequences, semi-continuous Profile HMMs were proposed. Compared to discrete Profile HMMs, their semi-continuous counterparts capture the biochemical properties of residues in their local neighborhood much better. Additionally, approaches for less complex model architectures were presented allowing robust protein family model estimation even when using only small training sets. Because of the major importance for high-throughput screening tasks of large databases the acceleration of the model evaluation at the algorithmic level was proposed. By means of a parallel model evaluation scheme in combination with state space pruning techniques fast model evaluation will become possible.

In this chapter the concepts for improved HMM based protein sequence analysis were presented in general. According to these concepts in the following chapter approaches and methods developed are presented in detail. All described methods were implemented in a prototypical HMM framework (the GRAS^{2}P system; cf. [Plö02]). By means of this toolkit the effectiveness of the new approaches were evaluated using representative sequence analysis tasks. The results of this experimental evaluations will be presented in chapter 6.

5 Advanced Probabilistic Models for Protein Families

Common practical experience of molecular biologists with regard to the effectiveness of even the most sophisticated state-of-the-art probabilistic sequence analysis techniques leads to the conclusion that current approaches addressing remote homology detection tasks as summarized in chapter 3 have reached their limits. The results of an experimental evaluation for a representative task of sequence comparison using Profile HMM based stochastic models of protein families (cf. section 4.1) demonstrate that improved techniques are required.

The goal of this thesis is the development of enhanced probabilistic methods for general improvements of sequence analysis results. Currently, the development of the most promising approaches is very goal oriented, which means that several concepts are almost exclusively influenced by the actual biological task. One prominent example is the complex Profile HMM architecture including three different kinds of states which reflects the traditional sequence alignment including insertions, deletions, and substitutions of sequence residues. Another example is the model regularization using mostly manually designed background distributions (cf. Dirichlet mixture based regularization in section 3.2.2 on page 59). Unfortunately, the generalization capabilities of such techniques, i.e. the effectiveness for obtaining really new knowledge regarding protein relationships, are apparently limited (cf. section 4.1). Since protein sequence analysis can generally be understood as a pattern recognition problem where more or less modified occurrences of patterns need to be assigned to the correct classes, the approaches of this thesis address the incorporation and adoption of general pattern recognition techniques into the bioinformatics domain. Due to a more abstract view at the protein sequence analysis tasks during modeling, enhanced probabilistic models for protein families become possible.

The general concepts for enhanced protein family models were presented in the previous chapter. Based on this, detailed explanations of the newly developed techniques are given in this chapter addressing the three basic issues as formulated in section 4.2:

1. General Performance Improvement,
2. Sparse Data Problem, and
3. Efficient Model Evaluation.

According to the concepts developed, this chapter is organized as follows: The fundamental idea for all developments in this thesis is a feature based protein sequence representation. In section 5.1 the approach for obtaining a rich sequence representation is presented. Based on this, section 5.2 deals with techniques for robust parameter estimation for feature based Profile Hidden Markov Models and new application concepts of such enhanced models. Following this and based on the feature representation of protein sequences, in section

5.3 the complex model architecture of Profile HMMs is discarded and replaced by topologies with reduced complexity. The focus is on an alternative model architecture, namely Bounded Left-Right models. Furthermore, a new concept of protein family modeling using small building blocks, so-called Sub-Protein Units extracted automatically and in an unsupervised manner from training samples, is discussed. Approaches for a general acceleration of the evaluation of protein family HMMs are presented in section 5.4. The assignment of the new developments to the three basic issues relevant for the successful application of new protein family modeling approaches is given at the appropriate passages in the text.

Descriptions of some of the new approaches developed for enhanced probabilistic protein family modeling can also be found in [Plö04].

5.1 Feature Extraction from Protein Sequences

As an example for protein grouping in (super-)families, or folds, the family of *Kinases* contains enzymes that transfer phosphate groups from, e.g. *Adenosine triphosphate (ATP)* to a specified substrate or target.[1] In fact such family affiliations are caused by functional similarities of particular proteins (here the phosphate groups transfer). Generally, the biological function of a protein is determined by its three-dimensional structure which is mainly influenced by the underlying linear sequence of amino acids. According to this well-known and accepted theory, protein sequence analysis is performed using primary structure data.

However, a large amount of proteins exists which are functionally similar but whose similarities at the primary-structure level are rather weak. This phenomenon can be observed at almost every level of granularity when analyzing protein data. Even when considering the smallest functional units which are usually modeled by Profile HMMs, namely protein domains, sequence similarity is not always given for the whole chain of amino acids. In fact, these remote homologues are problematic for the majority of current sequence analysis techniques.

In order to generally improve the performance of current probabilistic sequence analysis techniques which was defined as the first basic issue for enhanced HMM based protein family modeling (cf. section 4.2), the central aspect of the developments described in this thesis is the explicit consideration of biochemical properties of the protein families during modeling. Compared to the standard description of protein data using their underlying chains of amino acids, a rich feature based protein data representation is used for establishing stochastic models of protein families. In the following, this feature extraction approach is presented.

5.1.1 Rich Signal-Like Protein Sequence Representation

In section 2.2 the biochemical composition of proteins was described. Proteins are macromolecules composed by 20 standard amino acids. The amino acids differ by the side chain attached to the C_α atoms. Due to these different side chains, the amino acids themselves vary in their biochemical properties like hydrophobicity, electric charge etc. The biochemical

[1] This process is termed “phosphorylation”. Typically, the target is “activated” or “energized” by being phosphorylated (cf. [Str91, p.368f]).

properties of protein domains and furthermore of the complete protein is determined via the local combination of the residues' properties.

By means of the symbolic description of protein data using amino acid sequences the general biochemical properties of the residues are *summarized*. Every amino acid contains its specific biochemical characteristics. These numerous specialties are only implicitly considered by discriminating between the 20 standard amino acids. However, neither local contextual relationships between residues nor specific mutual relationships between actual biochemical properties, which might be important for the overall function of the protein (domain), are considered.[2] It seems unrealistic that all biochemical properties change radically from one residue to another which is suggested when different amino acid symbols are adjacent. Instead, less abrupt changes and especially higher level mutual relationships between different biochemical properties are expectable.[3] One hypothetical example could be that from one residue to the next the hydrophobicity does not change substantially while the electric charge does.

In these premises, an alternative protein sequence representation is developed which is based on protein specific primary structure data and general knowledge about biochemical properties of amino acids. Biologically meaningful biochemical properties of residues are explicitly considered.

Protein Sequence Encoding

Basically, the biochemical characteristics of amino acids were well investigated throughout the years. Numerous researchers performed countless wet-lab experiments measuring various properties of the standard amino acids. In the literature some general sequence analysis techniques were described exploiting selected individual biochemical properties (cf. section 3.3.1 on page 75). However, most of such techniques use biochemical properties for more or less technical reasons (like applying spectral analysis based on numerical sequence mapping) as alternative but completely equivalent representation compared to the conventional amino acid sequences. Mostly, no *further* information is incorporated into the protein data representation since the 20 standard amino acids are mapped to 20 (or slightly less) different numerical values.

Basically, it is impossible even for very experienced molecular biologists to determine a single biochemical property being *exclusively* responsible for protein family affiliations. Instead, *multiple* properties are responsible for the biological function of proteins and thus their family affiliation. Certainly hydrophobicity, electric charge, and residue size are rather important for the three-dimensional structure and the surface properties of proteins. Additionally, other biochemical properties are important, too.

Biochemical properties of amino acids are usually collected in so-called amino acid indices. Such indices contain mapping rules for every amino acid to numerical values repre-

[2] Profile HMMs for protein (domain) families cover *global* contextual relationships of the residues via the model architecture. However, the emissions are determined completely neglecting any residual neighborhood.

[3] For standard pairwise sequence analysis techniques substitution groups are implicitly defined using substitution matrices which alleviates the abrupt character of symbol changes by assigning similar scores to similar amino acids.

senting the appropriate biochemical property. By exploiting these indices, amino acid sequences are numerically encoded resulting in "signal"-like representations. As previously mentioned (section 3.3.1 on page 75) Shuichi Kawashima and Minoru Kanehisa compiled a large amount of such amino acid indices [Kaw00]. For a rich protein sequence representation, in the new approach presented here, the particular amino acids are mapped to *multiple* properties carefully selected from the almost 500 indices. The selection was performed in cooperation with several biologists, i.e. incorporating expert knowledge. It turned out that 35 indices are sufficient for describing the biochemical properties of amino acids which are assumed most relevant for family affiliation. The authors of the amino acid indices compilation additionally performed a clustering of their indices regarding certain categories of amino acid properties:

- Hydrophobicity indices,
- Composition indices,
- Indices covering α and turn propensities,
- β propensity indices,
- Indices for physiochemical properties, and
- Indices covering other properties.

The 35 indices selected for the biochemical property based sequence representation developed in this thesis cover these clusters reasonably. A more detailed description of the amino acid indices used for sequence mapping can be found in appendix C.

As the result of the mapping procedure, protein sequences are encoded into multi-channel signal representations, i.e. given the linear chain of amino acids for a particular protein, its biochemical properties are represented in a 35-channel signal of the same length. Note that all components of the resulting feature vectors, i.e. the numerical mappings for all channels, are normalized to the range $[-1 \ldots 1]$ which is mandatory for further processing.

The general process of representation change by amino acid mapping is graphically summarized in figure 5.1. It is the base for all further developments, namely the extraction of relevant features describing the essentials of protein sequences.

Analysis of Local Neighborhoods

The previously described sequence encoding method subsumes information regarding biochemical properties of the amino acids from various sources in a multi-channel representation. Generally, when using state-of-the-art Profile HMMs for protein family modeling, two levels of residual context are considered. The classification result determining the decision regarding the probable affiliation of the sequence analyzed to a particular protein family is performed using the complete sequence, i.e. the complete residual neighborhood. Thus, the global context is captured by the HMM. Contrary to this, for the estimation of the emission probabilities no residual context is used at all.

Neglecting any residual context for the estimation of the HMM state emissions seems rather crucial since the biochemical characteristics certainly do not abruptly change from

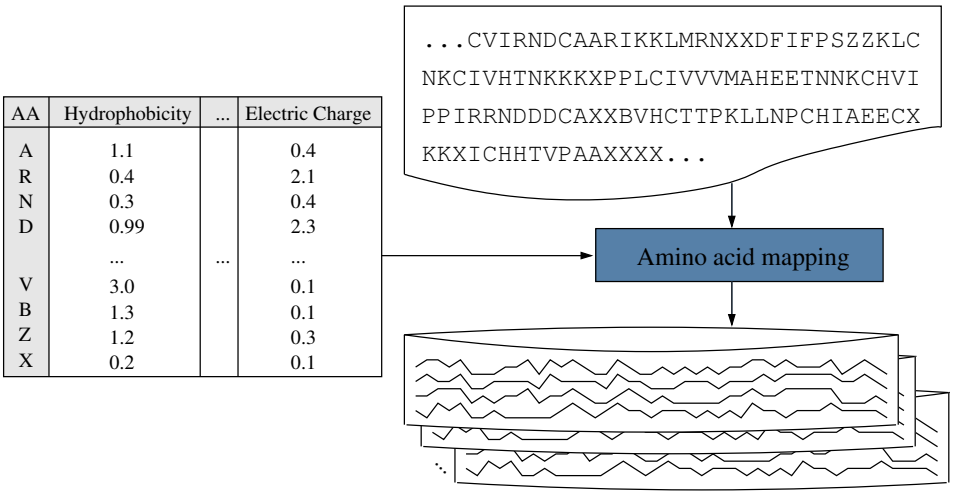


Figure 5.1: Overview of the approach for obtaining a biochemical property based protein sequence representation: Based on protein specific primary structure data (sequence at the upper part) and general knowledge about relevant biochemical properties of amino acids (left-hand side) the new representation is obtained by mapping the biochemical properties of residues to *multiple* numerical values (lower part). Note that all sequence and biochemical property data is hypothetical not corresponding to reality and for illustration purposes only.

one residue to another. Instead, they are dependent on their local environment since adjacent residues severely interfere with each other. As one example the electric charge of a residue is dependent on the electric charge of its immediate environment. Since amino acid indices like the symbolic description of amino acids themselves do not respect the residues' environment, these local characteristics are captured alternatively in the new approach for HMM state emissions.

In order to respect *local* signal characteristics already at the level of emission probabilities, in the feature extraction procedure local contexts of residues are considered. The emissions estimated on the base of the local neighborhoods of a particular residue contain much more information since they cover the residues' environment. Note that the global context, i.e. the structure of the protein family data, is still captured by the appropriate HMM. Generally, the HMM now describes the structure of protein data at the base of residues in their local neighborhood. In the new approach consecutive samples of the 35 channel signals are analyzed using a sliding window technique (extracting *frames*). These frames are used for short length signal analysis.

There are two general parameters for configuring the sliding window based context analysis. First, the length of the context analyzed for emission estimation needs to be determined. Certainly, it is dependent on the actual data analyzed. Thus, it generally needs to be treated as a parameter to be learned during training. Unfortunately, especially for HMM based modeling there are no techniques available for such parameter training. However, in informal experiments a fixed window size of 16 could be determined heuristically which is suitable for the majority of protein data. Thus, it was kept fixed for the general procedure. The sliding window containing the context of a particular residue is (almost) symmetrically organized, i.e. the context of an amino acid consisting of seven neighbors to the left and eight neighbors to the right is considered.

The second parameter determines the actual treatment of the sequence borders. Due to the (almost) centered context of a residue, at the beginning and at the end of a sequence the sliding window cannot be filled using real data – because in fact there is no data. This border problem is also known from general signal filtering approaches, e.g. within image processing applications when convolving images with specialized denoising filters. Since all residues have a special meaning for the particular protein, the borders cannot be skipped (as one common solution for image processing applications suggests) but three general "solutions" exist:

1. Zero-padding: The borders are filled with 0.
2. Distinct values-padding: The borders are filled with distinct values, optionally discriminating between beginning and end of the sequence.
3. Prior-padding: The sequence is extended at the beginning as well as at the end using prior knowledge about the general distribution of amino acids (i.e. filled with the amino acid wildcard 'X').

Note that all three options incorporate artificial, i.e. potentially erroneous, data into the protein sequence. Since the first two options seem to distort the sequence data too much, in the actual feature extraction approach the borders are treated using prior-padding. Generally, when modeling protein families the influence of the borders is not negligible but also not dramatic.

In figure 5.2 the sliding window approach, which is the prerequisite for the analysis of the residues' local neighborhoods, is schematically illustrated.

5.1.2 Feature Extraction by Abstraction

The plain multi-channel signal-like representation of residues in their local neighborhood as described in the previous sections is already a rich base for enhanced protein family models. They could be used as features for emission probability estimation for Profile HMMs. However, for remote homology detection the biochemical *essentials* of a particular protein family are of major interest. Thus, for good generalization, any putatively misguiding signal specialties in any of the channels encoding particular biochemical properties, relevant only for a minority of sequences belonging to the family of interest should be neglected.

In general pattern recognition applications, putatively misguiding characteristics of arbitrary signals evolving in time are usually identified and discarded by means of signal processing techniques. If a coarse but meaningful general shape of such a signal was desired, e.g. specific frequencies causing signal distortions will be eliminated. Therefore, the signal of interest is usually transformed into a more convenient but equivalent frequency representation. Generally, the standard transformation for signal analysis applications is the well-known Fourier transformation providing a frequency coefficients based signal representation. Every coefficient of the resulting spectrum represents a certain frequency part of the original signal. Once the Fourier coefficients are estimated, further frequency based signal analysis and thus filtering can be performed very easily by explicitly changing particular coefficients which results in the desired essentials of the signal processed. The spectral signal representation is completely equivalent to the original time-series based representation

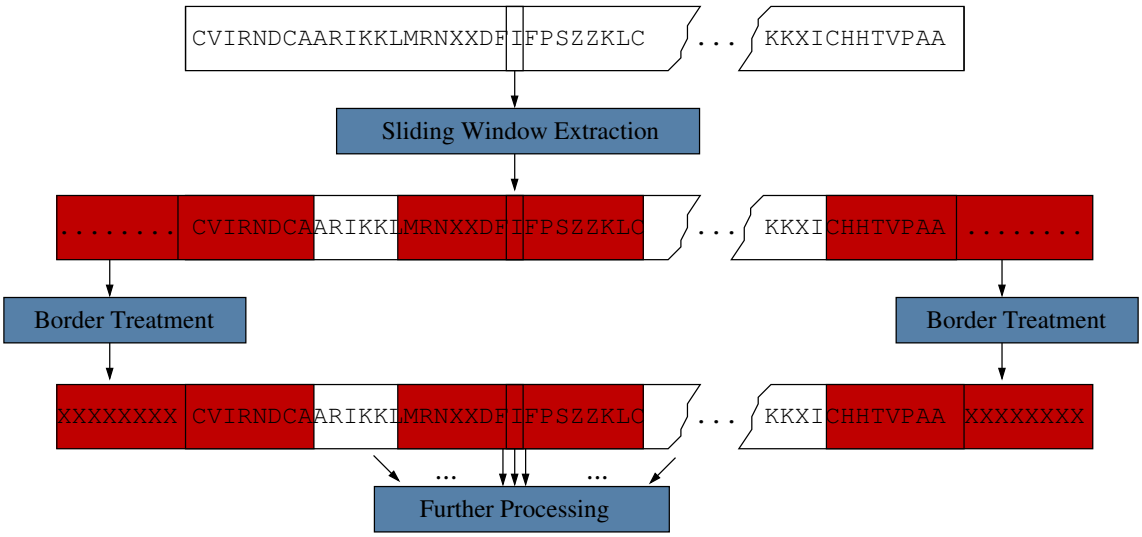


Figure 5.2: Sketch of the sliding window approach for the analysis of residues' local neighborhoods: For every residue of a protein sequence (upper part) the biochemical properties are analyzed with respect to the 15 surrounding amino acids captured by a sliding window which are illustrated by means of rectangles filled red (middle part). At the borders of the sequence, the windows are filled using prior knowledge about general amino acid distributions (lower part, left and right border). The filled windows are subject to further processing, i.e. multi-channel amino acid mapping and feature extraction.

if all coefficients are considered. This means, that the original signal can be reconstructed from the Fourier coefficients if all of them are used for the inverse function transformation.

Interpreting the multi-channel numerical protein sequence representation of a residue in its local neighborhood as a general signal evolving in time[4], the coarse shape of this "protein signal" can generally be obtained as described above for arbitrary signals. In fact, due to e.g. the elimination of specific "frequencies" of the residual context signal, putatively misguiding peaks, i.e. biochemical properties which are not relevant for the protein in general, can be eliminated. The resulting coarse shape of the protein signal will represent the biochemical essentials of the underlying actual protein. Note that the signal analysis is performed channel-wise, i.e. every single channel encoding a specific biochemical property is analyzed separately.

While analyzing signals of protein sequences it turned out that the standard spectral analysis approach using the Fourier transformation as described above is not suitable for biological signals subsumed in the frames covering the biochemical properties of residues' neighborhoods. This function transformation assumes periodic signals of infinite length which is in no way the case for the numerical representation of the biochemical properties of 16 adjacent amino acids. Furthermore, due to the rather artificial signal interpretation of the protein data, *explicit* signal analysis, e.g. the manual identification of specific frequencies not characteristic for the protein essentials, can hardly be performed.

Instead, abstraction from detailed signal shapes needs to be guided implicitly by the fre-

[4]Conceptually, here time is substituted by the amino acids' positions within the protein sequence, i.e. here within the context window, as usual for Profile HMM based sequence processing.

quency transformation used for signal analysis. A frequency transformation is required, which does not assume any signal specialties which are not fulfilled and which allows easy signal abstraction by e.g. discarding the first or the last k coefficients according to the amount of information they represent with respect to the original signal. A transformation which almost perfectly fulfills these constraints is the *Discrete Wavelet Transformation (DWT)*. After Wavelet transformation the coefficients are ordered with respect to the amount of information they particularly contain which is one reason for the better suitability of DWT compared to the traditional Fourier transformation. A detailed discussion of the DWT is far beyond the scope of this thesis. However, in appendix A the basic theory and some necessary essentials regarding Wavelets are described.

As usual for function transformations, the signal representation using all Wavelet coefficients is completely equivalent to the original time-series based signal – the original signal can be reconstructed without distortion. In addition to the previously mentioned improved suitability for protein signals, Wavelets are rather convenient for obtaining more abstract signals. In a Wavelet based signal representation, the coefficients are ordered according to their importance for the complete signal analyzed. First, the approximation coefficients represent the most relevant information necessary for the reconstruction of the original signal. Following these, detail coefficients describe signal specialties which generally do not contribute substantially to the coarse shape of the signal. Note that these detail coefficients do not necessarily correspond to a specific kind of frequencies, e.g. high frequencies.

By means of a Wavelet representation, the coarse shapes of the protein signals in any of the channels covering specific biochemical properties can easily by extracted by skipping a reasonable number of detail coefficients. For the extraction of relevant features from protein signals, a two-level Wavelet decomposition using second order Daubechies filter (length four) is performed, i.e. in addition to the basic decomposition of the signal, the approximation coefficients obtained are further decomposed which corresponds to a two scale analysis (cf. appendix A on pages 197ff). The actual parameterization of the feature extraction approach could be obtained in various informal experiments. Since pure signal *analysis* is performed (compared to signal *detection*), the actual choice of the Wavelet and scaling filter pair is of minor importance and a standard filter pair is used. Skipping the upper *five* coefficients is straightforward when inspecting the results with respect to the analysis of their average energy which is usually very low for all channels. The remaining coefficients substantially vary for different proteins and channels and thus cover the main energy of the signals, i.e. the most relevant information contained in the signals. They will be used for further processing. In figure 5.3 the general Wavelet based signal decomposition including the abstraction by discarding the upper five detail coefficients is graphically illustrated for one channel of one frame extracted from a hypothetical protein sequence.

Applying the above described feature extraction method, residues of protein sequences are represented as coarse shapes of signals obtained by multi-channel numerical encoding and signal abstraction using DWT. The 11 Wavelet coefficients for the 35 channels are concatenated to a 385-dimensional feature vector which represents the summary of relevant biochemical properties of a residue in its local neighborhood.

Processing such high-dimensional feature vectors is not feasible for remote homology detection applications where usually rather small training sets are available. *Robust* models

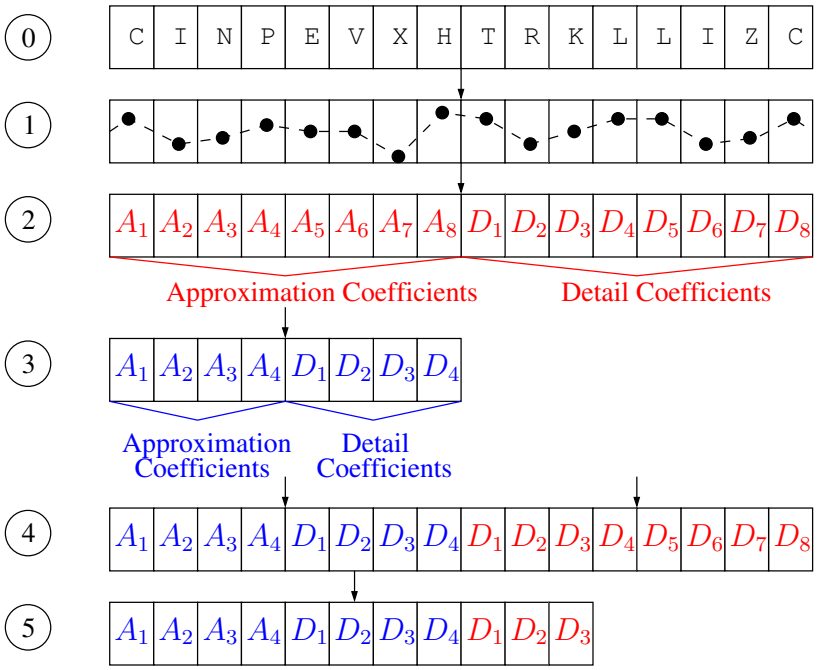


Figure 5.3: Illustration of a principle Wavelet based signal decomposition as performed for every channel of frames extracted from protein sequences: 0 + 1) illustration of one frame from a hypothetical protein sequence including *one* channel of the numerical encoding; 2) result of the first stage of two scale Wavelet analysis resulting in approximation and detail coefficients; 3) result of the further decomposition of the first level approximation coefficients; 4) concatenation of first and second stage coefficients; 5) final frame representation using eleven of 16 Wavelet coefficients.

based on a 385-dimensional feature representation can hardly be estimated. Thus, further reasonable dimension reduction needs to be performed.

Skipping the upper five Wavelet coefficients was quite straightforward. By means of informal experimental evaluations it turned out that the neighborhood of 16 residues can be described properly using eleven coefficients. Further reduction of the dimensionality could not be performed in this way. However, despite careful selection of the 35 channels actually used for the description of relevant amino acids' biochemical properties, redundancies within the multi-channel representation are more than expectable. Furthermore, since the sliding window approach is performed using an overlap of 15/16 of the frame size, redundancies are expectable here, too. Thus, further dimension reduction by automatic decorrelation seems possible without loosing too much information describing the essentials of protein data.

The standard procedure for automatic decorrelation and dimension reduction is applying the *Principle Components Analysis (PCA)*. Here, an N-dimensional feature space is projected onto an M-dimensional subspace ($M \ll N$) which covers the majority of data variance. This subspace is spanned by the M eigenvectors corresponding to the largest M eigenvalues of the covariance matrix of the sample data (the relevant theory of PCA is briefly summarized in appendix B). In these premises, the 385-dimensional feature vectors are projected onto their eigenspace. Inspecting the eigenvalue spectrum of the component wise normalized data, it becomes clear that a compact representation in a 99-dimensional subspace is sufficient. Note that this *substantial* dimension reduction is only possible when

using the two scale Wavelet analysis. Informal experiments showed that a single stage decomposition is less effective.

The overall procedure for feature extraction from protein sequences is summarized in figure 5.4. Based on this feature based protein sequence representation all further modeling approaches are performed.

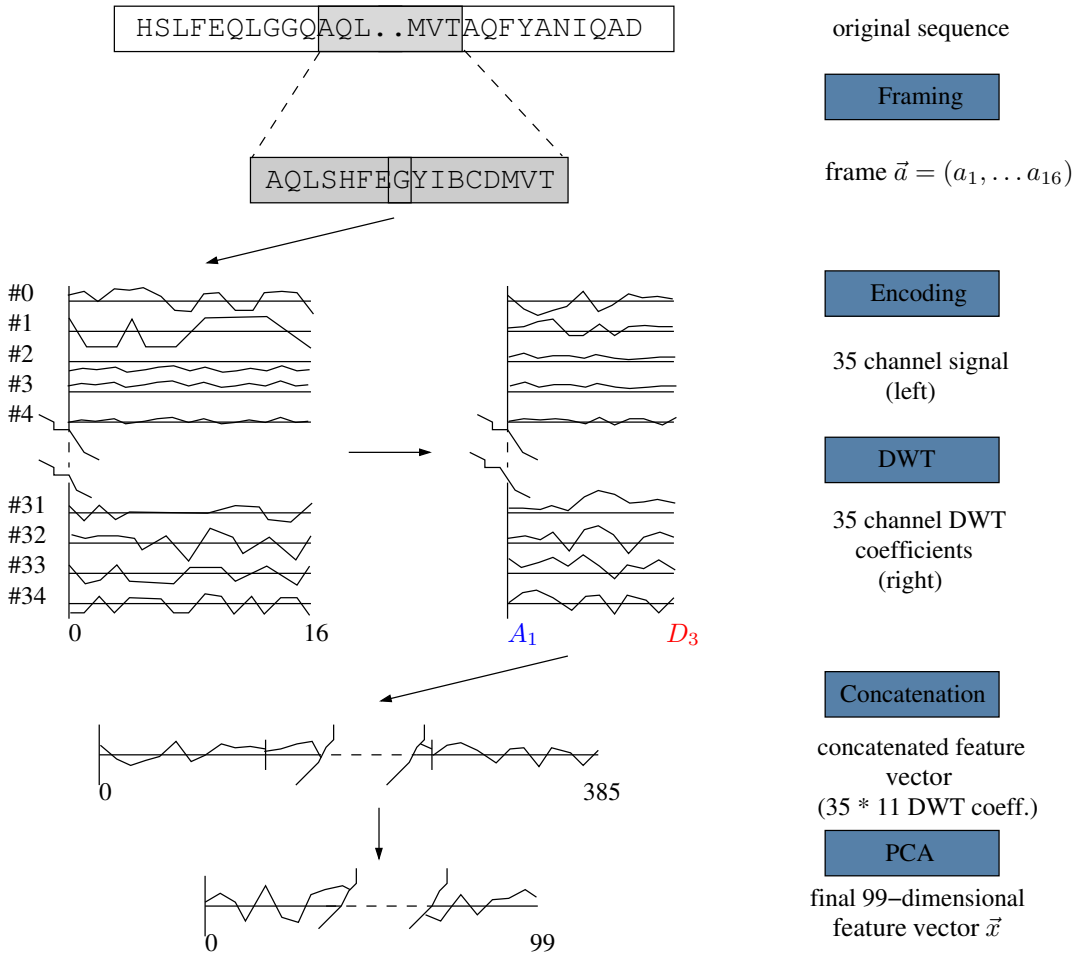


Figure 5.4: Overview of the feature extraction method for obtaining a rich protein sequence representation covering the most relevant biochemical properties: By means of a sliding window technique, frames $\vec{a} = (a_1, \ldots, a_{16})$ are extracted containing the 15 adjacent amino acids of a particular residue (upper part). Every frame $\vec{a}$ is mapped to a 35-channel numerical representation covering the most relevant biochemical properties of proteins (middle-left part). The coarse shapes of such "protein signals" are extracted by applying a Discrete Wavelet Transformation and skipping the upper five coefficients for signal abstraction (middle-right). The concatenation of the resulting 385 Wavelet coefficients is projected onto the 99-dimensional eigenspace resulting in the final feature vectors $\vec{x}$ (lower part).

5.2 Robust Feature Based Profile HMMs and Remote Homology Detection

In the previous section the new approach for obtaining a rich protein sequence representation based on feature extraction was presented. It is the central aspect of this thesis. In the following, by means of the new 99-dimensional vector representation of protein data, feature based Profile HMMs are developed. They represent a substantial enhancement of the basic method of protein family modeling using Profile Hidden Markov Models and serve as their replacements for general improvement of remote homology detection.

Compared to the 20 discrete standard amino acid symbols, the new feature representation of protein data corresponds to a 99-dimensional continuous feature space. When processing feature vectors, generally continuous instead of discrete HMMs are used (cf. section 3.2.2 on page 46). However, current Profile HMMs are defined for discrete data only. Thus, for the feature based sequence representation, Profile HMMs need to be modified for suitable processing of protein feature data. According to the general argumentation regarding proper model emission types, pure continuous models seem problematic because they require substantial amounts of training samples for robust model estimation. Every state includes its own individual feature space representation which covers the specifics of the feature vectors assigned to it. Especially for effective exploitation of small training sets, Xuedong Huang and colleagues proposed the concept of semi-continuous modeling where a common feature space representation is shared by all states of the particular model [Hua89]. The attractiveness of semi-continuous HMMs is also reasoned by the fact that a good estimation of the common feature space representation can be obtained in a separate estimation step using general feature data.[5] Since target specific data is not necessarily required for obtaining the general feature space representation, usually larger sample sets are available resulting in a more robust feature space representation. The separation of the estimation of a general feature space representation from position specific modeling is the basic advantage of semi-continuous HMMs which can be exploited especially for robust estimation of protein models using small sample sets. Note that the abovementioned separate model and feature space estimation is not mandatory for semi-continuous HMMs. Instead, both components can also be obtained from an integrated training procedure. However, the number of training samples required is substantially larger. Thus, semi-continuous Profile HMMs are developed by explicitly exploiting the separate feature space estimation.

In the following sections, first, an appropriate feature space representation using mixture components is presented. A general representation can be obtained using very effective standard estimation techniques. Following this, general semi-continuous Profile HMMs are introduced by means of the parametric description of the feature space. For robust protein family modeling, target family directed specialization is developed in section 5.2.3. Remote homology detection is improved using an explicit background model capturing all data not belonging to a particular protein family (section 5.2.4).

Corresponding to the three basic issues addressing improved probabilistic protein family models, the developments presented in this section are directed to general performance improvement and alleviating the sparse data problem (cf. section 4.2 on page 92).

[5] Note that *general* data denotes feature vectors which are not specifically assigned to a particular HMM state. However, the principle origin of all feature vectors used for the estimation of the feature space representation as well as for the specific modeling process needs to be identical (here: protein data).

5.2.1 Feature Space Representation

For discrete data the distribution of a random variable Y can generally be defined tabularly by assigning the probability $P(y)$ to every discrete event y, which Y can be assigned to.[6] This is the usual case for protein data where, given a set of sample sequences, the relative frequencies of every amino acid define the discrete distribution representing the data space of amino acids.

Due to their non-discrete character, the underlying probability density distributions of continuous data cannot be defined tabularly. Thus, parametric models are required for proper descriptions of continuous densities usually resulting in compact representations of the particular data spaces. According to the central limit theorem, most natural processes can be described using *normal distributions* if they were observed for reasonable time. Normal distributions $\mathcal{N}$ are mathematically very attractive since they contain only a small number of parameters for effective representation of unimodal densities:

$$\mathcal{N}(\vec{x}|\vec{\mu}, \mathbf{C}) = \frac{1}{|\sqrt{2\pi\mathbf{C}|}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu})^T \mathbf{C}^{-1}(\vec{x}-\vec{\mu})},$$

where $\vec{\mu}$ denotes the mean vector of the data and $\mathbf{C}$ the appropriate covariance matrix. General continuous density functions $p(\vec{x})$ can be approximated with arbitrary precision using linear combinations of normal distributions, so-called *mixture density models* [Yak70]. Usually, only a finite sum of K mixture components is used and the parameters of the model, namely the mixture weights, i.e. their prior probabilities c_i, the mean vectors $\vec{\mu}_i$, and the covariance matrices $\mathbf{C}_i$ of the particular normal densities, are summarized in a set of parameters θ:

$$p(\vec{x}|\theta) = \sum_{i=1}^{\infty} c_i \mathcal{N}(\vec{x}|\vec{\mu}_i, \mathbf{C}_i) \approx \sum_{i=1}^{K} c_i \mathcal{N}(\vec{x}|\vec{\mu}_i, \mathbf{C}_i).$$

Feature vectors extracted from protein sequences as introduced in the previous section *generally* represent continuous data. Certainly, they originate from discrete amino acid symbols but after their enrichment using biochemical property mapping and their signal-like treatment by applying Wavelet transformation and Principal Components Analysis to their local neighborhood, the character of the data becomes continuous. Thus, the 99-dimensional feature space is represented using mixture density distributions.

Estimation of the General Feature Space Representation

As previously mentioned, the general feature space representation can be estimated independently from the actual protein family modeling. In the approach presented here, mixture components are estimated by exploiting general protein data from major protein sequence databases. For the developments of this thesis, the SWISSPROT database [Boe03] was used for obtaining the general protein feature space representation. By means of its almost 90 000 sequences, 1024 normal densities are estimated which provide a sufficiently accurate feature space representation. The actual limitation to 1024 mixture components represents a

[6] The argumentation regarding the representation of general densities is mainly influenced by [Fin03, p.46f].

good compromise between suitably covering the general feature space and allowing further specialization as described later.

Due to this limitation the accuracy of the data space representation is strongly dependent on the actual choice of the mixture components, i.e. their "position" within the data space and their shape. For high-dimensional data these parameters correspond to mean vectors and covariance matrices of N-dimensional normal densities. Usually, they are estimated using the well-known *Expectation Maximization (EM)* procedure [Dem77]. By means of this iterative technique both mean vectors and covariance matrices of a predefined number of mixture components are estimated by maximizing the probability of training data $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T\}$ depending on the parameters of the mixture density model. Although widely used, the EM algorithm has severe drawbacks. First, it is in no way guaranteed that the global optimum for data space representation can be obtained. Instead, the local optimum which is closest to the initialization will be found. Thus, the quality of the solution is severely dependent on the initialization of the procedure. Especially random assignment of mean vectors and covariance matrices is problematic. Furthermore, most critically, the algorithm converges rather slowly, i.e. usually many iterations are required for estimating suitable mixture components. In figure 5.5 the EM algorithm for estimating mixture density models of probability distributions is summarized.

The EM algorithm is closely related to general vector quantization techniques where clusters within data spaces are searched. However, during clustering the data vectors are assigned deterministically to particular clusters compared to probabilistic assignment when using EM. Unfortunately, the EM algorithm is computationally expensive. Thus, instead of using EM in several practical applications a much faster procedure is applied for obtaining suitable parametric representations of high-dimensional continuous data spaces – a slightly modified version of the k-means vector quantization procedure which was originally developed by James MacQueen [Mac67].

The k-means algorithm is a non-iterative technique for estimating vector quantizers where based on a single-pass approach both cluster representatives, so-called prototypes, and the corresponding data space partition can be estimated very efficiently. In several informal experiments using different kinds of data vectors it turned out that the quality of the vector quantizers estimated using the k-means approach is at least comparable to those obtained when using iterative clustering techniques like the well-known Lloyd or LBG algorithms (cf. e.g. [Ger92, Fin03]). The crucial prerequisite for equivalent or even better vector quantizer design is the existence of reasonable numbers of sample data. This is the case when using general protein data from SWISSPROT. In figure 5.6 the standard k-means algorithm is summarized.

Once a vector quantizer has been obtained for the general data space using k-means, a mixture density can easily be estimated by exploiting the resulting prototypes as well as the partition of the data space. The prototypes of the clusters correspond to the mean vectors of the mixture components and the appropriate covariance matrices $\mathbf{C}_i$ can be calculated using the data vectors $\vec{x}$ assigned to particular cells of the data space partition (clusters) R_i:

$$\mathbf{C}_i = \sum_{\vec{x} \in R_i} (\vec{x} - \vec{\mu}_i)(\vec{x} - \vec{\mu}_i)^T.$$

Given a set of sample data $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T\}$ and the number K of desired base normal densities and a lower limit ΔL_{min} for the relative improvement of the likelihood function $L(\theta) = \ln P(\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T|\theta) = \sum_{\vec{x}\in X} \ln P(\vec{x}|\theta)$, where θ represents the parameter vector describing the mixture density model.

1. Initialization: Choose initial parameters $\theta^0 = (c_i^0, \vec{\mu}_i^0, \mathbf{C}_i^0)$ of the mixture density model, and initialize the iteration counter $m \leftarrow 0$.

2. Expectation Estimation (E-step): Determine estimates for the posterior probabilities of all mixture components ω_i (given the current model θ^m) for every data vector $\vec{x} \in X$:
$P(\omega_i|\vec{x}, \theta^m) = \frac{c_i^m \mathcal{N}(\vec{x}|\vec{\mu}_i^m, \mathbf{C}_i^m)}{\sum_j c_j \mathcal{N}(\vec{x}|\vec{\mu}_j^m, \mathbf{C}_j^m)}$.
Calculate the likelihood of the data given the current model θ^m:
$L(\theta^m) = \ln P(\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T|\theta^m) = \sum_{\vec{x}\in X} \ln \sum_j c_j^m \mathcal{N}(\vec{x}|\vec{\mu}_j^m, \mathbf{C}_j^m)$.

3. Maximization (M-step): Calculate updated parameters $\theta^{m+1} = (c_i^{m+1}, \vec{\mu}_i^{m+1}, \mathbf{C}_i^{m+1})$:

$$
\begin{aligned}
c_i^{m+1} &= \frac{\sum_{\vec{x}\in X} P(\omega_i|\vec{x}, \theta^m)}{|X|} \\
\vec{\mu}_i^{m+1} &= \frac{\sum_{\vec{x}\in X} P(\omega_i|\vec{x}, \theta^m)\vec{x}}{\sum_{\vec{x}\in X} P(\omega_i|\vec{x}, \theta^m)} \\
\mathbf{C}_i^{m+1} &= \frac{\sum_{\vec{x}\in X} P(\omega_i|\vec{x}, \theta^m)\vec{x}\vec{x}^T}{\sum_{\vec{x}\in X} P(\omega_i|\vec{x}, \theta^m)} - \vec{\mu}_i^{m+1}(\vec{\mu}_i^{m+1})^T.
\end{aligned}
$$

4. Termination: Calculate the relative change of the likelihood since the last iteration:
$\Delta L_m = \frac{L(\theta^m) - L(\theta^{m-1})}{L(\theta^m)}$.
If $\Delta L_m > \Delta L_{\text{min}}$ set $m \leftarrow m + 1$ and continue with step 2, terminate otherwise.

Figure 5.5: EM algorithm for estimating mixture density models (adopted from [Fin03, p.64], cf. also [Dem77, Ger92]).

Without further optimization a high quality representation of the biochemical feature space created for protein sequence data can be obtained which is used as starting point for feature based protein family HMMs.

In figure 5.7 the process of mixture density estimation for feature space representation using general protein data is summarized graphically.

Given a set of sample data $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T\}$ and the number K of desired clusters.

1. Initialization: Choose the first K vectors of the sample set as initial cluster prototypes $Y^0 = \{\vec{y}_1, \vec{y}_2, \ldots, \vec{y}_K\} = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_K\}$. Alternatively choose K data vectors randomly distributed over the sample set. Initialize the partition of the data set to $R_i^0 = \{y_i\}, i = 1, \ldots, K$.
Initialize counter $m \leftarrow 0$.

2. Iteration: For all data vectors $\vec{x}_t, K+1 < t < N$ not processed so far:

(a) Classification: Determine the optimal prototype $\vec{y}_i^m$ within the current set of cluster prototypes Y^m for $\vec{x}_t$ using some metric d (usually Euclidean distance, cf. equation 3.1 on page 22):
$\vec{y}_i^m = \underset{\vec{y} \in Y^m}{\operatorname{argmin}}\, d(\vec{x}_t, \vec{y})$.

(b) Re-partitioning: Change the partition of the data space by updating the cluster R_i belonging to the previously determined prototype $\vec{y}_i$:
$$R_j^{m+1} = \begin{cases} R_j^m \cup \{\vec{x}_t\}, & \text{if } j = i \\ R_j^m, & \text{otherwise.} \end{cases}$$

(c) Prototypes Update: Update the prototype $\vec{y}_i$ of the cluster which was changed in the previous step and keep all others:
$$\vec{y}_j^{m+1} = \begin{cases} \text{cent}(R_j^{m+1}), & \text{if } j = i \\ y_j^m, & \text{otherwise,} \end{cases}$$
where $\text{cent}(R_j^{m+1})$ designates the centroid of the current j-th cluster:
$\text{cent}(R_j^m) = \underset{\vec{y} \in R_j^m}{\operatorname{argmin}}\, \mathcal{E}\{d(\vec{x}, \vec{y}) | \vec{x} \in R_j^m\}$
which corresponds to:
$\text{cent}(R_j^m) = \mathcal{E}\{\vec{x} | \vec{x} \in R_j^m\} = \int_{R_j^m} \vec{x} p(\vec{x}) d\vec{x}$
when using elliptical symmetrical metrics like e.g. the Euclidean distance.

Update counter $m \leftarrow m + 1$.

Figure 5.6: k-means algorithm for vector quantization (adopted from [Fin03, p.61], cf. also [Mac67, Ger92]).

5.2.2 General Semi-Continuous Profile HMMs

When processing feature vectors, usually continuous instead of discrete HMMs are used. The emissions of HMM states are based on parametric representations of the underlying high-dimensional feature space. Linear combinations of normal densities are a suitable choice for easy mathematical treatment since the number of parameters required for a proper description is reasonable small (cf. section 3.2.2 on pages 46ff).

For the general case of continuous modeling, every state of such HMMs consists of its own set of mixture components. The major advantage of such a continuous modeling approach is the estimation of state-specific mixture components which allows sharp spe-

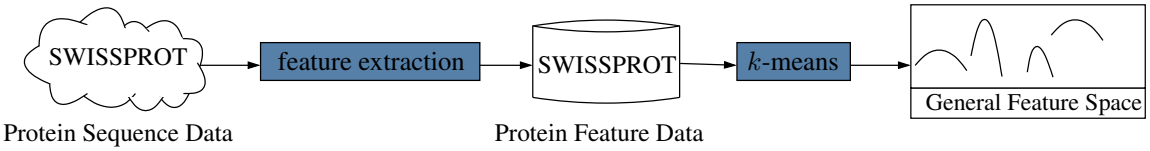


Figure 5.7: Estimation of mixture density for feature space representation: Using feature data extracted from general SWISSPROT protein sequences (left part), the k-means algorithm is applied for estimating the mixture density used (right part).

cialization individually for particular positions within the HMM. However, especially for HMMs consisting of complex model architectures and thus large amounts of states, the overall number of normal densities which are to be estimated is usually rather high. In order to obtain robust estimations of the mixture density, for all of these mixture components state specific training data is required. Unfortunately, for typical practical problems usually only small training sets are available (cf. the second issue relevant for the application of probabilistic protein families – the sparse data problem – in section 4.2 on page 92).

Since the amount of state-specific training samples is typically extremely small when modeling protein families using Profile HMMs, continuous emission modeling is out of question for feature based protein sequence representations. Instead, the semi-continuous modeling approach of Xuedong Huang and colleagues (cf. pages 46ff) is very attractive for feature based Profile HMMs. The training sets are effectively exploited by sharing a common set of normal densities between all HMM states. The state-dependent specialization is reached by the individual prior probabilities for the mixture components. Furthermore, as mentioned earlier one basic advantage of semi-continuous modeling is especially the principal possibility of dividing the model estimation process into two steps, namely obtaining the feature space representation and the actual model optimization which requires less *model specific* training data.

Thus, enhanced Profile HMMs which are based on continuous feature vector representations capturing the biochemical properties of amino acids in their local sequential neighborhood are of semi-continuous type. As described in the previous section, the general parametric feature space representation is efficiently determined by applying a modified version of the k-means algorithm for vector quantization to general protein data (approximately 90 000 sequences). Semi-continuous Profile HMMs themselves are derived from the architecture of discrete models, i.e. the general model architecture is kept fixed. Given the Profile structure, standard Viterbi training is performed using the component densities of the general feature space representation and small amounts of family specific data.

By means of this procedure, enhanced Profile HMMs for robust protein family modeling are estimated using SWISSPROT protein data for the general feature space representation (hence the name *general* semi-continuous Profile HMMs) and small amounts of protein family specific data. In figure 5.8 the new enhanced Profile HMMs are illustrated. The general model architecture is kept fixed whereas the emissions of Insert and Match states are now based on the underlying common mixture density.

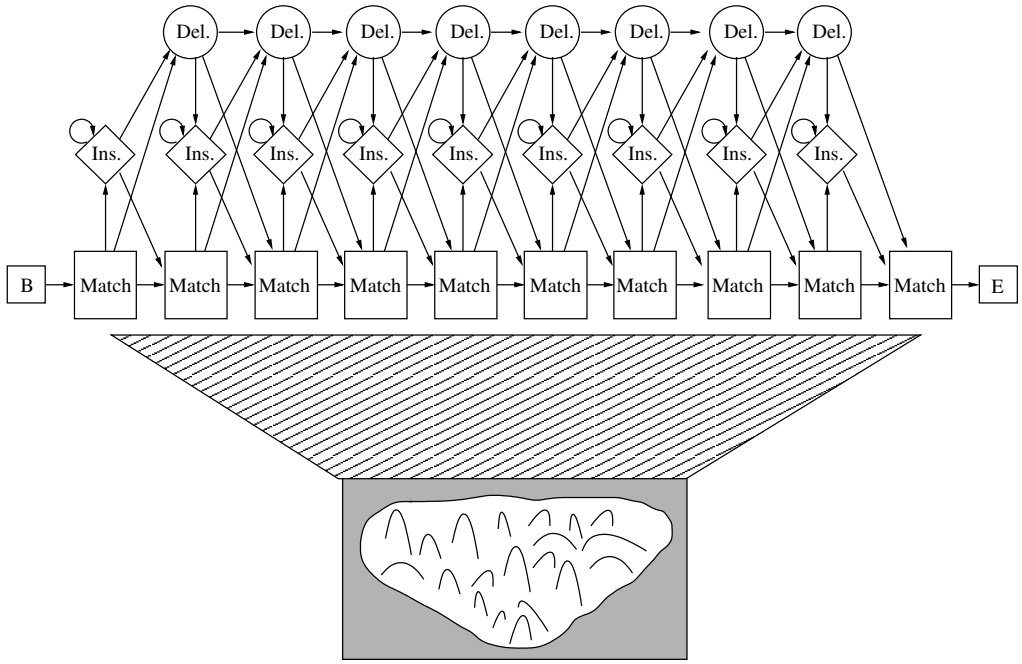


Figure 5.8: Semi-continuous feature based Profile HMMs for robust protein family modeling: The original model architecture of Profile HMMs is kept fixed (upper part). The emissions of both Insert and Match states are now based on a general mixture density representation of the new underlying feature space covering the essential biochemical properties of residues in their local neighborhood (lower part).

5.2.3 Specialization by Adaptation

The mixture density representation of the feature space obtained from SWISSPROT captures the global properties of general feature data. As mentioned earlier, family specific specialization of the general feature space representation is implicitly performed by estimating the state-dependent prior probabilities for all mixture components. In the previous section, semi-continuous Profile HMMs were developed by means of this concept. They are very efficient for robust protein family modeling if only little target specific data is available. However, mixture densities specifically estimated for particular protein family models, i.e. their HMM states, do *principally* better represent the specialties of the feature space covered by this family since the specialized normal densities are more focused on target data.

In order to explicitly focus the general feature space representation to the essentials relevant for a particular protein family, two general possibilities exist: Either family specific mixture density estimation is performed which is problematic when considering the sparse data situation as argumented above, or family specific mixture density *adaptation* is performed. Adaptation means explicitly changing the general mixture components, i.e. the mean vector $\vec{\mu}_i$ and covariance matrices $\mathbf{C}_i$, with respect to the (small amounts) of target specific data. Depending on the number of actually available family specific data, three different kinds of mixture adaptation can be applied:

1. The complete re-estimation of all family model parameters using the *Maximum Likelihood (ML)* approach,

2. The re-estimation of the mixture density with respect to optimization of the posterior probabilities of the mixture parameters for the adaptation samples – *Maximum A-Posteriori (MAP)* adaptation, or

3. The rotation and translation of the general feature space towards the family specific essentials by estimating affine transformations using the *Maximum Likelihood Linear Regression (MLLR)* approach on small adaptation sets.

In figure 5.9 the principle idea of mixture density adaptation is graphically summarized. Given a mixture density based representation of the general feature space, specialization techniques are applied in order to focus the general representation on the essentials relevant for a particular protein family. Both mixture density representations are generally similar but not identical which is symbolized in the figure by the sketched distortion of the distributions.

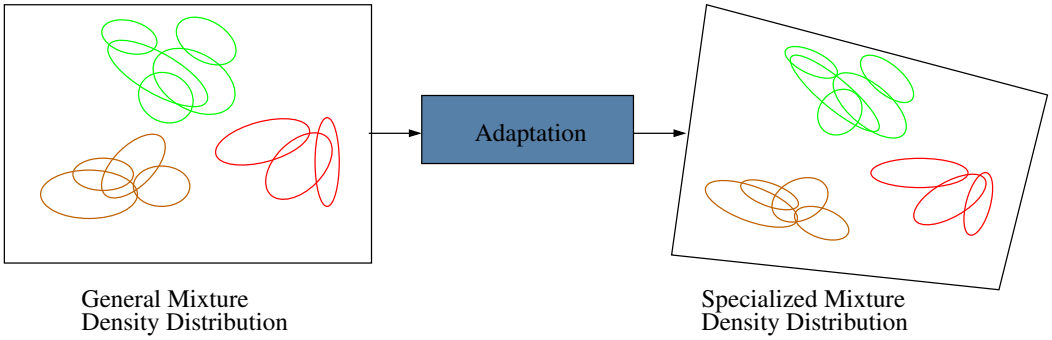


Figure 5.9: Principles of mixture density specialization by adaptation: The general feature space representation (left part) is focused on particular protein families (right part) by means of adaptation techniques.

In the following, all three adaptation techniques which are used for robust estimation of protein family models are discussed. The number of family specific training samples, i.e. the amount of adaptation data $X = (\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T)$ which is usually very small, is denoted by T. The adaptation techniques are applied to semi-continuous HMMs which already were described in section 3.2.2 on pages 46ff.

Maximum Likelihood (ML) Estimation

In the simplest case of target family based specialization of the general feature space, the adaptation is performed by maximizing the likelihood of the mixture density for the family specific sample sequences using the EM algorithm up to convergence. As previously mentioned (cf. section 5.2.1), Expectation Maximization is a standard technique for mixture density estimation. Its drawbacks (discussed on page 115) prevent it from general application to the mixture density estimation process. However, when an initial feature space representation based on a mixture density estimated using general protein data is available and furthermore small family specific training sets are used, EM can be applied for adaptation. In this case, the problematic initialization problem has already been solved by the

modified k-means approach alleviating the problem of finding only local optima, and slow convergence is not an issue since T is small.

Thus, given the initial mixture density representation of the feature space derived from SWISSPROT, the adaptation samples $\vec{x}_t$ are assigned probabilistically to all mixture components $g_k = \mathcal{N}(\vec{x}|\vec{\mu}_k, \mathbf{C}_k)$ in the iterative re-estimation of all parameters (depending on the whole mixture density parameters valid for the m-th iteration represented by θ^m):

$$\begin{aligned} \hat{c}_k^{m+1} &= \frac{1}{T}\sum_{t=1}^{T}\xi_t^m(k), \\ \xi_t^m(k) &= P(g_k|\vec{x}_t, \theta^m) \\ \theta^m &= (\hat{c}_{(\cdot)}^m, \hat{\vec{\mu}}_{(\cdot)}^m, \hat{\mathbf{C}}_{(\cdot)}^m) \end{aligned} \tag{5.1}$$

$$\hat{\vec{\mu}}_k^{m+1} = \frac{\sum_{t=1}^{T}\xi_t^m(k)\vec{x}_t}{\sum_{t=1}^{T}\xi_t^m(k)}, \tag{5.2}$$

$$\hat{\mathbf{C}}_k^{m+1} = \frac{\sum_{t=1}^{T}\xi_t^m(k)\vec{x}_t\vec{x}_t^T}{\sum_{t=1}^{T}\xi_t^m(k)} - \hat{\vec{\mu}}_k^{m+1}(\hat{\vec{\mu}}_k^{m+1})^T. \tag{5.3}$$

However, since the parameters of *all* mixture components are re-estimated by the ML procedure, usually still rather large sample sets are required for *robust* adaptation.

Maximum A-Posteriori (MAP) Adaptation

Although specialized mixture density representations can be obtained by means of the previously described ML estimation, the quality of the resulting transformed feature space drops significantly if too little adaptation data is available. Due to the complex model architecture and due to the modeling of large parts of proteins requiring large amounts of states, complete parameter re-estimation using pure ML adaptation of mixture components used for Profile HMMs can be problematic.

The principle idea of an alternative adaptation technique discussed here, is the suitable interpolation of existing parameter estimates obtained by exploiting the set of general protein data and the re-estimates derived when processing adaptation data. Within the Bayesian framework, the parameter estimation is now formally performed by maximizing the posterior probability of the statistical model for the given data (compared to maximum likelihood estimation as discussed in the previous section) – hence the name *Maximum A-Posteriori Adaptation*:

$$\hat{\theta} = \operatorname*{argmax}_{\theta} P(\theta|X) = \operatorname*{argmax}_{\theta} P(\theta)P(X|\theta).$$

The principle advantage of MAP adaptation is the balanced incorporation or prior information extracted from the larger set of unlabeled data. The more adaptation samples available,

the stronger their influence and vice versa, the smaller the amount of target specific data, the higher the influence of the background estimation. Note that when an infinite number of adaptation samples is available, the MAP adaptation degenerates to the ML estimation.

According to the detailed mathematical derivation by Jean-Luc Gauvain and coworkers in [Gau92], generally, prior parameter estimates $\hat{\vec{\mu}}_k$ and $\hat{\mathbf{C}}_k$ weighted by τ are combined with the re-estimation based on the family specific data by changing equations 5.2 and 5.3 to the following:

$$\hat{\vec{\mu}}_k^{m+1} = \frac{\tau\hat{\vec{\mu}}_k^m + \sum_{t=1}^{T} \xi_t^m(k)\vec{x}_t}{\tau + \sum_{t=1}^{T} \xi_t^m(k)} \tag{5.4}$$

$$\hat{\mathbf{C}}_k^{m+1} = \frac{\tau(\hat{\mathbf{C}}_k^m + \hat{\vec{\mu}}_k^m(\hat{\vec{\mu}}_k^m)^T) + \frac{\sum_{t=1}^{T} \xi_t^m(k)\vec{x}_t\vec{x}_t^T}{\sum_{t=1}^{T} \xi_t^m(k)}}{\tau + \sum_{t=1}^{T} \xi_t^m(k)} - \hat{\vec{\mu}}_k^{m+1}(\hat{\vec{\mu}}_k^{m+1})^T. \tag{5.5}$$

Note that MAP adaptation is performed iteratively, too. For the adaptation of feature based Profile HMMs using the MAP technique, τ is adjusted to the number of samples assigned to the mixture components as accumulated during the previous steps which allows robust mixture adaptation even for small training sets. The previously mentioned degenerated case of MAP behaving like standard ML estimation corresponds to adjusting τ to zero, i.e. do not respecting any prior knowledge.

Maximum Likelihood Linear Regression (MLLR)

Both specialization techniques explained in the previous sections are based on more or less sophisticated re-estimations of model parameters. Contrary to this, the third technique investigated for this thesis is based on a real *transformation* of the models' parameters, i.e. the particular normal densities.

Originally developed for rapid speaker adaptation of automatic speech recognition systems, Chris Leggetter and Phil Woodland proposed the modification of the mixtures' parameters using affine transformations [Leg95]. As the principle idea of the original work, the parameters of a speaker independent HMM based speech recognition system are optimized towards speaker specialization by exploiting small adaptation sets. The optimization criterion is the maximization of the generation probability $P(X|\lambda)$ (cf. section 3.2.2) of an HMM λ for the given set of adaptation data X. During adaptation the principle structure of the HMMs is kept fixed, only the mixture components are changed. Furthermore, the ML based re-estimations of mixture parameters belonging to HMM states which were covered by the actual adaptation data are generalized to the mixtures of related HMM states which were not observed during adaptation. The underlying mixture components are summarized in so-called *regression classes* and the adaptation is applied to *all* mixtures contained. The actual definition of regression classes is usually application dependent but it can be performed data driven, e.g. by clustering the mixtures' mean vectors. Since the transformation

is estimated only for the mean vectors of mixtures and generalized within a particular regression class neglecting transformations of the appropriate covariance matrices, MLLR adaptation requires very little target specific adaptation data for robust model specialization.[7] It is this small amount of required target specific adaptation data which makes MLLR attractive for the specialization of the general feature space for particular protein family HMMs. Conceptually, speech data is substituted by the feature representation of protein sequence data and speaker models correspond to particular protein family HMMs. Contrary to the specialization techniques presented in the previous sections, here, feature vectors $\vec{x}_t$ are deterministically assigned to particular mixture components. During model evaluation for the adaptation set every time a particular HMM state s is selected *one* mixture component out of the common set of normal densities is chosen. Hence, the selection of a particular state is conceptually equivalent to the selection of a mixture component.

As previously mentioned, the idea of MLLR adaptation is the modification of the mixtures' mean vectors by applying affine transformations:

$$\vec{\mu}_k' = \mathbf{A}_k \vec{\mu}_k + \vec{b}_k.$$

This affine transformation consisting of a translation vector $\vec{b}_k$ and a rotation matrix $\mathbf{A}_k$ can be summarized in a $N \times (N+1)$ dimensional transformation matrix

$$\mathbf{W}_k = \left[\vec{b}_k \mathbf{A}_k\right].$$

By means of this transformation matrix $\mathbf{W}_k$ and an augmented mean vector representation

$$\hat{\vec{\mu}}_k = (1, \mu_{k_1}, \mu_{k_2}, \ldots, \mu_{k_N})^T$$

the adapted mean vector can be defined as

$$\vec{\mu}_k' = \mathbf{W}_k \hat{\vec{\mu}}_k. \tag{5.6}$$

Using regression classes, which are individually defined by $R = \{\mu_1^R, \ldots, \mu_r^R\}$, the transformations defined in equation 5.6 are generalized to densities not covered by the adaptation set via linear regression.

The estimation of a transformation matrix $\mathbf{W}_s$ follows the Maximum Likelihood approach. In [Leg94] the inventors of MLLR define the auxiliary function Q, which needs to be maximized[8]:

$$Q(\lambda, \lambda') = \sum_{\vec{s} \in S^T} P(X, \vec{s}|\lambda) \log(P(X, \vec{s}|\lambda')), \quad \text{with } X = \{\vec{x}_1, \vec{x}_2, \ldots \vec{x}_T\},$$

where λ and λ' denote the model parameters of two consecutive iterations and S^T designates the set of all state chains with length T for a particular sequence of feature vectors

[7] The general idea of MLLR was further developed with respect to changing the mixtures' covariance matrices but the minor improvements generally do not justify the substantially larger adaptation sets required for robust model specialization.

[8] The notation used here corresponds to that which was introduced in section 3.2.2 for the formal definition of general HMMs.

with length T. The actual maximization of Q which is based on the substitution of the definition of the generation probability, and standard maximum determination using the first derivative, results in a non-iterative ML estimation rule:

$$\sum_{t=1}^{T} \gamma_t(s) \mathbf{C}_s^{-1} \vec{x}_t \hat{\vec{\mu}}_s^T = \sum_{t=1}^{T} \gamma_t(s) \mathbf{C}_s^{-1} \mathbf{W}_s \hat{\vec{\mu}}_s \hat{\vec{\mu}}_s^T,$$

where $\gamma_t(s)$ designates the probability for selecting state s at a given step t, and $\mathbf{C}_s$ and $\hat{\vec{\mu}}_s$ denote the covariance matrix and the appropriate mean vector of the (deterministically) assigned mixture for the feature vector $\vec{x}_t$. If $|R|$ densities (i.e. states) are grouped to a regression class R, the estimation of the corresponding transformation matrix $\mathbf{W}_R$ needs to be summed over the appropriate densities deterministically selected by the underlying states:

$$\sum_{t=1}^{T} \sum_{r=1}^{|R|} \gamma_t(s_r) \mathbf{C}_{s_r}^{-1} \vec{x}_t \hat{\vec{\mu}}_{s_r}^T = \sum_{t=1}^{T} \sum_{r=1}^{|R|} \gamma_t(s_r) \mathbf{C}_{s_r}^{-1} \mathbf{W}_R \hat{\vec{\mu}}_{s_r} \hat{\vec{\mu}}_{s_r}^T. \tag{5.7}$$

If identical covariance matrices are assumed for all mixtures assigned to a regression class R, equation 5.7 simplifies to:

$$\sum_{t=1}^{T} \sum_{r=1}^{|R|} \gamma_t(s_r) \vec{x}_t \hat{\vec{\mu}}_{s_r}^T = \sum_{t=1}^{T} \sum_{r=1}^{|R|} \gamma_t(s_r) \mathbf{W}_R \hat{\vec{\mu}}_{s_r} \hat{\vec{\mu}}_{s_r}^T.$$

It can be further simplified if deterministic state selections are assumed (e.g. when using the Viterbi algorithm):

$$\begin{aligned} \sum_{t=1}^{T} \vec{x}_t \hat{\vec{\mu}}_{s_t}^T \delta_{s_t} &= \mathbf{W}_R \sum_{t=1}^{T} \hat{\vec{\mu}}_{s_t} \hat{\vec{\mu}}_{s_t}^T \delta_{s_t} \qquad (5.8) \\ \delta_{s_t} &= \begin{cases} 1, & \text{if } s_t \in \{s_1, s_2, \ldots, s_{|R|}\} \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

If the matrices $\mathbf{X}$ and $\mathbf{Y}$ are defined as follows:

$$\begin{aligned} \mathbf{X} &= [\hat{\vec{\mu}}_{s_1}, \hat{\vec{\mu}}_{s_2}, \ldots, \hat{\vec{\mu}}_{s_T}] \\ \mathbf{Y} &= [\vec{x}_1 \delta_{s_1}, \vec{x}_2 \delta_{s_2}, \ldots, \vec{x}_T \delta_{s_T}], \end{aligned}$$

then equation 5.8 can be written as:

$$\mathbf{Y}\mathbf{X}^T = \mathbf{W}_R \mathbf{X}\mathbf{X}^T. \tag{5.9}$$

Hence, the transformation matrix $\mathbf{W}_R$ can then be defined as:

$$\mathbf{W}_R = \mathbf{Y}\mathbf{X}^T (\mathbf{X}\mathbf{X}^T)^{-1}. \tag{5.10}$$

The original derivation of MLLR adaptation was regarding multiple regression classes. However, Alexander Fischer and Volker Stahl successfully combined the simplifications as given above with the definition of only a single regression class, i.e. the generalization

of the transformation rule to the complete set of normal densities [Fis99]. This implies a dramatic reduction of the number of parameters required for mixture density adaptation to $N \times (N+1)$, where N denotes the dimensionality of the underlying feature data. The *global* transformation matrix which is applied to all augmented mean vectors $\hat{\vec{\mu}}_k$ is defined as follows:

$$\mathbf{W} = \left\{ \sum_{t=1}^{T} \vec{x}_t \hat{\vec{\mu}}_t^T \right\} \left\{ \sum_{t=1}^{T} \hat{\vec{\mu}}_t \hat{\vec{\mu}}_t^T \right\}. \tag{5.11}$$

By means of all adaptation techniques described here the general feature space representation is focussed on particular target families in a completely data-driven way. For both ML and MAP adaptation all mixture parameters are re-estimated, in the latter case in combination with prior estimates of the mixture parameters. For MLLR only the single transformation matrix $\mathbf{W}$ needs to be estimated which requires considerably smaller amounts of target family specific data. Therefore, MLLR is especially attractive for remote homology detection tasks as addressed in this thesis.

5.2.4 Explicit Background Model

When applying Profile HMMs to detection tasks as usual for target identification applications within the drug discovery pipeline (cf. figure 2.7 on page 18), the major difficulty is the discrimination between target hits and misses. Usually it is realized by threshold comparison of the scores and the analysis of the alignment scores regarding statistical significance.

For independence regarding the actual length of a query sequence and for robust separation of sequences belonging to the target model and those which are not, discrete Profile HMM evaluation is based on more or less sophisticated null models for log-odd scores. In section 3.2.2 on pages 61ff several options for background modeling were discussed.

For the application of semi-continuous feature based Profile HMMs as developed in this chapter, a null model based on the prior probabilities of the mixture components estimated during model building is used. In order to reduce the overall number of false detections further, a technique which is principally known from general pattern detection tasks is investigated. Considering e.g. the problem of automatic speaker detection, usually an additional non-target model is estimated which explicitly covers all data *not* belonging to the target class. According to Douglas Reynolds such a model is called *Universal Background Model (UBM)* [Rey97]. As an optional enhancement of the general UBM approach, the definition of the so-called *structured* background model developed in this thesis captures structural information using a left-right topology as outlined in figure 5.10. Furthermore, the original UBM approach proposed by Douglas Reynolds can be used, too.

In order to sharpen the detection results, for target identification, both the UBM and the particular target model are evaluated in a competitive manner which is combined with the log-odd scoring method described above. Thus, the overall remote homology detection procedure is two-stage: in the first step, the UBM and the target model are competitively evaluated limiting the number of resulting target candidates. Following this, the threshold based decision is performed for target identification by analyzing the significance of log-odd scores (via e.g. E-values). The UBM itself, consisting of $L_U = 30$ states, is estimated on the set of general SWISSPROT data by Baum-Welch training. The actual model length

was determined heuristically in informal experiments. Note that alternative background models are imaginable and their appropriate suitability needs to be evaluated for particular applications and target model types.

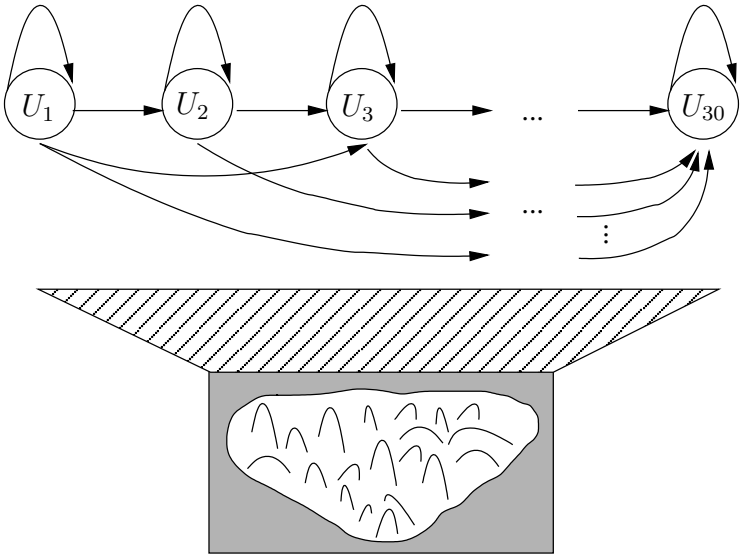


Figure 5.10: Explicit universal background model (structured UBM) covering all general protein data not explicitly assigned to a particular target family of interest. The UBM is of feature based semi-continuous type (lower part – mixture density representation of the general feature space) and evaluated competitively to the particular target Profile HMM.

Summary

In the previous sections a new method for probabilistic protein family modeling was developed. Based on a rich feature based protein sequence representation, semi-continuous Profile HMMs were presented. In figure 5.11 the approach for estimating semi-continuous Profile HMMs and an explicit UBM for robust remote homology detection is summarized graphically for hypothetical protein families $F_1, F_2, \ldots, F_N$. Based on the feature representation of general protein data obtained using the new extraction method, a mixture representation of the general feature space is estimated using k-means (upper-left part). For semi-continuous modeling, the separate optimization of the emission space representation using large amounts of general protein data and the family specific training of the model structure is possible. By means of standard discrete models λ_D estimated on family specific training samples (upper-right), and the general feature space representation, semi-continuous Profile HMMs λ_G are obtained via Viterbi training (middle-right). Then, the mixture representation is optimized for the target families by applying adaptation techniques resulting in family specific models λ_S (lower-right). Finally, on SWISSPROT data the UBM is estimated (lower-left part).

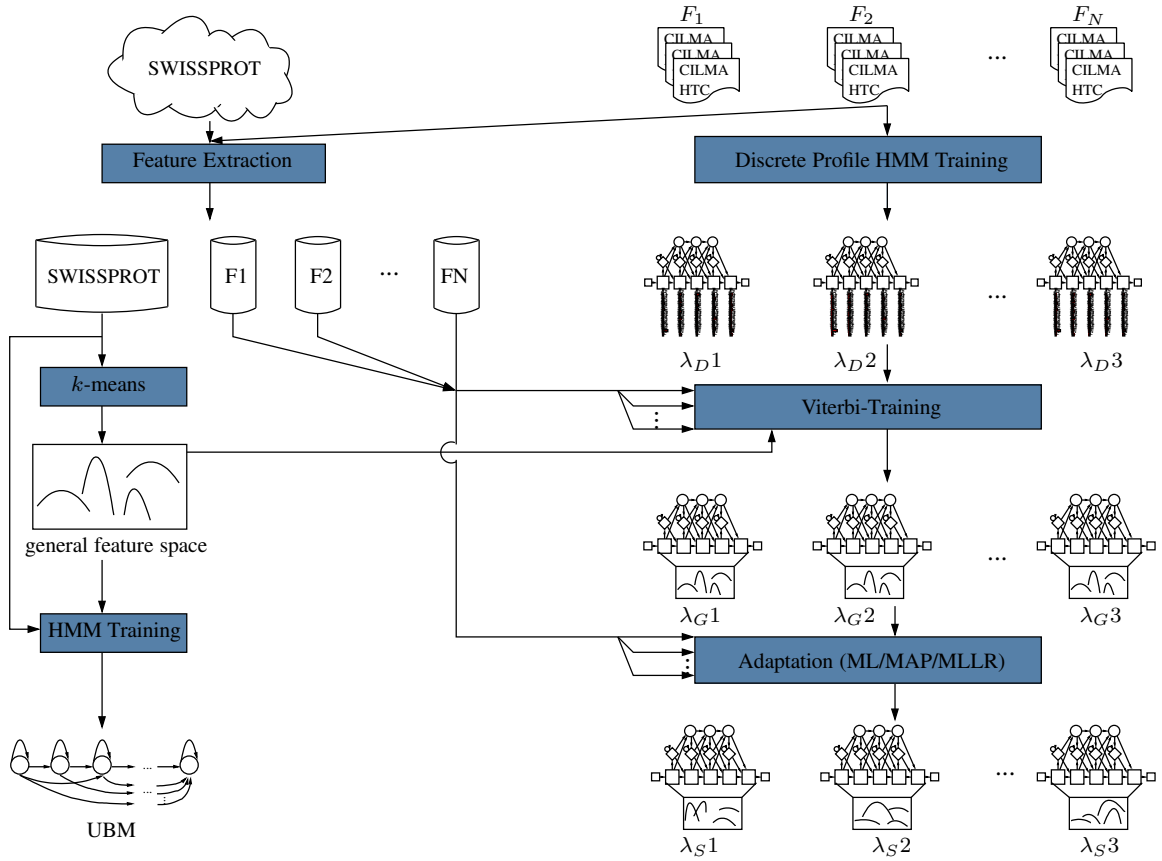


Figure 5.11: Overview of the complete model estimation process for obtaining robust feature based semi-continuous Profile HMMs of protein families – see text for explanation.

5.3 Protein Family HMMs with Reduced Complexity

In chapter 3 the state-of-the-art in computational protein sequence analysis was summarized. Strictly speaking, almost all methods described are based on more or less sophisticated variants of the general Dynamic Programming approach. Analyzing the technique currently most successful, i.e. sequence comparison using probabilistic models of protein families (namely Profile HMMs), the Dynamic Programming "roots" become obvious. Its principle operations, i.e. substitution or match, insertion, and deletion, are directly mapped to the probabilistic framework of Hidden Markov Models. They correspond to the three different kinds of states: Match, Insert, and Delete. By means of such stochastic models of protein families, sequence comparison is performed via traditional alignment of the sequence of interest to the particular family model, either locally or globally. Basically, the better effectiveness of Profile HMM based comparison techniques, i.e. the reason for outperforming traditional sequence to sequence approaches, is caused by their explicit consideration of family specific information in a *stochastic* model. Thus, sequence analysis is performed less strictly which is advantageous especially for remote homology detection.

Due to this principle equivalence of Profile HMM based sequence comparison techniques, current protein family HMMs require a rather complex topology. A Profile HMM

represents the stochastic analogy to a conventional multiple sequence alignment (MSA). For every column of an MSA, three HMM states (Match, Insert, Delete) are created which are almost fully connected to each other (cf. figure 3.14 on page 58 which illustrates the standard three-state architecture). Usually, even when modeling the smallest *functional* protein units, namely domains, the resulting Profile HMMs are very complex, consisting of a large number of states, i.e. model parameters (emission and transition probabilities).[9] In order to establish robust protein family models, *representative* training samples are required for all parameters. Unfortunately, especially for remote homology detection in pharmaceutical applications only small training sets are available which complicates robust model training. For alleviating this sparse data problem, usually prior expert knowledge is incorporated into the model building process (cf. section 3.2.2 on pages 59ff). However, since this expert knowledge is usually obtained more or less manually, the resulting models tend to capture facts which are already known before, i.e. their generalization capabilities are limited.

Addressing the alleviation of the sparse data problem for estimating robust probabilistic protein family models without loosing too much generalization effectiveness and thus to improve the general classification performance, in this section two principle approaches for protein family models with reduced complexity are discussed. First, protein family modeling using architectures beyond Profile HMMs is presented. Here, the modeling base is kept fixed as for usual Profile HMMs, i.e. global e.g. protein domain models are established. Due to the feature based protein sequence representation the complex three-state architecture can be discarded. Following this, an alternative idea for protein family modeling based on the concatenation of small so-called *Sub-Protein Units (SPUs)* is presented.

Note that all alternative modeling approaches presented in the following are based on the feature extraction procedure developed in the previous sections. When directly processing symbolic sequence data as in state-of-the-art techniques, the complex three-state architecture of conventional discrete Profile HMMs is absolutely necessary. The richer sequence representation which captures biochemical properties of residues in their local neighborhood allows abstraction from the strict position dependency of traditional Profile HMMs since now even the emissions cover broader parts of the original protein data. For robust model estimation using small training sets, the concept of semi-continuous HMMs is used for the non-Profile HMMs with reduced model complexity, too. Thus, all robust estimation and evaluation techniques including feature space adaptation and explicit background modeling are also applied as presented in the previous sections. The general model estimation process which was illustrated in figure 5.11 remains valid, whereas in the right part of the sketch the complicated Profile HMMs are conceptually substituted by less complex model architectures.

5.3.1 Beyond Profile HMMs

Current Profile HMMs are direct derivatives of the classical Dynamic Programming approach. Based on the analysis of symbolic representations of protein data the essentials of a protein family are captured by a stochastic model. In fact, the more or less soft position

[9] Note that this thesis is directed to protein *family* models which means that at least protein domains are covered by stochastic models. Compared to motif based Profile HMMs, the modeling base is significantly larger for protein domains for the majority of applications.

dependent modeling of amino acid distributions is the basic advantage of Profile HMMs compared to classical sequence to sequence alignment techniques. Similar to classic pairwise alignment techniques, for flexibility regarding residue insertion and deletion, special states (Insert and Delete) are incorporated. This is also important for local sequence to model alignments where only parts of the model match to (parts of) the sequence. Insert states also contain (more or less) position-dependent amino acid distributions.[10] The classical Profile HMM architecture and its derivatives guarantee highest flexibility for sequence to family alignment. Generally, the basic principle of Profile HMM evaluation corresponds to "probabilistic Dynamic Programming".

In this thesis, the usual application of Profile HMMs is considered. By means of representative sample sets of protein sequences stochastic models of protein families are estimated. Using these models, the affiliation of new protein sequences to a particular protein family is predicted, either by local or by global alignment. In classical three-state Profile HMMs the consensus of a multiple sequence alignment is modeled via a linear chain of Match states. Since the consensus represents the conserved parts of a protein family, the Match states contain the most relevant information for a particular protein family of interest. Thus, the amino acid distributions assigned to Match states are position specific for the columns of the MSA they represent. The amino acid distributions of Insert states are usually not position specific since that would assume conservation which is already covered by the appropriate Match state. However, Insert states' amino acid distributions are model specific. Since current Profile HMMs are based on discrete amino acid data, and every non-silent state emits amino acid symbols, the states' distributions are usually rather specific – they are estimated for a single column of an MSA. Thus, the global decision regarding family affiliation of a particular protein sequence (or parts of it) requires the complex three-state model topology for "probabilistic Dynamic Programming". In several informal experiments based on the SCOPSUPER95_66 corpus (cf. section 4.1.1) the superior performance of the three-state Profile HMMs compared to various alternative model topologies could be proved when processing *discrete* amino acid data.

Complexity Reduction due to Feature Representation: The central idea of this thesis is the explicit consideration of biochemical properties of residues in their local neighborhood for protein data classification (cf. section 5.1). By means of features covering these properties a richer sequence representation is used for protein sequence analysis. Emissions of protein family HMMs are now based on the mixture density representation of the new feature space. The resulting continuous emission probability distributions are much broader than the discrete amino acid distributions of current Profile HMMs while keeping the specificity necessary for sequence classification. If features properly match the emission probability distributions of a particular state, the resulting contribution to the overall classification score is rather high which corresponds to the Match case of Dynamic Programming. Contrary to this, if the features do not match the states' probability distribution, the local score will be small which corresponds to an Insertion. Thus, the *explicit* discrimination between Insert and Match states is not needed any longer because it is implicitly performed already

[10] For Insert states often more general, e.g. family specific, amino acid distributions are used instead of strict position dependent distributions.

on the emission level. Furthermore, explicit Deletes are only conceptual and can be replaced by direct jumps skipping direct neighbors. This means that at least two thirds of the states contained in Profile HMMs can be discarded. One third of the states, namely the Insert states, contain transition probabilities as well as emission probabilities. When skipping the Inserts, the model architecture becomes less complex and thus the number of parameters which need to be trained can be decreased substantially.

To summarize, due to the feature based representation, the *strict* position specificity of Profile HMMs for global protein family models can principally be discarded and alternative model topologies with reduced complexity can be used for protein family modeling – beyond Profile HMMs.

Bounded Left-Right Models for Protein Families: In figure 3.10 on page 45 various standard HMM topologies which are well-known from different general pattern recognition applications were presented. The flexibility of the models depends on their complexity. Simple linear models, where every state is adjacent to itself and to its immediate neighbor to the right, are very common e.g. in speech recognition applications for modeling small acoustic units (so-called triphones) which do not significantly vary in length. Bakis models, where every state is connected to three adjacent states (including itself), are the methodology of choice for the domain of automatic handwriting recognition where letters are modeled which can contain moderate variations in length depending on the actual writer. The most flexible model architecture which is non-ergodic, i.e. fully connected, represents the Left-Right topology.[11] Here, every state is connected to all states adjacent to the right. Thus, arbitrary jumps within the model are possible (including self-transitions) which allows covering signals of arbitrary length.

When modeling protein families containing related but highly diverging sequences, global models need to offer high flexibility for covering the length variance of the family members. Thus, generally (almost) arbitrary jumps within the model must be possible for directly accessing matching parts of the model and skipping parts of the model which are irrelevant for particular sequences of interest. Thus, Left-Right topologies are principally the methodology of choice for protein family modeling beyond Profile HMMs.

However, if arbitrary jumps within a protein family model are allowed, as defined for plain Left-Right topologies, especially for models covering larger protein families the number of parameters to be trained is still rather high. Since every state is connected to all states adjacent to the right and itself, the number of transition probabilities N_t for a model containing L states is defined as follows:

$$N_t = \sum_{i=1}^{L} i + 1 = \frac{L}{2}(L+1) + 1.$$

The additional offset is reasoned by the model exit transition from the last state. If, furthermore, every state contains a direct model exit transition (which can be favorable for local

[11] Ergodic models are of minor importance for the vast majority of applications because either backward jumps are useless (especially when signals evolving in time are analyzed), or the number of parameters is just exorbitant and thus models cannot be trained robustly.

alignments) N_t needs to be increased by $L-1$. For an exemplary protein family model consisting of 100 states, the number of transition probabilities is 5 051 for the basic Left-Right architecture and 5 150 if model exit is possible from every state.

Even for extremely diverging sequences belonging to a particular protein family it is rather unrealistic to assume *arbitrary* alignment paths through the appropriate protein family model which are allowed when using the plain Left-Right topology. The majority of state transitions will not be observed and can, therefore, not be trained. Thus, a variant of standard Left-Right models is developed for protein family modeling – so-called *Bounded Left-Right (BLR) Models*. The basic idea is to restrict direct state transitions to a local context of a particular state by finding a reasonable compromise between linear and complete Left-Right models resulting in significantly less transition parameters to be trained. The number of state transitions depends on the length of the underlying protein family model which is determined as follows.

When applying BLR models to global alignment tasks where protein families are completely evaluated for the whole sequence of interest the model needs to ensure that even the smallest relevant sample sequence can be completely assigned to it. In Profile HMMs and standard Left-Right models principally every state is connected to the model exit either via Delete states or explicit transitions as mentioned earlier. However, for the majority of cases these are only fallback solutions since the transitions certainly cannot be trained using real sample sets. Protein family models need to cover the majority of family members. Thus, the length of the models is determined by analyzing the training samples. The length of BLR models is determined as the median of the lengths of the training data. Compared to e.g. the arithmetic mean the median is more robust against outliers. Given the length of the model and the rule explained above that it must be possible to align even the smallest relevant sample sequence to it, the number of direct state transitions N_s^F for a particular state s of a given protein family F is adjusted as follows:

$$N_s^F = \min\left(\frac{\text{median(length of sample sequences)}}{\text{min(length of sample sequences)}}, \#\text{ states adjacent to the right}\right). \quad (5.12)$$

At the end of a model the number of successors can be smaller than the number of transitions calculated. By means of the selection in equation 5.12 ('min') it is technically guaranteed that all transitions of a particular state point to states which are actually existing. For local alignments optionally every state can serve as model entrance and exit. The corresponding transition probabilities are fixed by assuming uniform distributions which is reasonable according to [Dur98, p. 113ff]. The BLR architecture of protein family models is illustrated in figure 5.12.

Compared to the approximately 5 000 transitions for the complete Left-Right model architecture of the exemplary protein family given above, the number of parameters to be trained for the BLR topology is decreased to approximately 500 when assuming a median length of 100 and a minimum length of 20. For the three-state Profile HMM architecture the number of *transition* parameters for the given example is approximately 2 700. Additionally, the number of emitting states in BLR models is halved compared to standard three-state Profile HMMs. Note that due to respecting local amino acid contexts already at the level of emissions, usually feature based BLR models are significantly shorter than Profile HMMs.

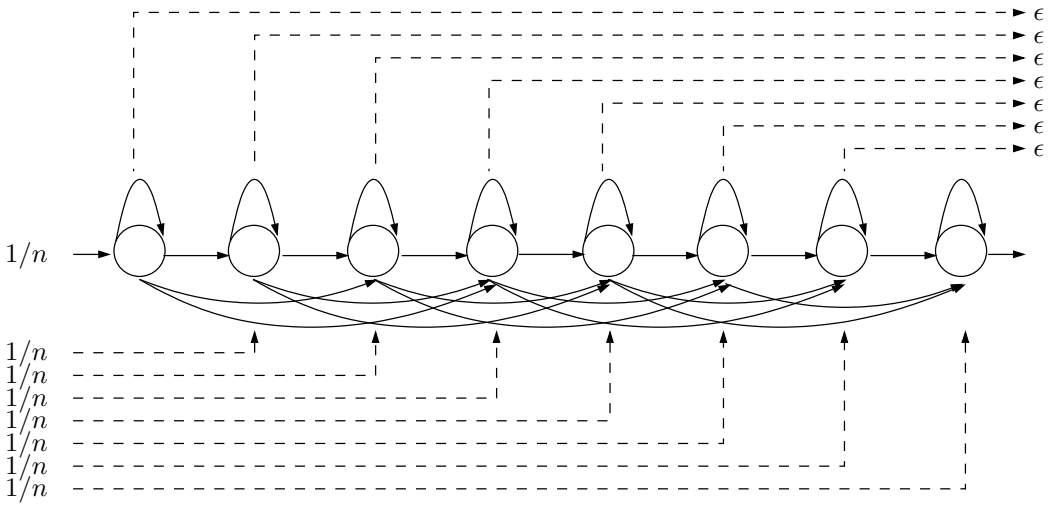


Figure 5.12: Bounded Left-Right model architecture for robust feature based protein family modeling using small amounts of training samples. The model length (here 8) is determined by the median of the lengths of the training sequences and the number of transitions per state is automatically bounded (here 4). For local alignments optionally arbitrary state entrances and exits are allowed (dashed lines) whose probabilities are fixed according to a uniform distribution. Exit probabilities are equally set to some small value ϵ.

Given the BLR topology for protein family HMMs, feature based models of semi-continuous type are initialized and trained using standard algorithms as discussed in section 3.2.2 on pages 47ff. For model initialization the feature data corresponding to a particular protein sequence is reasonably aligned to the models' states using simple interpolation. Standard Baum-Welch training as described on pages 51ff is performed for model optimization. Note that the process of BLR model estimation is significantly less complex and can thus be performed much faster than its analogy for Profile HMMs. In addition to the reduced number of parameters involved this is the major outcome of the new approach. As already mentioned earlier the procedures of general feature space estimation as well as model specialization by adaptation remain identical as described in the previous sections. To summarize, by means of the richer feature based protein sequence representation, less complex global protein family models become possible, namely Bounded Left-Right models as presented here. These models contain significantly less parameters which need to be trained for robust protein family modeling. By means of effective adaptation techniques, powerful probabilistic models of protein families can be estimated in a completely data-driven manner without incorporating explicit prior knowledge.

5.3.2 Protein Family Modeling using Sub-Protein Units (SPUs)

In addition to the enhanced protein family modeling techniques presented so far, in the following the concept of a more radical change within the overall modeling process is discussed – protein family modeling using building blocks which are obtained in a completely data-driven manner. Note that the focus is concentrated on the description of the *general* concept as an emerging field including one reference implementation. Due to the

conceptual character, various modifications and especially enhancements aiming at certain concrete tasks within molecular biology research beyond protein *family* analysis are imaginable. However, these applications and especially their evaluation with respect to biological (e.g. pharmaceutical) relevance are beyond the scope of this thesis.

Comparing most current protein family modeling with state-of-the-art approaches in different pattern recognition applications another fundamental difference becomes obvious. Usually, protein families are covered by global probabilistic models capturing complete sequences. Even when estimating models for the functionally smallest units, i.e. the protein domains as treated in this thesis, very large models consisting of more than hundred states are not the exceptional case. Especially for remote homologue sequences rather dissimilar parts of a particular protein family are integrated into a single probabilistic model. Contrary to this, for e.g. speech recognition systems word models are established by concatenations of significantly smaller building-blocks (usually triphones). This becomes reasonable when analyzing complex languages containing large numbers of words which cannot be trained since there are usually too few training samples available. However, triphones are suitable building blocks for even the most complex words which can be trained "easily".

According to the literature there are hardly any protein family modeling approaches following the paradigm of concatenating building blocks. One exception is the MEME system of William Grundy and colleagues which was already discussed in section 3.2.2 on pages 63ff including its general applicability for the remote homology analysis problem. MEME heuristically combines rather primitive motif HMMs to protein family models.

Principally, the idea of motif based protein family HMMs is very promising for tackling the sparse data problem as formulated on page 92. Since the new feature representation of protein sequence data is the fundamental approach of enhancements developed in this thesis the definition of building blocks directly at the residue level seems counterproductive. Instead, building blocks are defined directly on feature data. In analogy to sub-word models in automatic speech recognition applications, these building blocks are called *Sub-Protein Units (SPUs)*. The straightforward approach for modeling protein families using concatenations of some building blocks is to train models given training sets which are annotated with respect to the particular SPUs. Following this, variants of the protein family are extracted by analyzing the most frequent combinations of SPU concatenations. When classifying unknown sequences all protein family variants obtained during training are evaluated in parallel.

Unfortunately, SPU based annotations of training sequences are generally not available. The basic dilemma can be described as some kind of a "chicken and egg" problem: SPU models (later serving as building blocks for whole protein family models) can only be trained when SPU based annotations are available. However, these annotations can only be generated if suitable SPU models are available. In the thesis, this principle problem is tackled using an iterative approach which allows combined SPU detection and model training in an unsupervised and data-driven manner.

Once SPUs are found, which cover only the "interesting" or dominant parts of a protein family relevant for successful sequence classification, they are modeled using standard HMMs with reduced model complexity. Biochemical properties of the protein data analyzed are explicitly considered, and thus the resulting building blocks do not necessarily

correspond to motifs. Since the overall protein family model is reduced to small essential parts, significantly less training samples are sufficient. By means of the most frequent SPU occurrences within the particular training sets, protein family models are derived automatically by concatenation of the building blocks.

The overall process of modeling protein families using SPU based HMMs can be divided into three parts which are described in the following, namely SPU Candidate Selection, Estimation of SPU Models, and Building Protein Family Models from Sub-Protein Units.

SPU Candidate Selection

The feature extraction method developed in section 5.1 provides a richer sequence representation aiming at better remote homology analysis when using Profile HMMs. The selection of SPU candidates is directly based on the 99-dimensional feature vectors. In the first step, general SPU candidates need to be extracted from protein sequences, i.e. training sequences are annotated with respect to the binary decision whether the underlying frames are SPUs or General. The SPU based annotation of the sample data will be used for SPU-model training and protein family creation.

Various criteria for classifying parts of the overall feature representation of protein sequences as SPU or non-SPU (so-called *General Parts GP*) are imaginable. As already mentioned earlier the SPU based modeling approach discussed here represents a general framework for protein family modeling based on building blocks which are conceptually below the level of direct biological functions. In the exemplary version shown here SPUs are defined as high-energy parts of the protein sequence which will be motivated below. Note that alternative approaches for SPU candidate selection can be used equivalently within the overall framework.

All parts of the original training sample whose feature vectors' energy is below the average energy of the particular protein sequence are treated as General parts GP. The actual discrimination method based on the feature vectors' energy becomes reasonable when analyzing the feature extraction method in more detail (therefore, cf. its general description in section 5.1.2). In order to extract reasonable features, a Discrete Wavelet transformation is applied to the signal-like representation of biochemical properties of residues in their local neighborhood. Following this, the approximation and some detail coefficients are used as the base for further analysis. One fundamental property of the wavelet transformation is the concentration of signal energy in the upper coefficients (cf. appendix A). Thus, high feature vector energy is a reasonable indicator for relevance.

For robust SPU candidate selection the energy signal of the feature vectors corresponding to a particular protein sequence is post-processed using smoothing techniques (DWT smoothing and median filtering). By means of these techniques, protein sequences are principally sub-divided into SPUs and General parts GP which can be seen in figure 5.13 for an exemplary *Immunoglobulin* (d1f5wa_). SPUs are extracted from the energy signal of the protein sequence (solid line) where the average protein energy (dotted line) is below the actual feature vector energy. By means of post-processing two SPU candidates (dashed rectangles) are selected.

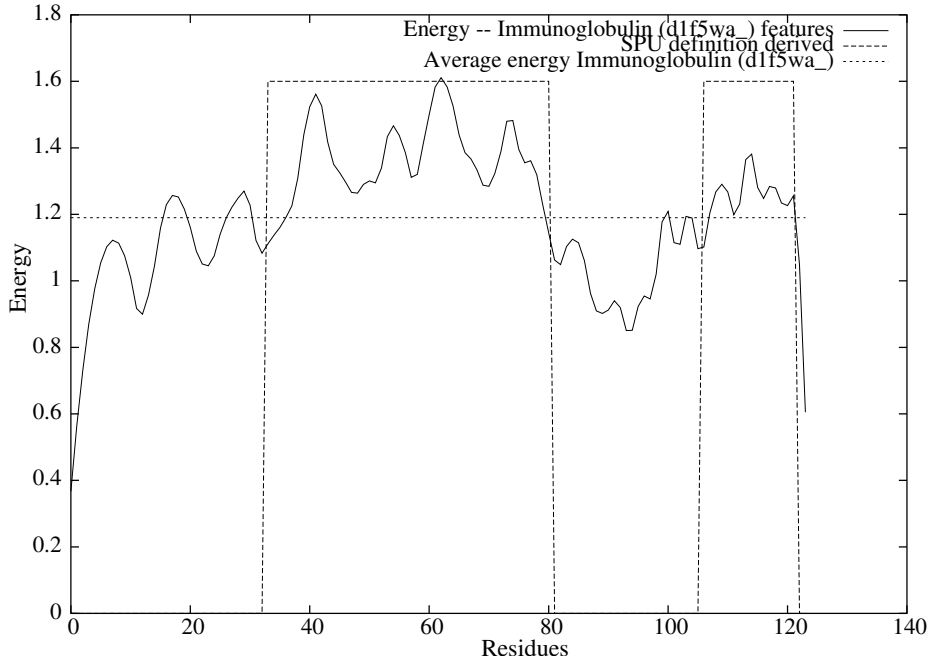


Figure 5.13: Example for SPU candidate selection via the energy criterion: All contiguous parts of the feature representation of an exemplary *Immunoglobulin* (d1f5wa_) whose energy (solid line) is higher than the sequence's average (dashed line) are marked as SPU candidates (rectangles).

Estimation of SPU-Models

In the first step of the new protein family modeling approach protein sequences are annotated with respect to the SPU candidates or General decision. Following this, corresponding SPUs need to be identified in order to train HMMs for a non-redundant set of SPUs relevant for the particular protein family.

The SPUs estimated for the protein family model, and the General model which is unique for every protein family, are modeled using linear, semi-continuous HMMs. Once the training set is finally annotated using the non-redundant set of SPUs, these models are trained with the standard Baum-Welch algorithm.

In the approach presented here, the final set of SPUs relevant for a particular protein family is obtained by applying a variant of the EM algorithm for agglomerative clustering of the initial (unique) SPU candidates. Therefore, model evaluation and training of SPU-HMMs is alternated up to convergence. Here, convergence means a "stable" SPU based annotation of the training set, i.e. only minor differences between the annotations obtained in two succeeding iteration steps. During the iterative SPU determination unique models for corresponding SPUs are estimated since redundant models will not be hypothesized. Thus the set of effective SPU candidates is stepwise reduced and the most frequent SPUs are used for the final annotation of the training set. This procedure, which is comparable to the k-means clustering for HMMs proposed in [Per00b], is summarized in figure 5.14.

1. Initialization:
Obtain initial set S_0 of SPU candidates by e.g. energy based annotation of training sequences.

---

2. Training
Perform Baum-Welch training for SPU candidate models (linear HMMs) using the current annotation of the training set.
3. Annotation
Use updated SPU models for obtaining new annotation of the training set – recognition phase.

---

4. Termination
Terminate if two subsequent annotations of the training set do not (substantially) differ, i.e. convergence, continue with step 2 otherwise.
5. SPU Candidate List Reduction
Reduce set of SPU candidates by discarding those elements which are not included in the final annotation of the training sequences. Perform final annotation of the training set using the remaining list of SPU candidates.
6. Final Training
Perform steps 2-4 until convergence and train SPU models using the final annotation of the training set.

Figure 5.14: Algorithm for obtaining a non-redundant set of SPUs which are used for final protein family modeling (comparable to [Per00b]).

Building Protein Family Models from Sub-Protein Units

Given the non-redundant set of SPUs relevant for the particular protein family, the global protein family model is finally created. The protein family itself consists of variants of SPU concatenations obtained during training. The N variants which are most frequently found within the annotation of the particular training sets, are extracted for the conceptual family definition. Here, optional parts as well as looped occurrences are possible. For actual protein sequence classification, all variants are evaluated in parallel and determine the final classification decision. Comparable to e.g. speech recognition applications the variants are mutually connected using pseudo states which gather transitions.

In figure 5.15 the complete concept for estimating SPU based protein family models is graphically summarized. The three steps described above directly correspond to the three parts marked. In the first row SPU candidates are highlighted red. For clarity the amino acid representation is shown. However, the selection is performed using the 99-dimensional feature vectors. Based on the initial list of SPU candidates a non-redundant set of SPUs is obtained by applying the iterative SPU estimation algorithm. In the middle part of the sketch the corresponding linear SPU HMMs are symbolized. For the global protein family model these SPUs are concatenated. The most frequently occurring SPU concatenations within the training set serve as base for the protein family variants which are evaluated in parallel when classifying unknown protein sequences (lower part in figure 5.15).

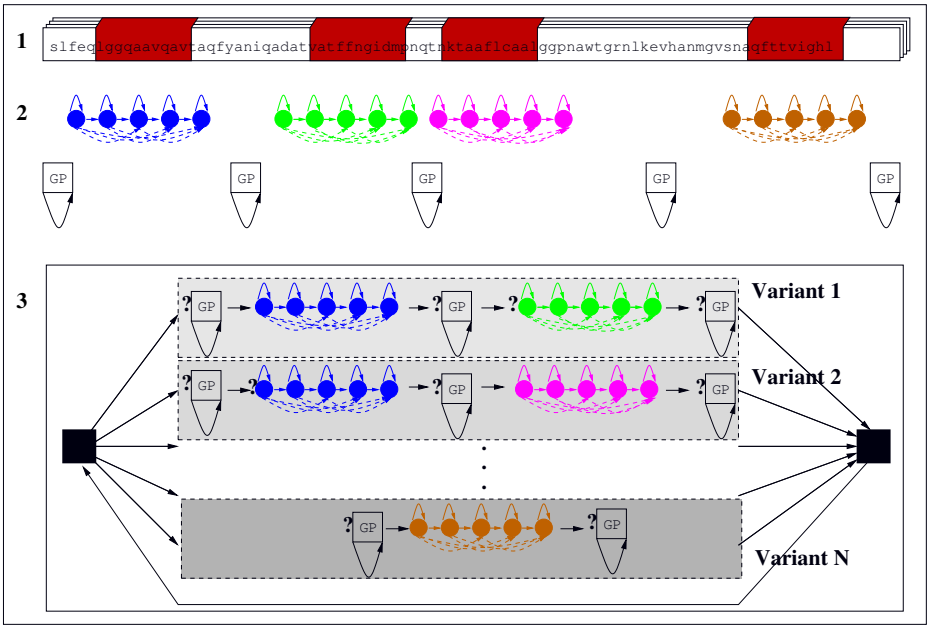


Figure 5.15: Overview of the SPU based protein family modeling process: 1) SPU candidate selection; 2) Estimation of the non-redundant set of SPU models; 3) Protein family modeling using variants of the most frequently occurring SPU combinations within the training set. All optional parts of the model variants are marked with '?"'.

5.4 Accelerating the Model Evaluation by Pruning Techniques

Protein sequence analysis techniques are generally the reason for the basic paradigm shift in research dedicated to the first steps of the molecular biology processing pipeline from manually analyzing specific proteins (e.g. in wet-lab experiments) towards high-throughput automatic screening techniques. Especially for pharmaceutical purposes, currently, target identification and verification would not have been imaginable without computational methods as summarized in chapter 3.

In the last few years the size of protein sequence databases has increased dramatically and it is still steadily growing. As an example in figure 5.16 the exponential growth of the PDB database is illustrated. Since the number of sequences is enormous, homology detection as well as homology classification became a very challenging task in terms of computational effort dedicated to the search and classification problem. Usually, research in molecular biology is performed in a more or less iterative manner, i.e. once new insights are obtained new questions are to be answered. Mostly this implies new database searches in order to find sequences belonging to a particular protein family which is defined by e.g. a certain biological function which was discovered in the "previous" iteration. Thus, efficient model evaluation techniques are mandatory when applying probabilistic protein family models to the task of protein sequence analysis for huge amounts of data.

With respect to the computational effort necessary for model evaluation, in different fields of pattern classification applications the situation is almost comparable. As an example state-of-the-art speech recognition systems consist of hundreds of HMMs each containing substantial amounts of parameters. These models need to be evaluated for every utterance to be recognized which is rather challenging especially when performing online, i.e. when the

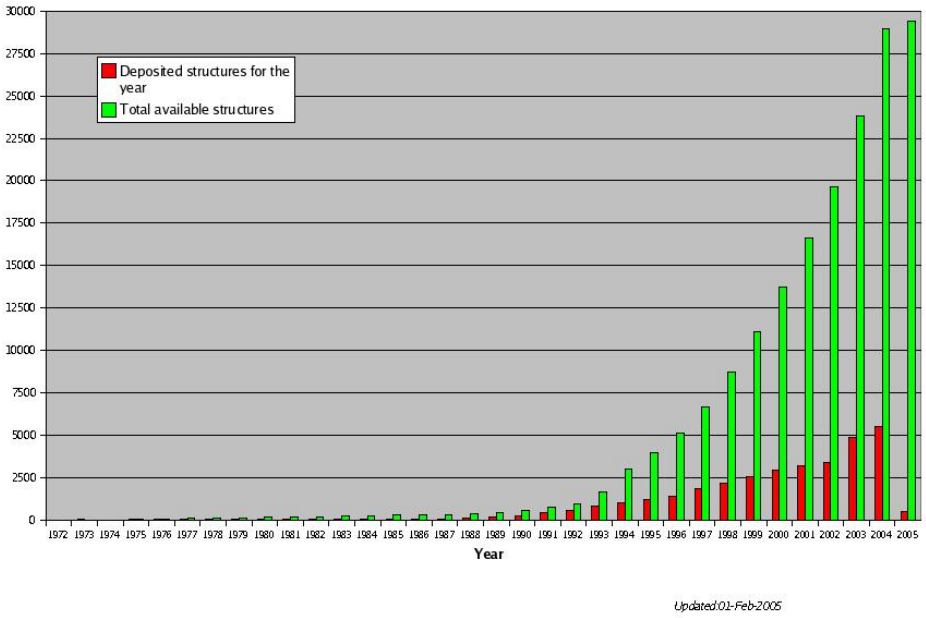


Figure 5.16: Growth of the PDB, the primary protein database. Especially in the last decade the number of entries grew exponentially (courtesy of [Ber02]).

transcription of the uttered speech is required in real-time as usual for e.g. dialogue systems used in automatic information desks.

Throughout the years, sophisticated model evaluation methods were developed aiming at limiting the computational effort for general pattern recognition applications of HMMs. Contrary to the bioinformatics domain for the majority of these alternative application fields the acceleration of the evaluation is performed algorithmically. Currently, biological sequence analysis is usually accelerated by increasing the computational power, i.e. by the deployment of more computers.

Unfortunately, the general problem of extraordinary computational effort still remains. “Brute Force” methods like using specialized hardware for massive parallel model evaluation only treat the symptoms whereas the reasons for computationally expensive database searches when applying Profile HMMs are usually not addressed. The dimensionality of the problem even increases when applying new methods for improving the general detection and classification performance as discussed in this thesis. Thus, in this section model evaluation optimizations are presented which address the third issue relevant for successful application of probabilistic protein family models (cf. page 92ff) algorithmically. Since all of these techniques can be used in parallel computation environments, too, even current solutions requiring specialized hardware[12] can benefit from it. The three approaches presented in the following, namely state-space pruning (section 5.4.1), combined model evaluation (section 5.4.2), and optimization of mixture density evaluation (section 5.4.3), were discovered in alternative pattern recognition fields. For this thesis they were adopted and transferred to the bioinformatics domain which results in efficient computational protein sequence analysis.

[12] As already mentioned before, one example is the GeneMatcher™ architecture of PARACEL©.

5.4.1 State-Space Pruning

Analyzing the state-of-the-art in HMM based protein sequence analysis and reconsidering the basic theory of Hidden Markov Models as summarized in section 3.2.2 it becomes clear that the evaluation of probabilistic protein family models is usually rather straightforward. This means that no optimizations either heuristic or theoretic are applied at all. Especially for automatic speech recognition the situation is rather different. Already in the late 1970s Bruce Lowerre proposed the so-called *Beam-Search* algorithm for heuristic state-space pruning during model evaluation [Low76]. By means of this technique substantial accelerations in HMM evaluation become possible.

In the following the Beam-Search approach is introduced and its general transfer to the bioinformatics domain is presented. Note that this optimization technique can be applied to all kinds of protein family HMMs including discrete, i.e. state-of-the-art, and semi-continuous feature based models. Due to the substantial additional computational effort required for mixture density evaluation the relevance of state-space pruning is extraordinary.

The Beam-Search Algorithm for Accelerated HMM Evaluation

In order to find the most probable path $\vec{s}^*$ through the whole state space $\mathbf{V}$ of an HMM λ_k producing the observation sequence $\vec{o}$, the Viterbi algorithm is widely used as discussed in section 3.2.2. Reconsidering the main idea, basically, each step t of the incremental algorithm consists of the calculation of *maximally* achievable probabilities $\delta_t(i)$ for partial emission sequences $\vec{o}_1 \ldots \vec{o}_t$ and state sequences $s_1 \ldots s_t$:

$$\delta_t(i) = \max_{s_1 \ldots s_{t-1}} \{P(\vec{o}_1, \ldots, \vec{o}_t, s_1 \ldots s_t | \lambda_k) | s_t = S_i\}.$$

Since the dependencies of the HMM states are restricted to their immediate predecessors (the so called Markov property) the calculation of $\delta_{t+1}(j)$ is limited to the estimation of the maximum of the product of the preceding $\delta_t(i)$ and the appropriate transition probability. Additionally the local contribution of the emission probability $b_j(\vec{o}_{t+1})$ is considered. Stepping through the state space recursively all $\delta_{t+1}(j)$ are calculated using the following rule:

$$\delta_{t+1}(j) = \max_i \{\delta_t(i) a_{ij}\} b_j(\vec{o}_{t+1}).$$

Figure 5.17 illustrates the recursive calculation of $\delta_t(i)$ during the Viterbi algorithm.

When analyzing the necessary computational effort for the Viterbi algorithm it becomes clear, that after only a few steps a large amount of possible paths needs to be considered. In figure 5.18 for two different model architectures, namely a classical linear model as used for speech recognition applications (upper part) and the standard three-state Profile HMM topology (lower part), an idea is given for the number of overall explored states while stepping through the state space. The more states there are that have to be explored at each step, the more continuations of all paths possible so far become reasonable. Thus, the amount of paths traced overall increases dramatically and as a consequence the processing time necessary for model evaluation, too. Alleviating the constraints on the model architecture implies increasing the decoding effort!

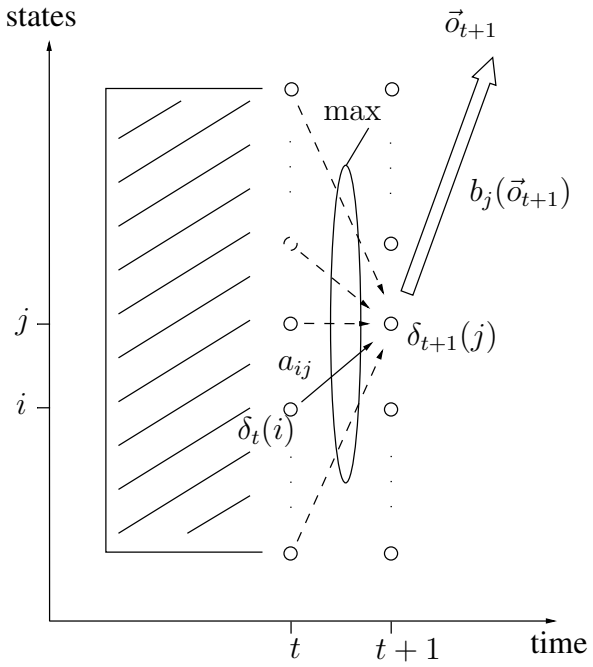


Figure 5.17: Illustration of the recursive calculation scheme for estimating the path probabilities by recombination (courtesy of [Fin03]).

Ergodic HMMs (cf. figure 3.10 on page 45) in principle allow arbitrary paths through the state space. Since the number of the state combinations which need to be explored reaches the extreme value of N^2 the computational effort is substantial. Linear model architectures moderately bound the number of possible paths by strongly restricting the successors of a given state to the appropriate state itself and its immediate successor. However, due to self-transitions the number of states to be considered increases rapidly when stepping through the model. Since discrete protein data covered by Profile HMMs usually varies in both length and constitution, the more flexible three-stage Profile HMM architecture was developed. Due to the Delete states principally every state (adjacent to the right of a particular one) of the HMM can be reached from anywhere in the model, which is comparable to the ergodic topology. Thus, a huge number of paths is generally required to be evaluated. On the right side of the sketches in figure 5.18 for both architectures the substantially differing number of states that need to be explored is shown. The more states are explored, the more computational effort is required for model evaluation.

In order to bound the problem Bruce Lowerre introduced the *Beam-Search* algorithm establishing a dynamic criterion for search space pruning based on the relative differences of the partial path scores [Low76]. The state space is pruned following the basic idea of restricting the search to ”promising” paths only. For obvious reasons the processing time for model evaluation is proportional to the amount of promising paths. Especially for large models consisting of numerous states, many of them cover quite different pattern families. So nearly always at least parts of the state space are quasi irrelevant for one particular final solution. The remaining states are *activated* regarding the most probable path. Formally all states $\{s_t\}$ are activated whose $\delta_t(i)$ are more or less similar to the locally optimal solution

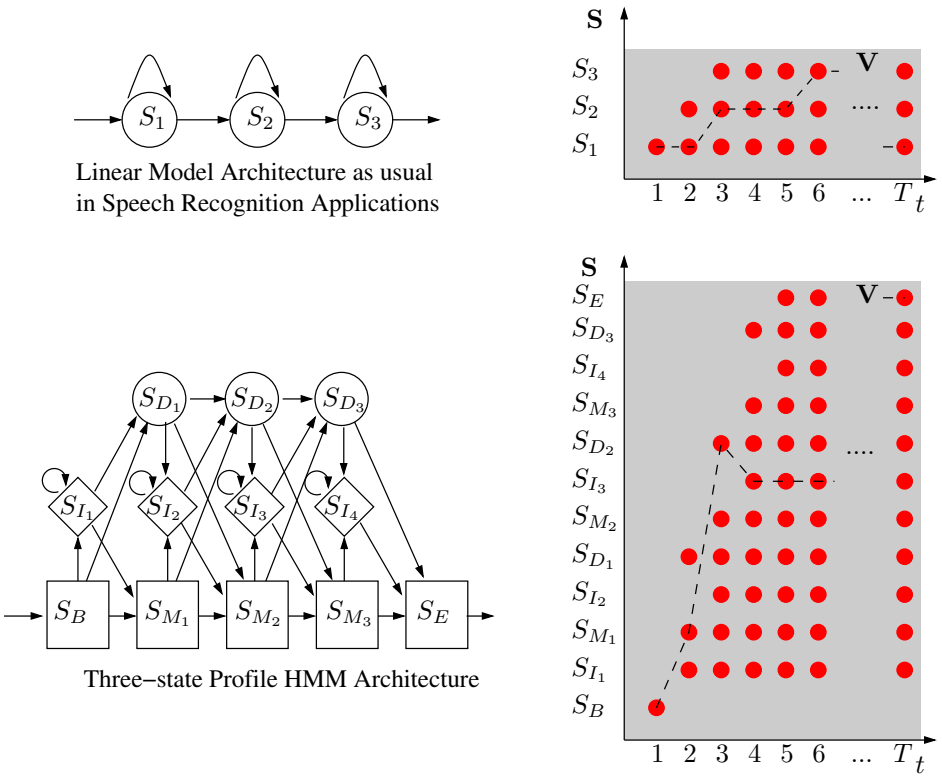


Figure 5.18: States that need to be explored at each step of the evaluation (right) of HMMs with different model architectures (left) – dashed lines: Viterbi paths through state spaces $\mathbf{V}$ for hypothetical sequences. Especially for Profile HMMs after very few evaluation steps a large number of states can be reached and thus need to be explored when using the standard Viterbi algorithm.

$\delta_t^* = \max_j \delta_t(j)$. The threshold of just acceptable differences in the local probabilities is defined proportional to δ_t^* by the parameter B. So the set of activated states $\mathcal{A}_t$ at a given time is located in a *Beam* around the optimal solution and determined by:

$$\mathcal{A}_t = \{i | \delta_t(i) \geq B\, \delta_t^*\} \quad \text{with} \quad \delta_t^* = \max_j \delta_t(j) \text{ and } 0 < B < 1.$$

The only parameter of this optimization technique is the *Beam-width* B. Exploring the Viterbi matrix $\mathbf{V}$ at the next step $t+1$ only these active states are treated as possible predecessors for the estimation of local path probabilities. Thus at every Viterbi step the following modified rule is used for the estimation of all $\delta_{t+1}(j)$:

$$\delta_{t+1}(j) = \max_{i \in \mathcal{A}_t} \{\delta_t(i) a_{ij}\} b_j(\vec{o}_{t+1}).$$

Note that the Beam-Search algorithm represents a heuristical approximation of the standard Viterbi procedure. Consequently, it is a sub-optimal solution of the decoding problem which is, however, of sufficient quality.

Acceleration of Protein Family Model Evaluation

In alternative pattern recognition domains normally *classification* is the primary application for HMMs (comparable to target validation for protein sequence analysis). Usually, small basic models are evaluated in parallel which technically corresponds to a combination of all states involved into a global state-space (see section 5.4.2). This allows global state-space pruning. However, if large patterns are modeled using HMMs the states *within* a particular model are likely to cover mutually different parts. Thus, Beam-Search is expected to be effective already for single model evaluation.

This thesis is directed to the enhancement of probabilistic protein family models. As previously mentioned the smallest protein unit for which a biological function can be assigned is usually the domain. Thus, model lengths of dozens or even hundreds of conserved parts are very common. There is strong evidence that parts of the model that are further apart do not necessarily interfere even for remote homologies. However, when using the complex three-state Profile topology exactly this fact is implicitly assumed since the chain of Delete states allows almost arbitrary state transitions within a particular model. It can be expected that most of the paths through a Profile HMM are not relevant for the final solution and state-space pruning is very effective when concentrating the model evaluation on the most promising paths only.

All modeling approaches developed in this thesis include the state-space pruning as described above. The Beam-Search algorithm needs to be configured specifically for the protein sequence analysis domain. This means that a suitable Beam-width needs to be determined which enables efficient model evaluation while keeping the detection, or the classification accuracy as high as possible. In section 6.4 the experiments which give hints for a proper choice of the Beam-width and its results are presented.

5.4.2 Combined Model Evaluation

As briefly mentioned earlier one basic difference for HMM applications within the bioinformatics domain compared to their use in different fields of general pattern classification is the serialized model evaluation. As an example currently all protein family models which are in any way relevant for the molecular biologists or for pharmaceutical purposes are generally evaluated separately when e.g. the whole genome is annotated with respect to them. Furthermore, even for pure classification tasks in terms of classical pattern recognition, e.g. when performing target verification, the scores produced for protein sequences of interest by every HMM are calculated separately. Finally, the highest scoring model determines the classification result.

Basically, when performing like this, automatic speech recognition is not imaginable for reasonable numbers of words to be recognized. There are too many models to be evaluated completely for too many uttered words. Therefore, the general procedure differs from the one performed for protein sequence analysis. Instead of fully evaluating every model separately, all relevant models are treated combined. Technically, this implies that the states of all HMMs are integrated into a *global* state-space which is conceptually segmented into the particular models.

When arranging all relevant protein family models like this, at the beginning of a particular sequence classification process the initial states of all models are activated, i.e. they are treated as starting points for Viterbi path-search. These paths including all their extensions which can be reached during the remaining steps of the Viterbi evaluation need to be considered in parallel which is basically no advantage compared to the usual separate evaluation. However, when applying the Beam-Search algorithm as discussed in the previous section further substantial savings of necessary computations can be obtained in addition to those implied by local state-space pruning for single models. Usually, after certain Viterbi steps huge amounts of HMM states can be skipped for further evaluation. Reasoned by the avoidance of the exploration of devious paths within the combined state space not necessarily all known profile HMMs need to be evaluated *completely*. Contrary to this, when performing serialized evaluation of multiple models for e.g. genome mapping at least on complete path through *all* particular models needs to be evaluated.

In figure 5.19 the accelerated model evaluation process for protein family HMMs is illustrated. All known models of protein families ($\lambda_1 \ldots \lambda_K$ on the left side of the sketch) are integrated into a single combined state space – the grey shaded box at the right of the diagram. As in figure 5.18 the evaluation process is shown in the diagram of the state space $\mathbf{V}$ on the right side. Following this approach large amounts of HMM states do not need to be activated – they are pruned (black circles). The sequence $\vec{o}$ is classified using the combined state space by finding the Viterbi path (dashed line) through all models. The effect of state space pruning can be noticed via the ratio of activated states (red circles) to the overall number of states (all circles). For every Viterbi step only a moderate "Beam" of states around the Viterbi path is activated. The smaller the percentage of activated states is, the more the model evaluation process itself is accelerated. Note that in figure 5.19 the effects of both local *and* global state-space pruning can be seen. After only a few Viterbi steps the lower model is no longer evaluated since all successor states are pruned (global pruning) and for the remaining models a certain number of assigned states is pruned as well.

At the end of combined model evaluation the index of the best fitting model is delivered including its score for the requested sequence. Compared to the conventional approach multiple repeats of this procedure are not necessary since a global classification is performed.

5.4.3 Optimization of Mixture Density Evaluation

In section 5.1 the new feature based protein sequence representation based on the analysis of biochemical properties of residues in their local neighborhood was presented as the fundamental approach of enhanced probabilistic protein family models developed in this thesis. Following the general theory of Hidden Markov Models it is rather counterproductive to use discrete models for continuous data like the new protein features. Thus, (semi-)continuous HMMs are applied for this kind of data.

The protein features span a 99-dimensional feature space which is best represented using mixture density distributions (cf. section 5.2). By means of both the protein features and the suitable feature space representation, enhanced protein family HMMs are developed. Unfortunately, the price for the enhancement of state-of-the-art protein family models in terms of computational complexity is rather high. Whereas in the discrete case only a single probability distribution needs to be evaluated for every emitting state, in the (semi-)continuous

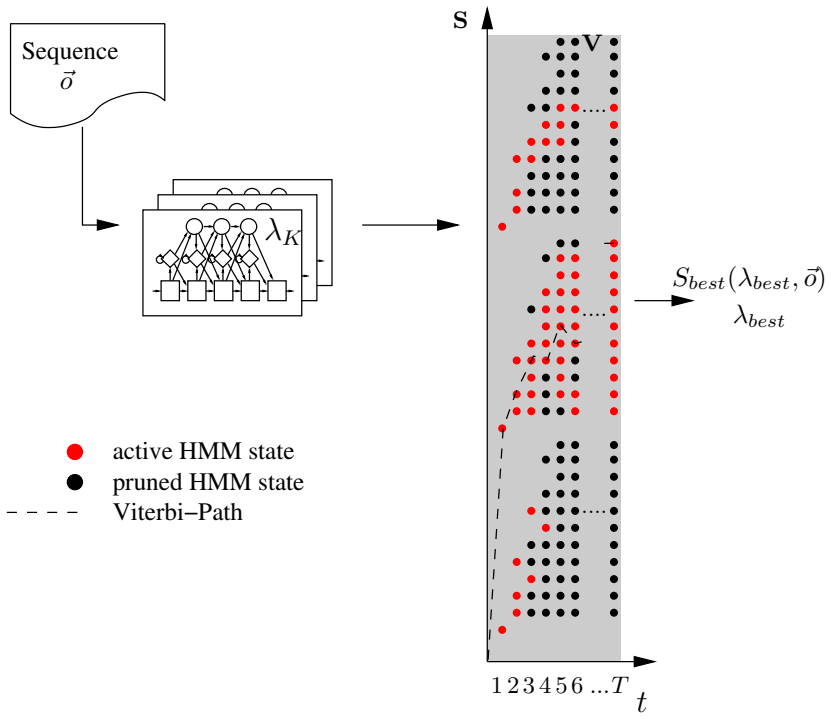


Figure 5.19: Accelerated Profile HMM evaluation due to the use of a combined, pruned state space: all states of all known models are integrated into **V** and only a small fraction of states needs to be explored (red solid circles).

case a mixture density distribution consisting of 1024 Gaussians needs to be examined for every emitting state. Thus, the evaluation of the new semi-continuous feature based protein family models is systematically slower than the evaluation of current discrete Profile HMMs.

The problem of computational expensive mixture density evaluation principally exists for all applications of (semi-)continuous Hidden Markov Models. Not surprisingly, especially within the speech recognition community several approaches were proposed aimed at reducing the computational complexity of mixture density evaluation.[13] Based on techniques provided by the ESMERALDA system [Fin99] the optimization of mixture density evaluation for protein family HMMs is concentrated on the following principles.

Efficient Normal Densities Classification: Since the protein feature space is parametrically represented using a mixture density consisting of K Gaussians (cf. section 5.2.1), strictly speaking, K normal density classifiers are required to be evaluated for every feature vector. The result of these evaluations provides probabilities for every component of the mixture density representation of the feature space. On the one hand, the larger K, the more accurate is the feature space coverage. However, on the other hand, the larger K, the larger the computational effort for mixture density evaluation.

[13] As an example in [Fin03, pp. 163ff] the most relevant techniques for the efficient mixture density evaluation are briefly summarized.

In his standard work on pattern classification, Heinrich Niemann describes the reduction of the computationally expensive problem of evaluating K normal densities $\mathcal{N}_k$ to the calculation of K dot products which is favorable for efficient normal densities classification [Nie83].

For efficient mixture density evaluation the abovementioned calculation of scalar products is used as default for the evaluation of advanced stochastic protein family models developed in this thesis.

Beam-Search for Mixture Densities: Originally, the Beam-Search algorithm as described in the previous section, was defined for state-space pruning. However, the principle idea of the pruning approach can also be generalized to the evaluation of mixture components. Instead of defining a set $\mathcal{A}$ of activated states, a set of activated Gaussians is used. Given the highest scoring mixture component for a particular Viterbi step, mixture components within a *Beam*, which is conceptually identical to the state based Beam, are defined and the further mixture density evaluation is limited to those components only.

Usually, the number of mixture components which need to be evaluated can be reduced substantially when applying the (modified) Beam-Search algorithm. The reason for the actual effectivity is given by the general structure of the underlying feature space. If the mixture density representation suitably covers the feature space it is very unlikely that all components are equally relevant for a single state. By means of the state specific prior probabilities of the particular Gaussians only a small group of components will significantly contribute to the final *local* score. Thus, following the Beam-Search idea the evaluation can be concentrated on these components.

The probability of missing relevant mixture components when using mixture density Beam-Search is almost negligible when configuring the Beam-width wisely. For protein sequence analysis the default values of ESMERALDA ($-\ln(B) = 14$ for logarithmic densities) could be proved sufficient in informal experiments. All modeling approaches developed in this thesis which are based on mixture density representation of the underlying protein feature space use the pruning technique as described here.

Multi-stage Classification: As discussed in section 5.2.1 the feature space representation, namely the particular normal densities, can be obtained by applying variants of standard vector quantization techniques to training data. Generally, the evaluation of a mixture density of Gaussians when aligning feature data to HMMs is equivalent to soft vector quantization using a specialized distance metric, namely the Mahalanobis distance.

Among others, Ernst Günter Schukat-Talamazzini and colleagues proposed methods for fast Gaussian vector quantization [Sch93]. One rather simple but very effective optimization technique is the so-called *sequential pruning*. Here, the basic idea is the decomposition of the vector quantization process into a sequence of "filters" VQ_i each computing probability scores for every mixture component of its stage-specific input candidate list. Furthermore, "...the VQ_i's are designed as Gaussian quantizers using the initial d_i coefficients of the input vector for probabilistic scoring, where d_i increases with i, and $d_I = N$" [Sch93]. This means that every stage of the decomposed vector quantizer operates on a subspace of the

original feature space only, whereas the number of Gaussians which need to be evaluated steadily decreases by candidate selection. As discussed in the previous section, the actual candidate selection can be realized by applying the Beam-Search algorithm to the mixture density evaluation.

In the original case, i.e. without any optimization, a single filter VQ_0 is applied to all Gaussians of the particular mixture density for complete feature vectors $\vec{x} \in \mathbb{R}^N$. A sequential vector quantizer is completely specified by fixing the number of stages I as well as the appropriate subspace dimensions for every stage. In informal experiments within the domain of automatic speech recognition it could be proved that a two-stage classifier is usually suitable when defining the particular subspaces wisely. In the first stage Gaussians are evaluated only for a lower-dimensional version of the original feature vector. Following this, the Beam-Search algorithm severely reduces the number of remaining Gaussian candidates which need to be evaluated in the second step for the complete feature vector, i.e. its representation in the original (high-dimensional) space.

As explained in section 5.1.2, the final step of the feature calculation process developed for protein data consists of a Principal Components Analysis (PCA). According to the theory of PCA, the components of the 99-dimensional feature data are ordered with respect to the percentage of variance they cover (cf. appendix B). The first dimensions of the feature vectors represent the most relevant information necessary for describing the feature space. Thus, the multi-stage mixture density classification is expected to be very effective when processing protein feature data. The dimensionality of the initial subspace representation of the particular feature data used within the first stage of the classification process is determined in various experiments which are described in detail in section 6.4.

5.5 Summary

In this chapter, advanced stochastic models were developed addressing both improved remote homology classification and detection performance. The foundation for all developments which can be used for tackling the sparse data problem as defined on page 92, is the paradigm shift from analyzing symbolic amino acid data towards a feature based representation of biochemical properties of residues in their local neighborhood. Therefore, a sliding window technique is applied to protein sequences for the extraction of frames which contain the residues of a local neighborhood. The particular amino acids are mapped to a signal-like multi-channel numerical representation by exploiting amino acid indices which encode various biochemical properties. Features relevant for the particular protein sequences are extracted by channel-wise signal abstraction via a Discrete Wavelet Transformation and a global Principal Components Analysis.

Based on the new feature based protein data representation, semi-continuous protein family HMMs were developed. Especially in those cases where only little training data is available, the separate estimation of a mixture density based feature space representation using general protein data, and the actual model creation using family specific protein data is favorable. For further specialization of the general feature space which is underlying semi-continuous protein family HMMs, various adaptation techniques, namely ML, or MAP re-estimation, and MLLR adaptation, are applied.

In addition to semi-continuous feature based Profile HMMs, which consist of the standard three-state model topology and semi-continuous emissions, alternative model architectures with reduced complexity were developed. Two variants were presented, namely Bounded Left-Right models, and protein family models based on concatenations of small building blocks, so-called Sub-Protein Units, which are automatically determined in an unsupervised and data-driven procedure. These new modeling techniques allow the estimation of advanced stochastic protein family models containing significantly less parameters which need to be trained. Thus, substantially less training sequences are required for robust protein family modeling.

In order to decrease the number of false positive predictions during remote homology detection target models are competitively evaluated with explicit background models covering general protein data. The reduction of false positive predictions is especially relevant for pharmaceutical applications where candidate sequences which are erroneously identified as targets increase the costs of the drug design process. For efficient model evaluation various optimization techniques known from general pattern recognition domains were transfered and adopted to the bioinformatics domain.

6 Evaluation

Currently, stochastic models of protein families, namely discrete Profile Hidden Markov Models, are the methodology of choice for protein sequence analysis. They are especially relevant for remote homology classification and detection tasks as for e.g. pharmaceutical applications within the drug discovery process as described in section 2.4.1 (target identification and target verification). However, the effectiveness of state-of-the-art modeling techniques is still insufficient as proved in the assessment presented in section 4.1.

The previous chapter of this thesis was directed to the developments of advanced stochastic protein family models. Several new approaches were presented which aim at more robust and thus more effective stochastic protein family models. In addition to the general improvement of protein family models the focus was on model estimation using small training sets. This is especially relevant since usually only very little *representative* data is available for therapeutically interesting protein families at the beginning of the drug design process.

In this chapter detailed experimental evaluations of the newly developed methods are presented. First, in section 6.1, the general evaluation methodology is discussed together with the data sets used. Following this, the three fundamental categories of enhancements are separately evaluated. Based on these results the complete system for enhanced protein sequence analysis using Hidden Markov Models is configured and finally evaluated as a whole in section 6.5.

6.1 Methodology and Datasets

The newly developed methods described in this thesis were implemented in the GRAS^{2}P system (cf. [Plö02]). Thus, all experiments were performed using it. For comparison with state-of-the-art techniques, the SAM system was used as described in section 4.1.

When assessing the capabilities of state-of-the-art methods for protein family modeling, in section 4.1.1 on page 85 the general "chicken and egg" problem for the evaluation of sequence analysis methods has already been mentioned. Generally, it is not useful to refer to automatically obtained sequence annotations as baseline for the evaluation of new techniques. The only conclusion which can be drawn when comparing the capabilities of new approaches to the results obtained when using alternative automatic analysis techniques is how "good" the new technique approximates the old one(s). Without applying further biological expertise it is very difficult to judge differences between both kinds of automatically generated annotations as erroneous or not. Thus, manually annotated sequence sets were used for the assessment of current discrete Profile HMM based protein sequence analysis. The detailed experimental evaluation of the new approaches of this thesis including their comparison to state-of-the-art techniques directly follows this argumentation.

The goal of this thesis is the development of protein sequence analysis methods which can generally be used for e.g. remote homology detection. Thus, concentrating the experimental evaluation on one or two selected protein families seems generally counterproductive. In-

stead, a broader evaluation based on database screening for a substantial number of protein families is more favorable. Corpora are required containing both training and test sets of sequences for which the appropriate protein family affiliations are definitely known. One basic source for protein data is certainly the Brookhaven Protein Data Bank (PDB) which was already briefly described in section 2.3.2. The PDB contains a wealth of information for a vast amount of protein sequences. Based on this database a structural classification of proteins was manually performed by Alexey Murzin and colleagues resulting in the SCOP database [Mur95]. Due to the manual curation of the datasets their quality is extraordinary good which has been widely accepted by the scientific community.[1] Except for SCOP there are hardly any alternative databases containing annotations of this superb quality. One almost comparable example is the CATH database of protein structure classification [Ore97]. In CATH principally the same data (PDB sequences) is classified, but unfortunately at least partially automatic annotation methods were used. Thus, the majority of experimental evaluations performed for this thesis are based on SCOP annotations allowing well founded conclusions about the general applicability of the new methods.

Following the descriptions of experiments which are directed to the separate evaluation of the effectiveness of methods addressing the three fundamental issues (feature based sequence representation, less complex model architectures, and acceleration of the model evaluation), the evaluation of the final complete system as a whole is presented. For both kinds of evaluations, different corpora were created which are summarized in the following. According to the putative application fields of the new methods, namely target verification and identification within the drug discovery process, the evaluation covers both classification accuracy and detection performance. The particular measurements are equally relevant for the overall assessment and the results obtained for target verification tasks usually give reasonable hints for the effectiveness of the appropriate method when it is used for target identification. In section 4.1.2 exact descriptions of the basic methodology used were given (measuring the classification error and ROC curves for estimating the detection performance). Especially for pharmaceutical applications concrete values within ROC curves are relevant for judging the effectiveness of methods addressing remote homology detection. Fixed working points at the ROC curves are analyzed where certain percentages of e.g. false negative predictions are allowed and the corresponding number of, in this case, false positive predictions is treated as characteristic value which concretely measures the effectiveness of the particular method. In practical applications the number of corresponding false predictions is analyzed when allowing five percent failures. Thus, a typical evaluation of detection methods can be formulated as follows:

> *How many false negative predictions (in percent) are delivered by the system, if five percent false positive predictions are acceptable?*

and vice versa:

> *How many false positive predictions (in percent) are delivered by the system, if five percent false negative predictions are acceptable?*

In addition to ROC curves demonstrating the general effectiveness of the methods, these characteristic values are presented in tabular form.

[1] In fact a large amount of publications are based on experiments performed using the SCOP database.

The SCOPSUPER95_66 Corpus

The SCOPSUPER95_66 corpus consists of 16 SCOP superfamilies each containing at least 66 sequences with residue based similarity values of not more than 95 percent. They were obtained from the SUPERFAMILY hierarchy created for the SCOP database by Julian Gough and coworkers [Gou01]. The sequences were randomly divided into disjoint sets of training (two thirds) and test data (one third). Note that this subdivision of the protein sequences is kept fixed, i.e. no further leave-N out tests are performed. SCOPSUPER95_66 was already used for the general assessment of the capabilities of state-of-the-art Profile HMMs in section 4.1 and its detailed description can be found on pages 85ff.

This corpus is mainly used for detailed evaluations of certain variants of the methods developed. In terms of general pattern recognition, the test cases originating from the use of the SCOPSUPER95_66 corpus can be understood as cross validation. Using its results, parameters can be adjusted and the final evaluation based on extended corpora (SCOPSUPER95_20 and PFAMSWISSPROT, see below) are performed using the configuration derived by analyzing the evaluation results based on SCOPSUPER95_66.

Note that contrary to SCOPSUPER95_66 where the suffix '66' designates the overall minimum of sequences per family (including both training and test), the names of the following corpora are determined with respect to the number of training samples they contain. This is reasoned by the fact that evaluations based on these corpora are more related to the sparse data problem. Thus, the suffix is directed to the number of training samples. The actual amount of test sequences is adjusted individually.

For the assessment of detection capabilities the approximately 8 000 sequences of the 95% similarity based SUPERFAMILY hierarchy of SCOP are analyzed for occurrences of the 16 superfamilies.

The SCOPSUPER95_44f Corpus

It is one major goal of this thesis to develop methods for estimating protein family HMMs using small training sets (cf. the second basic issue relevant for the successful application of new powerful probabilistic models – the sparse data problem – as defined in section 4.2 on page 92). Thus, the dependency of the new techniques on the number of training samples used is explicitly evaluated.

Therefore, the SCOPSUPER95_44f corpus was designed. It is a direct derivative of the previously described SCOPSUPER95_66 dataset, i.e. it contains the same 16 SCOP superfamilies extracted from the SUPERFAMILY hierarchy of SCOP limiting the family-wise sequence similarity to the maximum of 95 percent. Contrary to the previous corpus, here, the amount of training data is severely reduced in multiple steps resulting in 44 sub-corpora. The sub-corpus consisting of the maximum number of training samples contains 44 sequences per superfamily, hence the name SCOPSUPER95_44f. The almost uniform distribution of the similarity values across the whole range as illustrated in figure 4.1 is generally valid for all sub-corpora, too.

Contrary to the former dataset, in each of the SCOPSUPER95_44f sub-corpora, the number of training sequences is equal for all 16 superfamilies. Starting from 44 training sequences, this number is steadily decreased down to one sample for every superfamily an-

alyzed while keeping the testset fixed. Generally, two testsets can be analyzed. For direct comparison to SCOPSUPER95_66 where more or less substantial amounts of training material are available, the first testset of this corpus is identical to the one defined for SCOPSUPER95_66 which contains 566 sequences – the so-called *original* testset. Since some superfamilies of SCOPSUPER95_66 contain more than 44 training sequences (cf. table 4.1 on page 87) and the amount of training samples for SCOPSUPER95_44f is fixed with a maximum of 44 samples, a second testset, the so-called *extended* testset, can be defined. For all superfamilies originally containing more than 44 sequences, the additional sequences were added to the original testset resulting in a larger amount of testdata, namely 983 sequences. For those superfamilies with larger training sets the actual selection of the 44 sequences was performed randomly.

When reducing the number of training sequences by eliminating samples, the statistically correct method for evaluating the capabilities of models trained with respect to the remaining sequences is based on averaging the results obtained by leave-N out tests. This means that for a given number N of samples which are to be eliminated from the maximally 44 training sequences of every superfamily analyzed (16), *all* combinatorial possibilities of selecting those omitted sequences need to be addressed. The number C_N^M of possible combinations when selecting N sequences from a universe of M is defined as

$$C_N^M = \binom{M}{N} = \binom{M}{M-N} = \frac{M!}{N!(M-N)!}.$$

For the training sets of SCOPSUPER95_44f this results in the exorbitant number of $2.8 \cdot 10^{14}$ possible combinations. Even with the fastest computers available it is unrealistic to perform 280 trillion experiments. Thus, the statistically correct method cannot be used for SCOPSUPER95_44f based evaluations.

Reconsidering how the particular sub-corpora of SCOPSUPER95_44f were created, it becomes clear that even when performing only single experiments for every sub-corpus reliable conclusions can be drawn. The maximally 44 training sequences were selected by chance for those families consisting of more than 66 sequences. Furthermore, the actual reduction of the resulting training sets for creating the particular sub-corpora is performed by randomly selecting N samples from the rather homogeneous base set for omitting. Although artifacts are expectable, the *general* assessment of the models' capabilities for target identification and verification is possible. As an example, the classification errors obtained for the particular sub-corpora can be summarized in a smoothed curve which can be compared to the smoothed curves of classification errors obtained when using alternative techniques. Although the correct statistical evaluation is (for practical reasons) not possible, the general trend can be approximated rather accurately.

Note that for detection experiments again the complete 95% similarity based SUPERFAMILY hierarchy of SCOP (approximately 8 000 sequences) is analyzed with respect to the 16 superfamilies.

The SCOPSUPER95_20 Corpus

Both corpora described so far are based on 16 SCOP superfamilies. The basic purpose of experiments performed using SCOPSUPER95_66 or SCOPSUPER95_44f is to get some

reliable general idea about the effectiveness of the new techniques. Furthermore, some kind of cross validation is performed. By means of experiments related to one of these corpora (or both) certain parameters of methods can be adjusted (e.g. which adaptation method to use, cf. section 5.2.3) which will be described on the particular pages.

In order to obtain a broader overview of the effectiveness of enhanced protein family models two additional corpora where created. The final system is evaluated using these corpora without further optimization. In terms of general pattern recognition these corpora represent the actual testcase compared to the cross validations treated before.

First, the SUPERFAMILY hierarchy of the SCOP database is further exploited (again at the sequence similarity level of maximally 95 percent). Contrary to both previous corpora, the criterion for including particular superfamilies into this corpus is the existence of at least 40 family members which results in 34 superfamilies. These sequences are divided into training and test sets in (almost) equal shares, hence the suffix '_20' according to the minimum number of training samples per superfamily. Similar to the presentation of the SCOPSUPER95_66 corpus in section 4.1.1 (pages 85ff), in figure 6.1 as well as in table 6.1 the datasets are characterized. It can be seen that the properties of SCOPSUPER95_20 are rather similar to those of SCOPSUPER95_66 with respect to the similarity value distributions as well as to the sequence lengths and their variances.

The base for detection experiments is the same as for the previous corpora.

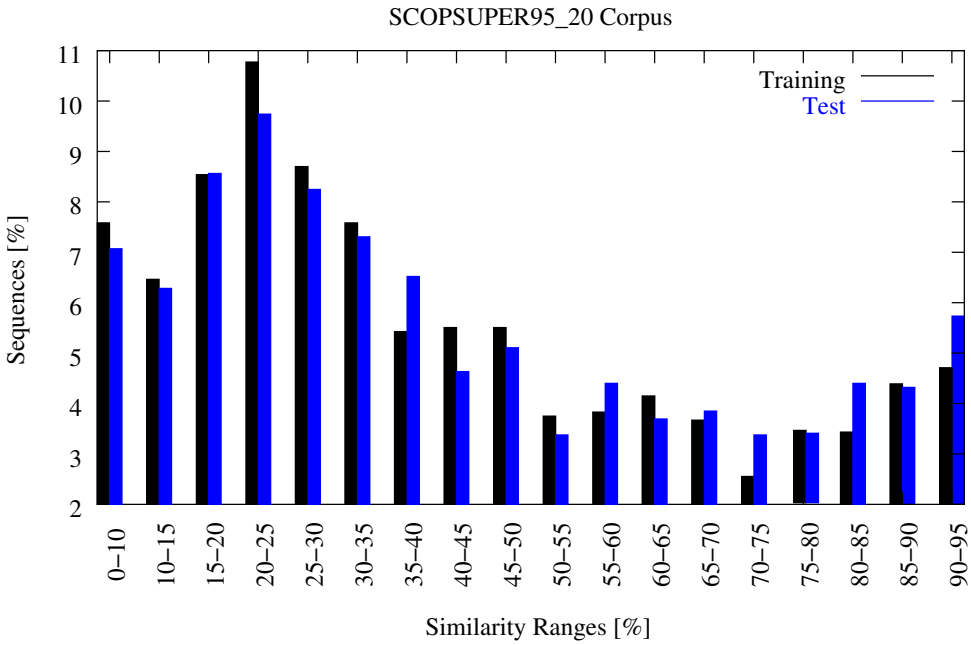


Figure 6.1: Histogram of similarity ranges for the SCOPSUPER95_20 corpus averaged over all 34 superfamilies involved illustrating the well balanced distribution of similarities all over the whole range slightly tending to smaller values (15-35%).

The PFAMSWISSPROT Corpus

The second kind of experimental evaluation of advanced stochastic protein family models as a whole is directed to some kind of a "real-world" scenario. For three exemplary protein families, training data is obtained from the Pfam database, namely the seed alignments of

SCOP Id	SCOP Superfamily Name	# Samples		Length (Mean/Std.-Derivation)	
		Training	Test	Training	Test
a.1.1	Globin-like	45	45	148.9 (12.8)	150.5 (12.9)
a.26.1	4-helical cytokines	20	19	151.8 (24.9)	154.3 (24.0)
a.3.1	Cytochrome c	33	33	101.9 (27.2)	111.7 (30.1)
a.39.1	EF-hand	37	37	129.3 (38.1)	134.1 (52.7)
a.4.1	Homeodomain-like	30	29	67.0 (19.3)	70.6 (21.2)
a.4.5	"Winged helix" DNA-binding domain	37	37	92.1 (28.4)	92.9 (22.1)
b.1.1	Immunoglobulin	156	155	106.7 (11.2)	107.6 (16.9)
b.1.18	E set domains	31	30	118.6 (37.4)	125.9 (47.2)
b.1.2	Fibronectin type III	26	25	98.8 (9.3)	100.7 (6.7)
b.10.1	Viral coat and capsid proteins	48	48	268.6 (95.1)	274.8 (86.3)
b.29.1	Concanavalin A-like lectins/glucanases	40	39	226.4 (58.8)	213.9 (60.9)
b.40.4	Nucleic acid-binding proteins	36	35	101.3 (33.3)	121.6 (48.6)
b.47.1	Trypsin-like serine proteases	42	41	227.1 (33.4)	230.1 (26.0)
b.6.1	Cupredoxins	38	38	136.2 (31.9)	146.3 (36.1)
b.60.1	Lipocalins	23	22	152.3 (18.9)	149.5 (19.8)
c.1.8	(Trans)glycosidases	47	46	386.1 (85.8)	379.1 (73.4)
c.2.1	NAD(P)-binding Rossmann-fold domains	77	76	197.0 (56.6)	214.3 (73.1)
c.3.1	FAD/NAD(P)-binding domain	34	34	221.5 (92.2)	226.8 (89.9)
c.37.1	P-loop containing nucleotide triphosphate hydrolases	96	95	262.9 (121.1)	249.8 (98.3)
c.47.1	Thioredoxin-like	42	42	109.4 (38.7)	107.9 (36.0)
c.55.1	Actin-like ATPase domain	21	20	202.6 (34.9)	209.5 (36.0)
c.66.1	S-adenosyl-L-methionine-dependent methyl-transferases	21	20	255.0 (55.1)	266.2 (44.2)
c.67.1	PLP-dependent transferases	25	25	404.5 (36.2)	404.4 (30.6)
c.69.1	alpha/beta-Hydrolases	39	38	348.9 (100.1)	331.7 (96.1)
c.94.1	Periplasmic binding protein-like II	21	20	305.0 (74.8)	341.3 (76.5)
d.144.1	Protein kinase-like (PK-like)	23	23	315.9 (28.7)	304.9 (31.8)
d.153.1	N-terminal nucleophile aminohydrolases (Ntn hydrolases)	23	23	273.0 (131.1)	292.3 (172.2)
d.169.1	C-type lectin-like	24	23	124.3 (21.5)	121.8 (15.0)
d.19.1	MHC antigen-recognition domain	26	25	141.1 (43.8)	140.9 (45.1)
d.3.1	Cysteine proteinases	20	19	283.0 (82.4)	269.4 (50.4)
d.92.1	Metalloproteases ("zincins"), catalytic domain	20	20	281.4 (135.2)	259.5 (161.8)
g.3.11	EGF/Laminin	25	24	46.8 (7.4)	44.6 (7.2)
g.3.7	Scorpion toxin-like	26	26	46.5 (13.8)	46.4 (13.3)
g.37.1	C2H2 and C2HC zinc fingers	21	20	31.4 (5.9)	31.6 (4.6)
	Total:	1273	1253		

Table 6.1: Overview of the SCOPSUPER95_20 corpus: For every superfamily the alpha-numerical SCOP Id as well as their real name as defined in the database are given. In the last row the total numbers of samples are summarized.

Pfam-A [Son98]. Using these training samples, the protein family models are estimated and family members are sought within the approximately 90 000 SWISSPROT sequences. The reference annotations are given by the direct link between Pfam and SWISSPROT. However, the quality of this annotation is not as clear as for the SCOP related experiments. Note that evaluations performed using the PFAMSWISSPROT corpus are of summarizing character – just demonstrating real-world applications of advanced stochastic protein family models. The actual selection of the three protein domains was performed rather arbitrarily and a systematic evaluation is actually not claimed in any way.

Compared to the previous experiments, in the final evaluation using PFAMSWISSPROT additional properties which are practically relevant are assessed:

Search for Complete Proteins: SWISSPROT contains arbitrary protein data, i.e. the sequences included are not limited to single protein domains. Proteins are annotated

with respect to Pfam-domains and modeled using stochastic models. When evaluating SWISSPROT sequences for the domain HMMs, the models need to find sequences where only parts (namely the appropriate domains) match. This corresponds to one common application for molecular biologists where they might be interested in occurrences of certain biological functions (represented by the particular domain) in larger protein environments.[2]

Evaluation of the Specificity: Since SWISSPROT contains substantial amounts of data (approximately 90 000 sequences), the specificity of the new models including the effectiveness of the Universal Background Model (UBM) can be evaluated.

In table 6.2 the characteristics of PFAMSWISSPROT are summarized. Three representative protein domains were selected.

Pfam Id	Pfam Name	# Samples Training	# Occurrences in SWISSPROT
PF00001	7tm_1 – GPCR 7 Transmembrane Receptor	64	1078
PF00005	PKinase – Protein Kinase Domain	63	507
PF00069	ABC_tran – ABC Transporter	54	1202

Table 6.2: Overview of the PFAMSWISSPROT corpus for final system evaluation. Members of the three Pfam domains listed here will be searched within the approximately 90 000 sequences of the general protein database.

6.2 Effectiveness of Semi-Continuous Feature Based Profile HMMs

The first category of experimental evaluations, whose results are presented in this chapter, is directed to the central concept of enhanced protein family models developed in this thesis, namely the feature based representation of biochemical properties of amino acids in their local neighborhood (cf. sections 5.1 and 5.2, respectively). Therefore, the classical three-state Profile HMM topology is kept fixed whereas the emissions of both Insert and Match states are based on the new representation.

As described in section 5.2.1 the general feature space representation is estimated using the complete SWISSPROT database and further specialized using adaptation techniques (cf. section 5.2.3). The analysis of informal experiments turned out that semi-continuous protein family models based on feature space representations, which were estimated by exclusively exploiting the particular target family specific training samples, require substantially larger sample sets for robust modeling. When restricting the estimation of the underlying mixture density to the family specific sequence data, no sufficient coverage of the protein feature space can be obtained, resulting in poor generalization capabilities of the corresponding HMMs which is problematic for the analysis of remote homologues. Thus, the separate mixture density estimation using general protein data (SWISSPROT) is of extreme importance together with the family specific adaptation.

The evaluation results presented in this section are based on SCOPSUPER95_66 experiments. In the following both classification and detection results are discussed.

[2] Note that this kind of evaluation principally corresponds to *spotting* in different general pattern recognition applications like automatic speech recognition.

SCOPSUPER95_66: Evaluation of Remote Homology Classification

In table 6.3 the capabilities of the semi-continuous feature based (SCFB) Profile HMMs are given for the SCOPSUPER95_66 based classification task. The results are illustrated by means of a comparison of classification errors obtained when using variants of SCFB Profile HMMs, or their discrete counterparts.

Modeling Variant (Profile HMMs)	Classification Error $\mathcal{E}$ [%]	Relative Change $\Delta\mathcal{E}$ [%] (Base: Discrete Profile HMMs)
Discrete	32.9	–
SCFB (ML)	37.6	+14.3
SCFB (MAP)	24.0	−27.1
SCFB (MLLR)	20.7	−37.1

Table 6.3: Classification results for SCOPSUPER95_66 comparing discrete Profile HMMs to semi-continuous feature based Profile HMMs (SCFB) obtained by the three variants of feature space adaptation (ML/MAP/MLLR). Whereas the ML variant performs worse than state-of-the-art, both MAP and MLLR based models significantly outperform it (the mean confidence range for this corpus is approximately $\pm$ 3.5%).

Reconsidering the fact that the underlying model architectures are identical for all experiments, namely the complex three-state Profile topology, and analyzing the relative changes of the classification error $\Delta\mathcal{E}$ it becomes clear that the new feature representation is very effective. When applying semi-continuous Profile HMMs which where estimated using MAP or MLLR adaptation, the classification error can be decreased significantly. In the best case (MLLR adaptation), the classification error could be reduced by more than one third relative. The classification capabilities of semi-continuous Profile HMMs which are based on ML estimation are worse than standard models since the number of adaptation samples is too small.

SCOPSUPER95_66: Evaluation of Remote Homology Detection

The evaluation based on the SCOPSUPER95_66 classification task gave a first clue regarding the general capabilities of the new feature based protein family modeling techniques. For the second major application field addressed by this thesis, namely target identification, the results of detection experiments are presented in figure 6.2. The complete 95% sequence identity based SUPERFAMILY hierarchy of SCOP was searched for occurrences of the 16 superfamilies. As another new concept developed in this thesis, during searching the particular target models and a structured Universal Background Model (UBM) which captures general protein data are competitively evaluated.

Analyzing the four ROC curves, the superior performance of both MAP and MLLR based SCFB Profile HMMs can also be confirmed for the detection tasks. As suggested by the classification experiments, ML adaptation is also not that effective for target identification.

In addition to the evaluation of the overall detection capabilities, the effectiveness of the explicit background model can be evaluated using ROC curves (figure 6.3). The ordinate of the diagram represents the number of false positive predictions, i.e. those sequences which

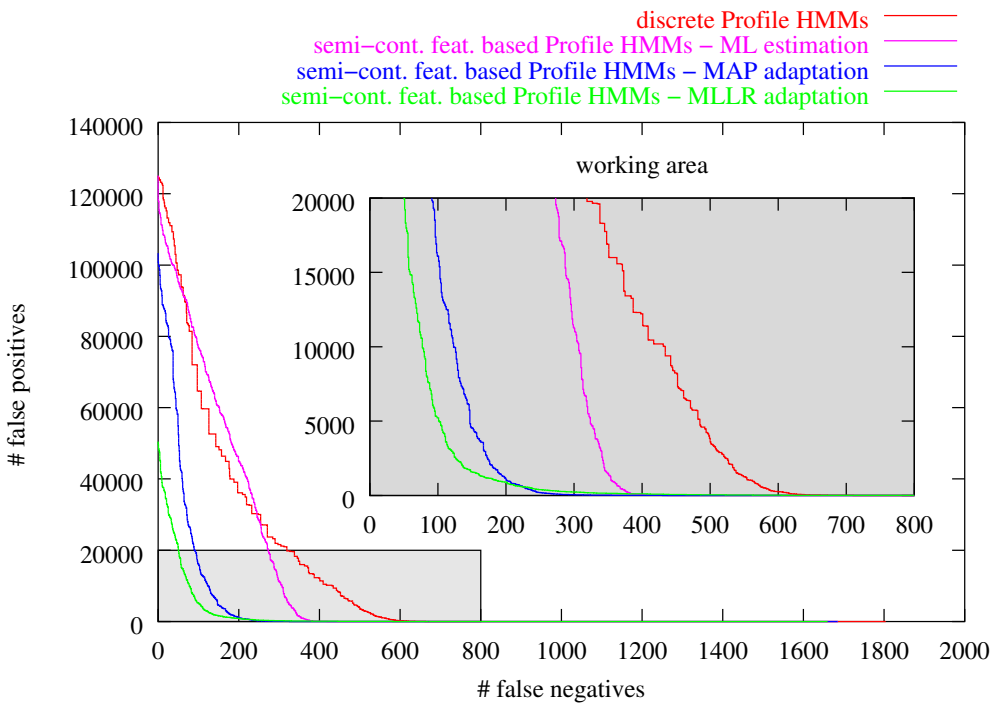


Figure 6.2: ROC curves illustrating the superior performance of feature based Profile HMMs compared to standard discrete models (red curve). The underlying experimental evaluation was performed using the SCOPSUPER95_66 corpus. It can be seen that all semi-continuous feature based Profile HMMs estimated using the particular adaptation techniques produce better detection results than their discrete counterparts – the area below the ROC curve is significantly smaller.

are actually not members of the particular target family but given the appropriate threshold they were falsely classified as members. The smaller the maximum number, the more effective the UBM. The ideal case is that the UBM based evaluation scores are better for all non-family sequences whereas the particular target family scores are better for all actual members. Traditionally, no explicit background model is applied for remote homology detection. Instead, usually some kind of post-processing of the detection results is performed by analyzing the significance of alignment scores and filtering properly. It can be seen that the combination of UBM and MLLR based SCFB Profile HMMs is superior. The maximum number of false positive predictions can be reduced by almost 66 percent.

The characteristic values which concretely specify the mutual dependencies of false positive and false negative predictions are summarized in table 6.4.

Conclusion

The experimental evaluation of the feature based representation of protein sequences presented in this section demonstrates the superior performance of semi-continuous feature based Profile HMMs. It could be shown that the richer sequence representation developed in this thesis is a basic foundation for enhanced probabilistic protein family models. Both application fields relevant for e.g. drug discovery, namely target verification and target iden-

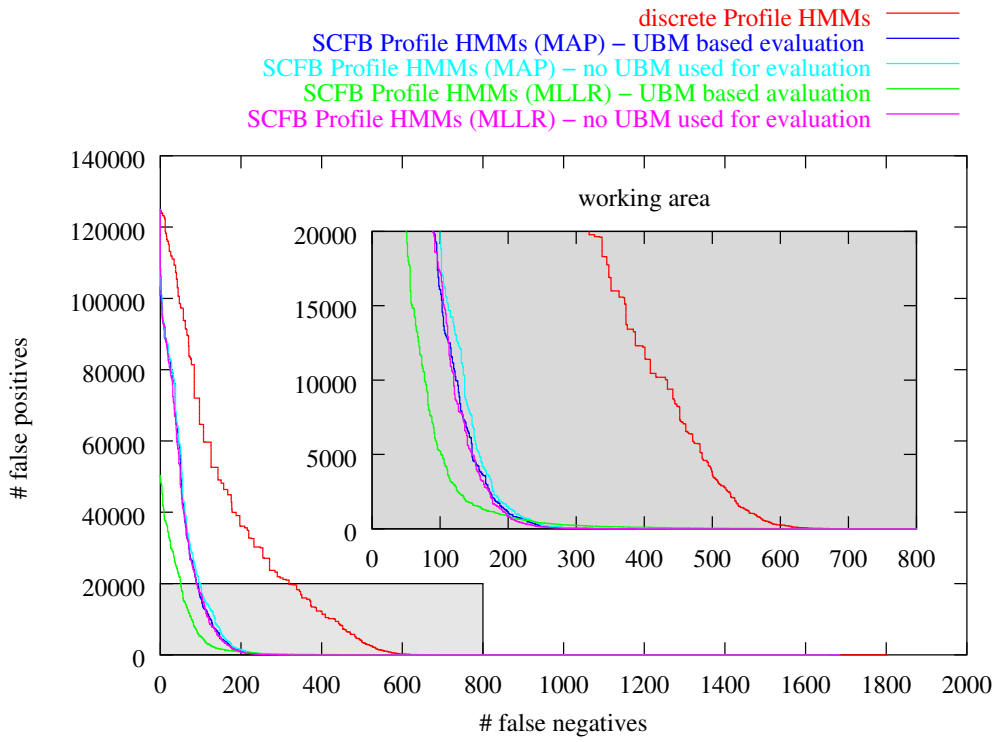


Figure 6.3: Direct comparison of SCFB Profile HMMs' detection performance when applying target models standalone (cyan, and pink ROC curves) and when competing with an explicit background model (blue, and green ROC curves). The superior performance of SCFB Profile HMMs (compared to state-of-the-art discrete models – red ROC curve) can further be improved when using an UBM.

Modeling Variant (Profile HMMs)	False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
discrete	26.1	57.6
SCFB (ML)	17.7	64.4
SCFB (MAP)	7.9	16.0
SCFB (MLLR)	5.1	5.5

Table 6.4: Characteristic values for SCOPSUPER95_66 UBM based detection experiments illustrating the relevance of the new feature based sequence processing for e.g. pharmaceutical applications: At fixed working points of the ROC curves allowing 5% false predictions, the numbers of corresponding false predictions decrease significantly for MAP, and MLLR variants of semi-continuous feature based Profile HMMs.

tification benefit from the new approach. For the latter task the effectiveness of an explicit background model (UBM) which covers all non-target data could be shown.

The third major outcome of these experiments is that ML estimation is not a proper method for specializing protein family HMMs towards the target they represent. Thus, in all further evaluations ML based specialization will not be respected any longer. Note that a detailed presentation of the experimental evaluation of this section is given in appendix D.

6.3 Advanced Stochastic Protein Family Models for Small Training Sets

In the previous section the new approach of feature based probabilistic protein family modeling was experimentally evaluated by means of Profile HMMs consisting of the standard three-state model topology. It could be seen that semi-continuous feature based Profile HMMs significantly outperform their discrete counterparts.

Whereas the experiments described in section 6.2 addressed the analysis of the general effectiveness of advanced stochastic protein family models, this section deals with the sparse data problem as defined on page 92. In addition to the techniques developed for robust estimation of semi-continuous feature based Profile HMMs by means of small sample sets, in section 5.3 two approaches were developed aiming at the reduction of the models' complexities – protein family HMMs which are based on Sub-Protein Units (SPUs), and Bounded Left-Right models. In the following sections all advanced stochastic modeling techniques are evaluated with respect to the number of training samples exploited.

First the experimental evaluation for the new building block based protein family modeling using SPUs is given in section 6.3.1. Following this, the results for BLR models are presented in section 6.3.2.

Both approaches are evaluated for classification as well as for detection tasks. The SCOPSUPER95_66 corpus is used for direct comparison of the results to the experiments discussed in the previous section serving as proof of concept for the general effectiveness of protein family models with reduced complexity. In order to demonstrate the actual effectiveness of all new modeling techniques for small training sets, the SCOPSUPER95_44f corpus is used.

6.3.1 Effectiveness of Sub-Protein Unit based Models

In section 5.3.2 the new idea of alternative protein family modeling using small building blocks, automatically estimated so-called Sub-Protein Units (SPUs), was presented. The idea of SPU based protein family models is to explicitly cover only those parts which are absolutely necessary for successful remote homology classification or detection.

In this section, the results of the first experimental evaluations of this new approach are summarized thereby serving as the general proof of concept. In addition to the prototypical implementation of the SPU approach alternative configurations are imaginable for the general framework. However, the *exhaustive* evaluation is beyond the scope of this thesis.

In order to prove the general applicability of the approach, the performance of SPU based protein family models is compared to both discrete Profile HMM based results and the results obtained when applying semi-continuous feature based Profile HMMs. Motivated by the results of the preceding experimental evaluations, the specialization of the feature space representation underlying the SPU models is limited to MLLR adaptation only.

SCOPSUPER95_66: Evaluation of Remote Homology Classification

Compared to current discrete Profile HMMs, remote homology classification using SPU based protein family models performs significantly better. This can be seen in table 6.5

where the classification errors for both Profile HMMs and SPU based protein family models are compared. The classification error is decreased by almost 29% relative. Compared to semi-continuous feature based Profile HMMs, the improvements are at an almost similar level, i.e. the proof of concept for the alternative protein family modeling approach could be given.

Modeling Variant	Classification Error $\mathcal{E}$ [%]	Relative Change $\Delta\mathcal{E}$ [%] Base: Discrete Profile HMMs
Discrete Profile HMMs	32.9	–
SCFB Profile HMMs	20.7	-37.1
SCFB SPU HMMs	23.5	-28.6

Table 6.5: Classification performance for discrete Profile HMMs, their semi-continuous feature based counterparts (MLLR adapted feature space representation), and the new modeling approach using SPUs. Target models estimated using feature based building blocks significantly outperform state-of-the-art models for SCOPSUPER95_66 while reaching comparable performance as SCFB Profile models..

SCOPSUPER95_66: Evaluation of Remote Homology Detection

Following the assessment of the classification capabilities of the new SPU based protein family models, their applicability for remote homology detection tasks is evaluated for the SCOPSUPER95_66 corpus. In analogy to preceding presentations of results for similar experiments, in figure 6.4 ROC curves are presented, and in table 6.6 the corresponding characteristic values for fixed working points within the curves are given.

Note that according to the results of informal experiments on different data the UBM architecture was changed from structured UBM containing 30 states to the classical UBM of Douglas Reynolds consisting of a single state (cf. section 5.2.4). The reason for this decision is as follows: When applying the structured UBM the selectivity is perfect, i.e. no false positive predictions are obtained at all. However, compared to the single state UBM rather large numbers of false negative predictions occur. Since the competitive model evaluation delivers hard decisions for a particular model (either target or UBM) this number cannot be reduced in any further step. The particular false negatives are actually rejected before the threshold based decision regarding log-odd scores is performed. In order to sharpen the specificity of the detection process the application of a single state UBM delivers slightly more false positives but the number of false negatives can be reduced drastically.

In the abovementioned figure and table, respectively, it can be seen that the improved performance of SPU based protein family models compared to discrete Profile HMMs that has been proved for the classification task can be generalized to the detection task, too. The percentage of false positive predictions can be substantially reduced, and the sensitivity is also improved in terms of reduced numbers of false negatives.

Note that the radically changed modeling approach includes still some optimization potential since the effectiveness of semi-continuous feature based Profile HMMs currently cannot be reached. However, since the evaluations presented here serve as the general proof

of concept of the new modeling technique, and the state-of-the-art discrete Profile HMMs are significantly outperformed, SPU based protein family modeling is very promising.

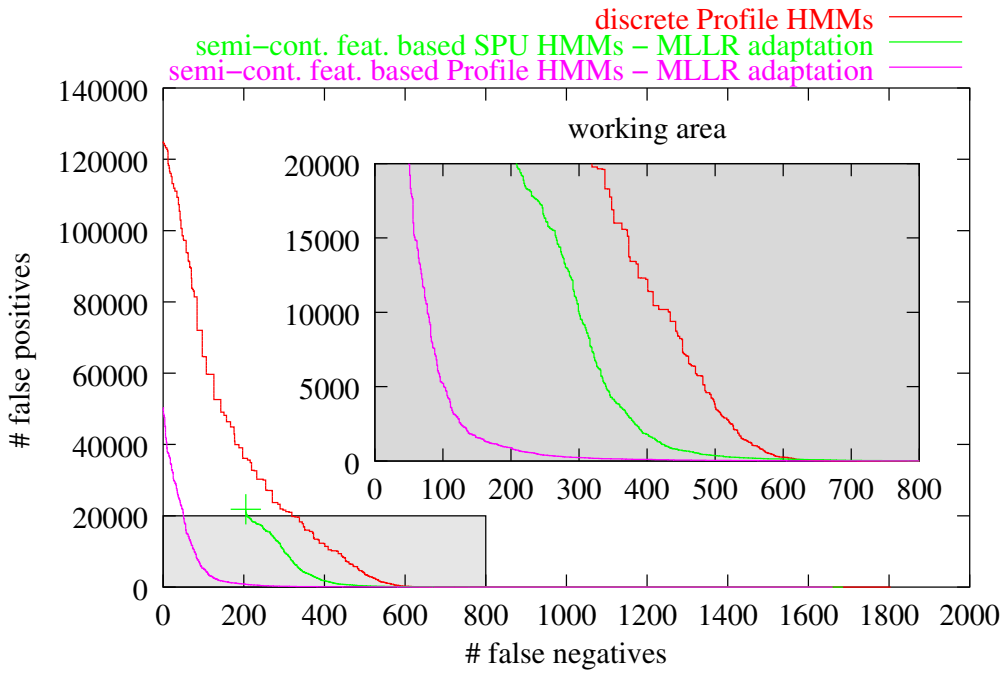


Figure 6.4: ROC curves illustrating the improved remote homology detection performance when applying SPU based protein family models (green curve) instead of current discrete Profile HMMs (red curve). The SPU based models are evaluated competitively to a single state UBM. False rejections are the reason for the curve's endpoint apart from the y-axis (marked with '+'). The ROC curves corresponding to semi-continuous feature based Profile modeling are given as reference (pink) illustrating their still better performance for detection tasks. Since the proof of concept for the SPU based modeling approach was addressed, their optimization potential is promising.

Modeling Variant	False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
Discrete Profile HMMs	26.1	57.6
SCFB Profile HMMs + UBM	5.1	5.5
SCFB SPU HMMs + UBM	18.2	0.0 (17.4)

Table 6.6: Characteristic values for SCOPSUPER95_66 detection experiments for Profile HMMs (discrete, and semi-continuous) and SPU based protein family models evaluated competitively to a single state UBM. The specificity as well as the sensitivity can substantially be improved compared to the state-of-the-art when using the SPU approach. For SCFB evaluations the corresponding limits of 5% false negative predictions were not reached. Thus, the appropriate global maxima at the end-points of the ROC curves are given in parentheses. Both feature based modeling variants include an MLLR adapted feature space representation.

SCOPSUPER95_44f: Evaluation of Remote Homology Classification

In addition to the general proof of concept for the applicability of SPU based protein family models which was given in the previous sections using the SCOPSUPER95_66 corpus, in the following the effectiveness of the new modeling approach for reduced training sets is evaluated. Again, the evaluation is concentrated on the general applicability of the new technique providing the proof of concept for the paradigm shift in protein sequence analysis using building blocks. Since the SPU framework can be configured and thus enhanced in various ways, the results presented give hints for further developments.

The SCOPSUPER95_44f corpus is used for the explicit assessment of the robustness of SPU based models depending on the number of training samples available. The first set of experiments is directed to the evaluation of the effectiveness of SPU based models for remote homology detection tasks. Therefore, in figure 6.5 the classification error rates are shown depending on the amounts of training samples used for model estimation. In the upper diagram the results for the original testset are given whereas in the lower chart the extended testset of SCOPSUPER95_44f is analyzed.

The actual training sets are obtained by randomly selecting sequences from the SCOP pool of the particular superfamily (cf. section 6.1 on page 151 for details about the corpus definition). Thus, certain "statistical noise" occurs when measuring the classification errors for the particular subsets of training samples. In order to allow easy comparison of the general effectiveness of the particular modeling methods for remote homology classification, the actual values are smoothed using Bezier interpolation. This results in continuous curves which can be analyzed by visual inspection as well as numerically for estimating the overall trend of the effectiveness.

It can be seen that SPU based protein family models are generally suitable for remote homology classification using small training sets up to a certain minimum amount of sample sequences. If the particular training sets contain at least (approximately) 20 sequences the classification error obtained for SPU based protein family models is comparable and slightly smaller, respectively, as when using current discrete Profile HMMs. Although the better performance of SCFB Profile HMMs cannot be reached yet, the results are very promising for the new concept of protein family modeling.

SCOPSUPER95_44f: Evaluation of Remote Homology Detection

The presentation of the classification performance given in the previous section provides an overview of the general effectiveness of SCFB SPU based protein family HMMs when only small amounts of training data are available. In order to prove the effectiveness for remote homology detection 44 ROC curves are necessary when proceeding similarly to the classification case. Since the presentation of such a large amount of diagrams for randomly selected training sets is in no relation to the knowledge gain which can be obtained from it, the following presentations are limited to three representative training sets of SCOPSUPER95_44f.

For the first set of experiments the subsets containing 20 training samples are selected whereas the second kind of training sets contain 30 sequences each. Finally, the upper limit of the number of training samples available is used, namely 44 sequences. Note that these

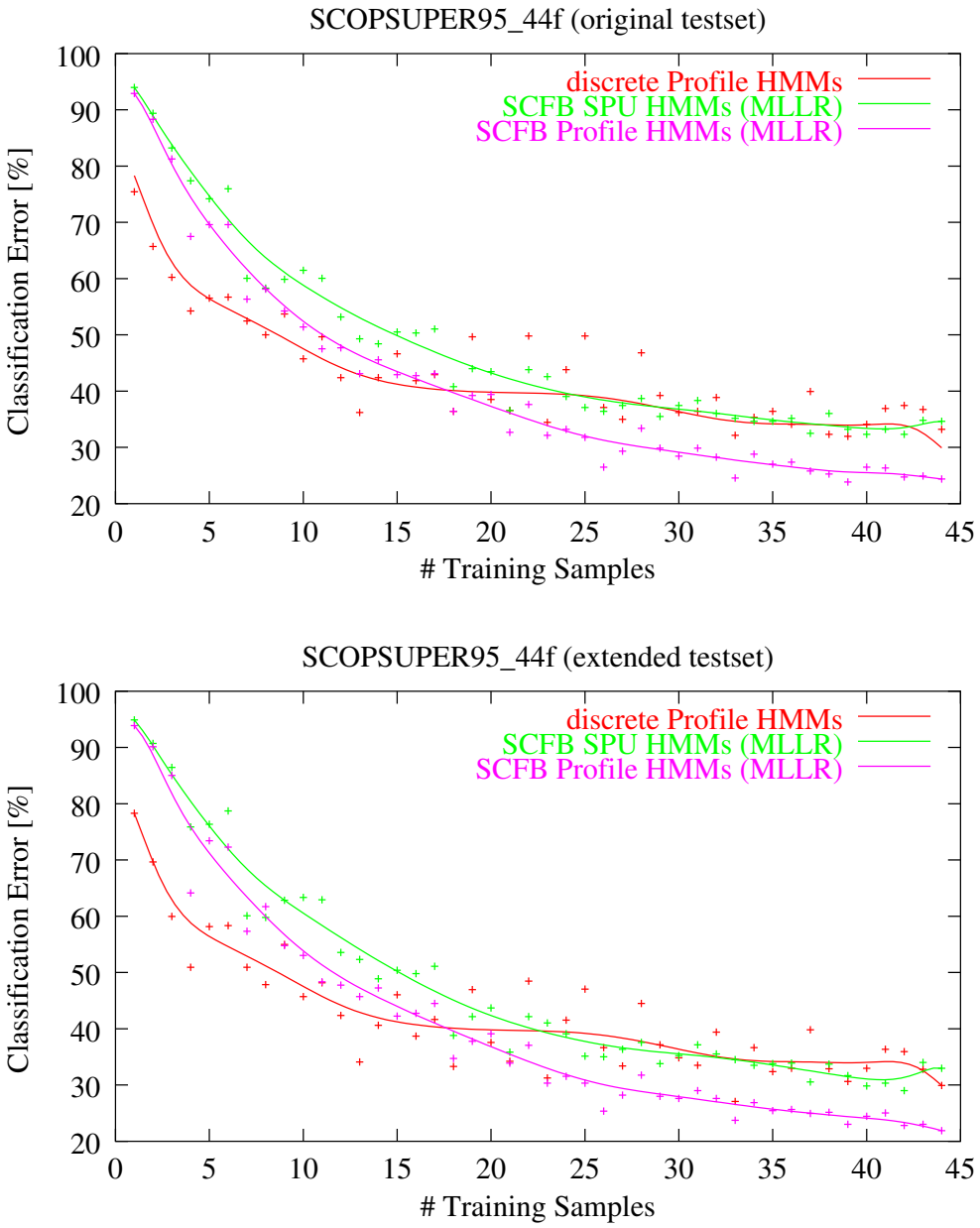


Figure 6.5: Classification error rates obtained when applying SPU based protein family models to the task of remote homology classification (SCOPSUPER95_44f). With respect to current discrete Profile HMMs, the diagrams illustrate the comparable and slightly reduced classification error rates obtained when a minimum of approximately 20 training sequences is available. Since the training samples are randomly picked the actual classification error rates (marked by '+') are smoothed using Bezier splines (solid lines). The results for semi-continuous feature based Profile HMMs are given here, too, illustrating their still better performance.

training sets are *not* identical to the ones from SCOPSUPER95_66 since the number of training samples is fixed to 44 for all 16 superfamilies (compared to the *minimum* number of 44 training sequences in SCOPSUPER95_66). The presentation of ROC curves is given in figures 6.6, and 6.7.

The particular ROC curves confirm the results obtained for classification experiments for remote homology detection. If less then the suggested minimum of approximately 20 training sequences are available, the detection performance of SPU based target models is worse compared to state-of-the-art discrete Profile HMMs (upper diagram of figure 6.6).

However, when 10 training samples more are used for model estimation the detection performance gets better on average and it is comparable to state-of-the-art. Furthermore, when 44 sample sequences are available for model training, SPU based protein family models in combination with a single state UBM significantly outperform discrete Profile HMMs. In fact 44 sequences are a reasonably small number. Thus, the SPU based modeling approach is very promising. For completeness in table 6.7 the characteristic values for fixed working points of the particular ROC curves are given.

Although the differences between SCFB Profile HMMs and SPU based protein family models are substantially, the evaluation of the detection performance for the new modeling approach is promising since they only represent the successful proof of concept. Further research activities should be directed to this new kind of protein family modeling in order to further improve their capabilities.

Modeling Variant	False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
20 Training Samples		
Discrete Profile HMMs	36.3	80.2
SCFB Profile HMMs + UBM	33.1	0.0 (13.6)
SCFB SPU HMMs + UBM	45.7	0.0 (8.6)
30 Training Samples		
Discrete Profile HMMs	31.6	78.2
SCFB Profile HMMs + UBM	19.0	0.0 (25.9)
SCFB SPU HMMs + UBM	34.0	0.0 (19.3)
44 Training Samples		
Discrete Profile HMMs	27.8	68.7
SCFB Profile HMMs + UBM	12.9	26.6
SCFB SPU HMMs + UBM	21.9	0.0 (29.1)

Table 6.7: Comparison of characteristic values for SCOPSUPER95_44f detection experiments (20, 30, and 44 training samples) at fixed working points for SPU based protein family models (MLLR adapted feature space representation) vs. Profile HMMs (discrete and semi-continuous feature based variants). For those values where the corresponding limit of 5% false predictions was not reached, the appropriate global maxima at the endpoints of the ROC curves are given in parentheses.

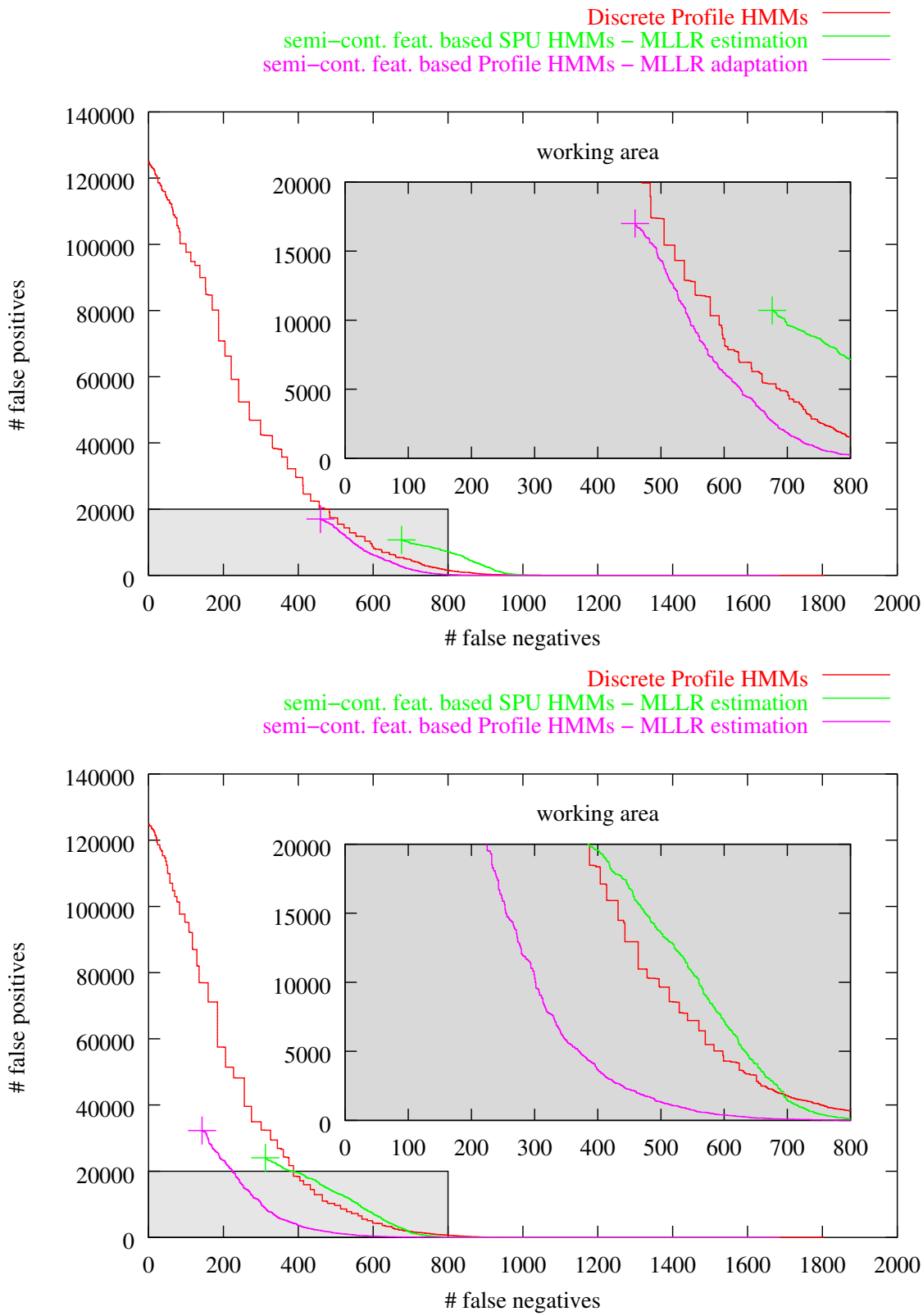


Figure 6.6: ROC curves illustrating the detection performance of SPU based target models competitively evaluated to a single state UBM for SCOPSUPER95_44f (upper diagram: 20 training sequences; lower diagram: 30 samples). A minimum of approximately 20 training sequences is required for reaching state-of-the-art (red), or outperforming it. The endpoints of the curves not crossing the y-axis caused by false rejections due to the UBM are marked with '+'.

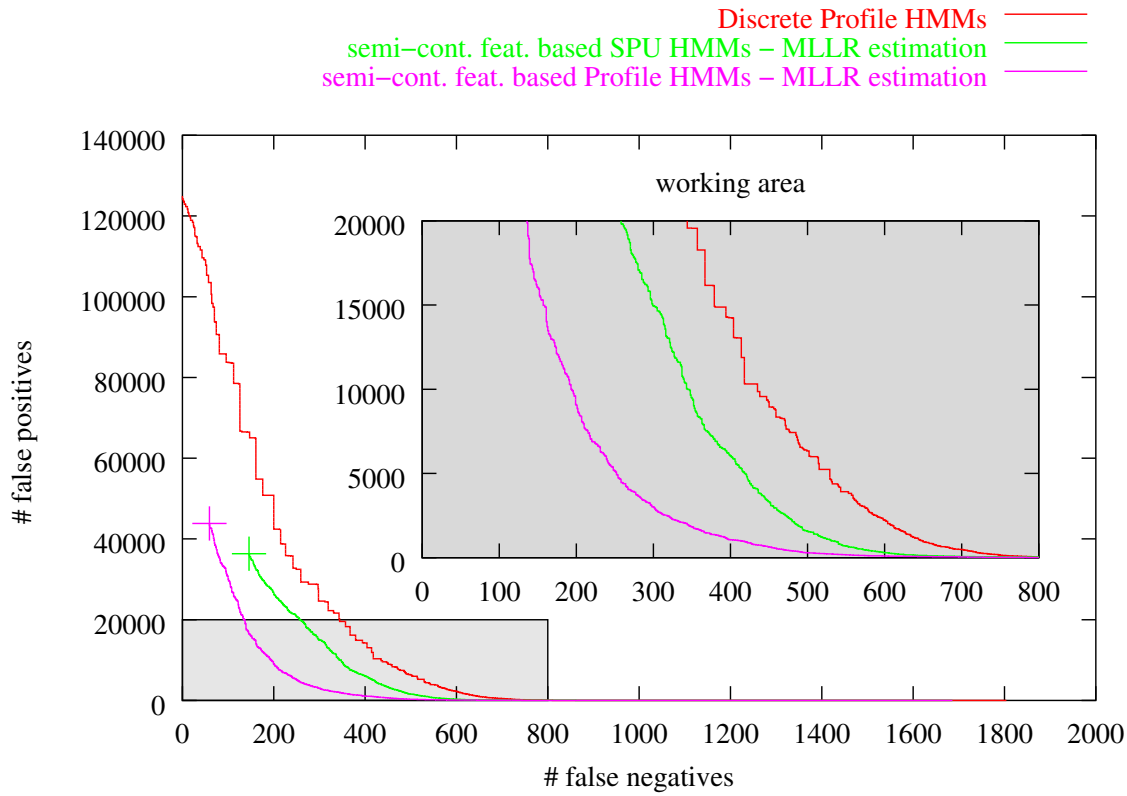


Figure 6.7: SCOPSUPER95_44f based comparison of detection results for the third set of experiments based on the use of 44 training samples each for the estimation of target family models. The combination of SCFB SPU models and single state UBMs performs best while obtaining small amounts of false rejections due to the UBM. Again the endpoints of UBM based curves not crossing the y-axis caused by false rejections are marked with '+'.

To summarize, the proof of concept for the general applicability of the new approach for protein family modeling using automatically derived building blocks was given. Both classification and detection performance of SPU based target models are improved in comparison to current discrete Profile HMMs when approximately 20 training sequences are available. If the amount of sample data is below this minimum the resulting SPUs are expected to be not as representative as necessary for the overall protein family. Thus, more flexibility of target models is required.

6.3.2 Effectiveness of Bounded Left-Right Models

Since the new feature based representation of protein sequences respects the biochemical properties of particular residues in their local neighborhood, the complex three-state model topology can be discarded. The reason for this is the incorporation of context information already at the level of HMM states' emissions and the flexible modeling using the Bounded Left-Right topology as developed in section 5.3.1 on pages 130ff. In the following, the results for the experimental evaluation of BLR protein family HMMs are presented.

Similarly to the evaluation of SPU based HMMs, the assessment of the effectiveness of the BLR approach is based on the SCOPSUPER95_66 as well as on the SCOPSUPER95_44f corpus. In the following the particular evaluation results are presented separately.

SCOPSUPER95_66: Evaluation of Remote Homology Classification

Table 6.8 contains the results of the experimental evaluation of the classification task performed for the SCOPSUPER95_66 corpus. The most effective variants of feature space adaptation as discussed in the previous section were applied and appropriate semi-continuous feature based Bounded Left-Right protein family HMMs were estimated – SCFB BLR HMMs (MAP/MLLR).

Analyzing the classification error rates of both SCFB Profile HMMs and SCFB BLR HMMs, it becomes clear that the new models with reduced complexity significantly outperform the feature based Profile models containing the standard three-state topology. For the best configuration, namely BLR HMMs based on the MLLR adapted feature space representation, the classification error decreases by approximately 20 percent relative. Compared to the corresponding state-of-the-art discrete Profile HMMs, this implies almost halving the classification error.

Note that the Bounded Left-Right model topology requires the new feature representation of protein sequences. In addition to the comparison of SCFB HMMs, in table 6.8 the classification error rate for discrete BLR models is given. Discrete BLR models perform significantly worse than their feature based counterparts and even worse than state-of-the-art discrete Profile HMMs. When processing discrete amino acid data, state-of-the-art discrete Profile HMMs are without doubt the methodology of choice.

Modeling Variant	Classification Error $\mathcal{E}$ [%]	Relative Change $\Delta\mathcal{E}$ [%] Base: Profile HMMs	
		Discrete	SCFB
Discrete Profile HMMs	32.9	–	–
SCFB Profile HMMs (MAP)	24.0	−27.1	–
SCFB Profile HMMs (MLLR)	20.7	−37.1	–
Discrete BLR HMMs	38.9	+15.4	–
SCFB BLR HMMs (MAP)	21.7	−34.0	−9.6
SCFB BLR HMMs (MLLR)	16.8	−48.9	−18.8

Table 6.8: Classification results for SCOPSUPER95_66 comparing Profile HMMs (both discrete and semi-continuous feature based) with protein family models with reduced model complexities based on the Bounded Left-Right architecture (mean confidence range: approximately ± 3.5%). Using the best configuration (BLR models including MLLR based feature space adaptation) the classification error can be further decreased by approximately 20 percent relative (compared to SCFB Profile HMMS) which implies halving the classification error obtained when using standard discrete Profile HMMs. The significantly worse results for discrete BLR models are given separately.

In figure 6.8 the transition probabilities of the SCOPSUPER95_66 SCFB BLR models are visualized using a greyvalue representation. According to the definition of Bounded Left-Right models (cf. page 130f.) the number of transitions varies for all models. Obviously,

the dominating state transition for all models is the direct connection between adjacent states (almost white stripes within all sub-images in the second rows). Additionally, the majority of states in all models contain non-vanishing transition probabilities to themselves and to farther adjacent states. It can be seen that the model topology is rather suitable since transitions to states which are not locally adjacent are very unlikely to occur (the lower rows of all sub-images are almost black).

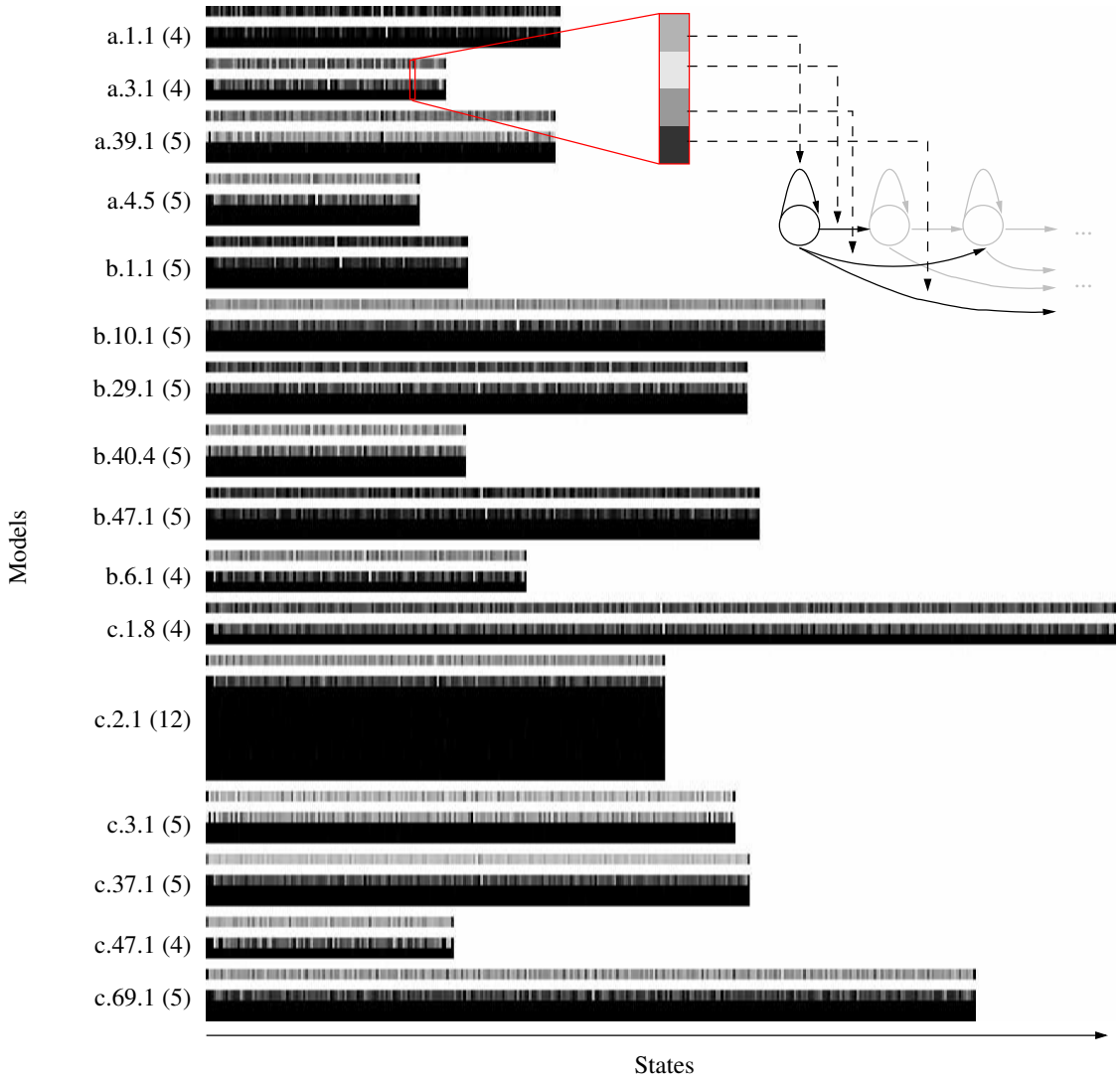


Figure 6.8: Visualization of transition probabilities for all superfamily BLR models of the SCOPSUPER95_66 corpus: Transition probabilities are mapped to greyvalues as illustrated by the inset (upper right); the lighter the shades, the higher the corresponding transition probabilities. The model names are written on the left side of the figure including the number of transitions created (cf. page 130f.) whereas the x-axis represents the states of the particular superfamily models.

SCOPSUPER95_66: Evaluation of Remote Homology Detection

In the previous section, evaluation results were presented which illustrate the superior performance of Bounded Left-Right HMMs for protein sequence classification. The effectiveness of the new modeling alternative for remote homology detection is now evaluated by means of the same methodology and datasets as previously mentioned for SCFB Profile HMMs. As suggested by the improved specificity of UBM based model evaluation (cf. figure 6.3), target BLR models based on either MAP, or MLLR based feature space adaptation were competitively evaluated to the single state Universal Background Model (UBM).

In figure 6.9 the results for the evaluation of the detection performance are presented by means of ROC curves. The curves for SCFB BLR models (blue and green) are directly compared to the curves of the corresponding SCFB Profile HMMs (cyan and pink). Furthermore, the baseline for all SCOPSUPER95_66 experiments discussed in this thesis, namely the ROC curve for discrete Profile HMMs, is shown in red. Note that some small amounts of false rejections are produced by the single state UBM. Thus, the corresponding ROC curves do not cross the y-axis.

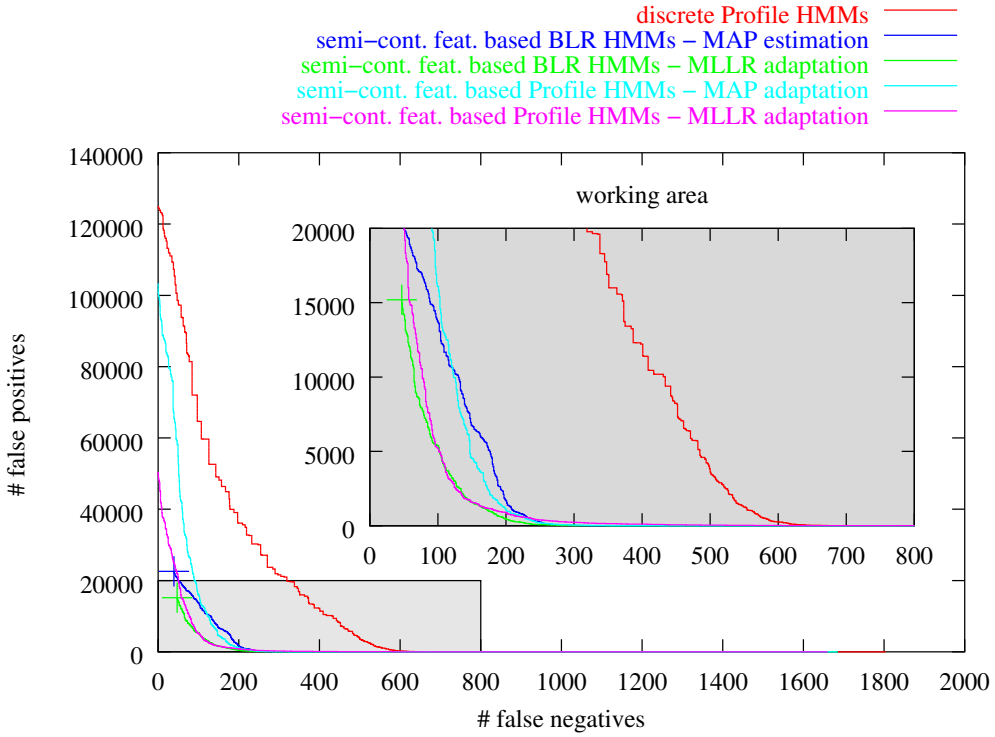


Figure 6.9: SCOPSUPER95_66 based comparison of the detection performance for both SCFB Profile HMMs (cyan, and pink ROC curves) and SCFB BLR models (blue, and green ROC curves) when evaluating the particular models competitively to an UBM. Both variants of the feature based BLR models show excellent performance for target identification tasks (red ROC curve: corresponding standard discrete Profile HMMs). Due to small amounts of false rejections produced by the single state UBM applied both BLR MAP and BLR MLLR curves do not cross the y-axis. For clarity the particular endpoints are marked with '+'.

It can be seen that Bounded Left-Right models are well suited for remote homology detection. The target models with reduced complexity, i.e. containing significantly less pa-

rameters which need to be trained, perform better than Profile HMMs. Thus, improved performance can be expected for smaller training sets, too. The combination of BLR target models and an UBM is superior for the SCOPSUPER95_66 based classification task.

In table 6.9 the corresponding percentages of false predictions for fixed points within the ROC curves allowing five percent false predictions are summarized. It becomes clear that when applying SCFB BLR protein family models to the practical task of remote homology detection for e.g. pharmaceutical applications where certain percentages of false classifications are allowed, the percentages of corresponding false classifications could have been decreased again (compared to the SCFB Profile HMMs).

Modeling Variant	False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
Discrete Profile HMMs	26.1	57.6
SCFB Profile HMMs (MAP)	7.9	16.0
SCFB Profile HMMs (MLLR)	5.1	5.5
SCFB BLR HMMs (MAP)	8.9	11.9
SCFB BLR HMMs (MLLR)	4.9	4.7

Table 6.9: Characteristic values for SCOPSUPER95_66 UBM based detection experiments: At the working points of 5 percent allowed false predictions only little corresponding false predictions are obtained when using the new SCFB BLR based protein families.

The evaluation of both classification and detection performance for SCFB BLR protein family HMMs using the SCOPSUPER95_66 corpus represents the successful proof of concept for the new modeling approach. Following this, the amount of training samples is explicitly reduced in order to evaluate the effectiveness of SCFB BLR HMMs for the sparse data problem. Therefore, experimental evaluations based on the SCOPSUPER95_44f corpus are presented.

SCOPSUPER95_44f: Evaluation of Remote Homology Classification

In order to illustrate the dependency of the classification error obtained for remote homology classification on the amount of training material available for model estimation, in figure 6.10 these values are presented for both SCFB Profile HMMs and SCFB BLR models. The baseline for discrete Profile HMMs is shown for completeness as well. As previously mentioned, generally two test sets exist for the SCOPSUPER95_44f corpus. Thus, the results are shown for both of them in separate diagrams.

Inspecting the curves in both diagrams of figure 6.10 it becomes clear that SCFB protein family HMMs show superior performance for almost the whole range of training subsets. Only for very small sample sets (less than five sequences), discrete Profile HMMs relatively outperform the new techniques. However, in this area the absolute classification error is out of any reasonable range (more than 60 percent). When applying SCFB BLR protein family HMMs the classification error can be halved for almost all subsets of training samples compared to the SCFB Profile HMMs.

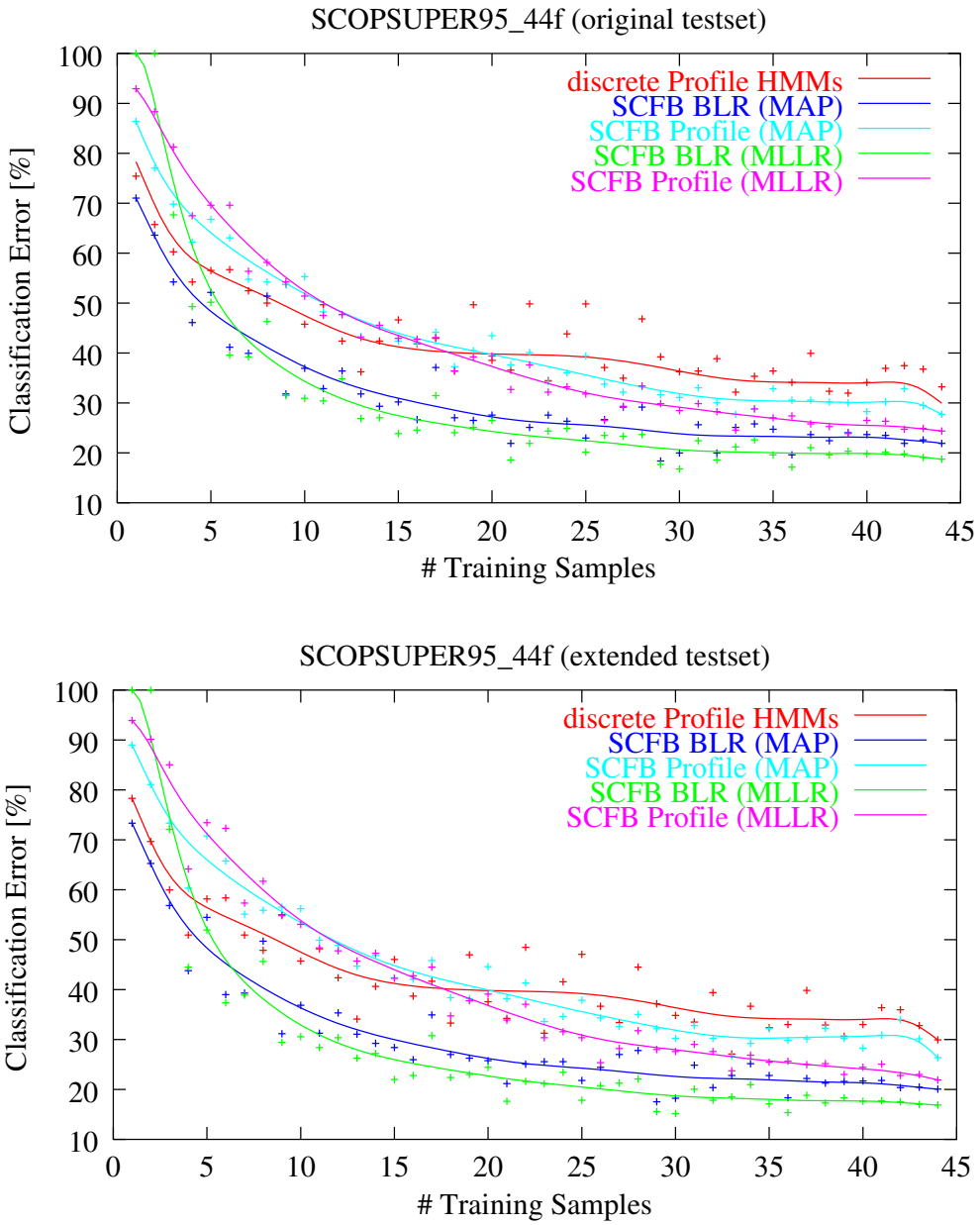


Figure 6.10: Illustration of the SCOPSUPER95_44f based classification errors (top: original testset; bottom: extended testset) depending on the number of training samples used for model training for Profile HMMs (MAP, and MLLR based SCFB, and discrete) as well as for BLR models. The models with reduced complexity show superior classification performance especially when less training material is available. Since the training samples are randomly picked the actual classification error rates (marked by '+') are smoothed using Bezier splines (solid lines).

SCOPSUPER95_44f: Evaluation of Remote Homology Detection

Following the assessment of the classification capabilities of SCFB BLR protein family HMMs in the following their effectiveness for detection tasks is considered. In analogy to the argumentation given on page 162, the presentation of ROC curves is limited to three kinds of experiments.

In figures 6.11 and 6.12 the detection results are presented by means of ROC curves illustrating the mutual dependencies of false predictions. All diagrams contain ROC curves for discrete Profile HMMs (serving as baseline reference), SCFB Profile HMMs evaluated competitively to a structured UBM, SCFB BLR HMMs which are applied in combination with an unstructured UBM, and the results for standalone evaluation of SCFB BLR HMMs, i.e. not competing with any kind of UBMs.

In the diagrams the improved performance of advanced stochastic protein family modeling techniques for remote homology detection can be seen even for small training sets. The ROC curves corresponding to semi-continuous feature based approaches lie almost everywhere below the reference curves for state-of-the-art discrete Profile HMMs.

Furthermore, it can be seen that SCFB Bounded Left-Right protein family models outperform their Profile topology based counterparts when evaluating them competitively to an UBM. Note that the specificity of this model combinations degrades proportional to the decreasing numbers of training samples used. The smaller the number of training samples, the larger the percentage of false negative predictions. In the diagrams this is illustrated by the positions of the endpoints of the particular ROC curves not crossing the y-axis (marked with '+'). The larger the horizontal distance of the endpoints from the point of origin, the larger the number of (fixed) false rejections.

However, the number of false positive predictions which is extremely relevant for e.g. pharmaceutical applications due to enormous costs linked to further expensive analysis of erroneously selected candidates (e.g. in wet-lab experiments) can substantially be reduced. In addition to identifying the limits of advanced stochastic modeling approaches developed in this thesis, the experiments clearly highlight their substantial benefits: Already when using 20 training samples (which is in fact a very small number) great improvements for remote homology detection tasks can be obtained. Further reductions of the amounts of training samples makes SCFB BLR protein family models' capabilities tend to the detection performance of current discrete Profile HMMs including slight improvements.

Similar to the presentations given in previous sections, in table 6.10 the characteristic values of the particular ROC curves which concretely specify the mutual dependencies of false predictions at reasonable working points are summarized. Due to the great effectiveness of the UBM the limit of five percent false positive predictions is often not reached. When keeping consistency regarding the definition of the characteristic values, the percentage of corresponding false negative predictions is 0.0. However, due to the (occasionally obtained) larger numbers of false rejections caused by the UBM, additionally, the percentages of these false rejections are given in parentheses. In analogy to the argumentation given above, when not reaching the limit of five percent false negatives, the percentage of false positive predictions for the endpoint of the particular ROC curve is given in parentheses, too.

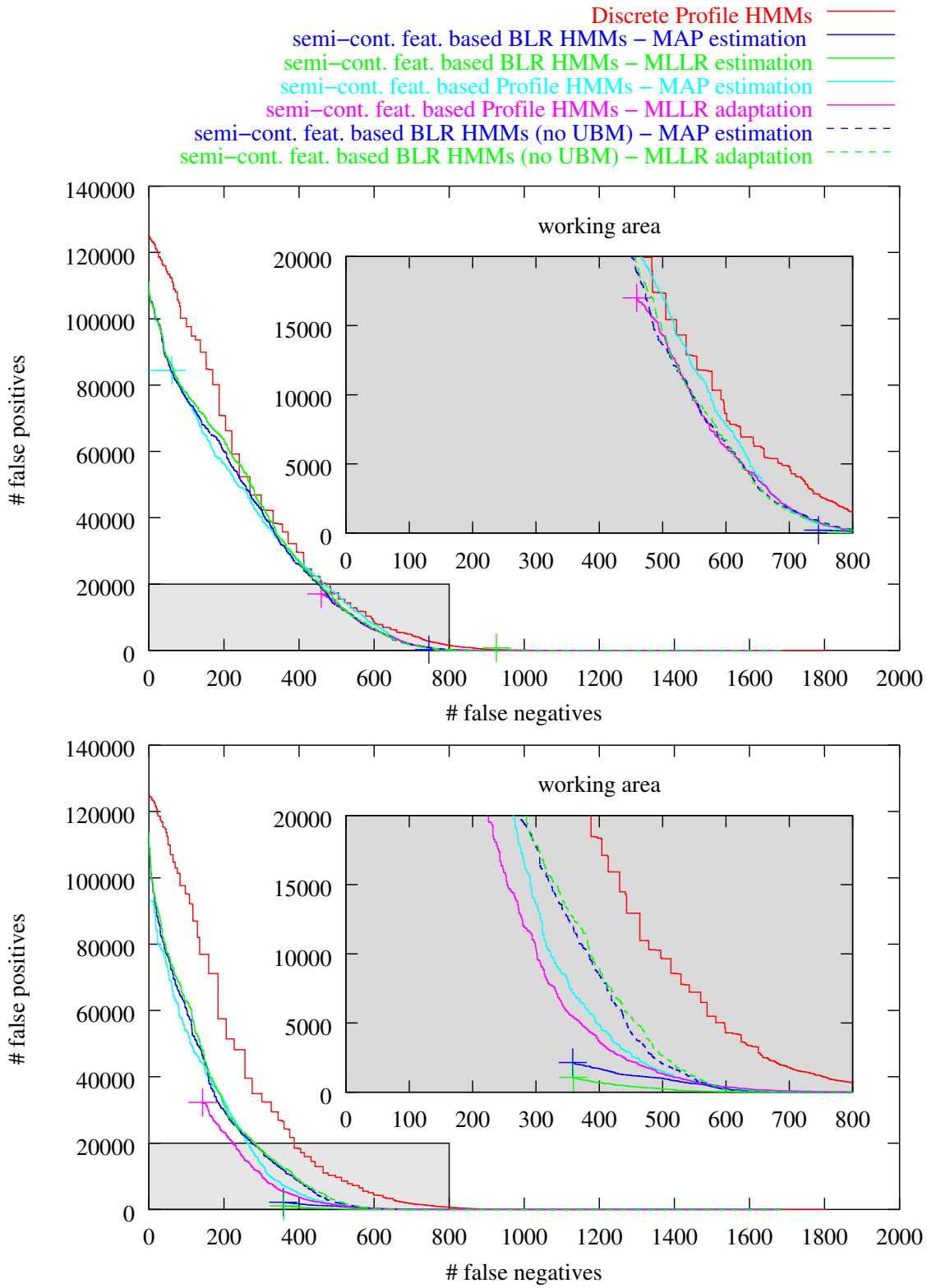


Figure 6.11: Detection results for SCOPSUPER95_44f experiments based on target family models estimated using 20 (upper diagram), and 30 training samples (lower diagram). Even for small training sets, new modeling techniques outperform discrete Profile HMMs (red). The endpoints of UBM based curves not crossing the y-axis caused by false rejections due to the UBM are marked with '+'.

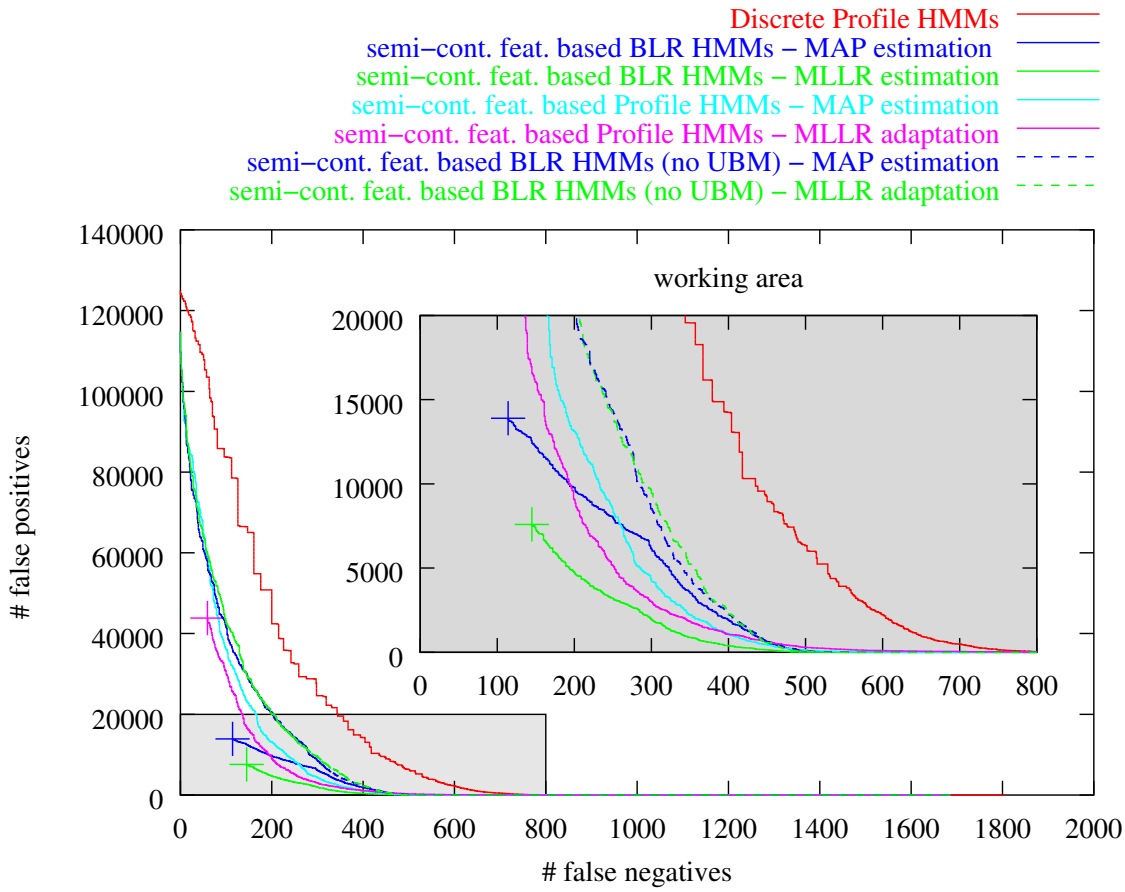


Figure 6.12: SCOPSUPER95_44f based comparison of detection results for the third set of experiments based on the use of 44 training samples each for the estimation of target family models. The combination of SCFB BLR models and single state UBMs performs best while obtaining small amounts of false rejections due to the UBM. Again the endpoints of UBM based curves not crossing the y-axis caused by false rejections are marked with '+'.

To summarize, semi-continuous feature based protein family HMMs with Bounded Left-Right topology are well suited for remote homology detection tasks where only little training data is available. Especially in combination with a single state UBM substantial reductions of false positive prediction rates are achievable which is relevant for e.g. pharmaceutical applications. Generally, advanced stochastic models of protein families as developed in this thesis outperform current discrete Profile HMMs particularly when only small sets of training samples are available.

6.4 Acceleration of Protein Family HMM Evaluation

The analysis of the general procedure for evaluating current Profile HMMs when performing protein sequence classification, or homology detection, showed that state-of-the-art techniques are rather straightforward. Especially when applying huge numbers of large protein family models to current databases which consist of tens of thousands of entries, inef-

Modeling Variant	False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
20 Training Samples		
Discrete Profile HMMs	36.3	80.2
SCFB		
Profile HMMs (MAP) + UBM	34.5	62.2
Profile HMMs (MLLR) + UBM	33.1	0.0 (13.6)
BLR HMMs (MAP)	33.5	62.1
BLR HMMs (MLLR)	31.8	63.2
BLR HMMs (MAP) + UBM	0.0 (41.4)	0.0 (0.2)
BLR HMMs (MLLR) + UBM	0.0 (51.3)	0.0 (0.6)
30 Training Samples		
Discrete Profile HMMs	31.6	78.2
SCFB		
Profile HMMs (MAP) + UBM	20.9	45.0
Profile HMMs (MLLR) + UBM	19.0	0.0 (25.9)
BLR HMMs (MAP)	23.7	50.5
BLR HMMs (MLLR)	24.2	51.9
BLR HMMs (MAP) + UBM	0.0 (19.8)	0.0 (1.7)
BLR HMMs (MLLR) + UBM	0.0 (19.9)	0.0 (0.9)
44 Training Samples		
Discrete Profile HMMs	27.8	68.7
SCFB		
Profile HMMs (MAP) + UBM	15.1	31.4
Profile HMMs (MLLR) + UBM	12.9	26.6
BLR HMMs (MAP)	18.0	35.4
BLR HMMs (MLLR)	18.7	38.3
BLR HMMs (MAP) + UBM	16.6	0.0 (11.1)
BLR HMMs (MLLR) + UBM	9.3	0.0 (6.1)

Table 6.10: Characteristic values for SCOPSUPER95_44f detection experiments (20, 30, and 44 training samples): When applying SCFB BLR HMMs for small training sets negligible percentages of corresponding false predictions are obtained for the fixed working points of 5% allowed false predictions. For those values where the corresponding limit of 5% false predictions was not reached, the appropriate global maxima at the endpoints of the ROC curves are given in parentheses.

ficient model evaluation is problematic. Currently, massive parallelization is the only optimization technique used for the acceleration of the model evaluation process. Furthermore, the computational effort required for the evaluation of the advanced stochastic techniques developed in this thesis increases due to the additional step of mixture density evaluation which is performed for semi-continuous Hidden Markov Models.

In section 5.4 different approaches for *algorithmic* accelerations of protein family model evaluation were discussed. In this section, the results of the experimental evaluations performed are presented. Since the general behavior of model evaluation is independent of the datasets which are actually used (as long as they are relevant in terms of the goal the thesis at hand is addressing) the benchmark of the efficiency of the particular techniques is limited

to the SCOPSUPER95_66 corpus. For systematic assessments the remote homology classification task is analyzed. Note that due to the *principal* similarity of the characteristics of all corpora used in this chapter the results obtained for SCOPSUPER95_66 can be generalized.

All experiments addressing the optimization of model evaluation were performed on COMPAQ Professional Workstations XP1000 with 21264 CPUs and 500 MHz clock speed running under OSF1 Tru64 Unix v4.0. However, most considerations are based on relative measurements of e.g. the number of HMM states explored which are in fact independent of the actual CPU speed.

6.4.1 Effectiveness of State-Space Pruning

When estimating stochastic models for complete protein families as addressed by this thesis, rather large models are usually created. Currently, these models are evaluated more or less straightforwardly since the whole state space is explored during Viterbi decoding.[3] As discussed in section 5.4.1, the situation is different in alternative pattern recognition domains. Here, state space pruning techniques are applied resulting in substantial savings of computational effort while keeping the classification error almost constant. The most promising technique is the Beam-Search algorithm aiming at limiting the Viterbi decoding to the most promising paths only (cf. pages 142ff).

In this section the effectiveness of Beam-Search pruning for protein sequence analysis is investigated. For the general proof-of-concept the state-space pruning technique is initially applied to discrete, i.e. state-of-the-art, Profile HMMs. Furthermore, for current protein family models the effectiveness of the combined model evaluation approach is investigated which is especially relevant for remote homology classification tasks. However, target detection applications can also benefit from it since the most promising results were obtained when competitively evaluating target models with an UBM, i.e. more than one model. Following this, the effectiveness of the Beam-Search algorithm for SCFB BLR models is evaluated. Note that the sub-optimal character of Beam-Search based model evaluation implies increasing classification error rates when limiting the state space explored improperly.

In figure 6.13 the percentages of states explored are compared to the corresponding classification error achievable when limiting the exploration of the state-space accordingly. It can be seen that for both separate and combined model evaluation large parts of the state-space do not need to be explored. For the separate evaluation of all 16 target models of SCOPSUPER95_66 the state-space exploration can be limited by 40 percent thereby keeping the classification error at the same level as for the baseline experiment where the whole state-space is explored (as in the state-of-the-art procedure). Further savings can be obtained when combining the 16 models into a common state-space. Here, more than 60 percent of the original combined state-space does not need to be explored.

For advanced stochastic modeling techniques developed in this thesis even more savings can be obtained. In figure 6.14 the evaluation of the effectiveness of Beam-Search pruning for the combined evaluation of SCOPSUPER95_66 SCFB BLR HMMs is visualized as in the preceding diagram. It can be seen that a limitation of the state-space exploration to approximately one fourth is sufficient for reaching the minimum of classification error achievable. Note that the absolute classification error of SCFB BLR HMMs is significantly

[3] The same remains true when performing Forward-Backward model evaluation.

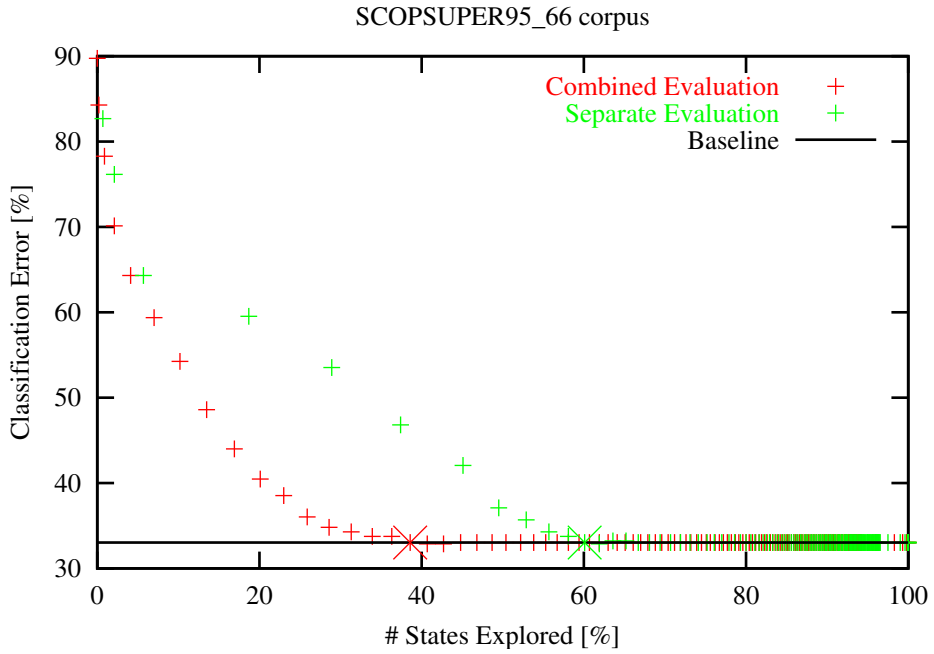


Figure 6.13: Effectiveness of state-space pruning when applying the Beam-Search algorithm to the evaluation of discrete Profile HMMs for SCOPSUPER95_66. If evaluating all models separately (green '+'-signs) only 60 percent of the overall state-space need to be explored for achieving the same classification error as when exploring the whole state-space (baseline: black solid line). Furthermore, for the combined model evaluation only less than 40 percent of the whole state-space need to be explored (red '+'-signs).

lower than the one for current discrete Profile HMMs (cf. table 6.8), and that the size of the overall state-space is significantly smaller. Whereas discrete Profile HMMs for SCOPSUPER95_66 consist of approximately 8 800 states, about 3 000 states are sufficient for SCFB Bounded Left-Right protein family models.

The reduction of the number of states to be explored for successful remote homology analysis is favorable for all kinds of modeling. Even existing state-of-the-art approaches can immediately benefit from it. Additionally, since single model evaluation can principally be accelerated by the Beam-Search algorithm, parallelization approaches can be further accelerated, too. Furthermore, when applying advanced stochastic models, state-space pruning is of major importance since for every state explored a set of mixture components needs to be evaluated. Although efficient techniques are used for this, it is the limiting factor for high-throughput applications. Thus, the smaller the state-space which actually needs to be explored, the faster the overall process of remote homology analysis.

6.4.2 Effectiveness of Accelerated Mixture Density Evaluation

The central aspect of advanced stochastic protein sequence analysis addressed by this thesis is the feature based representation of protein data. Contrary to current Profile HMMs, therefore, semi-continuous modeling techniques were applied instead of discrete approaches. For every state of an HMM an additional third stochastic stage is performed, namely the evalu-

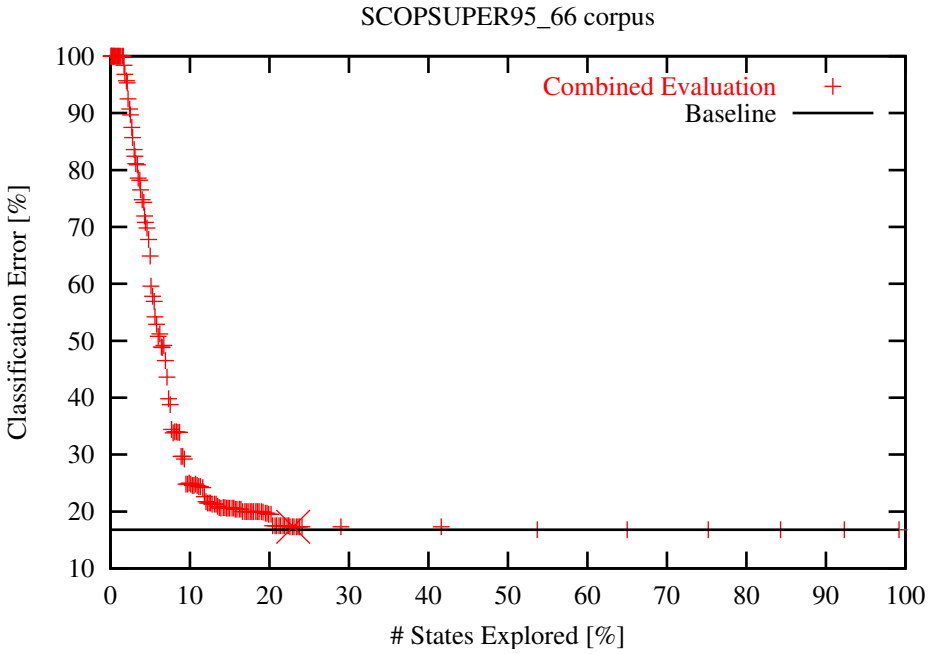


Figure 6.14: Effectiveness of Beam-Search based state-space pruning when applying SCFB BLR HMMs to SCOPSUPER95_66 based remote homology classification. Only approximately 25 percent of the whole state-space need to be explored for achieving the same classification error as the reference experiment where the complete state-space is evaluated (black solid line).

ation of mixture components representing the underlying feature space.

The evaluation of a mixture density is one substantial factor for the efficiency of the application of (semi)-continuous HMMs. Generally, when calculating density values for all 1024 mixture components (which is the number of components used for feature space representation) even in case of limiting the number of state explorations reasonably by applying the Beam-Search procedure, high-throughput database screening becomes impossible due to the enormous computational effort. Thus, further optimizations were discussed in section 5.4.3 whose effects are presented in the following.

Within the ESMERALDA system mixture density evaluation is principally performed using the efficient estimation technique described on page 144f. Applying this technique optimally limits the calculations. Furthermore, the Beam-Search idea is generalized to the mixture density evaluation. Informal experiments proved that the number of mixture components which need to be explored can be drastically reduced. Only approximately *one percent* of all mixture components need to be evaluated on average for successful remote homology classification. This implies strong compactness and locality of the protein data feature space.

Furthermore, according to the argumentation given on page 145f. the mixture classifier can be decomposed into multiple stages. In combination with the Beam-Search algorithm for mixture density pruning only small numbers of mixture components actually need to be explored for the whole 99 dimensional feature vectors. Informal experiments proved that more than two stages do not substantially influence the efficiency of mixture density

evaluation. However, the number of feature vector components used for the first stage of the classifier needs to be evaluated. Therefore, in figure 6.15 the number of components used for the first stage of the classifier is compared to the overall percentage of computational time required for the complete process of mixture density classification, i.e. including the second stage where complete feature vectors are used. Note that the diagram contains only those configurations where the resulting classification error does not significantly differ from the baseline result of 16.8% (the average confidence range for SCOPSUPER95_66 is approximately $\pm 3\%$). Further dimension reduction for the first stage of the classifier implies significant increase of the classification error.

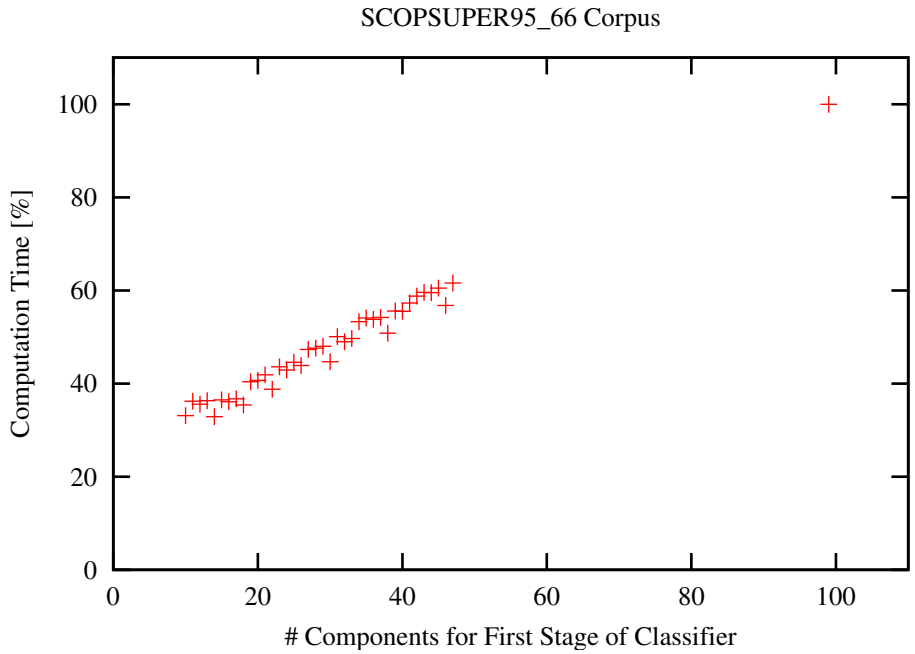


Figure 6.15: Effectiveness of the application of two-stage mixture density classifiers and appropriate Beam-Search pruning for SCFB BLR HMMs based SCOPSUPER95_66 classification experiments. The computation time for the classification can be reduced by more than 60 percent when using 20 of 99 feature vector components for the first stage of the classifier.

It can be seen that the two-stage mixture density classifier is very effective. The computational time can be reduced by more than 60 percent when using 20 feature vector components for the first classification stage and applying the Beam-Search algorithm for mixture components reasonably.[4]

By means of all optimization techniques used for accelerated model evaluation the computation times for SCOPSUPER95_66 classification experiments are as follows: Compared to the complete state-space evaluation of discrete Profile HMMs which takes approximately 23.9 minutes, the application of state-space pruning reduced the evaluation time to approximately 10.3 minutes. When applying the new advanced stochastic modeling techniques, the computation time required for decreasing the SCOPSUPER95_66 classification error by almost one half relative is approximately 27.8 minutes.

[4] In fact ESMERALDA's default pruning parameter for mixture density Beam-Search is applied, namely $-\ln(B) = 14$.

To summarize, due to efficient model evaluation techniques evaluated in this section, advanced stochastic protein family models can be evaluated in comparable time as discrete Profile HMMs. However, both classification and detection performance are improved.

6.5 Combined Evaluation of Advanced Stochastic Modeling Techniques

In the previous sections, the new techniques developed in this thesis were evaluated separately. Following this, the configuration of advanced stochastic protein family models which turned out to be most suitable for remote homology analysis is used for the combined evaluation. The final configuration of the particular protein family HMMs is as follows:

- Feature based protein data processing,
- Bounded Left-Right model topology,
- Competitive evaluation of target model and single state UBM,
- Semi-continuous Hidden Markov Models for protein family modeling,
- k-means based estimation of the general protein data feature space using approximately 90 000 sequences from SWISSPROT (resulting in 1024 mixture components),
- MLLR, or MAP based specialization of the protein data feature space, and
- Beam-Search based state-space and mixture density pruning (including a two-stage classifier).

For evaluation, the SCOPSUPER95_20 corpus as well as the PFAMSWISSPROT corpus are used whereas for SCOPSUPER95_20 the baseline results for the reference evaluation using state-of-the-art discrete Profile HMMs is given, too.

Evaluation based on the SCOPSUPER95_20 corpus

The first set of experiments performed for the combined evaluation of advanced stochastic modeling techniques is generally comparable to the evaluations described so far. The SCOPSUPER95_20 corpus is used for both remote homology classification and remote homology detection tasks.

In table 6.11 the results for the classification task are presented. Compared to current discrete Profile HMMs both variants of SCFB BLR HMMs, based on either MAP-, or MLLR-specialization of the general protein feature space, perform significantly better. The classification error can be decreased by approximately one fourth relative.

In figure 6.16, and table 6.12 the results for the remote homology detection based evaluation for SCOPSUPER95_20 are presented. The ROC curves illustrate the superior performance of the new approaches for protein family modeling compared to the state-of-the-art. Furthermore, the effectiveness of explicit background models can be seen since the UBM

Modeling Variant	Classification Error $\mathcal{E}$ [%]	Relative Change $\Delta\mathcal{E}$ [%]
Discrete Profile HMMs	29.5	–
BLR (MAP)	26.4	-10.5
BLR (MLLR)	22.6	-23.4

Table 6.11: Classification results for SCOPSUPER95_20 comparing discrete Profile HMMs with the protein family models as developed in this thesis in their final parametrization (semi-continuous feature based type, Left-Right model architecture, 1024 mixture components for feature space representation generally obtained from SWISSPROT – adapted using MLLR/MAP). The enhanced protein family models significantly outperform their discrete counterparts (average confidence range: approximately $\pm$ 2.4%).

based curves are below the corresponding curves for standalone target model evaluation. Note that only very few false positive predictions are obtained while reasonably limiting the percentage of false negative predictions.

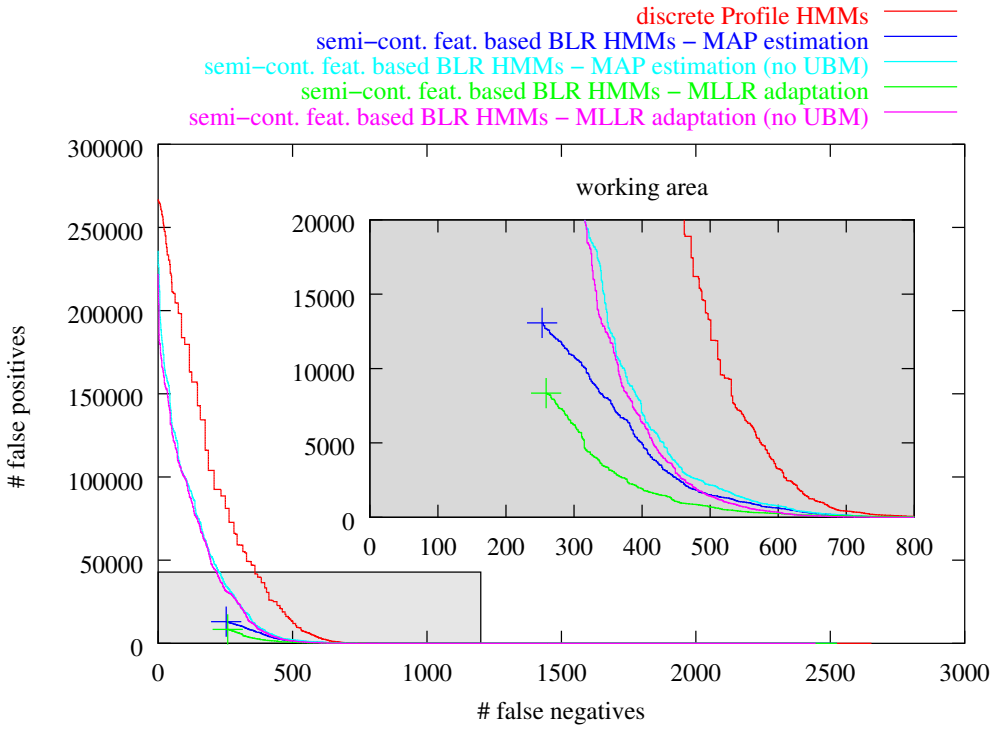


Figure 6.16: Detection results for the final evaluation experiments using the SCOPSUPER95_20 corpus. SCFB BLR models based on MAP (blue ROC curve), or MLLR adaptation (green curve) which are competitively evaluated to the single state UBM show superior performance compared to the state-of-the-art discrete Profile HMMs (red). Additionally, the ROC curves for standalone evaluation of SCFB BLR models, i.e. not applying any UBM, are given illustrating the effectivity of the background model for remote homology detection (cyan and pink curves). Due to small amounts of false rejections produced by the single state UBM applied both BLR MAP and BLR MLLR curves do not cross the y-axis. They are marked with '+' for clarity.

Modeling Variant	False Negative Predictions [%] for 5 % False Positives	False Positive Predictions [%] for 5 % False Negatives
Discrete Profile HMMs	18.7	58.9
SCFB BLR HMMs (MAP)	13.2	32.4
SCFB BLR HMMs (MLLR)	12.8	29.8
SCFB BLR HMMs (MAP) + UBM	0.0 (14.0)	4.9
SCFB BLR HMMs (MLLR) + UBM	0.0 (14.4)	3.1

Table 6.12: Characteristic values for SCOPSUPER95_20 UBM based detection experiments: At the working points of 5 percent allowed false positive/negative predictions the percentages of corresponding false negative/positive predictions obtained are substantially smaller for SCFB BLR models compared to state-of-the-art discrete Profile HMMs.

The characteristic values of false predictions at fixed working points within the ROC curves (given in table 6.12) illustrate the relevance of advanced stochastic protein family models especially for e.g. pharmaceutical purposes.

Evaluation based on the PFAMSWISSPROT corpus

The final set of evaluations is directed to some kind of "real-world" scenario. The PFAMSWISSPROT corpus consists of three exemplary protein domains which were picked from the Pfam database [Son98]. The actual selection of the particular protein domains corresponds to typical investigations performed for pharmaceutical research. Advanced stochastic models were estimated using the appropriate sets of training sequences.

Once the models are estimated, remote homology detection is performed for the approximately 90 000 sequences of the SWISSPROT database. Using the techniques and the configuration which proved most effective in the numerous preceding experimental evaluations (cf. page 180), and fixed thresholds determined from previous detection experiments, hard decisions regarding target hit or miss are performed. Generally, this is a reasonable procedure for e.g. pharmaceutical research.

In table 6.13 the corresponding results are summarized. The percentages for correct, and false predictions are calculated using the appropriate overall number of occurrences within SWISSPROT. Since SWISSPROT contains sequences for complete proteins, where the particular domains are actually only parts of them, local alignments are calculated. Analyzing the prediction results it becomes clear that advanced stochastic protein sequence models are suitable for real world scenarios of remote homology detection.

Additionally, in figure 6.17 the annotations of two SWISSPROT sequences corresponding to Pkinase hits are given. The comparison of the references annotation provided by the Pfam database and the one obtained when using the GRAS^{2}P system, i.e. advanced stochastic protein family models, turns out that real world applications can greatly benefit from the developments described in this thesis.

Pfam Id	Pfam Name	# Occurrences in SWISSPROT	# Predictions (Percentage) Correct	False
PF00001	7tm_1 GPCR 7 Transmembrane Receptor	1078	1024 (95.0%)	54 (5.0%)
PF00005	PKinase Protein Kinase Domain	507	493 (97.2)	14 (2.8%)
PF00069	ABC_tran ABC Transporter	1202	1090 (90.7%)	112 (9.3%)

Table 6.13: Summary of the detection results for the PFAMSWISSPROT corpus. For all exemplary domains, the percentages of correctly predicted occurrences within the approximately 90 000 SWISSPROT sequences of complete proteins is satisfactorily high. Note that the false predictions contain almost exclusively false negative predictions, i.e. the number of false positive predictions can almost be neglected.

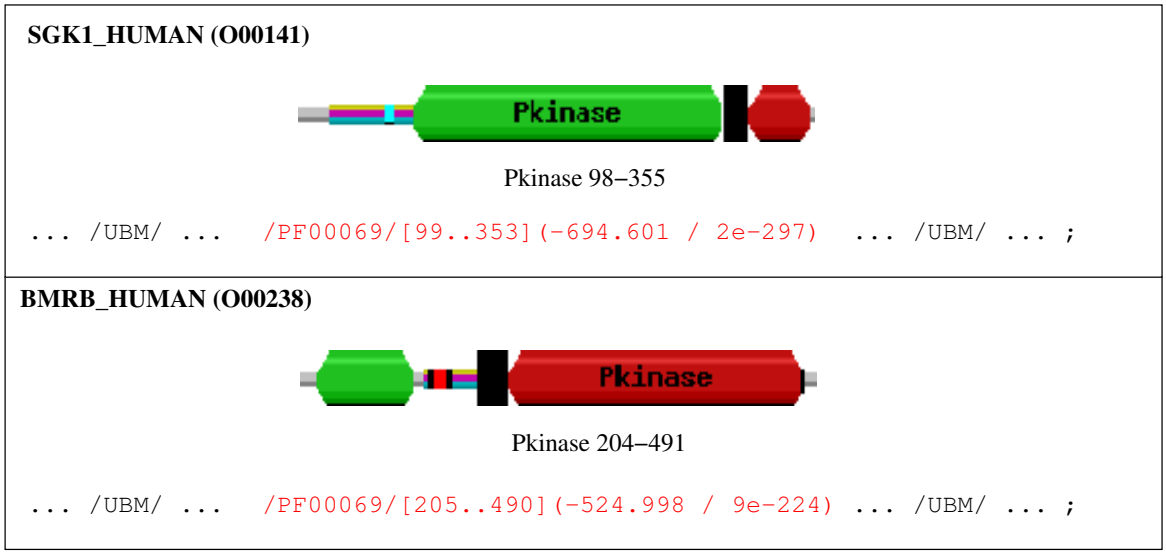


Figure 6.17: Illustration of the capabilities of advanced stochastic protein family models for detection tasks: Two exemplary target hits of SWISSPROT protein sequences containing the Pkinase domain (PF00069) are shown including the predicted localization of the target. According to the reference annotation provided by the Pfam database [Son98] and represented by the images given at the top of every frame, the positions of the domains within the protein sequences are predicted almost perfectly. The annotation provided by the GRAS^{2}P system are given below the reference annotation including the domain positions (square brackets) and the E-values of the particular models based on log-odd scores as negative log-likelihoods (given in paranthesis).

6.6 Summary

The three fundamental issues formulated for the successful application of advanced stochastic protein family models (cf. page 92) address the general performance improvement for both classification and detection applications, robust model estimation even when only little training data is available, and efficient model evaluation. Following the description of the developments directed to these constraints for new modeling techniques, in this chapter the particular approaches were extensively evaluated in a large number of experiments. For reliable conclusions regarding the effectiveness of the new approaches various data sets were analyzed aiming at separate evaluations of the new approaches. Furthermore, a combined assessment of these methods in their final configuration for typical sequence analysis tasks within molecular biology research was performed.

The concept of feature based protein sequence analysis using HMMs is the fundamental approach developed in this thesis. Since all further developments are based on the richer representation of biological data, initially the effectiveness of the new features was evaluated. Therefore, the model topology of state-of-the-art Profile HMMs, namely the complicated three-state architecture, was kept fixed whereas the base for emissions of the particular HMM states was changed to the features explicitly covering biochemical properties of residues in their local neighborhood. The resulting semi-continuous feature based Profile HMMs proved to be very effective for both remote homology classification and detection tasks. For the representative SCOPSUPER95_66 corpus which consists of sequences for 16 superfamilies the classification error rate could be decreased by more than one third relative (compared to the state-of-the-art). By means of this evaluation it could be seen that ML based feature space specialization is usually not suitable since too many training samples are required for robust adaptation. Instead, the application of MAP as well as MLLR adaptation greatly improves the quality of Profile HMMs. The significantly improved performance of feature based protein family models could be proved for detection tasks, too. In combination with an UBM which explicitly covers all non-target specific protein data, both specificity and sensitivity of stochastic protein family models could be improved drastically.

In order to tackle the so-called sparse data problem, protein family models with reduced complexity were developed. The experimental evaluation for both variants, namely global Bounded Left-Right models and protein family models created by concatenating automatically derived Sub-Protein Units, showed that models including less parameters which need to be trained can be estimated even when only little training data is available. Especially semi-continuous feature based BLR models are very effective. The SCOPSUPER95_66 based classification error could almost be halved compared to current discrete Profile HMMs. The experimental evaluation using the SCOPSUPER95_44f corpus, where the amount of training data is steadily decreased, showed that approximately 20 training samples are sufficient on average for both improved classification and detection performance. For the new concept of protein family modeling using small building blocks the proof of concept for their applicability to remote homology analysis could be given.

Since enhanced modeling techniques for protein families require substantially more computational effort (caused by the additional third stochastic stage during model evaluation, namely mixture density evaluation), acceleration techniques are mandatory. Based on pruning techniques addressing the reduction of state-space exploration as well as of mixture

density evaluation, the computational effort could be decreased significantly on average while keeping the superior classification or detection performance. The time required for current Profile HMM evaluation and pruned evaluation of SCFB BLR HMMs is almost comparable which implies a gain in classification, or detection accuracy without substantially increasing the processing time.

Finally, the configuration which proved most effective in the separate evaluation of the new techniques was used for combined evaluation using the SCOPSUPER95_20 as well as the PFAMSWISSPROT corpus. This configuration consists of:

- Feature based protein data processing,
- Bounded Left-Right model topology,
- Competitive evaluation of target model and single state UBM,
- Semi-continuous Hidden Markov Models for protein family modeling,
- k-means based estimation of the general protein data feature space using approximately 90 000 sequences from SWISSPROT (resulting in 1024 mixture components),
- MLLR, or MAP based specialization of the protein data feature space, and
- Beam-Search based state-space and mixture density pruning (including a two-stage classifier).

For both corpora the improved performance of advanced stochastic models for protein family models could be proved. Additionally, when occurrences of Pfam based models where searched within SWISSPROT, the great effectiveness of the new methods could be proved for real-world scenarios including domain spotting within large protein sequences.

7 Conclusion

In the last decade(s) the computational analysis of protein sequences has become the base for almost all fields of molecular biology research. Usually, the search for homologue sequences contained in one of the major sequence databases stands at the beginning of most investigations with respect to the gain of (further) biological insights. Especially for pharmaceutical research within the drug discovery process, broad scale protein sequence analysis is of fundamental scientific as well as commercial interest. Traditionally, pairwise alignment techniques are applied to this task but in the last few years analysis approaches based on probabilistic protein family models, most notably Profile HMMs, has become the methodology of choice. Unfortunately, although sophisticated methods for both robust model estimation and evaluation have been developed, the general problem of remote homology detection and classification is still far from being solved. Current stochastic protein family models estimated using small training sets for covering highly diverging sequence data tend to capture facts, i.e. protein family members, which were more or less known before. The generalization capabilities of state-of-the-art Profile HMMs are often not satisfactory. Since “post-genome” techniques may at least be partially doomed if the fundamental sequence analysis fails, new approaches are demanded.

In order to improve the effectiveness of remote homology analysis, in this thesis new approaches for stochastic protein family modeling were developed. Therefore, the problem of protein sequence processing was consequently treated as some general pattern recognition task. Based on this more abstract point of view, various new techniques were presented which were motivated from alternative pattern classification applications like automatic speech recognition. Consequently, a generally new and very effective kind of protein sequence analysis was developed. By means of the new approaches presented here substantial improvements for remote homology analysis could be achieved. In the following the thesis is summarized and the practical applicability of advanced stochastic protein family models is discussed.

Summary

Current protein sequence analysis approaches are mostly based on direct processing of amino acid data. Especially for the most successful method, namely probabilistic protein family modeling using Profile HMMs, no alternative procedures are known. The biological functions of proteins are determined by their three-dimensional structure which is the result of the underlying rather complex folding process. Similar structure corresponds to similar biological function which justifies the general approach of sequence comparison based protein analysis. The linear chain of the 20 standard amino acids, the so-called primary structure of proteins, dominates the folding process. Although the primary structure gives reasonable hints for the biological function, lots of information are discarded when limiting protein sequence analysis to it. Note that further protein data like secondary struc-

ture information is usually not available when processing sequences at the beginning of the molecular biology processing loop.

The biochemical properties of protein sequences are actually *summarized* only by the chain of amino acids. In order to explicitly capture the biochemical properties, the fundamental innovation of the thesis describes a paradigm shift towards processing protein data in a richer representation. Using a sliding window technique, frames based on 16 consecutive residues are created which consist of a multi-channel signal-like numerical representation of certain biochemical properties of the amino acids covered by the local context. Every channel contains a mapping of the 16 frame residues to numerical values that correspond to the particular property. These mappings are obtained by exploiting certain amino acid indices which are collected in the literature. By means of this procedure a very detailed description of proteins' biochemistry is obtained. In order to concentrate on the corresponding relevant essentials which determine the actual protein family affiliation, features are extracted from the new signal-like representation. Therefore, pattern recognition techniques are applied, namely a Discrete Wavelet Transformation as well as a Principal Components Analysis, aiming at the extraction of meaningful feature vectors which sufficiently describe the general protein signal shape. When applying this procedure, protein sequences' residues are converted into a 99-dimensional feature vector representation which is used for remote homology analysis.

Since more or less continuous data is processed when considering the new 99-dimensional feature vectors, discrete Profile HMMs are not suitable for robust protein family modeling. Instead, continuous modeling techniques are applied. As known from the literature, semi-continuous HMMs are very effective especially when only little training data is available. Their basic advantage is the principle possibility to separate the estimation of a feature space representation (using a Gaussian mixture density) from the model training itself. Only for the actual model estimation moderate amounts of target specific data are required. The general feature space representation can be obtained by applying mixture density estimation techniques to large amounts of *general* protein data. Based on the new feature representation of biological sequences, semi-continuous Profile HMMs were developed for remote homology classification. In the approach presented here, approximately 90 000 sequences obtained from SWISSPROT are exploited for robust mixture density estimation. Only small amounts of protein family specific data are required for model training. In order to specialize the mixture density based feature space representation with respect to a particular target family, adaptation techniques are applied. Using the family specific sample data for either MAP or MLLR adaptation, the focus of the general feature space can be effectively concentrated on a particular protein family which results in robust model estimation even for small training sets.

Compared to the rather complex three-state topology of Profile HMMs required when processing discrete amino acid data, by means of the richer protein sequence representation developed in this thesis, protein family HMMs with reduced complexity become possible. Their basic advantage is the smaller number of parameters required which need to be trained using representative sample data. Thus, the amount of training sequences necessary for robust model estimation can be further reduced. In this thesis two variants of protein family models with reduced complexity were developed. First, global protein family models as currently used are created containing a Bounded Left-Right (BLR) architecture. Basically BLR

HMMs consist of a linear model architecture with reasonably limited numbers of direct transitions to states adjacent to the right. The second variant represents a general paradigm shift in protein family modeling. Inspired by alternative pattern classification applications like automatic speech recognition, target family models are estimated using automatically derived building blocks, so-called Sub-Protein Units (SPUs). In the exemplary definition of SPU-candidates described in the thesis, high-energy parts of feature vector sequences corresponding to protein data are interpreted as building blocks. In an iterative procedure a non-redundant set of SPUs relevant for describing the essentials of complete protein families is extracted. Similar to the BLR models, the number of parameters which needs to be trained is substantially smaller compared to state-of-the-art Profile HMMs.

For remote homology detection tasks, actual target hits need to be discriminated robustly from non-targets. Therefore, as usual for protein sequence analysis, log-odd scores are used for threshold based decisions. The null model used for feature based modeling techniques developed in this thesis consists of the prior probabilities of the particular mixture components which describe the feature space. By means of this scoring method and the significance analysis using E-values, detection tasks can actually be solved. For a further reduction of the number of false positive predictions, which is especially relevant for e.g. pharmaceutical applications, in addition to the abovementioned log-odd scoring technique semi-continuous feature based protein family HMMs are evaluated competitively to a so-called Universal Background Model (UBM). Such an UBM explicitly covers general protein data and its application can thus be interpreted as some kind of pre-filtering stage.

When analyzing the common procedure of evaluating state-of-the-art protein family HMMs, its rather straightforward character becomes obvious. The only methods which are applied to the acceleration of e.g. database searches for remote homologies are based on more or less “brute-force” techniques, namely massive parallelization and adding more computers to the task. Currently, there are hardly any solutions available for the *algorithmic* acceleration of protein family model evaluation. Especially when applying advanced stochastic modeling techniques as described in the thesis, the problem of inefficient model evaluation becomes more manifest. Thus, acceleration techniques were adopted from general pattern recognition techniques and transferred to the protein sequence analysis domain. Most of these techniques can be applied to the current procedure, too, i.e. parallelization remains possible. Among others the HMM state-space which actually needs to be explored can be substantially reduced by applying pruning techniques like the Beam-Search algorithm. Furthermore, since the feature space representation is rather specific for particular HMM states, the evaluation of mixture densities can also be severely pruned. Protein family model evaluation is greatly accelerated resulting in comparable computation times as when processing discrete Profile HMMs, while simultaneously substantially improving their effectiveness.

The capabilities of the new modeling techniques were evaluated in numerous experiments. Therefore, the SCOP database was divided into various corpora which were used for both the separate evaluation of the new approaches as well as for the combined evaluation of the resulting advanced stochastic protein family models. It could be shown that state-of-the-art discrete Profile HMMs, which are in fact the currently most promising approach for remote homology analysis, are significantly outperformed when using the new techniques developed in this thesis.

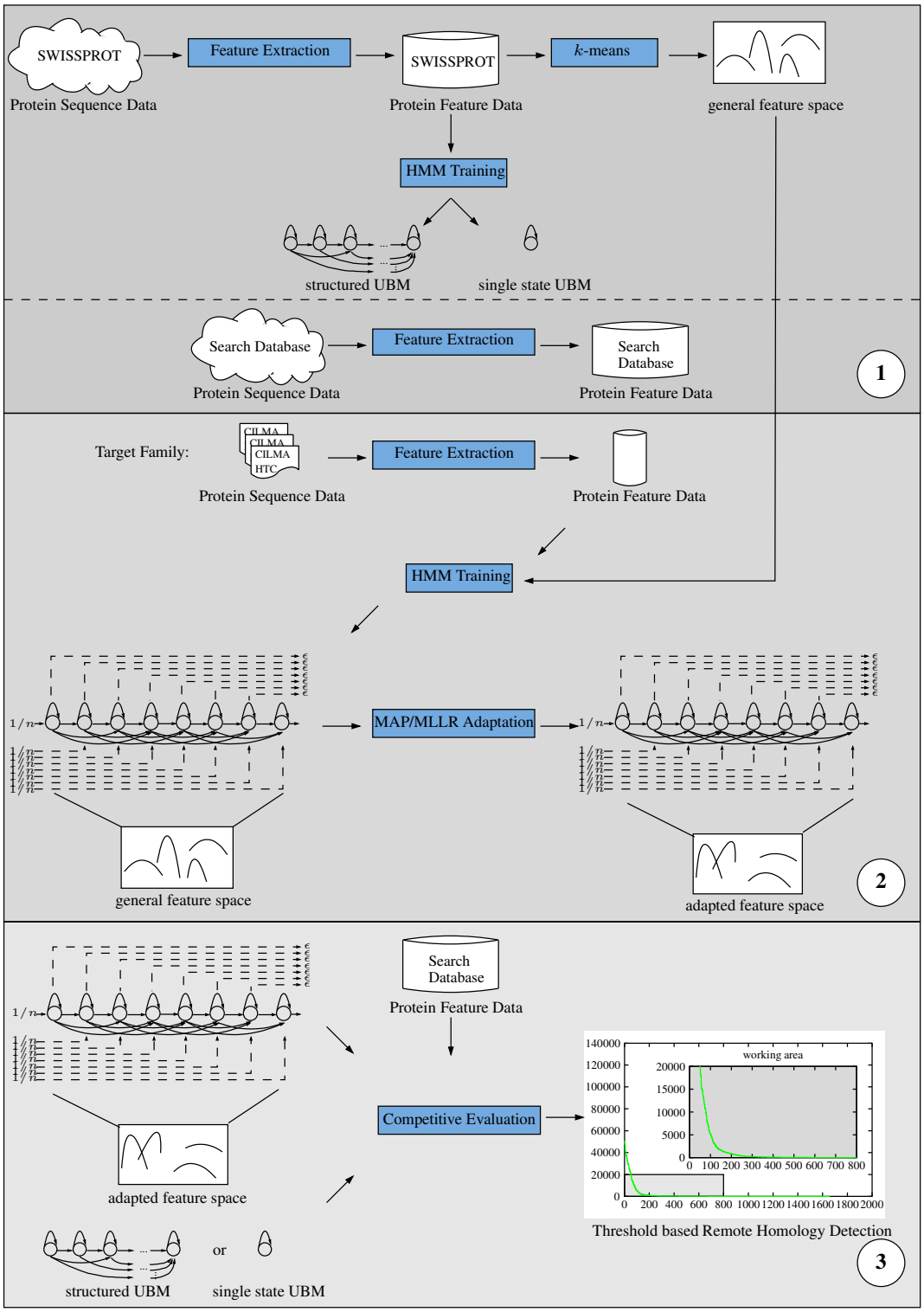


Figure 7.1: Application scheme of advanced stochastic protein family models developed in this thesis: 1) Estimation of a general mixture density based feature space representation using general protein data (SWISSPROT), and estimation of Universal Background Models and feature extraction for search database; 2) Training of semi-continuous feature based Bounded Left-Right protein family HMMs using small sets of target specific feature data, and feature space specialization by adaptation; 3) Competitive (accelerated) model evaluation of target and UBM HMMs, and threshold based remote homology detection.

The general application scheme of advanced stochastic protein family models is summarized in figure 7.1 by means of remote homology detection using semi-continuous feature based Bounded Left-Right HMMs.

In the first frame (upper block numbered with '1'), the general preprocessing steps necessary for the application of the new models are illustrated. In particular these are the estimation of the general mixture density feature space representation, UBM training, and the feature extraction for the sequences contained in the database which will be searched for remote homologues. These rather time-consuming steps are necessary to be processed only once. Following this, the actual target protein family model estimation procedure is shown in frame 2. Based on the features extracted from a small sample set, a general semi-continuous feature based protein family HMM is estimated. For further specialization of this particular model, either MAP, or MLLR adaptation is applied. The resulting target model is competitively evaluated to the UBM estimated in the initial preprocessing step. For efficient model evaluation, various acceleration techniques are applied. The actual remote homology detection process is summarized in the third block.

Discussion

Analyzing the current situation in computational protein sequence analysis, it becomes clear that especially the detection of remote homologue protein family members is still a very challenging problem. Apparently, state-of-the-art techniques, most notably the application of Profile HMMs as probabilistic protein family models, have reached their limits according to the general classification and detection performance. Since *successful* remote homology analysis is the fundamental prerequisite for all further processing steps within the molecular biology research loop, insufficient recognition results are rather problematic. Thus, new approaches are demanded addressing at least partial solutions to this problem. The developments of new methods for alleviating the abovementioned problems allowing further knowledge gains in molecular biology which is especially valuable for e.g. pharmaceutical applications was basically the motivation for this thesis. The conceptual idea for enhanced protein family modeling methods was the consequent treatment of the protein sequence analysis problem from a general pattern classification point of view.

By means of advanced stochastic protein family modeling techniques developed in this thesis, in fact substantial improvements for remote homology analysis, i.e. classification as well as detection, become possible. Due to the extensive experimental evaluation which actually proved the new approaches' effectiveness for various representative datasets, strong evidence could be given for the generalization of the improvements to related tasks. This is especially relevant when actually *new* members of certain protein families of interest are sought. When applying the methods developed in this thesis it is much more likely that the correct protein family affiliation of currently unknown sequence data is predicted reasonably accurately (compared to state-of-the-art techniques). The overall application scheme of the new techniques (which was summarized e.g. in figure 7.1) allows their easy integration into an already existing protein sequence analysis frameworks, e.g. within a concrete drug discovery pipeline. Thus, broad areas of molecular biology research can immediately benefit from the outcome of this thesis.

Furthermore, the new advanced stochastic protein family models can serve as the foundation for certain further developments. As an example, techniques for iterative model estimation were proposed in the literature (e.g. PSI-BLAST [Alt97] or SAM's target98 [Kar98]) aiming at successive enhancements of the training sets by alternating recognition and training phases. Since the methods described in this thesis address the basic modeling procedure, such iterative techniques can principally be realized as well. It can be seriously expected that iterative approaches will also benefit from the new techniques. Additionally, especially the SPU based modeling approach which was discussed and for which a general proof of concept for effectiveness was given, offers the opportunity for further developments. The idea of explicitly limiting the modeling base to small, automatically obtained building blocks can be enhanced in various ways. First, alternative criteria for SPU candidate selection can be applied, e.g. the incorporation of further statistical measures like entropy etc. Furthermore, general SPUs can be estimated on major protein data (like SWISSPROT) and the resulting building blocks can serve as some kind of "biological inventory" which is used for protein family modeling by concatenating the particular building blocks. Similar to e.g. automatic speech recognition applications, semi-automatic annotations of unknown protein sequences can be obtained by aligning it to the well trained SPU models. SPUs which are obtained by either the method outlined in this thesis or by the more general approach mentioned above include high potential for obtaining new biological insights. However, substantial effort needs to be dedicated to the *biological* analysis and interpretation of the particular results.

To conclude, advanced stochastic protein family models as developed in this thesis represent a major improvement for remote homology analysis tasks. Furthermore, they can serve as the foundation for various further developments which is very promising for general molecular biology research.

A Wavelets

Natural or artificial signals evolving in time are usually represented in their most obvious form, i.e. as a function f of a time-dependent variable t. Besides this, especially for applications in natural sciences and engineering tasks an alternative but completely equivalent representation is widely used – the frequency decomposition of the signal. Here, arbitrary signals f are interpreted as a superposition of basic functions ψ with specialized shapes and frequencies weighted by c_k:

$$f = \sum_k c_k \psi_k \tag{A.1}$$

Formally, these basic functions build up an alternative base defining a complete function space. The signal of interest is transformed into this function space. The alternative representation now consists of a function of frequency components k. This *spectral* representation contains exactly the same information as the standard signal view but it offers direct access to the frequencies of the data analyzed which is convenient for a wide range of applications.

Throughout the years a large number of transformations were developed (cf. e.g. [Pou00] for an extensive overview). In the following sections a very powerful technique which was successfully used for protein signal analysis in the thesis at hand, will be summarized – the Wavelet transformation. Since the Fourier Analysis is the base for a large variety of transformation techniques and basically the foundation for Wavelets, too, a brief overview of the principles of this fundamental technique will be given in section A.1. Throughout the whole chapter, the explanations are limited to one-dimensional signals for simplicity and clarity of argumentation. Nevertheless, they can easily be generalized to N-dimensional signals.

A.1 Fourier Analysis

The most prominent spectral representation of signals is based on the *Fourier* transformation (FT). Here, signals f are decomposed into basic sine and cosine oscillations, i.e. $\psi = \cos(\omega t)$ and $\psi = \sin(\omega t)$, respectively (cf. equation A.1). The Fourier transformation is defined as follows:

$$\begin{aligned} \mathcal{F}(\omega) &= \int_{-\infty}^{\infty} f(t) e^{-i2\pi\omega t} dt, && \text{for continuous signals} \\ \mathcal{F}(k) &= \frac{1}{N} \sum_{n=0}^{N-1} f(t) e^{\frac{-i2\pi k}{N}}, && \text{for discrete signals of length } N \end{aligned}$$

After transforming signals using the Fourier transformation, the proportions of single frequencies for the signal of interest are easily accessible via $\mathcal{F}$. The basic assumption for the

discrete Fourier transformation, which is especially relevant for the computational treatment of natural signal reasoned by the necessary discretization, is the existence of infinite and periodic signals. Unfortunately, this does not hold for the majority of actual signals. Furthermore, information concerning time-localization of single frequencies cannot be read off from $\mathcal{F}(k)$. So, for signal analysis the Fourier transformation is not always the methodology of choice.

In order to overcome the limitations of the standard Fourier transformation in time-localization, the windowed or Short-Time Fourier transformation (STFT) was developed. Generally, short windows are extracted at designated positions from the original signal by applying appropriate filters (e.g. Hamming- or Gauss-windows). The filtered signals are transformed using the standard procedure as described above (here exemplarily shown for continuous signals only):

$$\mathcal{F}(\omega, a) = \int_{-\infty}^{\infty} f(t)g(t-a)e^{-i2\pi\omega t}dt, \tag{A.2}$$

with e.g. a Gaussian window: $g(t) = e^{-kt^2}$

Depending on the size of the windows and the distance between two adjacent signal parts analyzed, the time-localization for single frequencies is mostly satisfying. Since the size of the windows extracted from the signal is fixed, the time-frequency resolution is also fixed. Therefore, frequent changes in signal characteristics, each requiring specialized frequency resolutions for successful detection, cannot be localized. If, for example, a signal is composed of small bursts associated with long quasi-stationary components (as e.g. shown in the outermost left diagram of figure A.3), then each type of component can be analyzed with good time resolution or frequency resolution, but not both [Rio91]. Furthermore, the basic assumption of infinite and periodic signals analyzed is still valid. So, although the STFT is applicable to a wide variety of signals, it still has some limitations.

A.2 Continuous Wavelet Transformation

In order to overcome the drawbacks and limitations of both standard and windowed Fourier transformation, the Wavelet Transformation (WT) was developed.

> "The Wavelet transformation is a tool that cuts up data or functions or operators into different frequency components, and then studies each component with a resolution matched to its scale."[Dau92, p. 1]

In contrast to the STFT (cf. section A.1) the time-frequency resolution of the WT is not fixed. The time resolution becomes arbitrarily good at high frequencies whereas the frequency resolution becomes arbitrarily good for low frequencies.[1] Different from the STFT, here the filter g (see equation A.2) itself is scaled. Figure A.1 compares the time-frequency resolutions of the Fourier- as well as of the Wavelet analysis techniques. Due to the flexible scaling of the *Mother-Wavelet* functions, the time-frequency localization of the WT is superior to all other techniques. The basic version of the filter function is called Mother-Wavelet

[1] The Heisenberg uncertainty principle ($\Delta t \Delta f \geq \frac{1}{4\pi}$) remains valid, though.

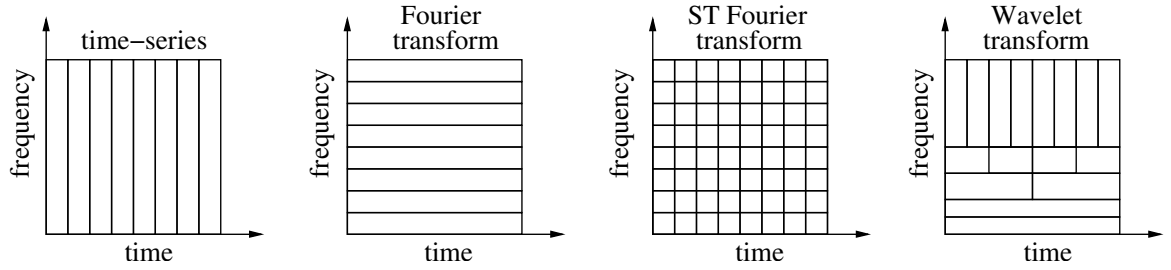


Figure A.1: Comparison of time-frequency resolutions for different signal analysis techniques: For the simplest kind of analysis (time-series at the most-left) at every time-step a continuum of frequency components exists corresponding to no frequency localization. Using the FT (second-left) frequency components can be determined but not located in time at all which can be done in fixed resolution steps when using the STFT (second-right). The squares represent particular frequency components at distinct time steps. The best time-frequency resolution is given by the WT (outer-right) where smaller time-windows are used for higher frequencies (adopted from [Fri01]).

ψ since all scaled and shifted versions are derived from it and its shape is similar to small waves.[2] Throughout the years a large variety of Mother-Wavelets were developed. Figure A.2 gives an overview of the most prominent filters. Formally, the Wavelet functions, which are now the basic functions mentioned earlier, used for the decomposition of the original signal, build up a complete function space. The actual shape of the Mother-Wavelet is important only for signal detection tasks. For general signal analysis the differences are negligible. Thus, for clarity the Haar-Wavelets are used in general explanations of the Wavelet transformation.

The continuous formulation of the Wavelet transformation is as follows:

$$\mathcal{W}(a,b) = \frac{1}{\sqrt{|b|}} \int_{-\infty}^{\infty} f(t)\psi\left(\frac{t-a}{b}\right) dt = \langle f, \psi_{a,b} \rangle \tag{A.3}$$

with

$$\int_{-\infty}^{\infty} \psi(t) = 0$$

In signal analysis applications using the general Fourier transformation, *spectrograms*, defined as the square modulus of the transformation, are a very common tool for visualization. In the two-dimensional spectrogram the abscissa represents the frequency components whereas the ordinate stands for its amplitude. A similar distribution can be defined in the Wavelet case – *Wavelet spectrograms* or *scalograms* as the squared modulus of the transformation, too. In contrast to the Fourier spectrogram, here the energy of the signal is distributed with different resolutions according to the scaling parameter b of equation A.3. Thus, the abscissa encodes the time-localization (parameter a of equation A.3) and the ordinate stands for the frequency resolution (the scaling defined by b). The amplitudes of the frequencies are visualized in the third dimension – the color or grey-value of the entries.

[2] In fact, the characteristic change of the sign of the filtering function for becoming a Wavelet is essential for WT which is proven e.g. in [Bän02].

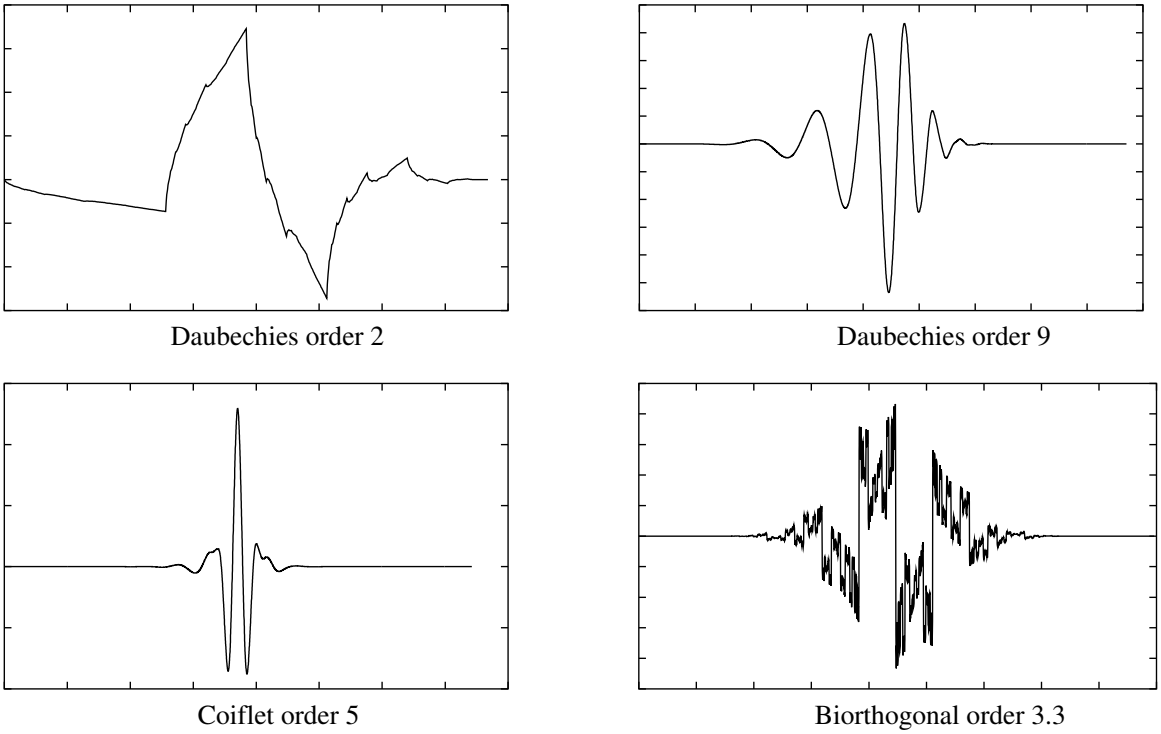


Figure A.2: Most prominent Mother-Wavelet shapes.

For the STFT similar spectrograms can be defined, too. Figure A.3 illustrates the differences between both visualization techniques. On the outer left-hand side an exemplary signal is visualized including an abrupt change of its frequency characteristics. The diagram in the middle of the figure visualizes the general Fourier spectrum. Obviously the characteristic frequencies of the signal can be figured out easily but they cannot be located. This can be done only in the Wavelet scalogram shown in the diagram at the right-hand side.

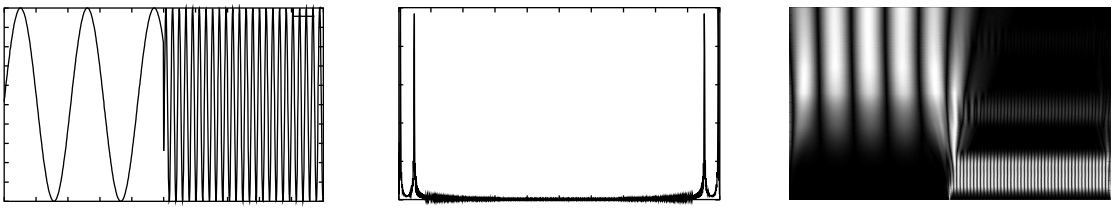


Figure A.3: Spectrogram of a Fourier analyzed signal vs. its scalogram visualizing the Continuous Wavelet transformation.

A.3 Discrete Wavelet Transformation

Restricting the parameters a for shifting and b for scaling the Mother-Wavelet ψ (cf. equation A.3) to discrete multiples of some initialization values a_0 and b_0, respectively, implies

the transition from continuous to discrete Wavelet functions. A very common choice for the initialization (especially for the implementation on a computer) is $a_0 = 1$ and $b_0 = 2$ defining a *dyadic* base. Thus, the discrete Wavelet functions $\psi_{a,b}$ are defined as follows:

$$\psi_{a,b} = \frac{1}{\sqrt{2^b}} \psi \left(\frac{t}{2^b} - a \right)$$

Figure A.4 illustrates the relationships between a Mother-Wavelet and its derivatives obtained by shifting and stretching it by means of the Haar-Wavelet. Using dyadic bases of Wavelet functions, very efficient implementations become possible. In the following section, the most common implementation of the Discrete Wavelet Transformation (DWT) is presented.

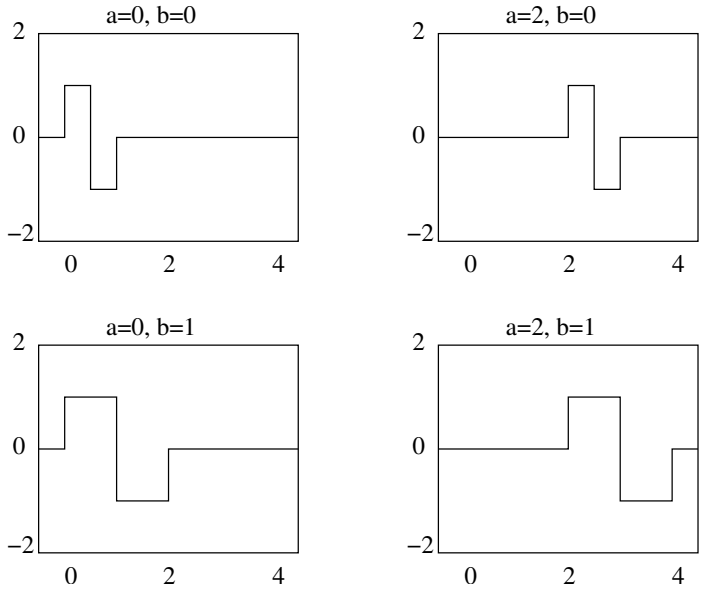


Figure A.4: Haar Mother-Wavelet and three derivatives obtained by shifting (parameter a) and scaling (parameter b) the base function.

Multiresolution Analysis (MRA)

The popularity of Wavelets for signal analysis applications is mainly reasoned by a very efficient technique developed by Mallat – the *Multi-Resolution Analysis (MRA)* (cf. e.g. [Mal98]). Generally, a sub-band analysis is performed using the Discrete Wavelet Transformation. Therefore, the pyramid algorithm known from image processing applications [Bur83] is generalized and applied to the signal of interest. The signal is analyzed iteratively using different resolutions at different time-locations.

By means of the MRA which is applicable to all functions f from $L^2(\mathbb{R})$ integrable quadratically, the signal-space is divided into nested sub-spaces, so-called scaling-spaces V_i:

$$\{0\} \subset \cdots \subset V_{i+2} \subset V_{i+1} \subset V_i \subset V_{i-1} \subset V_{i-2} \subset \cdots L^2(\mathbb{R})$$

Every sub-space describes the signal in a different resolution, whereby the larger i, the coarser the resolution. For all spaces, so-called scaling-functions exist which together with its translations span an orthonormal space.

Furthermore, Wavelet-spaces W_i are defined as the orthogonal complement of two adjacent scaling-spaces V_i and V_{i-1}, respectively. Thus, V_i can be defined as follows:

$$V_i = V_{i+1} \oplus W_{i+1} \quad \text{with} \quad V_{i+1} \perp W_{i+1},$$

where $\oplus$ denotes the additive combination of the particular Wavelet- and scaling-spaces. During transition from V_i to V_{i+1} detail information regarding the analyzed signal is lost. These details are projected onto the Wavelet-space W_{i+1}. Thus, in order to reconstruct an arbitrary signal f correctly, two signals are necessary: the signal based on the scaling-space V_{i+1} and the signal based on the Wavelet-space W_{i+1}. Similar to the scaling-functions, Wavelet-functions are defined spanning the Wavelet-space. Due to iteration, an arbitrary scaling-space is determined by the direct sum of all wavelet-spaces with indices larger than i. Consequently, a scaling-space of a given resolution can be described as the sum of the coarsest scaling-space – the approximation or trend V_T – and all wavelet-spaces up to the resolution level selected – the details or differences:

$$V_i = V_T \oplus W_T \oplus W_{T-1} \oplus \cdots \oplus W_{i+1}$$

Figure A.5 illustrates the decomposition of an exemplary signal s. Below the original signal the Wavelet decomposition using the Daubechies filter (fifth order) up to level five is shown. Here, initially the approximation of the signal is given which corresponds to the projection of s onto the fifth scaling-space ($s \in \mathbb{R} \rightarrow a_5 \in V_5$), succeeded by the five detail signals which are obtained by continuing the MRA (from top to bottom).

The practical relevance of nested sub-spaces spanned by scaling-functions as well as Wavelet-functions becomes clear by linking Discrete Wavelets using dyadic bases to the argumentation given above. Here, for signals of length T only a limited number of different resolutions exists, namely $\log_2 T$. If the signal decomposition at the finest level is selected as the original signal itself[3], the whole decomposition up to level i can be defined as a recursive filtering operation using analysis filters based on the scaling- and Wavelet-functions. If the DWT is applied completely, i.e. up to the last possible resolution $L = \log_2(T)$, $2^L - 1$ Wavelet-coefficients and one scaling-coefficient will be determined.

Independent of the resolution level i actually selected, the original signal f can be reconstructed identically by means of all coefficients. One of the most important properties of the DWT is the compactification of the energy of the signal which is mainly concentrated on the upper coefficients. Thus, for signal compression or smoothing applications the lower coefficients are simply discarded.

It can be shown, that the pairs of scaling- and Wavelet-functions define corresponding highpass- and lowpass-filters (for details, which are far beyond the scope of this thesis cf. e.g. [Bän02, Mal98, Per00a]). Additionally, due to the dyadic base of DWT, the Wavelet

[3] Strictly speaking, this initialization is an erroneous transition from the continuous towards the discrete world of Wavelets – the *Wavelet-crime*. Since in most applications no better clue exists it is taken for granted. In [Bän02, p. 73f.] the applicability is justified more theoretically alleviating the Wavelet-crime.

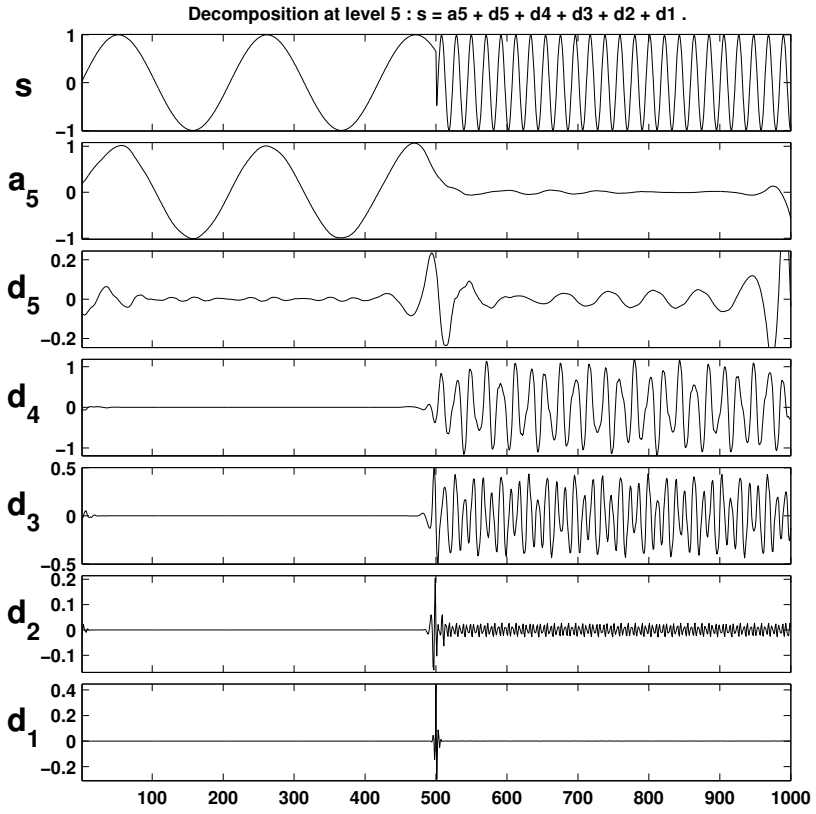


Figure A.5: Wavelet decomposition (up to level five) of an exemplary signal s using the Daubechies filters (fifth order); a_5 denotes the approximation and d_1 to d_5 the consecutive detail signals.

analysis can be formulated as the alternate application of a filtering as well as a downsampling process. Figure A.6 gives an overview about the resulting filtering scheme. On the left-hand side the filtering scheme of the analysis of an exemplary signal is shown, whereas on the right-hand side the synthesis is visualized making use of all coefficients determined during analysis. This scheme can be understood as a general sender-transmitter-receiver application.

Further issues

The brief overview given in this chapter captures the basic idea of the Wavelet transformation necessary to understand for its application to remote homology detection using probabilistic models for protein families. Besides this, countless enhancements and specialties exist. The very interesting mathematical foundations behind the theory of Wavelets are explained in numerous monographs. Especially the design of proper filter-pairs is of extreme interest for applications in various fields of research. Furthermore, variants within the filtering scheme were developed. Exemplarily, Wavelet-packets are a very powerful technique, where in addition to the exclusive decomposition of the lowpass-filtered signals, even the highpass-filtered parts are also further analyzed. Since further theoretical as well as practi-

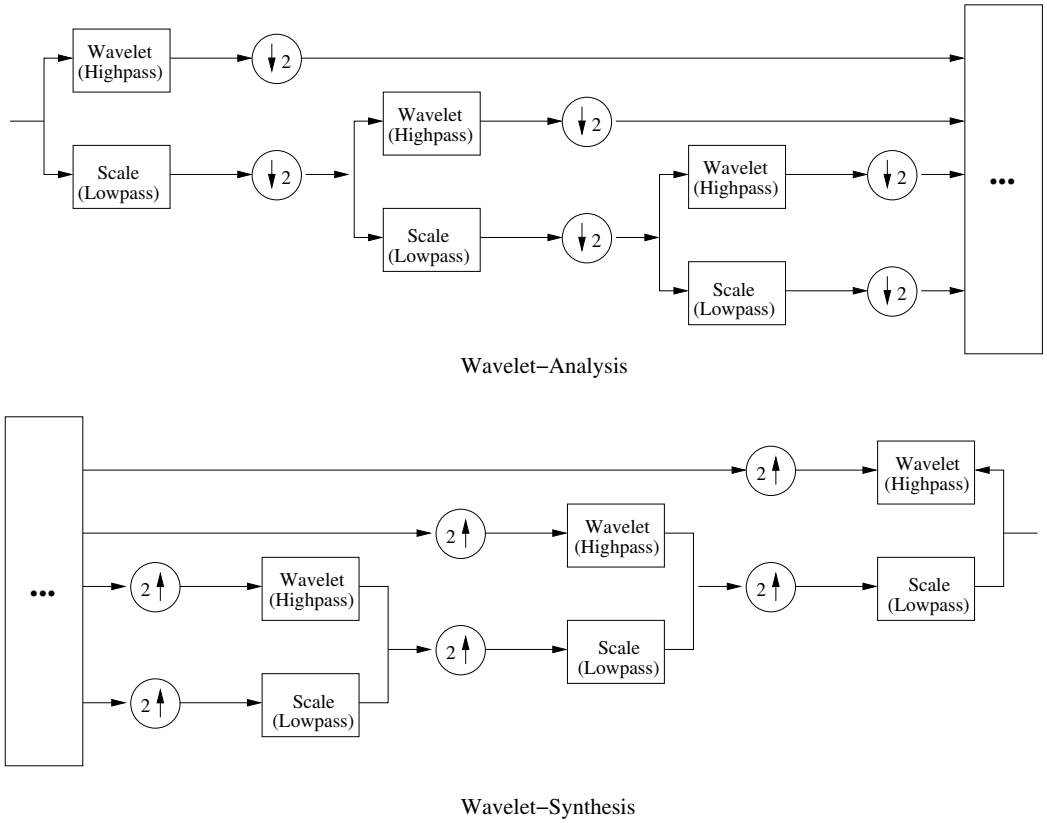


Figure A.6: Illustration of combined Wavelet-analysis and -synthesis by means of a sender-transmitter-receiver application: In the upper part the signal-decomposition at the sender is illustrated. The coefficients are transmitted via an arbitrary channel (rectangles at the right-hand side of the sender and the left-hand side of the lower part) to the receiver, which synthesizes the original signal more or less accurately, depending on the channel based distortions of the coefficients.

cal aspects regarding Wavelets is beyond the scope of this thesis, the reader is referred to the special literature (e.g. [Rio91, Dau92, Mal98, Per00a, Pou00]) for detailed insights.

B Principal Components Analysis (PCA)

Given an empiric or parametric distribution of N-dimensional data vectors, the general goal of the Principle Components Analysis (PCA) is the estimation of a new coordinate system where correlations between particular components are minimized. The axes of the new coordinate system are aligned in such a way that the maximum scattering of the underlying data occurs along the first axis. All following axes cover less variance of the data in descending order.

The estimation of the PCA is based on the analysis of the scattering of the analyzed data. They are characterized by so-called scatter matrices which are identical to covariance matrices if single distributions are considered. The total scatter matrix $\mathbf{S}$ of a sample set $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_T$ is defined as:

$$\mathbf{S} = \frac{1}{T}\sum_{i=1}^{T}(\vec{x}_i - \bar{\vec{x}})(\vec{x}_i - \bar{\vec{x}})^T, \quad \bar{\vec{x}} = \frac{1}{T}\sum_{i=1}^{T}\vec{x}_i,$$

where $\bar{\vec{x}}$ denotes the average vector of the sample set.

After decorrelation by applying a transformation matrix $\mathbf{\Theta}$, the scatter matrix of the transformed sample set is defined as:

$$\tilde{\mathbf{S}} = \frac{1}{T}\sum_{i=1}^{T}\mathbf{\Theta}(\vec{x}_i - \bar{\vec{x}})[\mathbf{\Theta}(\vec{x}_i - \bar{\vec{x}})]^T = \mathbf{\Theta}\mathbf{S}\mathbf{\Theta}^T.$$

Thus, in order to decorrelate the data, a transformation $\mathbf{\Theta}$ is required, which diagonalizes $\mathbf{S}$. Since the relative position of data vectors to each other must not change, only orthonormal transformations can be applied (cf. e.g. [Fin03, p.141]). It can be shown that the transposed matrix $\mathbf{\Phi}^T$ whose columns consist of the eigenvectors of $\mathbf{S}$ fulfills the constraint of orthonormality and can thus be used for automatic decorrelation of data vectors.

If the transformation is applied to the data vectors with zero mean

$$\vec{y} = \mathbf{\Phi}^T(\vec{x} - \bar{\vec{x}}),$$

than the transformed scatter matrix is received by

$$\tilde{\mathbf{S}} = \mathbf{\Phi}^T\mathbf{S}\mathbf{\Phi} = \mathbf{\Phi}^T\mathbf{\Phi}\mathbf{\Lambda}\mathbf{\Phi}^T\mathbf{\Phi} = \mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & & & \mathbf{0} \\ & \lambda_2 & & \\ & & \ddots & \\ \mathbf{0} & & & \lambda_N \end{bmatrix},$$

where $\mathbf{\Lambda}$ represents the eigenvalue matrix of $\mathbf{S}$. Thus, in the transformed space the variance of the data is given by the eigenvalues along the coordinate axes.

Effective dimension reduction including a reasonable estimation of the approximation error can be performed very easily when applying PCA to the sample set inspected. This becomes possible due to the previously mentioned fact that the eigenvalues of $\mathbf{S}$ denote the variance of the appropriate data vector components. If the transformation matrix Φ is created in such a way that its columns only consist of the eigenvectors corresponding to the M largest eigenvalues, data vectors $\vec{x} \in \mathbb{R}$ are projected onto the M-dimensional feature space ($M < N$) covering the majority of data variance. The reconstruction error ϵ obtained when using only M instead of N dimensions can be estimated as follows:

$$\epsilon = \mathcal{E}\{||\vec{x} - \vec{x}'||^2\} = \mathcal{E}\{||\sum_{i=M+1}^{N} y_i \phi_i||^2\} = \sum_{i=M+1}^{N} \lambda_i, \quad \vec{x}' = \sum_{i=1}^{M} y_i \phi_i.$$

By means of this selection of eigenvectors for creating the transformation matrix $\mathbf{\Phi}$ both the covered variance of the data analyzed is maximized and the reconstruction error resulting from skipping $N - M$ components is minimized.

C Amino Acid Indices

Table C.1 provides information about the 35 biochemical amino-acid properties, selected for the signal based protein sequence encoding. These indices were selected from the compilation of Shuichi Kawashima and Minoru Kanehisa [Kaw00].

Channel Index	Description	AAIndex Accession Key
0	Average flexibility indices	BHAR880101
1	Residue volume	BIGC670101
2	Transfer free energy to surface	BULH740101
3	Steric parameter	CHAM810101
4	Polarizability parameter	CHAM820101
5	A parameter of charge transfer capability	CHAM830107
6	A parameter of charge transfer donor capability	CHAM830108
7	Normalized average hydrophobicity scales	CIDH920105
8	Size	DAWD720101
9	Relative mutability	DAYM780201
10	Solvation free energy	EISD860101
11	Molecular weight	FASG760101
12	Melting point	FASG760102
13	pK-N	FASG760104
14	pK-C	FASG760105
15	Graph shape index	FAUJ880101
16	Normalized van der Waals volume	FAUJ880103
17	Positive charge	FAUJ880111
18	Negative charge	FAUJ880112
19	pK-a (RCOOH)	FAUJ880113
20	Hydrophilicity value	HOPT810101
21	Average accessible surface area	JANJ780101
22	Average number of surrounding residues	PONP800108
23	Mean polarity	RADA880108
24	Side chain hydropathy, corrected for solvation	ROSM880102
25	Bitterness	VENT840101
26	Bulkiness	ZIMJ680102
27	Isoelectric point	ZIMJ680104
28	Composition of amino-acids in extracellular proteins	CEDJ970101
29	Composition of amino-acids in anchored proteins	CEDJ970102
30	Composition of amino-acids in membrane proteins	CEDJ970103
31	Composition of amino-acids in intracellular proteins	CEDJ970104
32	Composition of amino-acids in nuclear proteins	CEDJ970105
33	Amphiphilicity index	MITS020101
34	Electron-ion interaction potential values	COSI940101

Table C.1: Biochemical properties selected for sequence representation.

At the website of the authors most indices of the database are clustered with respect to six coarse categories:

A. Alpha and turn propensities,

B. Beta propensity,

C. Composition,

H. Hydrophobicity,

P. Physiochemical properties, and

O. Other properties.

In figure C.1 an overview of the indices clustering is given. The indices used which were assigned by the authors to any of the categories are highlighted.

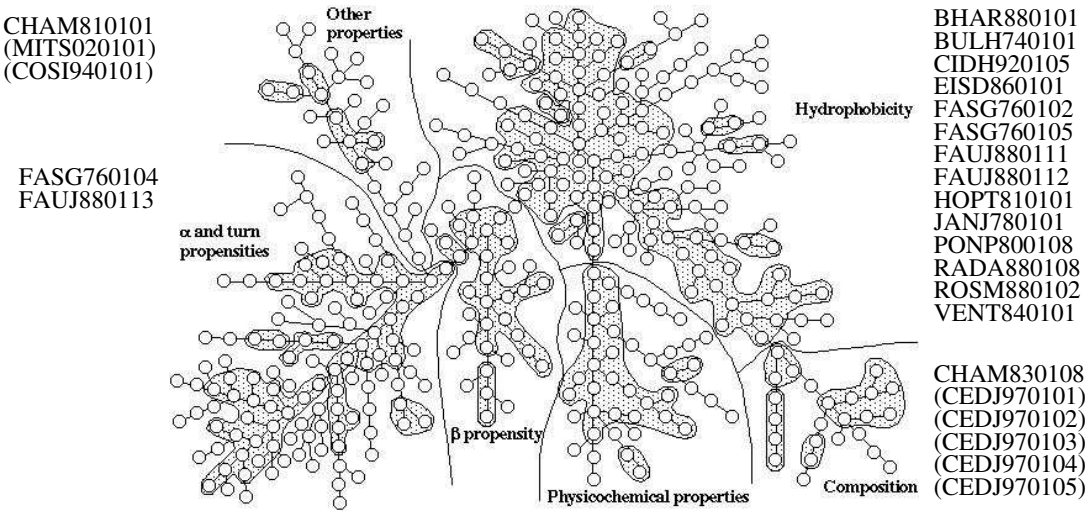


Figure C.1: Clustering of the amino acid index database in [Kaw00] as provided by the authors. The amino acid indices used for this thesis are marked explicitly. Note that for unknown reasons the cluster map does not include all indices used. These indices are manually assigned to the existing cluster map and written in parentheses.

D Detailed Evaluation Results

In chapter 6 the effectiveness of the methods developed for this thesis was presented. Especially for the detection task only summaries were given. For a more specific analysis of the detection results in this chapter they are presented in more detail.

SCFB Profile HMMs: SCOPSUPER95_66 Detection

In the following, the results of the experimental evaluation of the effectiveness of the new feature representation are given by detailed presentation of SCOPSUPER95_66 based ROC curves. For every superfamily contained in the corpus the appropriate ROC curves are given individually. Following this, in tables D.1 and D.2, respectively, the characteristic values of false prediction rates when limiting the appropriately corresponding rates to five percent are given for every superfamily.

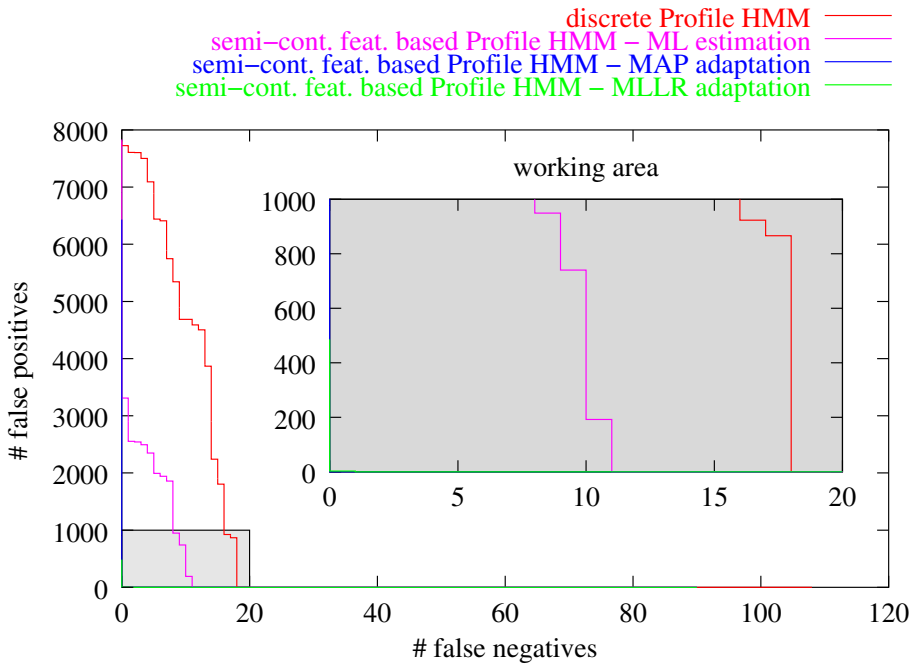


Figure D.1: ROC curve for Superfamily *Globin-like (SCOP-Id: a.1.1).*

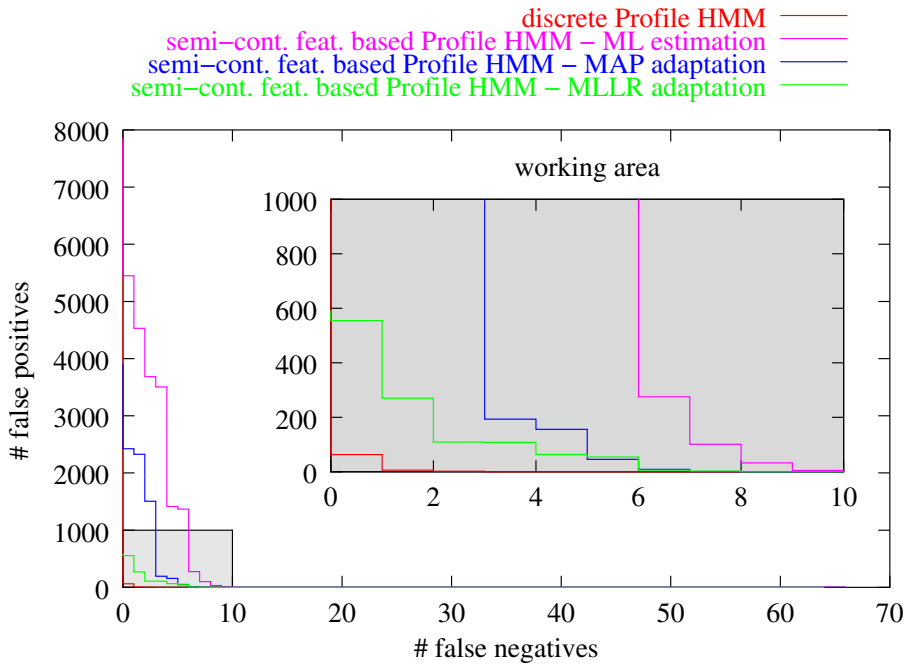


Figure D.2: ROC curve for Superfamily *Cytochrome C (SCOP-Id: a.3.1)*.

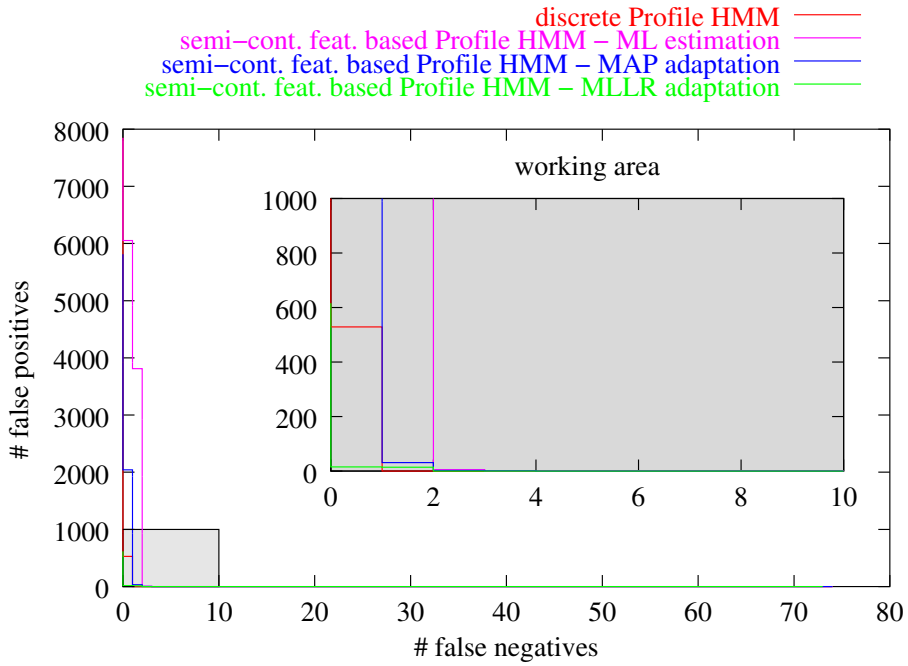


Figure D.3: ROC curve for Superfamily *EF-hand (SCOP-Id: a.39.1)*.

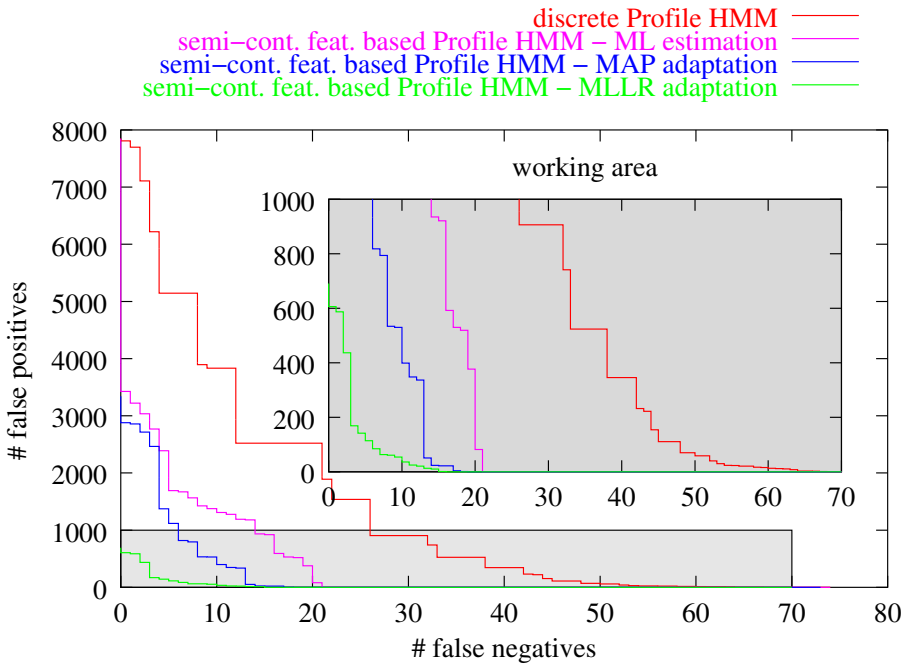


Figure D.4: ROC curve for Superfamily *"Winged helix" DNA-binding domain (SCOP-Id: a.4.5).*

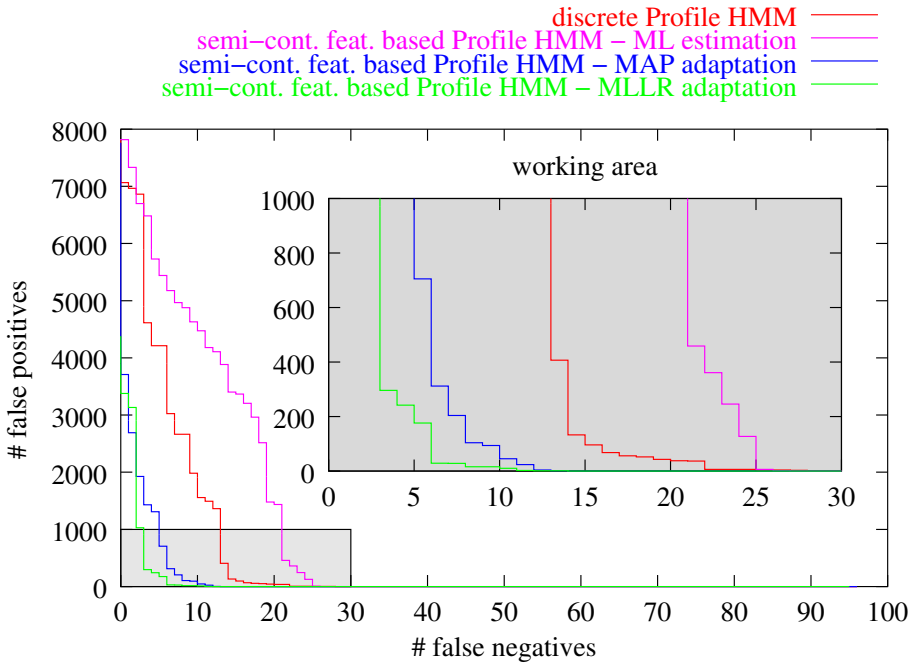


Figure D.5: ROC curve for Superfamily *Viral coat and capsid proteins (SCOP-Id: b.10.1).*

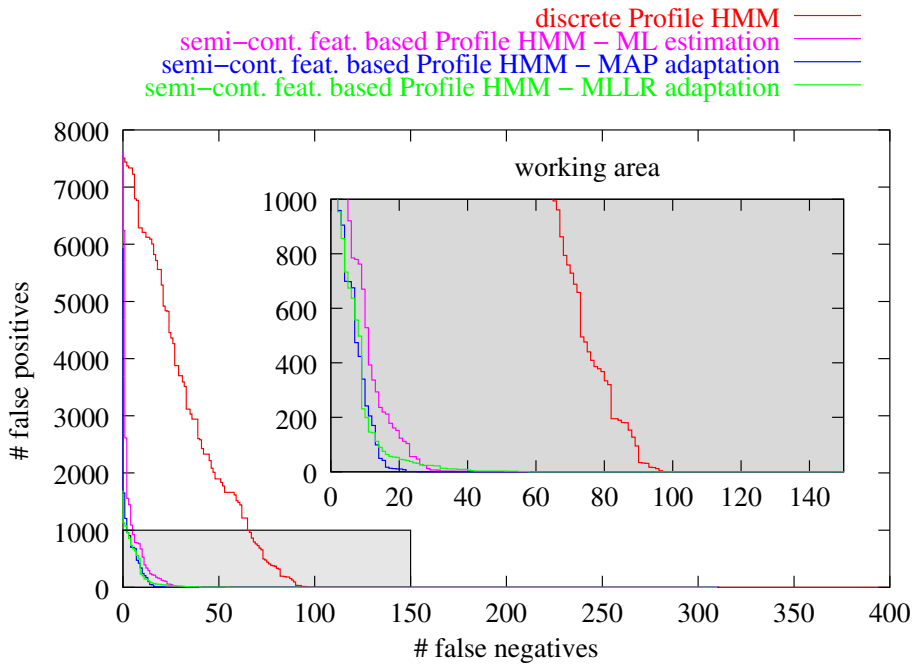


Figure D.6: ROC curve for Superfamily *Immunoglobulin (SCOP-Id: b.1.1)*.

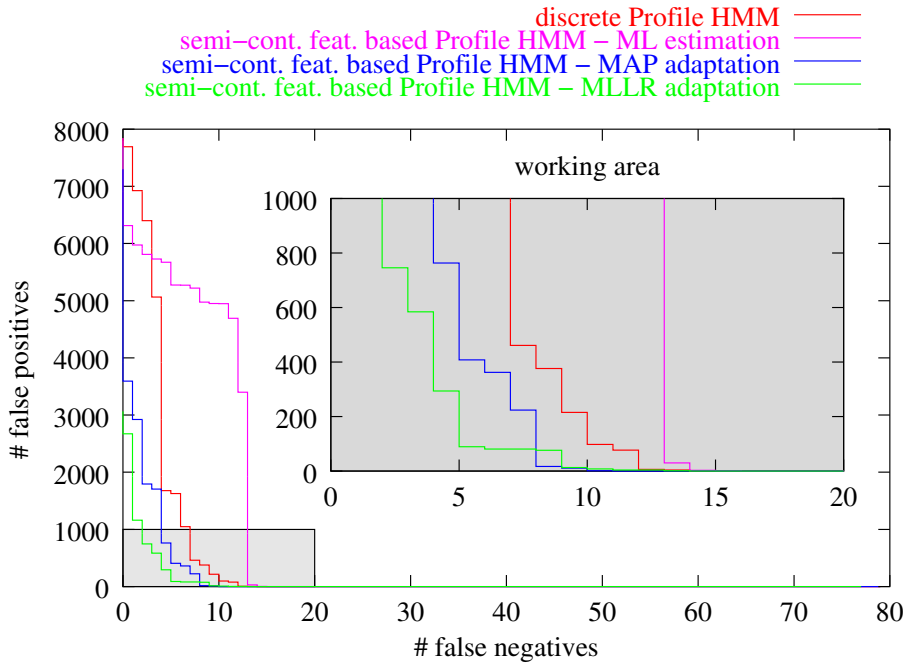


Figure D.7: ROC curve for Superfamily *Concanavalin A-like (SCOP-Id: b.29.1)*.

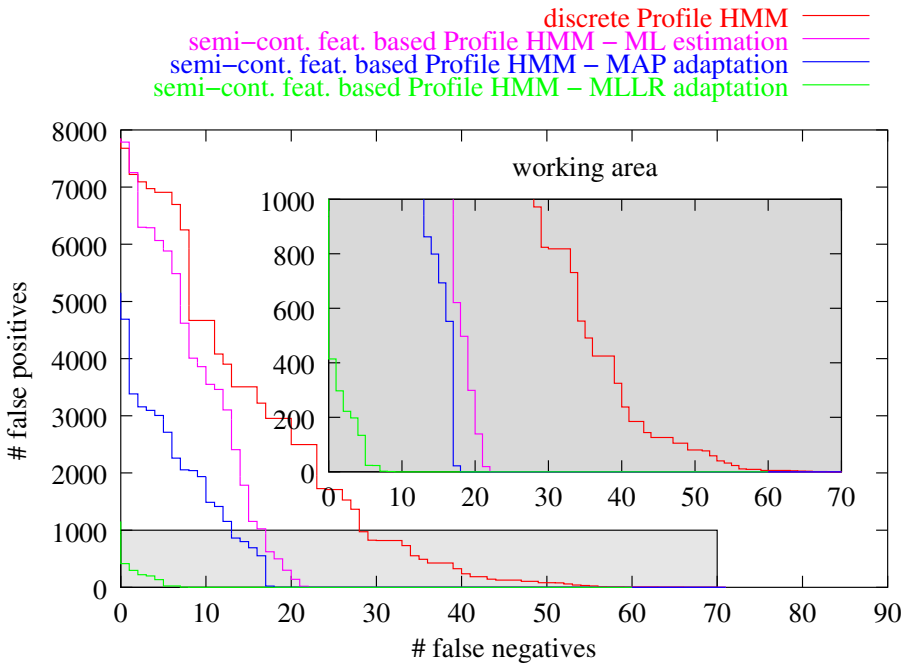


Figure D.8: ROC curve for Superfamily *Nucleic acid-binding proteins (SCOP-Id: b.40.4)*.

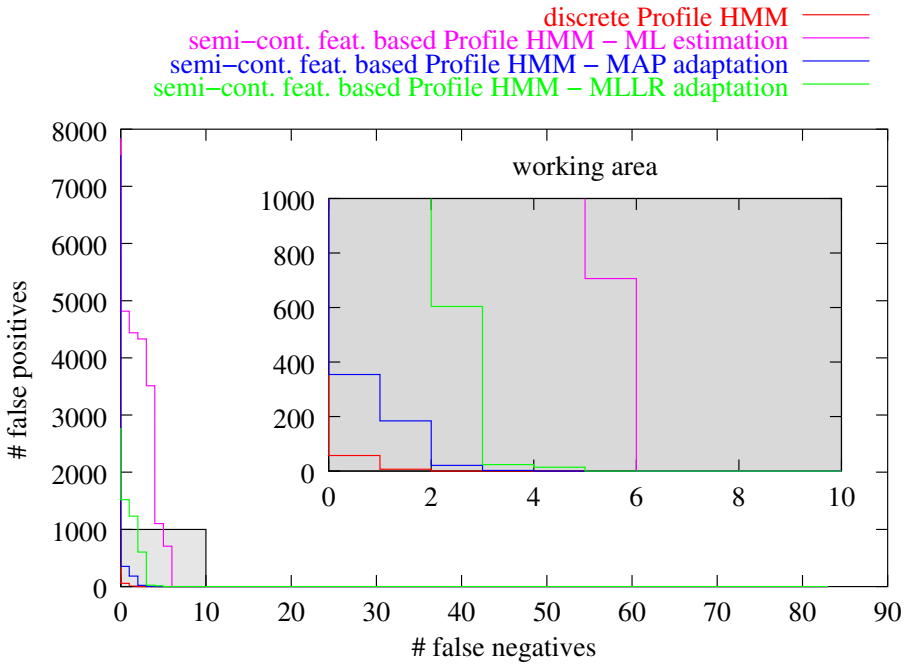


Figure D.9: ROC curve for Superfamily *Trypsin-like serine proteases (SCOP-Id: b.47.1)*.

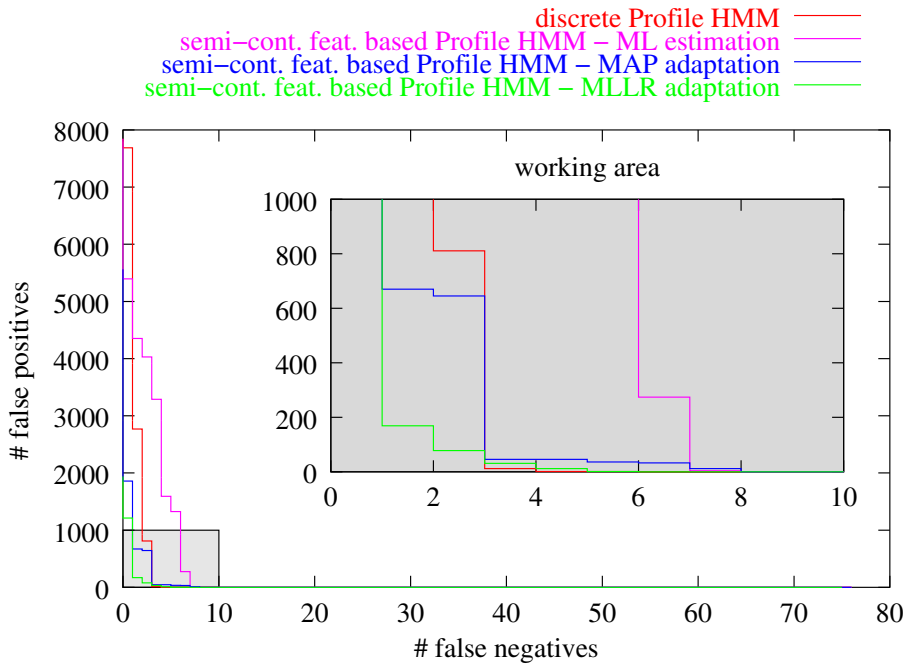


Figure D.10: ROC curve for Superfamily *Cupredoxins (SCOP-Id: b.b.1)*.

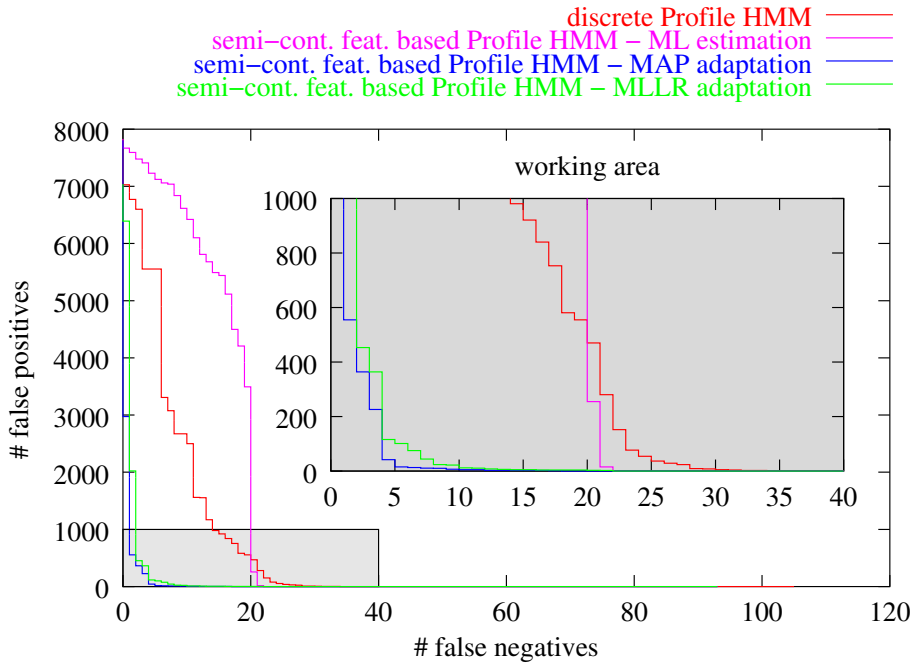


Figure D.11: ROC curve for Superfamily *(Trans)glycosidases (SCOP-Id: c.1.8)*.

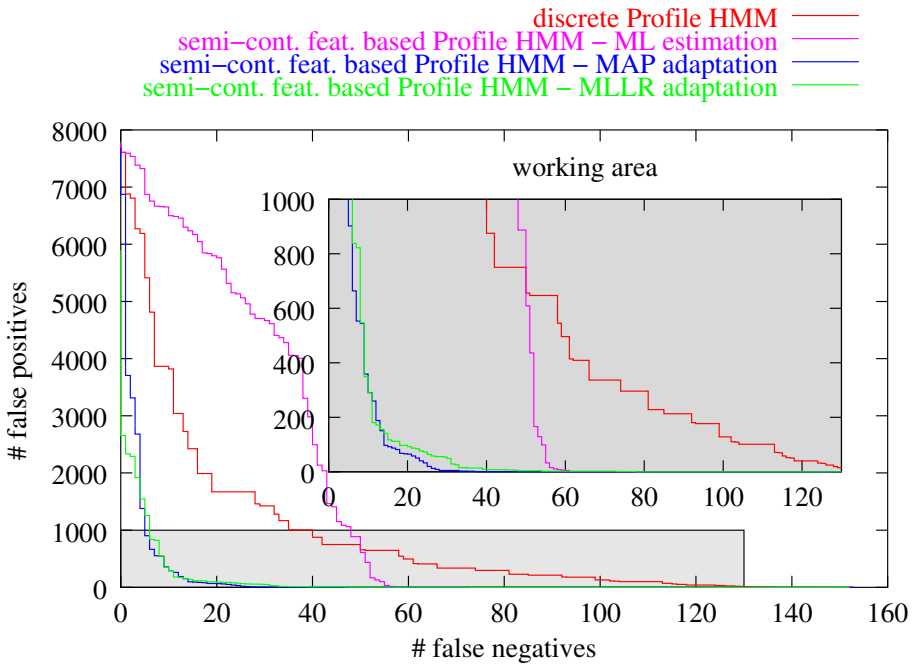


Figure D.12: ROC curve for Superfamily *NAD(P)-binding Rossmann-fold domains (SCOP-Id: c.2.1)*.

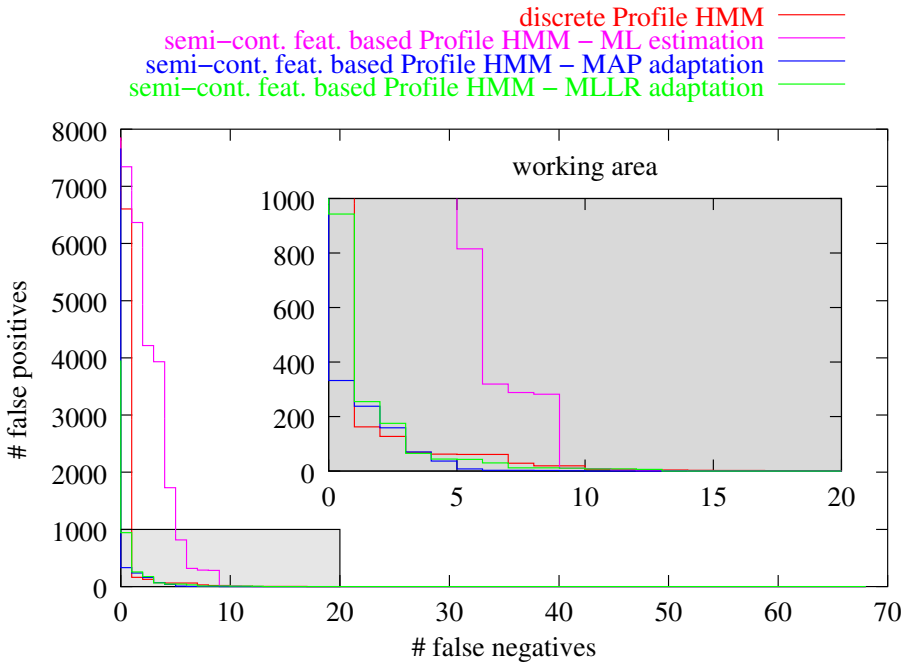


Figure D.13: ROC curve for Superfamily *FAD/NAD(P)-binding domain (SCOP-Id: c.3.1)*.

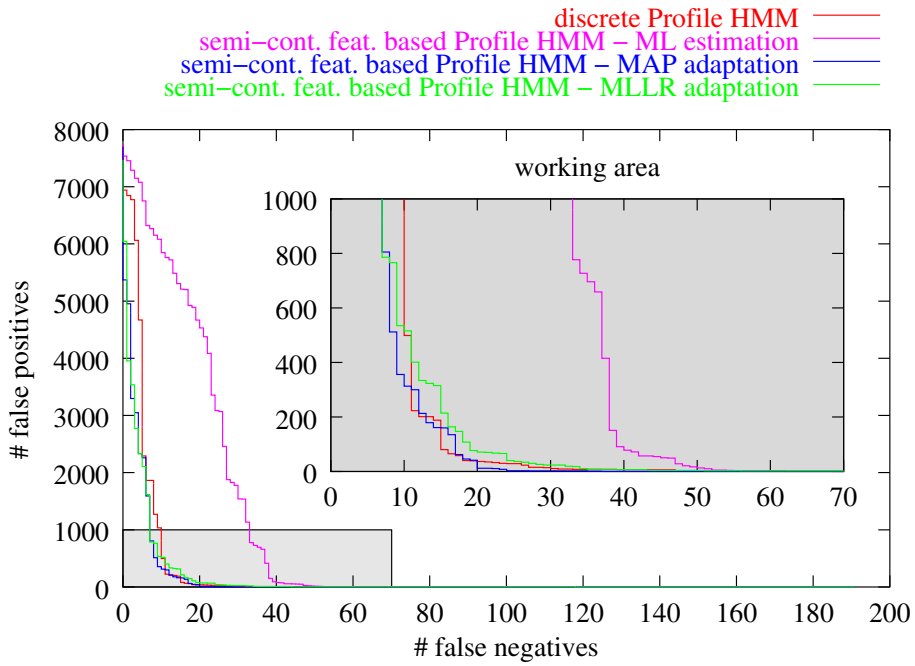


Figure D.14: ROC curve for Superfamily *P-loop containing nucleotide triphosphate hydrolase (SCOP-Id: c.37.1)*.

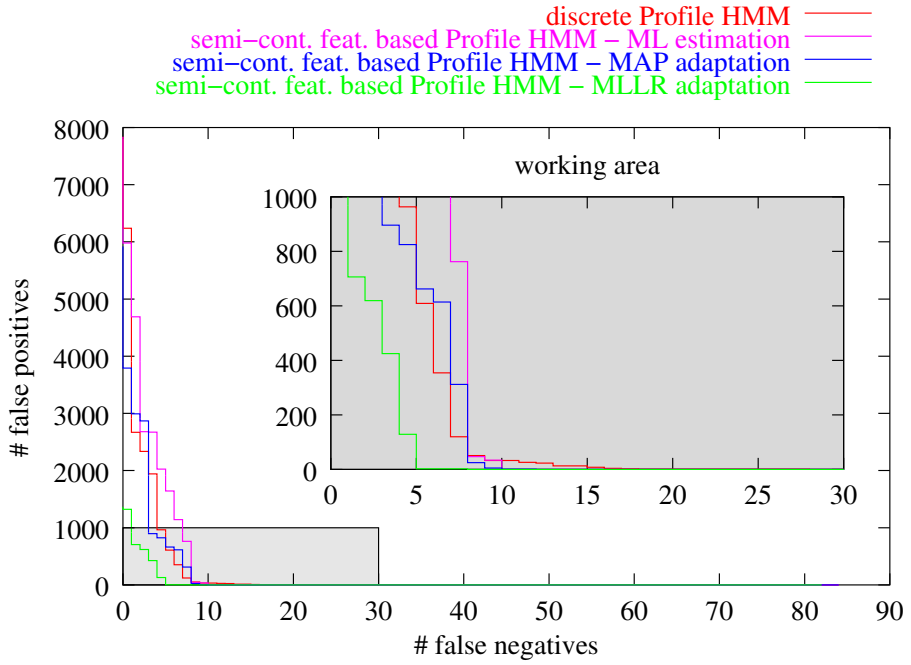


Figure D.15: ROC curve for Superfamily *Thioredoxin-like (SCOP-Id: c.47.1)*.

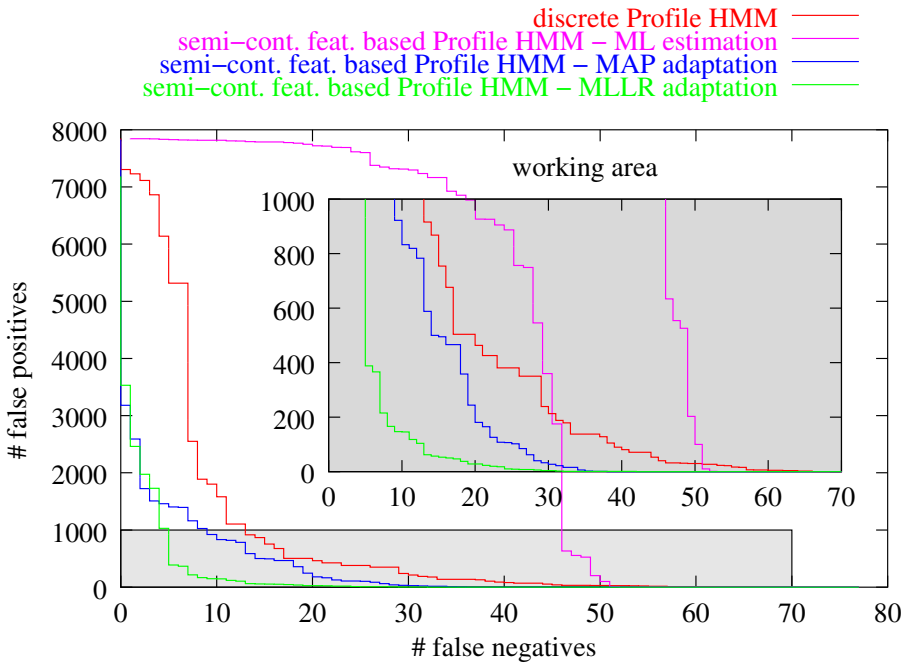


Figure D.16: ROC curve for Superfamily *Alpha/Beta-Hydrolases (SCOP-Id: c.69.1)*.

Superfamily	Profile HMM Variant (Adaptation)	false negative predictions [%] for 5 % false positives (**bold: best performing**)	false positive predictions [%] for 5 % false negatives (**bold: best performing**)
SCOPSUPER95_66 complete	discrete	26.1	57.6
	feat. based semi-cont. (ML)	17.7	64.4
	feat. based semi-cont. (MAP)	7.9	16.0
	feat. based semi-cont. (MLLR)	**5.1**	**5.5**
a.1.1	discrete	16.7	82.4
	feat. based semi-cont. (ML)	11.1	30.0
	feat. based semi-cont. (MAP)	**0.0**	**0.0**
	feat. based semi-cont. (MLLR)	**0.0**	**0.0**
a.3.1	discrete	**0.0**	**0.0**
	feat. based semi-cont. (ML)	9.1	44.6
	feat. based semi-cont. (MAP)	4.5	2.5
	feat. based semi-cont. (MLLR)	1.5	1.4
a.39.1	discrete	1.4	**0.0**
	feat. based semi-cont. (ML)	2.7	**0.0**
	feat. based semi-cont. (MAP)	1.4	**0.0**
	feat. based semi-cont. (MLLR)	**0.0**	**0.0**
a.4.5	discrete	51.4	79.2
	feat. based semi-cont. (ML)	25.7	35.3
	feat. based semi-cont. (MAP)	14.9	31.4
	feat. based semi-cont. (MLLR)	**4.1**	**2.2**
b.1.1	discrete	19.8	73.9
	feat. based semi-cont. (ML)	3.9	2.9
	feat. based semi-cont. (MAP)	**2.3**	**0.1**
	feat. based semi-cont. (MLLR)	**2.3**	0.7
b.10.1	discrete	14.6	53.8
	feat. based semi-cont. (ML)	22.9	73.2
	feat. based semi-cont. (MAP)	6.3	16.7
	feat. based semi-cont. (MLLR)	**3.1**	**3.1**

Table D.1: Characteristic values for SCOPSUPER95_66 Detection experiments (I).

Superfamily	Profile HMM Variant (Adaptation)	false negative predictions [%] for 5 % false positives (**bold: best performing**)	false positive predictions [%] for 5 % false negatives (**bold: best performing**)
b.29.1	discrete	10.1	64.5
	feat. based semi-cont. (ML)	16.5	73.0
	feat. based semi-cont. (MAP)	7.6	21.7
	feat. based semi-cont. (MLLR)	**5.1**	**7.4**
b.40.4	discrete	54.9	88.8
	feat. based semi-cont. (ML)	26.8	80.1
	feat. based semi-cont. (MAP)	23.9	39.4
	feat. based semi-cont. (MLLR)	**1.4**	**2.5**
b.47.1	discrete	**0.0**	**0.0**
	feat. based semi-cont. (ML)	7.2	14.0
	feat. based semi-cont. (MAP)	**0.0**	**0.0**
	feat. based semi-cont. (MLLR)	3.6	0.2
b.6.1	discrete	3.9	**0.2**
	feat. based semi-cont. (ML)	7.9	41.9
	feat. based semi-cont. (MAP)	3.9	0.6
	feat. based semi-cont. (MLLR)	**1.3**	0.4
c.1.8	discrete	20.0	71.0
	feat. based semi-cont. (ML)	21.5	92.3
	feat. based semi-cont. (MAP)	**1.9**	**0.2**
	feat. based semi-cont. (MLLR)	2.9	1.3
c.2.1	discrete	43.1	49.7
	feat. based semi-cont. (ML)	34.0	85.8
	feat. based semi-cont. (MAP)	**5.9**	**7.2**
	feat. based semi-cont. (MLLR)	**5.9**	10.6
c.3.1	discrete	1.5	0.9
	feat. based semi-cont. (ML)	8.8	50.1
	feat. based semi-cont. (MAP)	**0.0**	0.9
	feat. based semi-cont. (MLLR)	1.5	**0.8**
c.37.1	discrete	5.8	13.4
	feat. based semi-cont. (ML)	19.9	78.6
	feat. based semi-cont. (MAP)	**4.7**	**4.6**
	feat. based semi-cont. (MLLR)	6.3	6.9
c.47.1	discrete	7.1	12.3
	feat. based semi-cont. (ML)	9.5	25.8
	feat. based semi-cont. (MAP)	8.3	10.5
	feat. based semi-cont. (MLLR)	**4.8**	**1.6**
c.69.1	discrete	29.9	87.5
	feat. based semi-cont. (ML)	63.6	99.9
	feat. based semi-cont. (MAP)	23.4	**19.2**
	feat. based semi-cont. (MLLR)	**6.5**	22.0

Table D.2: Characteristic values for SCOPSUPER95_66 Detection experiments (II).

Bibliography

[Ale98] Alexandrov, N. and Go, N. On the number of structural families in the protein universe. In *Proc. Int. Conf. on Bioinformatics of Genome Regulation and Structure (BGRS'98)*. Novosibirsk, Altai mountains, Russia, Aug. 1998.

[Alt90] Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. Basic Local Alignment Search Tool. *J. Molecular Biology*, vol. 215(3):403–410, 1990.

[Alt91] Altschul, S. F. Amino acid substitution matrices from an information theoretic perspective. *J. Molecular Biology*, vol. 219:555–565, 1991.

[Alt96] Altschul, S. F. and Gish, W. Local alignment statistics. *Methods in Enzymology*, vol. 266:460–480, 1996.

[Alt97] Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., et al. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, vol. 25(17):3389–3402, 1997.

[Ana00] Anastassiou, D. Frequency-domain analysis of biomolecular sequences. *Bioinformatics*, vol. 16(12):1073–1081, 2000.

[Ana01] Anastassiou, D. Genomic signal processing. *IEEE Signal Processing Magazine*, vol. 18(4), 2001.

[Anf73] Anfinsen, C. B. Principles that govern the folding of protein chains. *Science*, vol. 181:223–230, 1973.

[Arn96] Arneodo, A., d'Aubenton Carafa, Y., Bacry, E., Graves, P., Muzy, J., et al. Wavelet based fractal analysis of DNA sequences. *Physica D*, vol. 96:291–320, 1996.

[Atk02] Atkins, J. F. and Gesteland, R. The 22nd amino acid. *Science*, vol. 296:1409–1410, 2002.

[Ave44] Avery, O. T., MacLeod, C. M., and McCarty, M. Studies on the chemical nature of the substance inducing transformation of Pneumococcal types. Induction of transformation by a Deoxyribonucleic acid fraction isolated from Pneumococcus type III. *Journal of Experimental Medicine*, vol. 79:137–158, 1944.

[Bai95] Bailey, T. L. and Elkan, C. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning Journal*, vol. 21:51–83, 1995.

[Bai98] Bailey, T. L. and Gribskov, M. Combining evidence using p-values: Application to sequence homology searches. *Bioinformatics*, vol. 14:48–54, 1998.

[Bal94] Baldi, P., Chauvin, Y., Hunkapiller, T., and McClure, M. A. Hidden Markov Models of biological primary sequence information. In *Proc. Nat. Academy of Sciences USA – Biochemistry*, vol. 91, pages 1059–1063. Feb. 1994.

[Bal95] Baldi, P., Brunak, S., Chauvin, Y., Engelbrecht, J., and Krogh, A. Periodic sequence patterns in human exons. In *Proc. Int. Conf. Intelligent Systems for Molecular Biology*, pages 30–38. 1995.

[Bal01] Baldi, P. and Brunak, S. *Bioinformatics – The Machine Learning Approach*. MIT Press Cambridge, Massachusetts; London, England, 2nd edn., 2001.

[Bän02] Bäni, W. *Wavelets*. Oldenbourg, Munich Vienna, 2002.

[Bar97] Barrett, C., Hughey, R., and Karplus, K. Scoring Hidden Markov Models. *Computer Applications in the Bioscience*, vol. 13(2):191–199, 1997.

[Bat00] Bateman, A., Birney, E., Durbin, R., Eddy, S. R., Howe, K. L., et al. The Pfam protein families database. *Nucleic Acids Research*, vol. 28(1):263–266, 2000.

[Bel57] Belman, R. E. *Dynamic Programming*. Princeton University Press, 1957.

[Ber77] Bernstein, F. C., Koetzle, T. F., Williams, G. J., Meyer, E. J., Brice, M., et al. The Protein Data Bank: A computer–based archival file for macromolecular structures. *J. Molecular Biology*, vol. 112:535–542, 1977.

[Ber02] Berman, H. M., Battistuz, T., Bhat, T. N., Bluhm, W. F., Bourne, P. E., et al. The Protein Data Bank. *Acta Crystallographica*, vol. D(58):899–907, 2002.
`www.pdb.org`.

[Bir95] Birge, R. R. Protein-Based Computers. *Scientific American*, pages 66–71, Mar. 1995.

[Bir01] Birney, E. and Copley, R. Wise2 – Documentation. European Bioinformatics Institute (EMBL-EBI), 2001.
`www.ebi.ac.uk/Wise2/documentation.html`.

[Böc91] Böck, A., Forchhammer, K., Heider, J., Leinfelder, W., Sawers, G., et al. Selenocysteine: The 21st amino acid. *Molecular Microbiology*, vol. 5(3):515–520, 1991.

[Boe03] Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M.-C., Estreicher, A., et al. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Research*, vol. 31(1):365–370, 2003.

[Bra98] Branden, C. and Tooze, J. *Introduction to Protein Structure*. Garland Publishing Inc., 2nd edn., 1998.

[Bre86] Breslauer, K. J., Frank, R., Blocker, H., and Marky, L. A. Predicting DNA duplex stability from the base sequence. *Proc. Nat. Academy of Sciences USA*, vol. 83(11):3746–3750, 1986.

[Bre98] Brenner, S. E., Chothia, C., and Hubbard, T. J. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Nat. Academy of Sciences USA*, vol. 95:6073–6078, 1998.

[Bro91] Bronstein, I. and Semendjajew, K. *Taschenbuch der Mathematik*. B.G. Teubner Verlagsgesellschaft, 25th edn., 1991.

[Bro93] Brown, M. P., Hughey, R., Krogh, A., Mian, I. S., Sjölander, K., et al. Using Dirichlet mixture priors to derive Hidden Markov Models for protein families. In *Proc. Int. Conf. Intelligent Systems for Molecular Biology*, pages 47–55. 1993.

[Bur83] Burt, P. J. and Adelson, E. H. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, vol. COM-31,4:532–540, 1983.

[Bur97] Burge, C. and Karlin, S. Prediction of complete gene structures in human genomic DNA. *J. Molecular Biology*, pages 78–94, 1997.

[Car88] Carrillo, H. and Lipmann, D. The multiple sequence alignment problem in biology. *SIAM Journal on Applied Mathematics archive*, vol. 48(5):1073–1082, Oct. 1988.

[Cho92] Chothia, C. One thousand families for the molecular biologist. *Nature*, vol. 357:543–544, 1992.

[Chr00] Christiani, N. and Shawe-Taylor, J. *An introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[Chu89] Churchill, G. A. Stochastic models for heterogeneous DNA sequences. *Bulletin of Mathematical Biology*, vol. 51:79–94, 1989.

[Chu92] Churchill, G. A. Hidden Markov Chains and the analysis of genome structure. *Computers and Chemistry*, vol. 116(2):107–115, 1992.

[Con02] Conte, L. L., Brenner, S. E., Hubbard, T. J., Chothia, C., and Murzin, A. G. SCOP database in 2002: Refinements accomodate structural genomics. *Nucleic Acids Research*, vol. 30(1):164–267, 2002.

[Cor95] Cortes, C. and Vapnik, V. Support-Vector Networks. *Machine Learning*, vol. 20(3):273–297, 1995.

[Cos97] Cosic, I. *The Resonant Recognition Model of Macromolecular Bioactivity – Theory and Applications*. Birkhäuser Verlag, Basel, 1997.

[Cri01] Cristea, P. D. Genetic signals: An emerging concept. In *Proc. Int. Workshop on System, Signals and Image Processing IWSSIP 2001, Bucharest, Romania*. Jun. 2001.

[Cri03] Cristea, P. D. Large scale features in DNA genomic signals. *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 83:871–888, 2003.

[Dau92] Daubechies, I. *Ten Lectures on Wavelets*. CBMS-NSF Regional Conference Series In Applied Mathematics. Society for Industrial and Applied Mathematics, 1992.

[Day78] Dayhoff, M. O., Schwartz, R. M., and Orcutt, B. C. A model of evolutionary change in proteins. In M. O. Dayhoff, ed., *Atlas of Protein Sequence and Structure*, vol. 5, supplement 3 chapter 22, pages 345–352. The National Biomedical Research Foundation, 1978.

[De 00] De Trad, C. H., Fang, Q., and Cosic, I. The resonant recognition model (RRM) predicts amino acid residues in highly conserved regions of the hormone prolactin (PRL). *Biophysical Chemistry*, vol. 84:149–157, 2000.

[De 01] De Trad, C. H., Fang, Q., and Cosic, I. An overview of protein sequence comparison using Wavelets. *Proc. 2nd Conf. of the Victorian Chapter of the IEEE EMBS*, pages 115–119, 2001.

[De 02] De Trad, C. H., Fang, Q., and Cosic, I. Protein sequence comparison based on the Wavelet transform approach. *Journal of Protein Engineering*, vol. 15(3):193–203, 2002.

[Dem77] Dempster, A., N.M., L., and Rubin, D. Maximum likelihood from incomplete data via the *EM* algorithm. *Journal of the Royal Statistical Society*, vol. 39:1–38, 1977. Series B (methodological).

[Dre00] Drews, J. Drug discovery: A historical perspective. *Science*, vol. 287:1960–1964, 2000.

[Dur98] Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998.

[Edd95a] Eddy, S. R. Multiple alignment using Hidden Markov Models. In *Proc. Int. Conf. Intelligent Systems for Molecular Biology*, pages 114–120. 1995.

[Edd95b] Eddy, S. R., Mitchison, G., and Durbin, R. Maximum discrimination Hidden Markov Models of sequence consensus. *Journal of Computational Biology*, vol. 2:9–23, 1995.

[Edd01] Eddy, S. R. HMMER: Profile Hidden Markov Models for biological sequence analysis. *http://hmmer.wustl.edu/*, 2001.

[Edd04] Eddy, S. R. What is dynamic programming? *Nature Biotechnology*, vol. 22:909–910, Jul. 2004.

[Fin99] Fink, G. A. Developing HMM-based recognizers with ESMERALDA. In V. Matoušek, P. Mautner, J. Ocelíková, and P. Sojka, eds., *Text, Speech and Dialogue*, vol. 1692 of *Lecture Notes in Artificial Intelligence*, pages 229–234. Springer, Berlin Heidelberg, 1999.

[Fin03] Fink, G. A. *Mustererkennung mit Markov-Modellen*. Teubner Verlag, 2003.

[Fis99] Fischer, A. and Stahl, V. Database and online adaptation for improved speech recognition in car environments. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*. 1999.

[Fri01] Fritzsche, M. *Anwendung von Verfahren der Mustererkennung zur Detektion von Landminen mit Georadaren*. Ph.D. thesis, Fakultät für Elektrotechnik der Universität Fridericiana Karlsruhe, 2001.

[Gar99] Garzon, M. H. and Deaton, R. J. Biomolecular computing and programming. *IEEE Trans. on Evolutionary Computation*, vol. 3(3):236–250, 1999.

[Gau92] Gauvain, J.-L. and Lee, C.-H. MAP estimation of continuous density HMM: Theory and applications. In *Proc. DARPA Speech and Natural Language Workshop*. 1992.

[Gei96] Geiger, D. and Heckerman, D. A characterization of the Dirichlet distribution through global and local parameter independence. Tech. Rep. MSR-TR-94-16, Microsoft Research Advanced Technology Division, Mar. 1996.

[Ger92] Gersho, A. and Gray, R. *Vector Quantization and Signal Compression*. Communications and Information Theory. Kluwer Academic Publishers, 1992.

[Gon01] Gonik, L. and Wheelis, M. *Genetik in Cartoons*. Parey Buchverlag Berlin, 5th edn., 2001.

[Got82] Gotoh, O. An improved algorithm for matching biological sequences. *J. Molecular Biology*, vol. 162:705–708, 1982.

[Gou01] Gough, J., Karplus, K., Hughey, R., and Chothia, C. Assignment of homology to genome sequences using a library of Hidden Markov Models that represent all proteins of known structure. *J. Molecular Biology*, vol. 313:903–919, 2001.

[Gri28] Griffith, F. The significance of Pneumococcal types. *Journal of Hygiene*, vol. 64:129–175, 1928.

[Gri87] Gribskov, M., McLachlan, A. D., and Eisenberg, D. Profile analysis: Detection of distantly related proteins. In *Proc. Nat. Academy of Science USA – Biochemistry*, vol. 84, pages 4355–4358. Jul. 1987.

[Gru97] Grundy, W. N., Bailey, T. L., Elkan, C. P., and Baker, M. E. Meta-MEME: Motif-based Hidden Markov Models of protein families. *Computer Applications in the Bioscience*, vol. 13(4):397–406, 1997.

[Hal02] Halperin, I., Ma, B., Wolfson, H., and Nussinov, R. Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins: Sructure, Function and Genetics*, vol. 47:409–443, 2002.

[Hau93] Haussler, D., Krogh, A., Mian, I. S., and Sjölander, K. Protein modeling using Hidden Markov Models: Analysis of Globins. In T. Mudge, V. Milutinovic, and L. Hunter, eds., *Proc. Twenty-Sixth Ann. Hawaii Int. Conf. System Sciences*, vol. 1, pages 792–802. 1993.

[Hel75] Helck, W., ed. *Lexikon der Ägyptologie*. Harrassowitz – Wiesbaden, 1975.

[Hen92] Henikoff, S. and Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proc. Nat. Academy of Sciences USA*, vol. 89:10915–10919, 1992.

[Hen96a] Henderson, J., Salzberg, S., and Fasman, K. Finding genes in human DNA with a Hidden Markov Model. In *Proc. Int. Conf. Intelligent Systems for Molecular Biology*. AAAI Press., Jun. 1996.

[Hen96b] Henikoff, S. Scores for sequence searches and alignments. *Current Opinion in Structural Biology*, vol. 6:353–360, 1996.

[Hen99] Henikoff, S., Henikoff, J. G., and Pietrokovski, S. Blocks+: A non-redundant database of protein alignment blocks dervied from multiple compilations. *Bioinformatics*, vol. 15(6):471–479, 1999.

[Hen00] Henikoff, J. G., Greene, E. A., Pietrokovski, S., and Henikoff, S. Increased coverage of protein families with the blocks database servers. *Nucleic Acids Research*, vol. 28:228–230, 2000.

[Her89] Hering, E., Martin, R., and Stohrer, M. *Physik für Ingenieure*. VDI-Verl., Düsseldorf, third edn., 1989.

[Hil03] Hillisch, A. and Hilgenfeld, R., eds. *Modern Methods of Drug Discovery*. Birkhäuser Verlag, Basel – Boston – Berlin, 2003.

[Hog01] Hogenesch, J. B., Ching, K. A., Batalov, S., Su, A. I., Walker, J. R., et al. A comparison of the Celera and Ensembl predicted gene sets reveals little overlap in novel genes. *Cell*, vol. 106(4):413–415, Aug. 2001.

[Hua89] Huang, X. D. and Jack, M. A. Semi-Continuous Hidden Markov Models for speech signals. *Computer Speech & Language*, vol. 3:239–251, 1989.

[Hua01] Huang, X., Acero, A., and Hon, H.-W. *Spoken Language Processing – A Guide to Theory, Algorithm, and System Development*. Prentice Hall PTR, 2001.

[Hud99] Hudak, J. and McClure, M. A comparative analysis of computational motif-detection methods. In *Proc. Pacific Symposium on Biocomputing*, vol. 4, pages 138–149. 1999.

[Hug96] Hughey, R. and Krogh, A. Hidden Markov Models for sequence analysis: Extension and analysis of the basic method. *Computer Applications in the Bioscience*, vol. 12(2):95–108, 1996.

[IHG04] IHGSC. – International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, vol. 43:931–945, Oct. 2004.

[Jaa98] Jaakkola, T., Diekhans, M., and Haussler, D. A discriminant framework for detecting remote protein homologies. *Journal of Computational Biology*, vol. 7(1,2):95–114, 1998.

[Jaa99] Jaakkola, T., Diekhans, M., and Haussler, D. Using the Fisher kernel method to detect remote protein homologies. In *Proc. Int. Conf. Intelligent Systems for Molecular Biology*, pages 149–158. 1999.

[Jac77] Jacob, F. Evolution and tinkering. *Science*, vol. 196:1161–1166, 1977.

[Joh72] Johnson, N. L. and Kotz, S. *Distributions in Statistics*. John Wiley & Sons, Inc., 1972.

[Jon04] Jones, N. C. and Pevzner, P. A. *An Introduction To Bioinformatics Algorithms*. MIT Press Inc., 2004.

[Jua90] Juan, B.-H. and Rabiner, L. The segmental *K*-means algorithm for estimating parameters of Hidden Markov Models. In *IEEE Trans. on Acoustics, Speech, and Signal Processing*, pages 1639–1641. 1990.

[Kar90] Karlin, S. and Altschul, S. F. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Nat. Academy of Sciences USA*, vol. 87:2264–2268, 1990.

[Kar95] Karplus, K. Evaluating regularizers for estimating distributions of amino acids. In C. Rawlings, D. Clark, R. Altman, L. Hunter, T. Lengauer, and S. Wodak, eds., *Proc. Int. Conf. Intelligent Systems for Molecular Biology*, pages 188–196. 1995.

[Kar98] Karplus, K., Barrett, C., and Hughey, R. Hidden Markov Models for detecting remote protein homologies. *Bioinformatics*, vol. 14(10):846–856, 1998.

[Kau03] Kauer, G. and Blöcker, H. Applying signal theory to the analysis of biomolecules. *Bioinformatics*, vol. 19(16):2016–2021, 2003.

[Kaw00] Kawashima, S. and Kanehisa, M. AAindex: Amino acid index database. *Nucleic Acids Research*, vol. 28(1):374, 2000.

[Kri04] Krishnan, A., Li, K.-B., and Issac, P. Rapid detection of conserved regions in protein sequences using Wavelets. *In Silico Biology*, vol. 4, 2004.

[Kro94a] Krogh, A., Brown, M., Mian, I. S., Sjölander, K., and Haussler, D. Hidden Markov Models in computational biology: Applications to protein modeling. *J. Molecular Biology*, vol. 235:1501–1531, 1994.

[Kro94b] Krogh, A., Mian, I., and Haussler, D. A Hidden Markov Model that finds genes in E. coli. *Nucleic Acids Research*, vol. 22:4768–4778, 1994.

[Kro97] Krogh, A. Two methods for improving performance of an HMM and their application for gene finding. In *Proc. Fifth Int. Conf. Intelligent Systems for Molecular Biology*, pages 179–186. 1997.

[Kul96] Kulp, D., Haussler, D., Reese, M. G., and Eeckman, F. H. A generalized Hidden Markov Model for the recognition of human genes in DNA. In *Proc. Int. Conf. Intelligent Systems for Molecular Biology*. 1996.

[Lan01] Lander, E. S. et al. Initial sequencing and analysis of the human genome. *Nature*, vol. 409:860–921, 2001.

[Las02] Lassmann, T. and Sonnhammer, E. L. Quality assessment of multiple alignment programs – Minireview. *Fed. of Europ. Biochem. Societies Letters*, (529):126–130, 2002.

[Leg94] Leggetter, C. J. and Woodland, P. C. Speaker adaptation of HMMs using linear regression. Tech. rep., Cambridge University Engineering Department, Jun. 1994.

[Leg95] Leggetter, C. J. and Woodland, P. C. Maximum likelihood linear regression for speaker adaptation of continuous density Hidden Markov Models. *Computer Speech & Language*, pages 171–185, 1995.

[Lej04] Leja, D. et al. Talking glossary of genetic terms – illustration. National Human Genome Research Institute (NHGRI), 2004. `www.genome.gov/Pages/Hyperion/DIR/VIP/Glossary/`.

[Les02] Leslie, C., Eskin, E., and Noble, W. S. The spectrum kernel: A string kernel for SVM protein classification. In *Proc. Seventh Pacific Biocomputing Symposium*, pages 566–575. 2002.

[Les04] Leslie, C., Eskin, E., Cohen, A., Weston, J., and Noble, W. S. Mismatch string kernels for discriminative protein classification. *Bioinformatics*, vol. 20(4):467–476, 2004.

[Lev66] Levenshtein, V. Binary codes capable of correcting insertions and reversals. *Soviet Physics Doklady*, vol. 10:707–710, 1966.

[Lew94] Lewin, B. *Genes V*. Oxford University Press, 1994.

[Lia02] Liao, L. and Noble, W. S. Combining pairwise sequence similarity and support vector machines for remote homology detection. In *Proc. Sixth Ann. Int. Conf. Computational Molecular Biology*, pages 225–232. 2002.

[Lin80] Linde, Y., Buzo, A., and Gray, R. An algorithm for vector quantizer design. *IEEE Trans. on Communications*, vol. 28(1):84–95, 1980.

[Lin01] Lin, K., May, A. C., and Taylor, W. R. Amino acid substitution matrices from an artificial neural network model. *J. Molecular Biology*, (8):471–481, 2001.

[Liu04] Liu, X., Fan, K., and Wang, W. The number of protein folds and their distribution over families in nature. *Proteins: Structure, Function, and Bioinformatics*, vol. 54:491–499, 2004.

[Low76] Lowerre, B. *The Harpy Speech Recognition System*. Carnegie-Mellon University, 1976.

[Luk98] Lukashin, A. V. and Borodovsky, M. GeneMark.hmm: New solutions for gene finding. *Nucleic Acids Research*, vol. 26(4):1107–1115, 1998.

[Mac67] MacQueen, J. Some methods for classification and analysis of multivariate observations. In L. M. L. Cam and J. Neyman, eds., *Proc. Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pages 281–296. 1967.

[Mal98] Mallat, S. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[Mam96] Mamitsuka, H. A learning method of Hidden Markov Models for sequence discrimination. *Journal of Computational Biology*, vol. 3(3):361–373, 1996.

[Man97] Mandell, A. J., Selz, K. A., and Shlesinger, M. F. Wavelet transformation of protein hydrophobicity sequences suggest their membership in structural families. *Physica A*, vol. 244:254–262, 1997.

[Mer03] Merkl, R. and Waack, S. *Bioinformatik Interaktiv – Algorithmen und Praxis*. Wiley-VCH, 2003.

[Mor99] Morgenstern, B. DIALIGN 2: Improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, vol. 15(3):211–218, 1999.

[Mou04] Mount, D. W. *Bioinformatics: Sequence and Genome Analysis*. Cold Spring Harbor Laboratory Press, second edn., 2004.

[Mur95] Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C. SCOP: A structural classification of proteins database for the investigation of sequences and structures. *J. Molecular Biology*, vol. 247:536–540, 1995.

[Mur02] Murray, K. B., Gorse, D., and Thornton, J. M. Wavelet transforms for the characterization and detection of repeating motifs. *J. Molecular Biology*, vol. 316:341–363, 2002.

[Nee70] Needleman, S. B. and Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Molecular Biology*, vol. 48:443–453, 1970.

[Nie83] Niemann, H. *Klassifikation von Mustern*. Berlin: Springer, 1983.

[Not00] Notredame, C., Higgins, D. G., and Heringa, J. T-Coffee: A novel method for multiple sequence alignments. *J. Molecular Biology*, vol. 302:205–217, 2000.

[Opp89] Oppenheim, A. V. and Schafer, R. W. *Discrete-Time Signal Processing*. Prentice Hall, Englewood Cliffs, New Jersey, 1989.

[Ore97] Orengo, C., Michie, A., Jones, S., Jones, D., Swindells, M., et al. CATH – A hierarchic classification of protein domain structures. vol. 5(8):1093–1108, 1997.

[Pea88] Pearson, W. R. and Lipman, D. J. Improved tools for biological sequence comparison. In *Proc. Nat. Academy of Sciences USA – Biochemistry*, vol. 85, pages 2444–2448. Apr. 1988.

[Ped03] Pedersen, J. S. and Hein, J. Gene finding with a Hidden Markov Model of genome structure and evolution. *Bioinformatics*, vol. 19(2):219–227, 2003.

[Per00a] Percival, D. B. and Walden, A. T. *Wavelet Methods for Time Series Analysis*. Cambridge Series in Statistical and Probabilistical Mathematics. Cambridge University Press, 2000.

[Per00b] Perrone, M. P. and Connell, S. D. K-means clustering for Hidden Markov Models. In L. Schomaker and L. Vuurpijl, eds., *Proc. Int. Workshop on Frontiers in Handwriting Recogntion*, pages 229–238. Sep. 2000.

[Pir01] Pirogova, E. and Cosic, I. Examination of amino acid indexes within the Resonant Recognition Model. In *Proc. IEEE Engineering in Medicine and Biology Society Conf.* Feb. 2001.

[Plö02] Plötz, T. Genetic Relationships Analysis based on Statistical Sequence Profiles – The GRAS^{2}P project, 2002.
`www.techfak.uni-bielefeld.de/ags/ai/projects/grassp`.

[Plö04] Plötz, T. and Fink, G. A. Feature extraction for improved Profile HMM based biological sequence analysis. In *Proc. Int. Conf. on Pattern Recognition*. 2004.

[Pou00] Poularikas, A. D., ed. *The Transforms and Applications Handbook*. CRC Press LLC, 2nd edn., 2000.

[Qiu03] Qiu, J., Liang, R., Zou, X., and Mo, J. Prediction of protein secondary structure based on continuous Wavelet transform. *Talanta*, vol. 61:285–293, 2003.

[Rab89] Rabiner, L. R. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, vol. 77(2):257–286, 1989.

[Red01] Redford, D. B., ed. *The Oxford Encyclopedia of Ancient Egypt*. Oxford University Press, 2001.

[Rey97] Reynolds, D. A. Comparison of background normalization methods for text-independent speaker verification. In *Proc. European Conf. on Speech Communication and Technology*, vol. 1, pages 963–966. Rhodes, Greece, 1997.

[Rio91] Rioul, O. and Vetterli, M. Wavelets and signal processing. *IEEE Signal Processing Magazine*, pages 14–38, Oct. 1991.

[Sal98] Salzberg, S. L., Searls, D. B., and Kasif, S., eds. *Computational Methods in Molecular Biology*. Elsevier, 1998.

[San91] Sander, C. and Schneider, R. Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, vol. 9:56–68, 1991.

[Sch93] Schukat-Talamazzini, E. G., Bielecki, M., Niemann, H., Kuhn, T., and Rieck, S. A non-metrical space search algorithm for fast gaussian vector quantization. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 688–691. Minneapolis, 1993.

[Sch95] Schukat-Talamazzini, E. G. *Automatische Spracherkennung*. Vieweg Verlag, 1995.

[Sch96] Schlittgen, R. *Einführung in die Statistik*. Oldenbourg Verlag, 6th edn., 1996.

[Sjö96] Sjölander, K., Karplus, K., Brown, M., Hughey, R., Krogh, A., et al. Dirichlet mixtures: A method for improved detection of weak but significant protein sequence homology. *Computer Applications in the Biosciences*, vol. 12(4):327–345, 1996.

[Smi81] Smith, T. and Waterman, M. Identification of common molecular subsequences. *J. Molecular Biology*, vol. 147:195–197, 1981.

[Son98] Sonnhammer, E., Eddy, S., Birney, E., Bateman, A., and Durbin, R. Pfam: multiple sequence alignments and HMM-profiles of protein domains. *Nucleic Acids Research*, vol. 26(1):320–322, 1998.

[Spa02] Spang, R., Rehmsmeier, M., and Stoye, J. A novel approach to remote homology detection: Jumping alignments. *Journal of Computational Biology*, vol. 9(5):747–760, 2002.

[Sto82] Stormo, G. D., Schneider, T. D., Gold, L., and Ehrenfeucht, A. Use of the 'Perceptron' algorithm to distinguish translational initiation sites in *E. coli*. *Nucleic Acids Research*, vol. 10(9):2997–3011, 1982.

[Str91] Stryer, L. *Biochemie*. Spektrum Akademischer Verlag, 1991.

[Swa86] Swannell, J., ed. *The Little Oxford Dictionary*. Oxford University Press, 1986.

[Tho94] Thompson, J. D., Higgins, D. G., and Gibson, T. J. CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, vol. 22:4673–4680, 1994.

[Tol01] Tollman, P., Guy, P., Altshuler, J., Flanagan, A., and Steiner, M. A revolution in R&D: How genomics and genetics are transforming the biopharmaceutical industry. Boston Consulting Group, Nov. 2001.

[Vai04] Vaidyanathan, P. Signal processing problems in genomics. Slides of a presentation given at IEEE Int. Symp. on Circuits and Systems, 2004. `www.iscas2004.org/VaidyanathanPlenary.pdf`.

[Vel72] Veljkovic, V. and Slavic, I. Simple general-model pseudopotential. *Physical Review Letters*, vol. 29(2):105–108, 1972.

[Ven01] Venter, J. C. et al. The sequence of the human genome. *Science*, vol. 291:1304–1351, 2001.

[Vit67] Viterbi, A. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Trans. on Information Theory*, vol. 13:260–269, 1967.

[Wan01] Wang, J., Ma, Q., Shasha, D., and Wu, C. H. New techniques for extracting features from protein sequences. *IBM Systems Journal*, vol. 40(2):426–441, 2001.

[Wat53] Watson, J. D. and Crick, F. H. C. Molecular structure of nucleic acids. *Nature*, vol. 171(4356):737f., Apr. 1953.

[Wie03] Wienecke, M. *Videobasierte Handschrifterkennung*. Ph.D. thesis, Faculty of Technology, Bielefeld University, 2003.

[Wu00] Wu, C. H. and McLarty, J. W. *Neural Networks and Genome Informatics*. Elsevier, 2000.

[Yak70] Yakowitz, S. J. Unsupervised learning and the identification of finite mixtures. *IEEE Trans. on Information Theory*, vol. 16(3):330–338, 1970.

[You99] Young, W. T. A working guide to glues. *Fine Woodworking Magazine*, vol. 134:60–67, 1999.

[Zel97] Zell, A. *Simulation neuronaler Netze*. Addison-Wesley, Bonn [u.a.], 1997.

[Zha97] Zhang, C.-T. Relations of the numbers of protein sequences, families and folds. *Protein Engineering*, vol. 10(7):757–761, 1997.