From the Center for Genomics and Bioinformatics,
Karolinska Institutet, Stockholm, Sweden

Predicting transmembrane topology and signal peptides with hidden Markov models

Lukas Käll

Stockholm, 2006

Karolinska Institutet

Published and printed by Larserics Digital Print, Sundbyberg
ISBN 91-7140-719-7

Abstract

Transmembrane proteins make up a large and important class of proteins. About 20% of all genes encode transmembrane proteins. They control both substances and information going in and out of a cell. Yet basic knowledge about membrane insertion and folding is sparse, and our ability to identify, over-express, purify, and crystallize transmembrane proteins lags far behind the field of water-soluble proteins.

It is difficult to determine the three dimensional structures of transmembrane proteins. Therefore, researchers normally attempt to determine their topology, *i.e.* which parts of the protein are buried in the membrane, and on what side of the membrane are the other parts located.

Proteins aimed for export have an N-terminal sequence known as a *signal peptide* that is inserted into the membrane and cleaved off. The same mechanism that inserts transmembrane proteins into their membranes also handles the export of protein with signal peptides. Transmembrane helices and signal peptides thus have many features in common.

In silico methods for predicting transmembrane topology and methods for predicting signal peptides from amino acid sequence are a fast and relatively accurate alternative to biochemical experiments. A methodology called *hidden Markov models* (HMMs) has proved particularly useful for these and other prediction tasks.

In this thesis, properties of transmembrane topology predictors and signal peptide predictors are investigated. It includes three novel HMM based prediction methods.

i) A combined transmembrane topology and signal peptide predictor, Phobius. The paper shows that cross predictions, *i.e.* signal peptides predicted as transmembrane helices and *vice versa*, are a common problem. About 10% of the genes in *E.coli* have overlapping signal peptide and transmembrane helix predictions by conventional predictors. We were able to dramatically lower these false cross predictions.

ii) A method for detecting remote G protein-coupled receptor (GPCR) families, GPCRHMM. GPCRs are a very large and divergent superfamily of transmembrane proteins. We designed a hidden Markov model based on the topological regions of the superfamily. We searched five genomes and predicted 120 previously not annotated sequences as possible GPCRs. The majority of these predictions (102) were in C. elegans, but 4 were found in human and 7 in mouse. We as well conclude that a family of odorant receptors in *Drosophila* are not GPCRs.

iii) A method to improve predictions with HMMs of generic sequence features (such as transmembrane segments or signal peptides) by including homologs. We show that the performance of Phobius using this decoder was significantly better than with other decoders.

We also assessed the difficulty of benchmark sets used in transmembrane topology prediction. By studying the level of agreement between different predictors applied to typical benchmark sets and whole proteome sets, we concluded that the benchmark sets are far easier to predict than reality. In other words, the accuracies reported in benchmark studies are exaggerated.

This thesis also includes a paper presenting a hypothesis of the transmembrane topology of presenilin, a protein involved in the development of Alzheimer's disease. By comparing the output of several transmembrane topology predictors with experimental results from previous studies, a novel nine-transmembrane topology with an extracellular C-terminus was elucidated.

List of publications

Publications included in this thesis

I **Lukas Käll** and Erik L.L. Sonnhammer.
Reliability of transmembrane predictions in whole-genome data.
FEBS Letters, **532**:415-418, Dec 2002.

II **Lukas Käll**, Anders Krogh and Erik L.L. Sonnhammer.
A combined transmembrane topology and signal peptide prediction method.
Journal of Molecular Biology, **338**(5):1027-1036, May 2004.

III Anna Henricsson, **Lukas Käll** and Erik L.L. Sonnhammer.
Transmembrane topology of presenilin by reconciling experimental and computational approaches.
FEBS Journal, **272**(11):2727-2733, June 2005.

IV **Lukas Käll**, Anders Krogh and Erik L.L. Sonnhammer,
An HMM posterior decoder for sequence feature prediction that includes homology information.
Bioinformatics, **21**(Suppl 1):i251-i257, June 2005.

V Markus Wistrand*, **Lukas Käll*** and Erik L.L. Sonnhammer,
A general model of G protein-coupled receptor sequences and its application to detect remote homologs.
Protein Science, **15**(3):509-521, Mars 2006.
* These authors contributed equally to this work.

Other Publications

- **Lukas Käll** and Erik L.L.Sonnhammer.
  Predicting membrane proteins.
  Encyclopedia of Genetics, Genomics, Proteomics and Bioinformatics, John Wiley & Sons.

Prediction servers

Three of the methods developed during the Ph.D. project are available to the public through web servers:

Phobius Prediction of transmembrane topology and signal peptides
`http://phobius.cgb.ki.se/` or `http://phobius.binf.ku.dk/`.

PolyPhobius Prediction of transmembrane topology and signal peptides aided by homolog sequences
`http://phobius.cgb.ki.se/poly.html`.

GPCRHMM Prediction of existence and transmembrane topology of GPCRs.
`http://gpcrhmm.cgb.ki.se/`.

Software packages

The following software package has been designed as a result of the project:

HomologHMM An HMM decoder that can handle homolog sequences.
`http://phobius.cgb.ki.se/data.html`

Contents

Abbreviations

ANN	Artificial Neural Network
CML	Conditional Maximum Likelyhood
DNA	Deoxyribonucleic acid
ER	Endoplasmic reticulum
GFP	Green fluorescent protein
GPCR	G protein-coupled receptor
HMM	Hidden Markov model
mRNA	Messenger Ribonucleic acid
SRP	Signal Recognition Particle
SP	Signal Peptide
TM	Transmembrane

Chapter 1

Biomembranes

Life depends on both interaction and isolation[1]. A cell needs to keep the integrity of its essential processes and retain its reactants and enzymes. But at the same time it is dependent on the ability to absorb new nutrition and deposit waste to its environment. Also, it has to gather information about its environment. In a cell, this dilemma is solved by a surrounding semipermeable plasma membrane.

Biological membranes consist of amphipatic molecules, *i.e.* molecules that contain both a hydrophilic and a hydrophobic end, most often phospholipids. The molecules form a bilayer, where they are aligned in such a manner that the hydrophobic tails of the molecules in each layer are facing the other layer, and that the hydrophilic parts are facing away from the center of the membrane. The bilayer effectively blocks the passage of hydrophobic molecules, or molecules larger than a couple of Ångström. However, biological membranes contain a large number of transmembrane (TM) proteins, proteins that span the membrane, and hence are able to serve as a bridge between the cytoplasm (the inside of the cell) and the extracellular world. Transmembrane proteins are included in a wide variety of pathways, and serve among other things as transporters of ions and molecules across the membrane and as chemosensors and hormone receptors. They are of high importance for medicine due to their strategic role. More than half of the protein targets of commercially available drugs are transmembrane proteins[2, 3], even though only a fifth of all human proteins are transmembrane proteins[4].

Eukaryote cells harbor organelles, compartments enclosed by a membrane, where milieus suitable for more specialized processes are kept. In a similar manner as for the entire cell, these compartments are shielded off by lipid bilayers which also are bridged by transmembrane proteins.

Most transmembrane proteins are α-helical, *i.e.* the segments of the protein crossing the membrane form 18-35 amino acid long hydrophobic α-helices. There are also β-barrel membrane proteins, where the segments of the protein crossing the membrane form β-sheets. However, the first class is far more common[5], hence I will only refer to these when describing TM proteins in this text.

1.1 Translocon

An interesting transporter is the *translocon* and its core components the protein-conducting channels (PCCs)[6]. Translocons are able to translocate amino acid chains across membranes, that

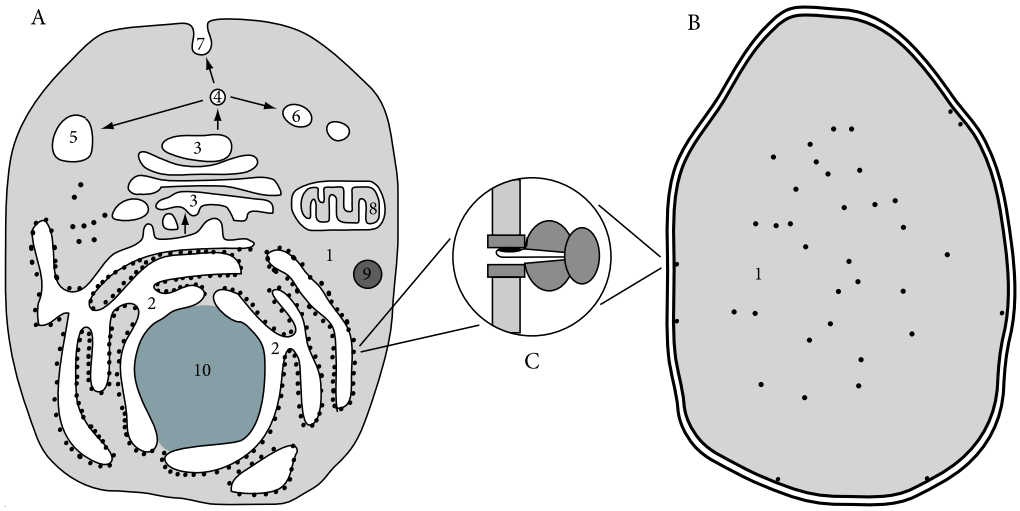


Figure 1.1: The secretory pathway in (A) eukaryotes and (B) bacteria are both dependent on (C) translocon/ribosome complexes. In eukaryotes the nascent proteins are translocated from the (1) cytosol into the (2) endoplasmic reticulum (ER). The proteins are then transported in (4) vesicles through (3) the Golgi apparatus. The proteins may continue to a (5) lysosome or (6) endosome, or they are (7) secreted from the cell. Proteins targeting (8) mitochondria, (9) peroxisomes, or the (10) nucleus are exported by other mechanisms. In bacteria nascent proteins are translocated over the plasma membrane by the translocon. Other mechanisms handles the transport over the bacterial cell wall (the outer membrane).

later form fully folded proteins in their new environment. They are also able to insert the TM helices of TM proteins into membranes, and transport their translocated loops.

The PCC is, in all kingdoms of life, built up from three proteins. In mammalians, they are named Sec61$\alpha/\beta/\gamma$, in yeast Sec61p/Sbh1p/Sss1p, in archea SecY/β/E, and in eubacteria SecY/G/E.

The heterotrimeric PCCs form dimers. It is debated if the dimers in their turn form dimers[7] or not[8]. So in each translocon there are at least two copies of the PCC. Since there is only room for one PCC to be active at the time, the purpose of the other copies are unknown, even though there are speculations that they are used in the insertion of transmembrane proteins[8] or that they have a structural role or that they recruit accessory factors[7].

1.2 The general secretory pathway

The translocons of the general secretory pathway are located in the plasma membrane of prokaryotes and in the endoplasmic reticulum (ER) of eukaryotes. This implies that in prokaryotes translocated proteins are excreted from the cell, while in eukaryotes they enter the ER. However many proteins are moved from the ER to other organelles or excreted, by vesicular transport, a process where a part of the membrane bud from the ER lumen to later fuse with another organelle (See Figure 1.1). In the same manner most TM proteins in eukaryotes are inserted by translocons into the ER membrane and transported to other membranes by vesicular transport.

All protein chains that are transported through a translocon are distinguished by a signal

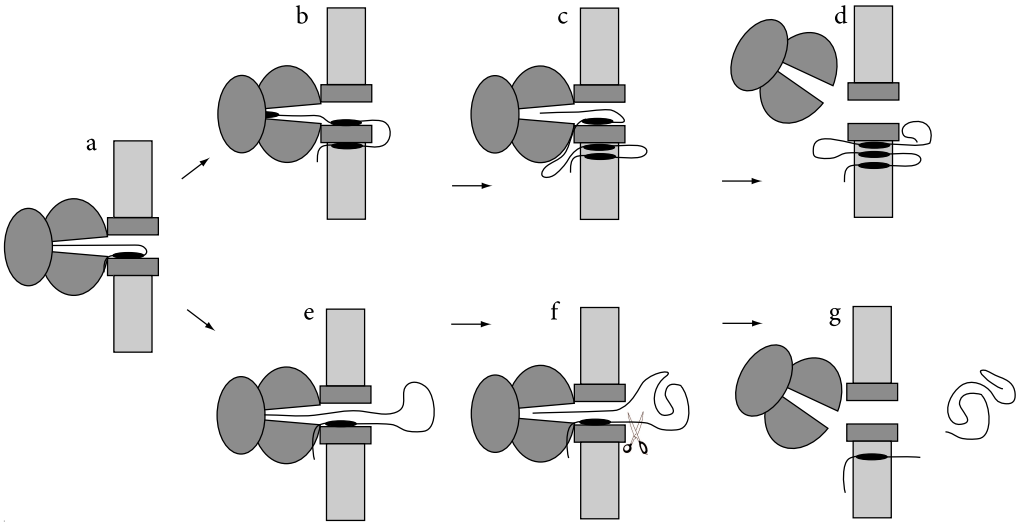


Figure 1.2: Translocon/ribosome complex inserting a transmembrane protein (a-b-c-d) and translocating a protein with signal peptide (a-e-f-g).

sequence, a hydrophobic stretch of amino acids. The signal sequence could either form a transmembrane helix or an *signal peptide* (SP). An SP is an N-terminal 15-30 amino acid long recognition sequence, the larger mid part being a hydrophilic α-helix. Unlike transmembrane segments, SPs are normally cleaved off from the rest of the protein, by the enzyme signal peptidase, during translocation.

The translocation process[9, 10, 11] follows the same pattern for TM proteins and proteins with an SP. When a ribosome has translated a part of a mRNA coding for a hydrophobic region, it will be caught by a signal recognition particle (SRP), a protein complex that identifies the nascent hydrophobic region and mediates docking of the ribosome to the translocon. A PCC in the translocon will then open up and enable translocation.

There are so called SRP-independent pathways, where translocation is done after translation is completed. In eukaryotes there is the membrane bound Sec62/63 complex that form a complex together with a translocon, which in turn binds and translocate already translated peptide chains with SPs[12]. The ratcheting mechanism that drives the translocation in this case consists of the ATPase BiP binding to the translocated chain disabling any Brownian motion acting backwards. Similarly, in eubacteria the soluble SecA can bind to SPs and dock to translocons, and aid the translocation process by pushing the polypeptide through the translocon[13]. The SRP-independent pathway is generally targeting less hydrophobic regions, mainly SPs and not TM segments, than the SRP-dependent pathways.

It should be noted that there are other types of translocons in eukaryotes, than the ones of the general secretory pathway. In mitochondria the TOM/GIP complex governs the import over the two membranes to the mitochondria, as well as the TIM complex controlling the re-export from the mitochondria to inter membrane space. Sequences destined for the mitochondria have a mitochondrial transfer peptide, which is cleaved off during translocation. The re-export signals, the inter membrane space targeting peptide, share important features with an SP, even though they are hidden behind a N-terminal import signal in the precursor protein. Similarly, the chloroplasts

and thylacoids of plants and algae have a Toc/Tic complex that imports sequences with chloroplast transit peptides and mechanisms that might import them further into the thylacoid. In bacteria there are also the twin arginin translocation pathway[14] and Type I[15] and III[16] secretion, which are using other means for translocation. In addition, in bacteria there is lipoprotein peptidase, which targets sequences with somewhat different motifs than normal signal peptidase. Throughout this text, I have used the term SP for describing signal peptides cleaved by signal peptidase targeting the general secretory pathway.

1.3 Insertion of transmembrane proteins

Depending on the orientation of the hydrophobic region either the N-terminal or C-terminal part of the polypeptide will be translocated. In the latter case (Figure 1.2a), the further translated peptide will be translocated until the sequence ends, and the PCC will shut and the ribosome will release, or if another TM helix enters the PCC the PCC will be blocked and the further translated peptide chain will be exposed to the cytosol (Figure 1.2b). If once again a TM helix is encountered this will open up the PCC again (Figure 1.2c). SPs and TM segments with their N-terminal part facing the cytosol are called start-transfer sequences, due to their property of opening up the PCC. TM segments with their C-terminal part facing the cytosol, stopping translocation, are called stop-transfer sequences[17]. Previously translocated helices will be shunted out perpendicularly to the PCC into the lipid bi-layer through a slit in the translocon. The helices remain within the vicinity of the translocon until the whole protein is inserted[18].

Recently, it has been shown that the probability of a hydrophobic region to be inserted into the lipid bilayer is proportional to the difference in free energy between the region being inserted in the membrane or it being exposed to the cytosol[19]. This suggests that direct protein-lipid interaction plays an important role in the recognition process of TM helices.

Chapter 2

Machine Learning and Biological Sequences Analysis

The rate at which data is generated in genome sequencing projects is enormous. Just during the four years of my Ph.D. project, both the number of known protein sequences* and known protein structures† have roughly doubled. This avalanche of data makes it impossible to characterize all novel proteins by experimental means, and the scientific community is heavily relying on computer based methods to characterize proteins. This is often done with machine learning, an area of artificial intelligence concerned with the development of methods to make computers "learn" from examples. The area include techniques such as: Bayesian networks[20]; genetic algorithms[21]; support vector machines[22]; artificial neural networks[23](ANNs); and hidden Markov models (see Chapter 3)

We normally divide machine learning techniques into supervised learning, unsupervised learning, and partially supervised learning. The difference is that supervised learning requires that we have annotated our examples with a desired prediction. In contrast, unsupervised learning techniques are able to draw conclusions from unannotated examples.

Some of the text in this chapter is generic for machine learning, but I have tried to narrow the content to issues that are specific to analysis of biological sequences and not described in the machine learning[24, 25] literature.

2.1 Training and Testing sets

Machine learning methods all require representative data to learn from. In biological sequence analysis this means that we need sets of sequences containing a feature that we would like to predict, and sometimes a set of sequences that does *not* contain the feature. The idea is that the system from the given examples should be able to learn what is common for sequences having a feature, and when presented with a new sequence, it should be able to extrapolate a prediction; does the new sequence have the feature or not?

A common concern in machine learning is that the system will "overfit" to the examples given, so that it will recognize all the given examples but not new sequences. To address this problem

*194 thousand protein sequences in Swissprot ver. 48 to be compared to 102 thousand protein sequences in Swissprot ver. 40

†35 thousand protein structures in January 2006 in PDB as compared to 17 thousand in January 2002

one usually divides the example sequences into two sets, a training set and a testing set. If the system would "overfit" to the training set, this would be recognized when testing the system's performance on the test set.

2.2 Homology reduction

It is important that we remove sequences that are to similar from our data sets. In training we do not want to incorporate non-generic patterns from an overrepresented group of proteins, and in testing we worry about too high influence of overrepresented features, or even worse, testing on the same sequences as we trained on.

A commonly used procedure to remove homologs, which we used in Paper II and V, is the remove until done or Hobohm algorithm 2 reduction[26]. It requires a measure of similarity between two sequences (*e.g.* sequence identity as reported from Blast[27]), and a target maximal similarity within the reduced set. The remove until done procedure starts by calculating the similarity of all pairs of sequences in the originating data set. For each sequence the number of sequences to which it is too similar to are counted. Then the sequence with the highest number of too similar sequences is removed. The procedure is then iterated until there are no sequences left that are too similar.

2.3 Weighting sequences

An alternative way to homology reduction is to weighting the sequences. Here we assign a weight to each sequence depending on its similarity to other sequences, instead of removing sequences. Thus, a sequence that is similar to other sequences gets a lower weight than unique sequences.

There are many different weighting schemes, for a good review see Durbin *et al.*'s Biological sequence analysis book[28]. Since I use the Henikoff and Henikoff scheme in Paper IV, I will describe it here. The scheme requires that the sequences we want to weight are aligned. Weights are assigned for each column individually, by giving each different type of residue an equal share of the weight, and then to divide up that weight equally among the sequences sharing the same residue[29]. So if we got 10 sequences with an alanine, 3 with a cysteine and 1 with a glycine, each sequences with an alanine get a weight of $1/30$, the ones with cysteine a get weight of $1/9$ and the sequence with glycine get a weight of $1/3$. The average weight over the alignment is then assigned as a final weight for the sequence. The original article does not give an answer to how inserts/deletes should be handled. Different approaches have been taken including: ignoring positions in the alignment where any sequence has a gap; treating gaps as a twenty first amino acid; or giving zero weight to sequences with gaps at a position, compensating for different sequence length by dividing the weight with the sequence length. In Paper IV I chose the latter.

We can use sequence weights both in training and when predicting with homologs (as in Paper IV, see as well 2.6). Sequence weighting is seldom used in testing, mainly due to the pedagogic problem of explaining exactly what is meant by a weighted number of correct or false predictions.

In some applications one can argue that it is important to have training sets that reflect the true distributions of amino acids, and it might be that sequence weighting would draw attention on less important cases. Furthermore, it makes all training and testing more sensitive to any deficiencies in training data, as they will get a high weight due to their deviation from the rest of the sequences.

2.4 Cross validation

In practice we seldom have enough sequences to spare some of the sequences to assemble pure test sets. We therefore often use a technique called cross validation. The idea is to divide a combined train and test set in to K equally sized subsets. We would then train on $K-1$ of the subsets and test on the Kth set. We then permute the sets and redo the procedure K times, so that all sequences are tested. This way we avoid testing on the same sequences we trained on.

It is essential that the division of the subsets is done in such a manner that the similarity between the sequences in the different subsets is kept at a low level. Otherwise we risk testing on sequences similar to those we trained on. We therefore normally chose a lower inter-set similarity threshold than the intra-set similarity for the cross validation subsets.

2.5 Significance tests

The significance of a difference between two machine learning algorithms can be determined by a paired Student's t-test[25]. In such tests the differences Δ_k in the number of errors made by the algorithms is measured for each of the cross-validation sets separately. The average difference in errors $\bar{\Delta}$ is then calculated. Under the assumption that the binomial distributions of the number of errors made can be approximated with a normal distribution (which is a good approximation for cross-validation sets of more than approximately 30 samples) we can calculate a Z% confidence interval of the difference in error rate between two machine learning algorithms as

$$\Delta = \bar{\Delta} \pm t_{Z,K-1} \sqrt{\frac{1}{K(K-1)} \sum_{k=1}^{K} (\bar{\Delta} - \Delta_k)} \tag{2.1}$$

where $t_{Z,K-1}$ is the distribution function of a t-distribution with $K-1$ degrees of freedom, and K is the number of cross validation sets.

2.6 Predictions supported by homologs

It is beneficial to make predictions not only based on a query sequence in itself, but also include homolog sequences in our predictions, as done in *e.g.* Paper IV. Proteins sequences separated during evolution[32] have often diverged to a state where we can no longer recognize their relation by sequence alone. This is generally a faster process than their divergence in function[33], structure[33, 34], or their features[35, 36]. It follows that the reversion of this is true; if two proteins are similar in sequence, they are likely to share function, structure and features. We therefore search for homologs with homology searching techniques, such as BLAST[27], and integrate the sequences as a part of the predictions. This generally gives better performance to the predictions[37, 38, 39], as we get more information to predict from. We get a better signal to noise ratio by averaging over more samples.

Figure 2.1: An illustration of conserved sequence features among homologs: a cut out from an alignment of transmembrane topology predictions of Drosophila odorant receptors. Predicted extracellular residues have white background, cytosolic residues light gray background, and TM regions dark gray. Even though the sequences have very low sequence similarity, the pattern of topological regions is quite conserved. Note that two sequences have a slightly deviating pattern. Since the deviating pattern is in minority, this is probably due to an erroneous prediction. The sequences where aligned with KALIGN[30] and displayed by KALIGNVU[31]. Their topologies were predicted by Phobius (See Paper II).

Chapter 3

Designing hidden Markov models of sequence features.

3.1 Background

3.1.1 Applications of hidden Markov models

Hidden Markov models is a framework to make discrete classifications of ordered series of observations. The basic techniques were developed during the late 1960's and the early 1970's[40, 41]. From the mid 1970's and on, they where successfully applied to speech recognition, where HMMs are used at different stages in the process of recognizing spoken language[42].

They were first employed within biological sequence analysis[43] by Churchill in 1989, who investigated the heterogeneity of DNA composition[44]. However, the technique did not become widespread until the introduction of profile HMMs[45] in the mid 1990's. A major difference from classical HMM applications, such as speech recognition and motion recognition, is that in biological sequence analysis, classifications are based on observations separated in space (position in a string) not in time.

HMMs can be used in two conceptually different ways within biological sequence analysis. Firstly, as in the case of detecting sequence homology, we ask how well a model could explain a query sequence[45, 46]. This is known as the *evaluation problem*. The most renowned usage of this technique is the PFAM database[47]. Here a set of pre-estimated models corresponding to different protein families is curated. The probability that a query sequence was generated by a (profile) HMM is then calculated for each HMM in the set. If a good enough match to a HMM of a protein domain family is found, the query protein is classed as being a member of that family. This is an analog to the problem of recognizing isolated words in speech recognition[42], where a registered speech signal is compared to a set of pre-estimated HMMs of words in a vocabulary.

Secondly, to determine the most likely state paths through the model that generated a query sequence. This is the *decoding problem*. We can use the path information to predict biochemical properties, *sequence features*, to parts of a sequence, a process that often is referred to as sequence feature prediction. A feature is predicted if a probable path passes through a submodel of the feature. The use of HMMs for sequence feature prediction include transmembrane topology predictors[48, 49, 50, 38], signal peptide predictors[51], coil-coil protein predictors[52], gene predictors[53, 54], secondary structure predictors[55]. The technique is used in sequence alignment programs as well[56, 57, 58], even though this is seldom considered as a feature prediction.

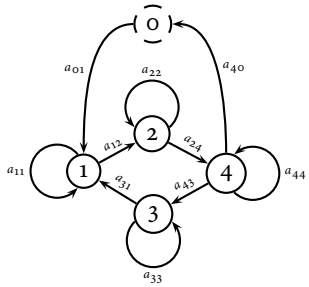


Figure 3.1: A simple (hidden) Markov model.

3.1.2 What is hidden in hidden Markov models?

Consider a system that at any given time, t, is in any of a set of states is $\sigma = \{0, \ldots, M\}$. At regular points in time the system undergoes a change in state according to a set of probabilities associated with the state. If the sequence of random variables, $\mathbf{\Pi} = (\Pi_t)_{t=0}^{T+1}$, representing the sequence of observed states, $\boldsymbol{\pi} = (\pi_t)_{t=0}^{T+1}$, has the Markov property,

$$P(\Pi_t = \pi_t | \Pi_0 = \pi_0, \ldots, \Pi_{t-1} = \pi_{t-1}) = P(\Pi_t = \pi_t | \Pi_{t-1} = \pi_{t-1}), \tag{3.1}$$

then the sequence is a discrete first order Markov chain. The sequence $\boldsymbol{\pi}$ is often referred to as the state path. The transition probabilities between the states can then be described by $\boldsymbol{a} = \{a_{ij}\}$, where $a_{ij} = P(\Pi_{t+1} = j | \Pi_t = i)$. In this text I have used the convention that state 0 is a "start and stop state", *i.e.* the state the system is in before the observations begin and the state it returns to when the observations end.

We can extend the concept of Markov chains to hidden Markov model by considering systems where the current state itself is not observable (is hidden), but the observation instead is a probabilistic function of the state. So the sequence of random variables, $\boldsymbol{X} = (X_t)_{t=1}^{T}$, representing the observations, $\boldsymbol{x} = (x_t)_{t=1}^{T}$, is described by the emission probabilities $\boldsymbol{e} = (e_{ik})$, where $e_{x\pi} = P(X_t = x | \Pi_t = \pi)$ when the state path is given. There is no mapping between the observables and the states in an HMM, and we can therefore not tell which state path that generated a sequence of observables. This is what the word 'hidden' in hidden Markov model describes.

Now for biological sequences feature prediction, we regard $\boldsymbol{x} = (x_t)_{t=1}^{T}$ as an amino acid or DNA sequence, in which we want to predict existence and location of a set of sequence features. We hence see the index t as a spatial position in the sequence, and T as the length of the sequence.

3.1.3 The probability of an observed sequence

For a query sequence $\boldsymbol{x} = (x_t)_{t=1}^{T}$, we can express the probability that such a sequence was generated when taking the path $\boldsymbol{\pi} = (\pi_t)_{t=0}^{T+1}$, where $\pi_t \in \sigma$ and $\pi_0 = \pi_{T+1} = 0$ as

$$P(\boldsymbol{X} = \boldsymbol{x}, \mathbf{\Pi} = \boldsymbol{\pi} | \boldsymbol{a}, \boldsymbol{e}) =$$

$$= P(\Pi_1 = \pi_1 | \Pi_0 = \pi_0) \prod_{t=1}^{T} P(\Pi_{t+1} = \pi_{t+1} | \Pi_t = \pi_t) P(X_t = x_t | \Pi_t = \pi_t) =$$

$$= a_{\pi_0 \pi_1} \prod_{t=1}^{T} a_{\pi_{t+1} \pi_t} e_{\pi_t x_t} \tag{3.2}$$

Since we seldom know which path generated the sequence we have to sum the probabilities of all possible paths.

$$P(\boldsymbol{X} = \boldsymbol{x}|\boldsymbol{a}, \boldsymbol{e}) = \sum_{\boldsymbol{\pi}} P(\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{\Pi} = \boldsymbol{\pi}|\boldsymbol{a}, \boldsymbol{e}) \tag{3.3}$$

However, the number of possible paths grow according to the power of the length of sequence, so the calculation of Equation 3.3 is seldom practically feasible. We instead use a calculation procedure known as the forward algorithm. By defining the forward variables $\alpha_{i,t} \equiv P(\Pi_t = i, X_1 = x_1 \dots X_t = x_t|\boldsymbol{a}, \boldsymbol{e})$, we can formulate the recursion

$$\alpha_{i,t} = \begin{cases} \delta_{i0} & t = 0 \\ e_{ix_t} \sum_{j \in \sigma} \alpha_{j,t-1} a_{ji} & t = 1, \dots, T \\ \delta_{i0} \sum_{j \in \sigma} \alpha_{j,T} a_{j0} & t = T + 1 \end{cases} \tag{3.4}$$

I have here used the Kronecker's δ_{ij} defined as being one when $i = j$, and zero otherwise. Now, by iterating over each position in a sequence $\boldsymbol{x}$, we can derive $P(\boldsymbol{x}|\boldsymbol{a}, \boldsymbol{e}) = \alpha_{0,T+1}$.

3.1.4 Posterior state probability

Similarly to the forward algorithm we can use the backward algorithm, that uses the backward variables, $\beta_{i,t} \equiv P(\Pi_t = i, x_{t+1} \dots x_T|\boldsymbol{a}, \boldsymbol{e})$, to calculate the probability of a sequence, by using the recursion

$$\beta_{i,t} = \begin{cases} \delta_{i0} & t = T + 1 \\ a_{i0} & t = T \\ \sum_{j \in \sigma} a_{ij} e_{jx_{t+1}} \beta_{j,t+1} & t = T - 1, \dots, 1 \\ \delta_{i0} \sum_{j \in \sigma} a_{0j} e_{jx_1} \beta_{j,1} & t = 0 \end{cases} \tag{3.5}$$

By combining the forward and backward variables we can as well calculate the probability that we are in a state, i, at position t in a sequence, given the whole observed sequence. This is the posterior state probability,

$$\gamma_{i,t} \equiv P(\Pi_t = i|\boldsymbol{x}, \boldsymbol{a}, \boldsymbol{e}) = \frac{P(\Pi_t = i, \boldsymbol{x}|\boldsymbol{a}, \boldsymbol{e})}{P(\boldsymbol{x}|\boldsymbol{a}, \boldsymbol{e})} =$$

$$= \frac{P(\pi_t = i, x_1 \dots x_t|\boldsymbol{a}, \boldsymbol{e}) P(\pi_t = i, x_{t+1} \dots x_T|\boldsymbol{a}, \boldsymbol{e})}{P(\boldsymbol{x}|\boldsymbol{a}, \boldsymbol{e})} =$$

$$= \frac{\alpha_{i,t} \beta_{i,t}}{\alpha_{0,T+1}} \tag{3.6}$$

Variations of equation 3.6 is of high importance when calculating the estimated number of times an event occurs, which is of high importance in parameter estimation (see next section).

3.2 Parameter estimation

To estimate the parameters of an HMM we need a set of sequences that are representative for what we want to model. We call this our training set. If we for each of our sequences know the state path that generated the sequence, we can simply assign probabilities according to the relative frequencies of a certain transition or emission event. For instance we can set the probability of

a certain amino acid being emitted by a state to the number of times the amino acid was found emitted from the state divided by the number of times the state was visited in the training set.

If we do not know the state path for the training set, we can instead use the Baum-Welch algorithm[40]. The algorithm uses the forward-backward algorithm to calculate the estimated number of times a state would be reached and the estimated number of times an event occurs in the state, given an HMM and the sequences. We can now calculate new model parameters the same way we did in the case where we knew the state paths, but using estimated frequencies instead of measured frequencies. It can be shown that the likelihood of the data given the new model always will be higher than or equal to the likelihood of the data given the original model. So if we select a random model we are guarantied to approach a (possibly local) maximum as we iteratively improve our model by the algorithm. The Baum-Welch algorithm is a special case of the more general parameter estimation method, expectation-maximization[59], which is not limited to HMMs. Note though that expectation-maximization was (independently) written down at a later point in time.

Sometimes we do have the exact state path of the sequences in our training set, but we still have some knowledge limiting the number of states that could have generated a certain symbol. As an example we could know that a certain amino acid lies in a TM helix. We can then set the forward and backward variables of all the states not representing TM helices to zero at this position, but in all other perspectives follow the Baum-Welch procedure to improve a model. This method to put constraints on the forward-backward procedure is easiest to implement using labels, see Section 3.3, and it is as well possible in the same manner to limit the possible paths in a constrained decoding as described in Section 3.3.2.

We can use conditional maximum likelihood (CML) to improve a model[60]. Here model parameters are changed proportionally to the difference between constrained forward-backward estimations and normal unconstrained forward-backward estimations. The idea is to see to that paths given by the training data should be favored over other paths. Unfortunately the technique is sensitive to annotation errors in the training data, since such regions are likely to have a higher difference between constrained and unconstrained estimations than correctly annotated sequences.

3.3 Decoding

How do we find a state path that, by some criteria, is likely to have generated an observed sequence $\boldsymbol{x}$? A straight forward solution is to select the Viterbi path,

$$\boldsymbol{\pi}^{Viterbi} = \text{argmax}_{\pi} P(\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{\Pi} = \boldsymbol{\pi}), \tag{3.7}$$

i.e. the state path with the highest probability. By defining a recursion over t,[28] the calculations of this path can be done with $O(MT)$ complexity[61].

Since we seldom have enough information about a sequence to recognize the exact state path for a sequence used for training, and as well seldom are interested in the exact path that generated a query sequence, we instead group states together based on a feature they represent. We hence introduce the notion of *labels*[60, 62]. The label l of a state i is given by the mapping $\Lambda(i) = l$ and the set of states that have label l is called $\boldsymbol{\sigma}_l \subset \boldsymbol{\sigma}$, so $i \in \boldsymbol{\sigma}_l \iff \Lambda(i) = l$. Often, each different label represents a sequence feature. In this setting the decoding problem instead turns into predicting a sequence of labels $\boldsymbol{l} = (l_t)_{t=1}^{T}$ that by some criteria is likely to have generated a query sequence $\boldsymbol{x}$.

Finding $\text{argmax}_l P(\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{L} = \boldsymbol{l})$ is a NP-hard problem[63], and we are in need of approximations. Easiest is to calculate the Viterbi labeling $\boldsymbol{l}^{Viterbi} = \left(\Lambda(\pi_t^{Viterbi})\right)_{t=1}^{T}$. However, to profit from the notion of labels we often use the N-best algorithm[62, 64] or its special case the 1-best algorithm. Here we recursively for each position t calculate the probability of different labelings ending up in a certain state. To reduce the computational complexity of the problem, we limit the number of different labelings to the N most probable at each state for each recursive step. The algorithm guaranties us to find a labeling with at least as high probability as the Viterbi labeling.

An interesting observation is that unsupervised Baum-Welch procedure often is more sensitive to path information than most decoders, since it takes all possible paths in account by relying on the forward-backward procedure. We can hence retrain our model with the query sequence as a preparing step before decoding the query sequence with the retrained model. In Paper IV I call this parameter re-estimation decoding. The technique has been used in the transmembrane topology predictor HMMTOP[49].

3.3.1 Decoding with homologs

The features we want to predict are often important for function and hence the existence of the features themselves are more conserved through evolution than the sequence of the feature (see Section 2.6). It is therefore often useful to incorporate signals from homologs[65] into sequence feature predictions. Different approaches for doing so with HMMs have been taken. A common approach is to first build a multiple sequence alignment of the homologs (see Figure 2.1). We may see each amino acid in a column as an independent sample from a common underlying distribution. We can hence replace the emission probability in equation 3.2 with the product of the emission probabilities of the residues in a column of the alignment[66]. A drawback when dealing with sequence feature prediction is that there is no clear way how to deal with the state transitions for the homologs. One approximation that has been taken is to let the path follow the query sequence and ignore the gaps and inserts of the homologs[38]. In such case the probability of an alignment $\boldsymbol{y} = \{\boldsymbol{x}^{(1)} \dots \boldsymbol{x}^{(M)}\}$, with coordinates according to the query sequence, and a path $\boldsymbol{\pi}$, can be expressed as

$$P(\boldsymbol{y}, \boldsymbol{\Pi} = \boldsymbol{\pi} | \boldsymbol{a}, \boldsymbol{e}) = P(\boldsymbol{X}^{(1)} = \boldsymbol{x}^{(1)}, \dots, \boldsymbol{X}^{(M)} = \boldsymbol{x}^{(M)}, \boldsymbol{\Pi} = \boldsymbol{\pi} | \boldsymbol{a}, \boldsymbol{e}) = \prod_{m=1}^{M} P(\boldsymbol{X}^{(m)} = \boldsymbol{x}^{(m)}, \boldsymbol{\Pi} = \boldsymbol{\pi} | \boldsymbol{a}, \boldsymbol{e}) =$$

$$= \prod_{m=1}^{M} \left(P(\Pi_1 = \pi_1 | \Pi_0 = \pi_0) \prod_{t=1}^{T} P(\Pi_{t+1} = \pi_{t+1} | \Pi_t = \pi_t) P(X_t^{(m)} = x_t^{(m)} | \Pi_t = \pi_t) \right) =$$

$$= (a_{\pi_0 \pi_1})^M \prod_{t=1}^{T} \left((a_{\pi_{t+1} \pi_t})^M \prod_{m=1}^{M} e_{\pi_t x_t^{(m)}} \right) \tag{3.8}$$

It is in my opinion harder to probabilistically motivate approaches where single sequence emission probability is replaced by the sum of emission probabilities of the amino acids in a column of the multiple sequence alignment[67, 68].

We can also use parameter re-estimation decoding (see previous section) to include homologs. Then we simply include all the homologs in the Baum-Welch estimation step, and finally decode the query sequence with our re-estimated model.

In Paper IV we describe an "optimal accuracy decoder" for homologue sequences.

3.3.2 Constrained decoding

Sometimes we do have prior knowledge about the prediction. We might have located a feature to a certain position in the sequence by an experiment or other type of observation. This can be seen as a labeling of the position.

So could we incorporate such constraints into our decoding procedures? Yes, for all the decoding algorithms I have mentioned here, this can be done by assigning zero probability to the paths passing through states with other labels than the one given for the position.

3.4 Architecture

Usually some of the transition probabilities are set to zero in advance to avoid "illegal" transitions. The non-zero transition probabilities define the underlying graph of the model. This graph structure restricts the possible paths that could be taken through the model, and we will refer to it as the architecture of the model.

One of the strong arguments for using HMMs as models of sequence features, is the possibility to model the length of a feature separate from its amino acid distribution (as opposed to using *e.g.* neural networks, or support vector machines). By using the definition of conditional probability we can divide the likelihood of a certain path and sequence into two parts, as

$$P(\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{\Pi} = \boldsymbol{\pi}) = P(\boldsymbol{\Pi} = \boldsymbol{\pi})P(\boldsymbol{X} = \boldsymbol{x}|\boldsymbol{\Pi} = \boldsymbol{\pi}). \tag{3.9}$$

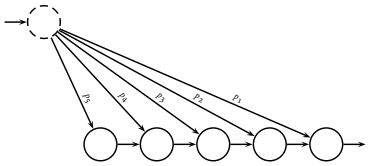
So we can see the modeling of a path, controlled by the probability $P(\boldsymbol{\Pi} = \boldsymbol{\pi})$, as a separate problem from that of modeling the symbols a state path emits, controlled by the probability $P(\boldsymbol{X} = \boldsymbol{x}|\boldsymbol{\Pi} = \boldsymbol{\pi})$. This notion is particularly useful when modeling sequence features that can be approximated as having uniform amino acid distributions, *e.g.* TM helix cores, or repetitive amino acid distribution, *e.g.* coil-coil structures or amphipatic helices, where the modeling of path can be reduced to modeling the length of a feature. We can do this in two different ways. We can use an explicit length model for each state in what is known as a Generalized HMMs[42, 54, 49]. Or we can set the same emission probabilities for a set of states in a normal HMM and connect them in a way that they implicitly model a length distribution. There are three reasons why I here will focus on the later alternative. Firstly there are only straight-forward solutions how to implement 1-best, CML and other training and decoding techniques for this kind of HMM. Secondly the architectural patterns for normal HMMs are not well described elsewhere. Thirdly this is the chosen alternative in my publications.

3.4.1 Modeling sequence feature length distributions

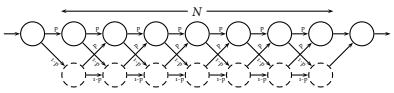
Here a couple of patterns that model length distributions are listed. They are grouped according to if they are limited in length or if they could model infinitely long sequence features. It should be noted that some of the patterns require that information from more than one path is taken in account and hence are not suitable for Viterbi decoders.

Sequence features with limited length

Due to their nature, some sequence features are limited in length. An example is TM segments. We know that they must be long enough to span the phospholipid bilayer, which corresponds to a lower limit of about 15 amino acids. They are seldom longer than about 35 amino acids.



(a) Cascade



(b) Binomial

Figure 3.2: Architectural pattern that model a distributions limited in length. (a) An example that span between one to five amino acids is modeled. (b) Pattern modeling binomial length distributions. Here circles drawn with dashed lines represent non-emitting states.

A straight forward way to model this flexibility in length is to use a pattern depicted in Figure 3.2(a), where an initial state is connected to all of the states in a chain of forward connected states. We can then estimate the transition probabilities according to the observed feature lengths in the training set. An alternative approach that was used in paper II, is to fit a discrete probability distribution to the training data and calculate the transition probabilities. By doing so we can normally lower the number of estimated parameters of the model, since we often find a probability distribution that is controlled by lower number of parameters than the number of transitions probabilities that should be estimated.

A binomial distribution can be generated by using the pattern in Figure 3.2(b). Two parallel linear chains of N states, one chain emitting amino acids and one silent, are connected so that it is possible to pass from each state to the next emitting or silent state. All transition probabilities to an emitting state are set to p and all transition probabilities to a silent state are set to $1 - p$. The length of a sequence can range from 0 to N amino acids and there are $\binom{N}{l}$ ways to produce a sequence with length l. Hence the probability to generate a sequence of length l follows a binomial distribution:

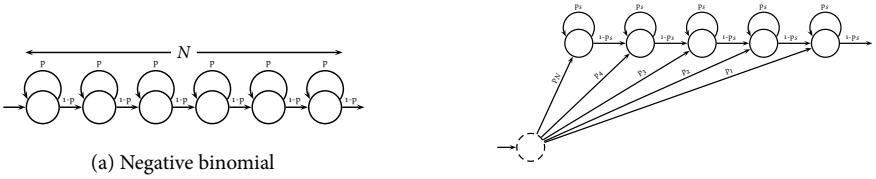
$$P(l) = \binom{N}{l} p^l (1-p)^{N-l} \tag{3.10}$$

Sequence features that are not limited in length

For sequence features with unlimited maximum length, the layout is shown in Figure 3.3(a). It consists of a linear chain of N emitting states that all have self-transitions. Again, transition probabilities are set equal throughout the structure so that there is a probability p of staying in the current state and a probability $1 - p$ of continuing to the next state. The emitted sequence can be no shorter than N amino acids but there is no upper limit. There are $\binom{l-1}{N-1}$ paths through the model with length l. If the individual amino acid emission probabilities are disregarded, the probability to generate a sequence of length l follows a negative binomial distribution[28].

$$P(l) = \binom{l-1}{N-1} p^N (1-p)^{l-N} \tag{3.11}$$

An interesting pattern that has been used in gene prediction[69] is the pattern illustrated in Figure 3.3(b). The distribution belongs to the acyclic phase type distributions[70] and seems to able to take very different shape, and mimic quite varying types of distributions. The probability of



(a) Negative binomial

(b) Phase type

Figure 3.3: Architectural patterns that can model a distributions that is not limited in length. (a) Pattern for negative binomial distribution (b) Phase type distribution.

generating a sequence of length l is:

$$P(l) = \sum_{i=1}^{N} \binom{l-1}{i-1} p_s^i (1-p_s)^{l-i} p_i \tag{3.12}$$

3.5 Implementation

Currently available software packages for modeling biological sequence features include:

ANHMM During the project I have been heavily reliant on Anders Krogh's proprietary HMM package, ANHMM. Even though it is most flexible and it contains many good features, it has the disadvantage of not being publicly available.

HomologHMM As a part of the study described in Paper IV I implemented some decoders: N-best; Viterbi; Max PLP; and optimal accuracy decoding, all the decoders have the option to include homologs in the predictions. The software is available under GPL*.

GHMM This software from Max Planck Institute for Molecular Genetics in Berlin contains the most essential algorithms needed†. It includes a graphical user interface for designing HMM.

modhmm This package, written by Håkan Viklund at Stockholm Bioinformatics Center, contains all the basic HMM algorithms as well as functionality to include homologs and the possibility of using different alphabets in parallel‡.

*`http://phobius.cgb.ki.se/data.html`
†`http://ghmm.org/`
‡`http://www.sbc.su.se/modhmm/`

Chapter 4

Transmembrane Topology

Transmembrane proteins make up about a fifth* of all protein sequences known, yet less than one percent† of all the known structures. This discrepancy is due to the fact that TM proteins are hard to over-express and crystallize, and therefore difficult to examine with X-ray diffraction or NMR. In fact, when the first larger structure of a membrane protein was determined, the photosynthetic reaction center, it rendered a Nobel Prize‡. It is however much easier to determine the TM topology. That is localizing all TM segments as well as determining which sub-cellular compartment to which the loops between the TM segments are exposed.

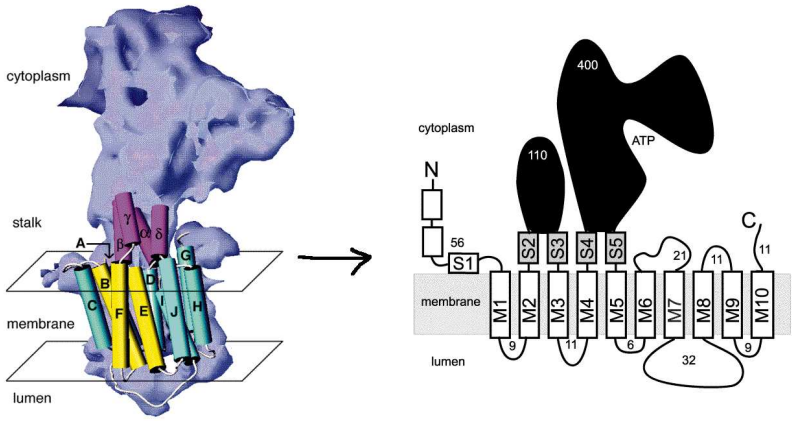


Figure 4.1: Transmembrane topology is a projection or conventionalization of the protein's structure, telling where different parts of the protein are located, *i.e.* in the membrane; or translocated or not translocated across the membrane. The figure is adopted from Zhang *et al.*'s study of the calcium pump from sarcoplasmic reticulum[71].

*19.9% of all proteins in Swissprot release 41.0 classed as TM proteins by TMHMM.
†151 TM entries of the 21819 entries with chain size over 50 in PDB.
‡`http://nobelprize.org/chemistry/laureates/1988/press.html`

4.1 Prediction by experimental means

4.1.1 Reporter fusions

Reporter fusions are frequently used to elucidate topological information. The experimenter make use of a reporter protein with properties, *e.g.* enzymatic activity or fluorescence, that depend on its extracellular or intracellular location. The reporter protein is attached to a hydrophilic domain of a membrane protein. Since the location of the hydrophilic region dictates the location of the reporter, we can extract topogenic information by studying the reporter gene. By repeating the procedure for different positions in the membrane protein, the topology can be derived.

There are two types of fusion studies. There are C-terminal deletion fusions, where the C-terminal part of the membrane protein is replaced by the reporter domain, and there are insertional or sandwich fusion[72], where the reporter gene is fused into the middle of the membrane protein, leaving the C-terminus intact.

A commonly used reporter protein in topological studies in bacteria is alkaline phosphatase, encoded by the *E. coli phoA* gene, which only can fold correctly and form enzymatic active dimers in the periplasm[73]. We can hence determine the location of the reporter gene by measuring the enzymatic activity of the fused protein. Another reporter protein is the periplasmic active β-lactamase, encoded by the *bla* gene[74]. When active, the enzyme gives ampicillin-resistance to its host. Since it is only active in the periplasm, it is used as an alternative to PhoA.

A nice complement to PhoA or β-lactamase fusions are fusions of the E. coli enzyme β-galactosidase, LacZ. Inversely to PhoA, LacZ exhibits enzymatic activity only in the cytoplasm [75]. Later studies often use the cytoplasmic active green fluorescent protein (GFP)[76] as a reporter. The reporter is fluorescent when properly folded, which only happens in the cytosol.

PhoA or β-lactamase and LacZ or GFP are, due to their activity in complementary location, often used in the same studies. Hereby mutual exclusive results can be obtained, and the experimenter does not have to rely on negative results.

However, conflicting results, *i.e.* high or low activity at the same fusion site by reporters active on complementary sides of the membrane, have been reported in a number of membrane topology studies. This indicates that the assumption that the fusions do not affect the topology of the examined protein might not always be true, and hence the accuracy of reporter fusions has been questioned. Furthermore, when examining previously fusion-assessed topologies, that later have been chrystalographed, there is not any significant difference in accuracy between the fusion studies and TM Topology predictors[37].

4.1.2 Site Tagging

As an alternative to reporter fusions, membrane protein may be fused with a site. Typically we can add a N-glycosylation site to a sequence, *i.e.* the amino acids N-X-S/T, where X could be any amino acid except for proline[77]. N-glycosolation is an ER lumenal process, so if the protein with a tag is larger than the wild type, we can conclude that the site was glycosylated and hence a part of a translocated loop. Normally the difference between wild type and tagged protein is measured on an SDS-PAGE gel.

4.1.3 Antibodies

We can design antibodies directed against a short stretch of a hydrophilic region of a TM protein. If we expose an organelle where the TM protein is located to the antibodies we can see if they attach or not to the proteins in the membrane, and from this elucidate the location of the region.

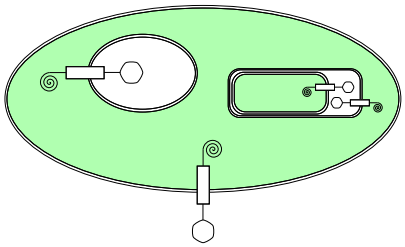


Figure 4.2: An illustration of why the commonly used ”inside” and ”outside” notation of loops is unsuitable when dealing with eukaryote cells. The rounded domains of the TM proteins in the figure are all translocated and hence located on the ”outside”. So the inside of the ER, Golgi or peroxisome will all be classified as ”outside”. I prefer the terms ”translocated” and ”not-translocated loops” or maybe ”cytosolic” and ”non-cytosolic” loops.

Both monoclonal and polyclonal approaches have been made. In general the problem with the approach is to assure that the antibodies are specific for the region.

4.1.4 Mass spectrometry

Recently a method deducing topological information by using ”shot-gun” proteomics was presented[78]. Here the TM proteins are digested by a protease while they are still embedded in their membrane. By analyzing the resulting peptide mixture with tandem mass spectrometry, the membrane proteins as well as the topological localization of the parts of the proteins that were exposed to the digestion, may be identified.

In particular, proteinase K turns out to be useful in such studies. At neutral pH this protease is extremely robust and often results in the complete digestion of proteins into dipeptides. However, high pH attenuates proteinase K's activity to levels at which 6- to 8-residue peptides are formed. The procedure involves first exposing a cell or an organelle to proteinase K at the neutral pH level, leading to digestion of all external domains of its membrane proteins. By then exposing the organelle/cell to high pH the membrane disrupts, and a subsequent treatment with proteinase K digests the interior domains into peptides suitable for identification by tandem mass spectrometry analysis. This procedure enables us to identify interior loops, since exterior loops were previously removed and membrane helices are still buried in the membrane.

The performance of the method has not been evaluated, but it is clear that the approach represents an attractive way to elucidate topological information in large scale.

4.2 *In Silico* Topology Prediction

4.2.1 Location terminology

TM topology predictors normally assume that all cellular membranes can be treated equally, regardless of which organelle or cell type they surround. The location of a loop can be classed as being on the originating side – the side of a membrane from which the TM protein was inserted (normally the cytoplasm), or the translocated side – the opposite side of the membrane. The commonly used ”inside” and ”outside” notation is confusing and should be avoided. For instance, the

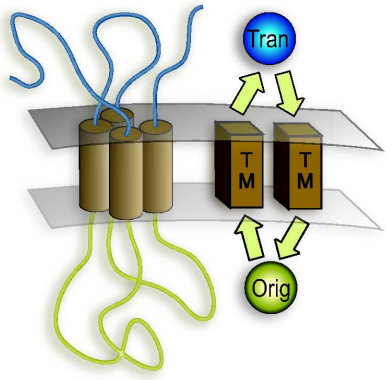


Figure 4.3: Transmembrane protein topology predictions are based on the amino acid composition of the regions of a transmembrane protein. A prediction corresponds to the path through the model with the highest score (probability) of having generated the query sequence.

lumenal inside of an organelle is the translocated side, which is normally considered the "outside" (See Figure 4.2).

4.2.2 Prediction principles

Early TM helix prediction methods were based on theoretically or experimentally determined hydrophobicity indices of hydrophobic properties for each amino acid. For the examined protein a hydrophobicity plot was calculated by summing the hydropathy indices over a window of a fixed length. A heuristically determined cut-off value was then used to indicate possible TM segments[79, 80]. An important improvement to this strategy came from the observation that positively charged amino acids (arginine and lysine) are overrepresented near the TM helices on the originating side loops of TM proteins (The positive inside rule)[81]. This gives an indication about the orientation of the helices and led to the development of the first automated full TM topology prediction methods *e.g.* TOPPRED[82]. This method first scans a sequence for certain and putative TM segments and then selects the putative segments that maximize the difference in charged amino acids in loops, summed over each side separately. Instead of only using a hydropathy index, some methods use a combination of this and indices for amino acids known to be frequent near the end of membrane helix ends, e.g. SOSUI[83]. Other methods are letting a sequence profile, e.g. DAS[84], or an Artificial Neural Network, e.g. PHDHTM[85], detect potential TM segments.

Integrated predictors

A more integrated approach could be taken to the problem. Instead of first scanning the sequence for TM segments and sort out the topology as a second step, the search for TM segments can be integrated with the evaluation of possible topologies in one step. The amino acid distribution of the investigated sequence is compared to pre-calculated expected amino acid distributions in each type of topologically distinct region (TM helices, originating side loops, and translocated side loops) of a TM protein (see Figure 4.3). Given correlation measurements between the amino

acid distributions of the examined protein and the expected amino acid distributions in different topological regions, the most likely topology can be predicted. A nice feature of this approach is the ability to model all parts of the protein so that all topogenic signals are properly weighted, which is preferable to giving priority to the hydrophobic signal. This was first done by a dynamic programming algorithm in the method MEMSAT[86]. The parameters of MEMSAT where estimated by Expectation Maximization[59] so one could say that it is highly related to hidden Markov models (HMMs, see Chapter 3). Pure HMM approaches to the problem have followed. Some popular HMM-based predictors are TMHMM [48, 50] and HMMTOP[49, 87], and the recently published PRODIV-HMM[38] and TMMOD[88].

β-barrel TM proteins seem to be hard to predict with the classical TM prediction methods since their TM segments generally are shorter and with a different amino acid composition than α-helical TM segments. There are dedicated predictors available for these kind of proteins; B2TMR-HMM[67] and BBF[89].

Predictions supported by homologs

As mentioned in Section 2.6, a common way to improve the performance of a predictor is to not only look at the examined sequence, but instead to first find homologs using homolog retrieval tools like BLAST[27] and then predicting the topology of the whole alignment. The idea is that topology should be conserved in a family, and by looking at the entire family there is less chance of mispredicting single atypical members. Examples of such methods are TMAP[90], PHDHTM[85] and PRODIV-HMM[38].

Consensus prediction

A good practice when predicting TM topology is to compare the results from different predictors. This is most easily done by running a Meta server, i.e. a server that runs a number of different prediction programs. The results may be delivered in the form of e-mails from the different underlying predictors, like for META-PP[91], or they may be displayed side by side graphically as by SFINX[92]. The results from multiple methods may also be combined by a consensus predictor[93]. Such a predictor only contains a weight for each method and heuristics for combining the results. An example is CONPRED II[94].

4.2.3 Benchmarking

To be able to compare different TM prediction methods one normally assembles a test set with known topologies and examines how well each prediction agrees with it. For such benchmark experiments it is important to collect adequate test sets since a biased test set easily could favor some of the predictors. During the years a couple of such test sets has been assembled, *e.g.* the MPtopo[95], Möller[96], and TMPDB[97] sets.

Most modern TM topology predictors are based on machine learning algorithms. When comparing the performance of different methods it is therefore essential not to include the training data of any of the compared methods in the test. This is however problematic, because so few TM topologies are known that, when removing all the training set proteins, only a few proteins with dubious topologies are left. In addition, benchmarking sets generally seem to be easier to predict than genomic data (See paper I).

It is therefore maybe not surprising that different benchmark studies come to different conclusions about which TM topology prediction method is better. Möller and colleges rate TMHMM as the best method[98], while Ikeda and colleagues rate HMMTOP best[99]. Chen and colleagues use

Table 4.1: The fraction of correctly predicted topologies as reported in different benchmarking studies. Values marked with asterisk (*) were measured on older versions of the method.

Method	Möller[98]	Ikeda[99]	Chen high-resolution[37]	Chen low-resolution[37]
TMHMM 2.0	47%	48.4%	45%*	85%*
HMMTOP 2.1	45%*	54.1%	61%	79%
PHDhtm - single	18%	-	54%	68%
PHDhtm - multiple	-	-	66%	67%
Memsat 1.5	41%	45.1%	-	-
Number of proteins	188	122	36	165

two different test sets, one containing topologies with known 3D-structures (high-resolution) and one containing topologies without known 3D structure (low-resolution)[37]. PHDhtm scored best on the high resolution set while TMHMM scored best against the low-resolution set. Table 4.1 lists a few different assessments of accuracy by a number of prediction methods. As the reader can see, the reported performance differs substantially. Hence, reported performance figures for TM prediction methods should be interpreted with caution, both in terms of absolute and relative accuracy.

4.2.4 Constrained Prediction

Lately an interesting type of bioinformatics and experimental hybrid technique has been used to determine the topology of large sets of *E. coli* TM proteins[100, 101]. By fusing a set of inner membrane proteins with with LacZ and GFP their C-terminus can be located as cytoplasmic or periplasmic. This piece of topogenic signal was used an input to a constrained prediction by TMHMM[102]. Full topological models of 601 *E. coli* TM proteins were proposed. By imposing the same topologies for homologs in other bacterias the findings were extended to models of 51208 TM proteins[103].

Chapter 5

Signal peptides

When trying to determine the function of a protein, an important question to answer is where in the cell is it located. The location of a protein governs what other types of proteins or other molecules it will be able to interact with. A first step in this process is to determine if it has an SP or not, since that will tell us if it is a cytosolic protein or not. In addition it is often valuable to know where the mature protein starts, so there is an interest in localizing the cleavage site of an SP.

About 16% of the proteins in the human proteome have an SP (See Paper II).

5.1 Characteristics of a signal peptide

Similar to the TM segment, one of the strongest indications of an SP is a hydrophobic α-helical region. It is called the h-region of the SP. However, the hydrophobic region is generally shorter for an SP (approximately 7-15 residues) than for a TM helix (See Figure 5.1). The h-region is near the N-terminal of the protein but it is preceded by a slightly positively charged n-region with high variability in length (approximately 1-12 amino acids). Between the h-region and the cleavage site a somewhat polar and uncharged 3-8 amino acid long c-region is situated. Another clear motif of the SP is the presence of small, neutral residues at the -3 and -1 relative to the cleavage site[104, 105]. We often see helix-breaking amino acids, *i.e.* proline, serine, or glycine, in between the h- and c-region[104, 105].

5.1.1 Kingdom specific variations

SPs are generally longer in Gram-positive bacteria than in other bacteria, and SPs of eukaryotes are on average shorter than SPs of bacteria[106]. The difference in length can most prominently be found in the h-region[51]. There is as well a difference in the preference in amino acids, *e.g.* the h-regions of eukaryotic SPs have a higher content of leucine than h-regions from bacteria[107].

However, in my work I found no difference in prediction performance when training different SP models for different kingdoms of life (see Paper II).

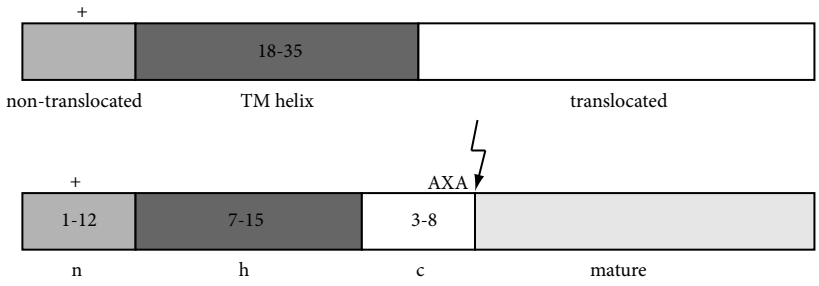


Figure 5.1: A comparison of a TM segment with N-terminus in the cytosol (above) and an SP (below). Both the TM helix and the h-region of the SP are preceded by a slightly positively charged region and succeeded by a hydrophilic region. However the h-region is generally shorter than a TM segment, and there are small and neutral residues at -3 and -1 position relative to the cleavage site (often alanine).

5.2 Predicting signal peptides by experimental means

Experimental efforts to identify SPs are often a reversion of the location problem. If we have seen that the cell is exporting a protein from the cytosol, it is probably a protein with SP. The techniques include:

- Fusing the 5'-end of a gene coding to a gene with a reporter protein. For example one can use a known essential secreted protein in yeast[108]. If a transfected colony survives the gene codes for a protein with signal peptide. Alternatively, one can use fluorescent labels[109].
- We can isolate organelles containing exported proteins and then separate its proteins on *e.g.* 2D-gels. A subsequent Edman degradation of the N-terminal of an isolated protein reveals if the amino acid sequence is found downstream of the translation start site. We are then likely to have found a protein with its signal peptide cleaved off.

5.3 *In Silico* prediction of signal peptides

Most available SP prediction methods use weight matrices [110, 111], Artificial Neural Networks (*e.g.* SignalP-NN[106, 112]), HMMs (*e.g.* SignalP-HMM[51], Sighmm[113], and LipoP[114]), or Support Vector Machines[115, 116]. The perhaps most popular method, SignalP-NN, has trained one ANN for detection of cleavage site motifs (the C-score), and one ANN to detect the existence of an SP (The S-score). The prediction scores are calculated for each position in the sequence sequentially. Finally cleavage sites are predicted by regarding a Y-score, a geometrical mean between the C-score of the position and the difference in S-score before and after the position. Existence of an SP is predicted by the value of the average S-score from the start of the sequence till the maximal Y-score[106]. An additional criteria is introduced in SignalP-NN 3.0 where the average S-score is replaced by a D-score, that is defined as the average of the average S-score and the maximal Y-score[112]. The HMMs have, thanks to their ability to model length distributions, the advantage of easily modeling all regions of an SP in a single model. Hence the prediction of cleavage site is predicted at the same time as the existence of an SP, and we will get one single answer, as to whether an SP is present or not[51](See Paper II).

5.3.1 Benchmarks

A number of independent evaluation studies of SP prediction have been published[117, 118, 119]. The methods of collecting test sets in the studies are either to trust the annotation given in the SWISSPROT database as Menne and colleagues[117] and Klee & Ellis[119] do, or else like Zhang & Henzel[118] who use experimentally determined SPs. The evaluations quite uniformly show that different versions of SignalP is the most accurate method, both in predicting existence and cleavage site of SPs. Only Menne *et al.* include membrane proteins in their negative test data. Klee & Ellis benchmark comprises the method described in Paper II, Phobius, and rates it as less accurate than both SignalP 2.0 and 3.0, but more accurate than PrediSi[111]. This stands in contrast to Paper II where we argue that the accuracy of predictions of SPs are improved compared to SignalP since we are able to discriminate SPs from TM segments. So the quote "*...N-terminal transmembrane domains were intentionally not included in the test set...*" from Klee & Ellis publication might explain the contradiction. Not including TM proteins in the test set does, in my opinion, bias the test.

5.4 Discriminating transmembrane helices and signal peptides

TM helices and SPs tend to confuse predictors. Because they have similar composition TM topology predictors often classify SPs as TM helices and SP predictors often classify N-terminal TM helices as SPs.

Different strategies for obtaining better discrimination have been tested. Lao and colleagues investigated strategies for topology prediction of multi-spanning TM proteins with SP[120, 121]. They propose three different strategies and measure the difference in accuracy between them: remove the part of the sequence in an SP before TM topology prediction; running TM topology prediction first and remove any predicted TM segments overlapping with the SP from the final prediction; and to ignore the problem and not to include information of SPs. They found a significant performance increase between both the strategies that removed SPs compared to not removing the SPs, but no significant difference between doing it before and after prediction. In one of the studies the existence of SP was given as a prerequisite[120] while in the other study they use a SP predictor[121]. But in both cases the examined proteins contained SPs. What if they do not? In Paper II we show that this kind of combined prediction have drastically lower performance on proteins not containing SPs, as SP predictors have to many false positive predictions on such data.

Is this a common problem? Yes, in Paper II we report that 5% of the proteins in human and 10% of the proteins in *E. coli* have overlapping predictions from SignalP and TMHMM. So how do we overcome these discrimination problems? Two of the previously mentioned SP prediction methods SignalP-HMM[51] and LipoP[114] contain models of signal anchors, *i.e.* N-terminal TM segments with their N-terminus located in the cytosol, to be able to discriminate such sequences. This is of course useful, but to quote Henrik Nielsens Ph.D. thesis "*SignalP-HMM does a fairly good job in discriminating between SPs and signal anchors, but this solves only a part of the problem, since signal anchors only constitute a minor fraction of TM proteins. When scanning genome data it would be desirable to distinguish SPs not only from signal anchors, but also from other types of TM helices.*"[122]. In the same section Dr. Nielsen also mentions the possibility to integrate the models of SignalP-HMM and TMHMM, which in essence is what Paper II does.

Chapter 6

Present investigation

In this chapter I will sketch background and summarize the results of the papers that are included in this thesis. The aims of this thesis have been to characterize TM topology and SP predictors, and develop a methodology to obtain better predictions. In particular, the methods developed have been shown to obtain better discrimination between the two types of predictions.

Papers II, IV and V present a new methodology for TM and SP predictions, while Paper I discusses the accuracy of TM predictors. Paper III concerns the TM topology prediction of a specific protein.

6.1 Paper I – Reliability of transmembrane predictions in whole-genome data

As pointed out in section 4.2.3 it is challenging to assess the accuracy of TM topology predictors due to test data-related issues. Benchmarks suffer from their test data i) overlapping with training data of the tested methods to various degree, and ii) being biased towards easily predicted topologies. In this study we try to give an indication of how well TM predictors would perform when applied to genomic data, by comparing predictions from different methods. We came to the conclusion that accuracy is far lower when examining data in general as compared to a commonly used test set, a conclusion that was later confirmed in an independent study[102].

6.2 Paper II – A combined transmembrane topology and signal peptide prediction method

As pointed out previously in this thesis, a common problem for TM predictors is that they have a tendency to falsely classify SPs as TM segments and SP predictors often mis-classify N-terminal TM helices as SPs. This is a natural consequence of the fact that a signal peptide, as well as a TM segment, may contain a hydrophobic α-helical region.

An illustrative example is given in this paper where we perform TM topology TMHMM and SP predictions SignalP on the whole proteomes of human and *E. coli*. We find that there is an overlap between the predictors of 5% of the human and 10% of the *E. coli* proteins. The proteins that are predicted to have both an N-terminal TM segment as well as SPs by the methods can not

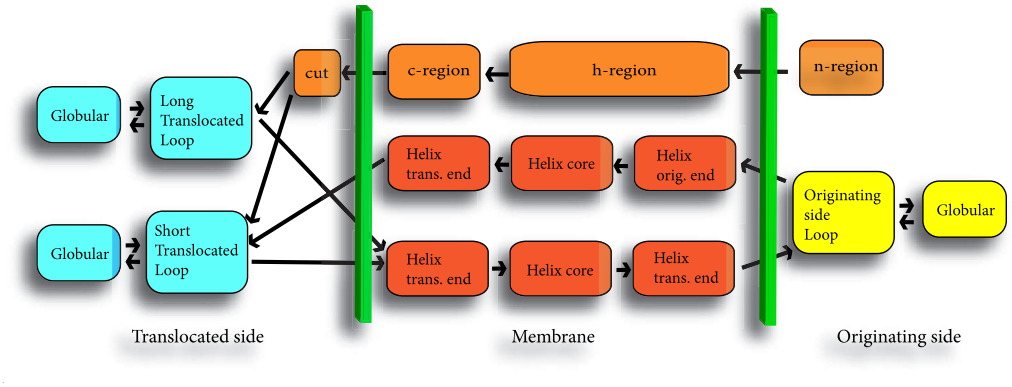


Figure 6.1: The Phobius model.

be discriminated as both types of predictions, as previously pointed out, are known to have high false positive rates.

To remedy the situation we designed an HMM, PHOBIUS, which is a combined TM topology and SP predictor. It can be seen essentially as an assembly of TMHMM[48] and SIGNALP-HMM[51].

The resulting method reduces false classifications of SPs from 26.1% to 3.9% and false classifications of TM helices were reduced from 19.0% to 7.7%, compared to TMHMM and SIGNALP-HMM respectively.

The method is publicly available through a web based prediction server*.

6.3 Paper III – Transmembrane topology of presenilin by reconciling experimental and computational approaches

Presenilin is a part of the γ-secretase complex, a protease active against TM regions, involved in the cleavage of the amyloid β precursor protein(APP). Mutations in presenilin are causing APP to be cleaved at the wrong position, and this is believed to be one of the mechanisms involved in the development of Alzheimer's disease.

Throughout the years a large number of studies have been published in which the topology of presenilin is examined, all arriving at contradictory conclusions. Our approach to the problem has been to reconcile the topology by studying i) TM topology predictors, ii) the experimental results of the previous studies, and iii) predictions constrained by the functional sites in presenilin.

We derive a nine TM helix topology with the N-terminus in the cytosol, a topology which was later confirmed by other studies[123, 124].

*`http://phobius.cgb.ki.se/`

6.4 Paper IV – An HMM posterior decoder for sequence feature prediction that includes homology information

As mentioned in section 2.6, performance of sequence feature prediction can be increased by including homologs in the predictions. There are different ways this could be done with HMMs as pointed out in section 3.3.1. Our approach in this study was to apply the HMM to each sequence individually before weighting the results together and making a final prediction (See Figure 6.2).

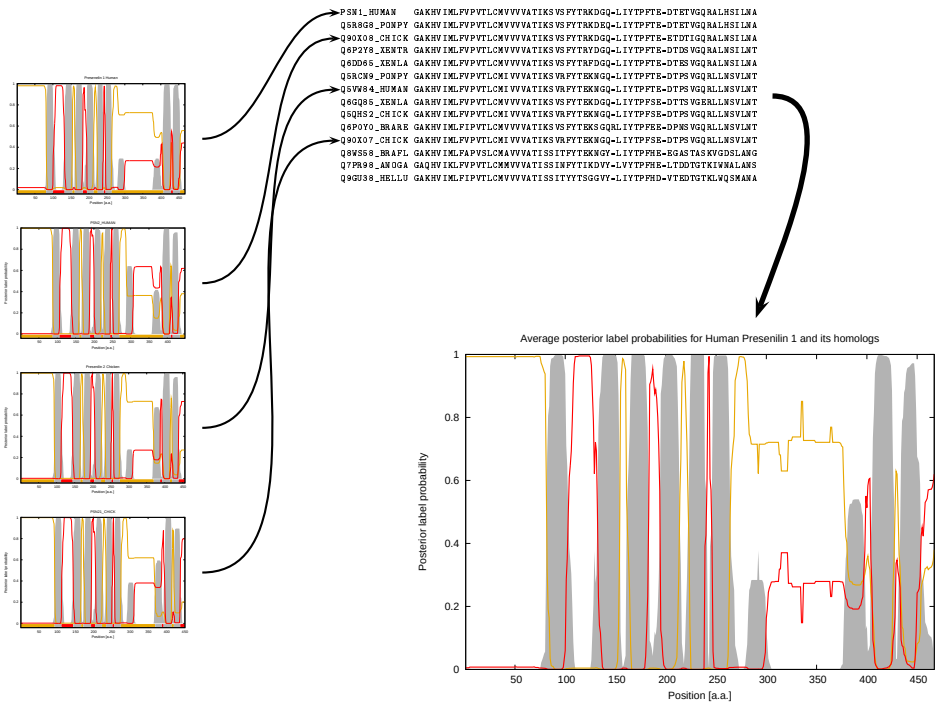


Figure 6.2: The decoder principles. First the posterior label probabilities were calculated for each homolog independently. The probabilities were then superimposed by the guidance of a multiple sequence alignment. An average posterior probability was calculated for each position which was then used for assigning one label to each amino acid in the query sequence.

The method was aimed to be generic for sequence feature prediction, even though it was only tested for Phobius. After the acceptance of our paper, a part of the algorithm was published independently, but evaluated in the context of secondary structure prediction[39], where increased performance was claimed.

We showed that we could obtain significantly better accuracy in TM topology predictions and increased performance in SP predictions compared to other decoding techniques – both including and not including homologs. The method is publicly available through a web-based prediction server*.

*`http://phobius.cgb.ki.se/poly.html`

6.5 Paper V – A general model of G protein-coupled receptor sequences and its application to detect remote homologs

G protein-coupled receptors (GPCRs) is the largest eukaryotic protein superfamily[†] and it is as well very divergent, including both odorant and hormone receptors. However all GPCRs share the same TM topology; seven TM segments and a extracellular N-terminal.

Already in 1994 Baldi & Chauvin published a study on how to model GPCRs with an HMM[125]. Here the authors have trained a profile HMM on sequences from the whole superfamily. As they were unable to build an initial multiple alignment of the sequences they relied upon Viterbi-training on the full length sequences. The HMM is aimed at discriminating GPCRs from non-GPCRs.

Here we examined the length distributions of the different loops and TM segments of our training data, and by combining architectural patterns, as described in Section 3.4, we constructed an HMM suitable of discriminating GPCRs from non-GPCRs. We named the method GPCRHMM. It was our intention to be able to detect novel GPCR subfamilies with low sequence similarity to other GPCRs, as sequences similar to other GPCRs are easily are detected with conventional sequence homology detection. The method is publicly available through a web-based prediction server[*].

We evaluated all the sequences in five proteomes with the HMM – worm, fly, fish, mouse and human. Even though we found previously non-GPCR annotated sequence families, most notably in worm, perhaps our greatest finding was that we were unable to detect a large family of odorant receptors in fly. In the article we speculate that odorant receptors in fly are not GPCRs.

After the acceptance of our paper, Vosshall and co-workers published an extensive analysis of the drosophila odorant receptors[126]. Their analysis indicated that the family does not have a GPCR topology but rather an inverted GPCR topology, *i.e.* including seven TM segments but an extracellular C-terminus. The authors argue that odorant receptors in Drosophila are not GPCRs. Hence the arthropods odorant reception is not functionally homologous to mammalian[†] odorant reception.

[†] 3.4% of the sequences in the human proteome, and 5.8% of mouse are GPCRs

[*] `http://gpcrhmm.cgb.ki.se/`

[†] `http://nobelprize.org/medicine/laureates/2004/press.html`

Chapter 7

Remarks and Future Perspectives

This chapter contains some reflections on the work performed for this thesis. Some of these remarks may be seen as ideas for possible follow-up projects.

HMMs and transmembrane helices The work of Hessa and colleagues[19] have shown that the probability of a potential TM helix being inserted into the membrane is proportional to the difference in free energy between the helix being inserted or not. Furthermore the authors indicate that each amino acid individually contributes to the energy , without any cross-linking terms from neighboring amino acids. The energy contribution is only dependent upon the depth within the membrane into which it is buried. Under those assumptions we can say that the probability of occurrence of an amino acid in one position of a TM helix is not correlated to the presence of specific amino acids in neighboring positions of the TM helix. Hence the amino acid distributions are conditionally independent, and are ideal to model with HMMs. The PHOBIUS TM helix model currently assumes three different regions of amino acid compositions of a TM helix; one amino acid distribution for the near the translocated end, one for the core, and one for the untranslocated end of the TM helix. It might be that better accuracy of the predictions can be obtained by allowing a 'sliding scale' in amino acid composition of a TM helix.

Proteomics and transmembrane predictions The work of Daley and colleagues[101] has shown that it is feasible to ramp-up constrained TM predictions to proteome scale. By cloning a large set of *E. coli* genes, coding for TM proteins, into phoA and GFP fusion vectors the authors were able to determine the location of the C-terminus of the proteins. However, the technique of cloning individual genes, is quite labor intensive. As mentioned in section 4.1.4, Wu and colleagues[78] have, elucidated the locations of loops of TM proteins using shot-gun proteomics, a comparatively easy process. It would be interesting to combine the techniques, *i.e.* perform constrained TM topology predictions based on proteomics data.

Partially supervised training The estimation procedure of Phobius can be described as 'supervised'. Except for the boundaries between the different regions of the the data, *e.g.* TM helices and loops, we specifically give the location of each amino acid in the training sequences. As described above, there are currently large sets of partially described topologies[101, 78]. We can use such data in the parameter estimation of an HMM. Both the Baum-Welch procedure and the extended Baum-Welch of CML estimation, as described in Section 3.2, can make use of partially annotated data. A study in an other area of bioinformatics claims

increased performance when including completely unannotated data together with their annotated data in the Baum-Welch estimation[127].

Data sets Since the publication of Paper II more data have been made available that probably would be beneficial for a re-estimation of the HMM.

SignalP 3.0 set When collecting SP data for Paper II we employed the so called Menne procedure[117]. It has been pointed out by Bendtsen and colleagues in SignalP 3.0 that this procedure has some limitations. The procedure is reliant on database annotations which often have wrongly annotated cleavage sites and sometimes contains SPs that not are cleaved by signal peptidases, such as lipoprotein and twin-arginine signal peptides[112]. The authors have manually curated a large set of signal peptides. If a new version of Phobius was to be trained on this data set we would probably observe an increase in the accuracy of SP cleavage site prediction.

3D data The number of transmembrane protein structures measured by X-ray crystallography has increased since I collected the data for Phobius.

Other types of signal sequences In Paper II we only include models for the SPs of the general secretory pathway. It would make sense to also include models of other types of signal sequences in the model. Lipoprotein signal peptides and twin-arginine signal peptides both contain hydrophobic regions and are hence sometimes predicted by TM topology predictors as TM helices. It can be argued that it makes sense to include models of mitochondrial transfer peptides and chloroplast transit peptides, as they may reveal the location on the N-terminus of the mature protein.

Modeling transmembrane super families In our work with modeling GPCRs, we showed that it is possible to model TM protein super families with one single HMM. GPCRs are probably an extreme but they still share their TM topology. It would probably be beneficial to model other TM protein families as well, such as voltage gated ion channels or ABC transporters.

Multiple Sequence Feature Alignments In Paper IV we show that the accuracy of sequence feature prediction is improved when taking in account multiple sequence alignments of homologs. The reason why this works so well is that sequence features generally are conserved even though the sequences have diverged. Due to the same reason it would be reasonable to assume that multiple sequence alignment methods could increase their performance by using information from predicted sequence features parallel with the plain amino acid sequence.

Chapter 8

Acknowledgments

This work has been supported by the Swedish Knowledge Foundation and Pfizer Inc. through the Industrial Ph.D. program in Medical Bioinformatics at the Center for Medical Innovations (CMI) at the Karolinska Institute.

I would like to express my gratitude to everybody that has supported and helped me during my Ph.D. project. First of all my supervisor **Erik Sonnhammer**. Thank you for encouraging me to start as a Ph.D. student in your group and for ideas and support over the years. A sincere thank you to my co-authors: **Anders Krogh** who always given me excellent advice, often sharing his knowledge, good ideas and enthusiasm. **Markus Wistrand** who has been the ideal roommate and support. He has not yet given up teaching me about the Swedish football league and results of various cross-country races. **Anna Henricson**, who has taught me about all the latest nifty home electronic devices and the cells biological mechanisms.

Members of the lab: **Timo Lassmann**, thanks for your persistence, and for always taking care of me whenever I was a grass widower. **Abhiman Saraswathi** for teaching me about India and biology, thanks for all your lovely food. The pragmatic attitude of **Alistair Chalk** has helped me a lot, thanks for all the pubs you arranged. **Carsten Daub** for always being helpful and friendly. **Andrey Alexeyenko** for the help with all Statistica graphs. **Volker Hollich** for teaching me quite a number of things I did not know about Sweden. **Christian Storm** for not giving up. **Kevin O'Brien**, who's smiling face and sarcastic tongue made my life simpler during the couple of years we shared a room. **Åsa Perez-Bercoff** and **Julia Lindberg** for giving new energy and perspectives to our group. **Lars Arvestad** for helping me out with Linux when I first came to the lab.

The SBC/CGB membrane protein club: **Gunnar von Heijne** for taking his time and having the capacity to listen to everybody. Thanks for your help when writing applications and papers. **Arne Elofsson** for all the long discussions on the Waxholm-boats. You helped me see things clearer. **Håkan Viklund** for co-organizing the "Ph.D. student course in hidden Markov models" with me, and for all the fruitful discussions of HMMs and transmembrane topology prediction. **Karin Melén**, **Johan Nilsson**, **Erik Granseth** and **Andreas Bernsel** for good and friendly discussions.

The bioinformatics unit at CGB, which have been an inspiring forum for discussion: **Pär Engström**, **Ying Sheng**, **Albin Sandelin**, **Boris Lenhard**, **David Fredman**, **Erik Arner**, **Ellen Kindlund**, **Daniel Nilsson**, **Martti Tammi** and **Björn Andersson**. Work at the CGB would be hard without the Linux guru **Bent Terp**, who showed us that there is such a thing as danish granite. Thank you all other inspiring personalities at the CGB, **Josefin Friberg**, **Rikard Dryselius**, **Shane McCarthy**, **Marcela Ferella**, **Hagit Katzov** and **Emily Hodges**, just to mention a few.

Some external people have contributed: **Henrik Nielsen** at DTU who's vast knowledge of and

profound interest in signal peptides inspired me. Thank you for being supporting and enthusiastic. **IngMarie Nilsson** and **Carolina Lundin** at DBB for investigating the *Drosophila* odorant receptor family.

I would also like to thank all the administrative personnel that have helped me unwind the red tape, most notably: **Elisabeth Grenmyr** & **Gitt Elsén** at CGB, **Matti Nikkola** & **Christine Jansson** at CMB and **Per-Erik Jansson** & **Lena Lewin** at CMI.

I am also grateful for all feedback from Phobius users around the world.

Finally a big thank you to my family: My Father **Stig** for frequent cooking and babysitting, and for keeping summerhouse and boat in good condition. My Mother **Lena** for moral support. **Ludvig** for letting me play with his electric race track. **Valter** just learn to walk when I was writing this. That gives perspective. **Stina**, thank you for being a wonderful wife and mother.

Bibliography

1. Jansson, T. *Who will comfort Toffle?* Schildts, 1960.
2. Drews, J. Drug discovery: a historical perspective. *Science*, **287**(5460):1960–1964, Mar 2000.
3. Hopkins, AL and Groom, CR. The druggable genome. *Nat Rev Drug Discov*, **1**:727–30, 2002.
4. Liu, J and Rost, B. Comparing function and structure between entire proteomes. *Protein Sci*, **10**(10):1970–1979, Oct 2001.
5. Schulz, GE. The structure of bacterial outer membrane proteins. *Biochim Biophys Acta*, **1565**(2):308–317, Oct 2002.
6. Van den Berg, B, Clemons, WMJ, Collinson, I, Modis, Y, Hartmann, E, Harrison, SC, and Rapoport, TA. X-ray structure of a protein-conducting channel. *Nature*, **427**(6969):36–44, Jan 2004.
7. Ménétret, JF, Hegde, RS, Heinrich, SU, Chandramouli, P, Ludtke, SJ, Rapoport, TA, and Akey, CW. Architecture of the ribosome-channel complex derived from native membranes. *J Mol Biol*, **348**(2):445–457, Apr 2005.
8. Mitra, K, Schaffitzel, C, Shaikh, T, Tama, F, Jenni, S, Brooks, CLr, Ban, N, and Frank, J. Structure of the E. coli protein-conducting channel bound to a translating ribosome. *Nature*, **438**(7066):318–324, Nov 2005.
9. Luirink, J, von Heijne, G, Houben, E, and de Gier, JW. Biogenesis of inner membrane proteins in Escherichia coli. *Annu Rev Microbiol*, **59**:329–355, 2005.
10. White, SH and von Heijne, G. Transmembrane helices before, during, and after insertion. *Curr Opin Struct Biol*, **15**(4):378–386, Aug 2005.
11. Osborne, AR, Rapoport, TA, and van den Berg, B. Protein translocation by the Sec61/SecY channel. *Annu Rev Cell Dev Biol*, **21**:529–550, 2005.
12. Matlack, KE, Misselwitz, B, Plath, K, and Rapoport, TA. BiP acts as a molecular ratchet during posttranslational transport of prepro-alpha factor across the ER membrane. *Cell*, **97**(5):553–564, May 1999.
13. Mori, H and Ito, K. The Sec protein-translocation pathway. *Trends Microbiol*, **9**(10):494–500, Oct 2001.
14. Robinson, C and Bolhuis, A. Protein targeting by the twin-arginine translocation pathway. *Nat Rev Mol Cell Biol*, **2**(5):350–356, May 2001.

15. Salmond, GP and Reeves, PJ. Membrane traffic wardens and protein secretion in gram-negative bacteria. *Trends Biochem Sci*, **18**(1):7–12, Jan 1993.

16. Hueck, CJ. Type III protein secretion systems in bacterial pathogens of animals and plants. *Microbiol Mol Biol Rev*, **62**(2):379–433, Jun 1998.

17. Alberts, B, Johnson, A, Lewis, J, Raff, M, Roberts, K, and Walter, P. *Molecular biology of the cell*. Garland Science, 4th edition, 2002.

18. Sadlish, H, Pitonzo, D, Johnson, AE, and Skach, WR. Sequential triage of transmembrane segments by Sec61alpha during biogenesis of a native multispanning membrane protein. *Nat Struct Mol Biol*, **12**(10):870–878, Oct 2005.

19. Hessa, T, Kim, H, Bihlmaier, K, Lundin, C, Boekel, J, Andersson, H, Nilsson, I, White, SH, and von Heijne, G. Recognition of transmembrane helices by the endoplasmic reticulum translocon. *Nature*, **433**(7024):377–381, Jan 2005.

20. Jensen, FV. *Bayesian Networks and Decision Graphs*. Springer, 2001.

21. Whitley, D. A genetic algorithm tutorial. *Statistics and Computing*, **4**:65–85, 1994.

22. Burges, CJC. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, **2**(2):121–167, 1998.

23. Baldi, P and Brunak, S. *Bioinformatics : the machine learning approach (2nd Edition)*. MIT Press, 2001.

24. Duda, RO, Hart, PE, and Stork, DG. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

25. Michel, TM. *Machine Learning*. McGraw-Hill Book Co., Singapore, 1997.

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

27. Altschul, SF, Gish, W, Miller, W, Myers, EW, and Lipman, DJ. Basic local alignment search tool. *J Mol Biol*, **215**(3):403–410, Oct 1990.

28. Durbin, R, Eddy, SR, Krogh, A, and Mitchison, G. *Biological Sequence Analysis. Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

29. Henikoff, S and Henikoff, JG. Position-based sequence weights. *J Mol Biol*, **243**(4):574–578, Nov 1994.

30. Lassmann, T and Sonnhammer, ELL. Kalign–an accurate and fast multiple sequence alignment algorithm. *BMC Bioinformatics*, **6**:298, 2005.

31. Lassmann, T and Sonnhammer, ELL. Kalign, Kalignvu and Mumsa: web servers for multiple sequence alignment. Submitted.

32. Lamarck, JB. *Philosophie zoologique, ou Exposition des considérations relatives à l'histoire naturelle des animaux*. Dentu, Paris, 1809.

33. Chothia, C and Lesk, AM. The relation between the divergence of sequence and structure in proteins. *EMBO J*, **5**(4):823–826, Apr 1986.

34. Murzin, AG. How far divergent evolution goes in proteins. *Curr Opin Struct Biol*, **8**(3):380–387, Jun 1998.

35. Jensen, LJ, Gupta, R, Blom, N, Devos, D, Tamames, J, Kesmir, C, Nielsen, H, Staerfeldt, HH, Rapacki, K, Workman, C, Andersen, CAF, Knudsen, S, Krogh, A, Valencia, A, and Brunak, S. Prediction of human protein function from post-translational modifications and localization features. *J Mol Biol*, **319**:1257–65, 2002.

36. Jensen, LJ, Ussery, DW, and Brunak, S. Functionality of system components: conservation of protein function in protein feature space. *Genome Res*, **13**(11):2444–2449, Nov 2003.

37. Chen, CP, Kernytsky, A, and Rost, B. Transmembrane helix predictions revisited. *Protein Sci*, **11**(12):2774–2791, Dec 2002.

38. Viklund, H and Elofsson, A. Best alpha-helical transmembrane protein topology predictions are achieved using hidden Markov models and evolutionary information. *Protein Sci*, **3**(7):1908–1917, Jul 2004.

39. Martin, J, Gibrat, J, and Rodolphe, F. HMM for local protein structure. In *Proceedings of ASMDA 2005*, pages 180–187. ENST Bretagne, Brest, France, May 2005.

40. Baum, LE and Petrie, T. Statistical inference for probabilistic functions of finite-state Markov chains. *Annals of Mathematical Statistics*, **37**(6):1554–1563, Dec 1966.

41. Baum, LE, Petrie, T, Soules, G, and Weiss, N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, **41**:164–171, 1970.

42. Rabiner, L. A tutorial on hidden Markov models and selected aplications in speech recognition. In *Proceedings of the IEEE.*, volume 77, pages 257–286, Feb 1989.

43. Koski, T. *Hidden Markov Models for Bioinformatics*. Kluwer Academic Publishers, 2001.

44. Churchill, GA. Stochastic models for heterogeneous DNA sequences. *Bull Math Biol*, **51**(1):79–94, 1989.

45. Krogh, A, Brown, M, Mian, IS, Sjölander, K, and Haussler, D. Hidden Markov models in computational biology. applications to protein modeling. *J Mol Biol*, **235**(5):1501–31, Feb 1994.

46. Eddy, SR. Profile hidden markov models. *Bioinformatics*, **14**(9):755–63, 1998.

47. Sonnhammer, EL, Eddy, SR, and Durbin, R. Pfam: a comprehensive database of protein domain families based on seed alignments. *Proteins*, **28**(3):405–420, Jul 1997.

48. Sonnhammer, EL, von Heijne, G, and Krogh, A. A hidden markov model for predicting transmembrane helices in protein sequences. *Proc Int Conf Intell Syst Mol Biol*, **6**:175–82, 1998.

49. Tusnady, GE and Simon, I. Principles governing amino acid composition of integral membrane proteins: application to topology prediction. *J Mol Biol*, **283**(2):489–506, Oct 1998.

50. Krogh, A, Larsson, B, von Heijne, G, and Sonnhammer, EL. Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol*, **305**(3):567–80, Jan 2001.

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

52. Delorenzi, M and Speed, T. An HMM model for coiled-coil domains and a comparison with PSSM-based predictions. *Bioinformatics*, **18**(4):617–25, Apr 2002.

53. Krogh, A, Mian, IS, and Haussler, D. A hidden Markov model that finds genes in E. coli DNA. *Nucleic Acids Res*, **22**(22):4768–78, Nov 1994.

54. Burge, C and Karlin, S. Prediction of complete gene structures in human genomic DNA. *J Mol Biol*, **268**(1):78–94, Apr 1997.

55. Bystroff, C, Thorsson, V, and Baker, D. HMMSTR: a hidden Markov model for local sequence-structure correlations in proteins. *J Mol Biol*, **301**(1):173–90, Aug 2000.

56. Needleman, SB and Wunsch, CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, **48**(3):443–53, Mar 1970.

57. Holmes, I and Durbin, R. Dynamic programming alignment accuracy. *J Comput Biol*, **5**(3):493–504, Fall 1998.

58. Do, CB, Mahabhashyam, MSP, Brudno, M, and Batzoglou, S. ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome Res*, **15**(2):330–340, Feb 2005.

59. Dempster, AP, Laird, NM, and Rubin, DB. Maximum likelihood from incomplete data via the EM algorithm. *J Royal Stat Soc B*, **39**(1):1–38, 1977.

60. Krogh, A. Hidden Markov models for labeled sequences. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition.*, pages 140–44. IEEE Computer Society Press., Los Alamitos, California, Oct 1994.

61. Viterbi, AJ. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, **IT-13**:260–269, 1967.

62. Krogh, A. Two methods for improving performance of an HMM and their application for gene finding. *Proc Int Conf Intell Syst Mol Biol*, **5**:179–86, 1997.

63. Brejova, B, Brown, DG, and Vinar, T. The most probable labeling problem in HMMs and its applications to bioinformatics. In I Jonassen and J Kim, editors, *Algorithms in Bioinformatics (WABI 2004)*, volume 3240 of *Lecture Notes in Bioinformatics*, pages 426–437. Springer, Bergen, Norway, 2004.

64. Schwartz, R and Chow, Y. The N-best algorithm: An efficient and exact procedure for finding the N most likely sentence hypotheses. In *Proceedings of ICASSP 1990.*, pages 81–84. IEEE., Albuquerque, New Mexico, Apr 1990.

65. Fitch, WM. Homology a personal view on some of the problems. *Trends Genet*, **16**(5):227–231, May 2000.

66. Edgar, RC and Sjolander, K. COACH: profile-profile alignment of protein families using hidden Markov models. *Bioinformatics*, **20**(8):1309–1318, May 2004. Evaluation Studies.

67. Martelli, PL, Fariselli, P, Krogh, A, and Casadio, R. A sequence-profile-based HMM for predicting and discriminating beta barrel membrane proteins. *Bioinformatics*, **18 Suppl 1**:S46–53, 2002.

68. Soding, J. Protein homology detection by HMM-HMM comparison. *Bioinformatics*, Nov 2004.

69. Munch, K. *Transcript prediction in eukaryotes using hidden Markov models*. Ph.D. thesis, University of Copenhagen, Nov 2005.

70. Bobbio, A, Horvàth, A, Scarpa, M, and Telek, M. Acyclic Discrete Phase Type Distributions: Properties and a Parameter Estimation Algorithm. *Performance Evaluation*, **4**:1–32, 2003.

71. Zhang, P, Toyoshima, C, Yonekura, K, Green, NM, and Stokes, DL. Structure of the calcium pump from sarcoplasmic reticulum at 8-Åresolution. *Nature*, **392**(6678):835–839, Apr 1998.

72. Ehrmann, M, Boyd, D, and Beckwith, J. Genetic analysis of membrane protein topology by a sandwich gene fusion approach. *Proc Natl Acad Sci U S A*, **87**(19):7574–7578, Oct 1990.

73. Manoil, C and Beckwith, J. A genetic approach to analyzing membrane protein topology. *Science*, **233**(4771):1403–1408, Sep 1986.

74. Broome-Smith, JK, Tadayyon, M, and Zhang, Y. Beta-lactamase as a probe of membrane protein assembly and protein export. *Mol Microbiol*, **4**(10):1637–1644, Oct 1990.

75. Silhavy, TJ, Shuman, HA, Beckwith, J, and Schwartz, M. Use of gene fusions to study outer membrane protein localization in Escherichia coli. *Proc Natl Acad Sci U S A*, **74**(12):5411–5415, Dec 1977.

76. Feilmeier, BJ, Iseminger, G, Schroeder, D, Webber, H, and Phillips, GJ. Green fluorescent protein functions as a reporter for protein localization in Escherichia coli. *J Bacteriol*, **182**(14):4068–4076, Jul 2000.

77. Hart, GW, Brew, K, Grant, GA, Bradshaw, RA, and Lennarz, WJ. Primary structural requirements for the enzymatic formation of the N-glycosidic bond in glycoproteins. Studies with natural and synthetic peptides. *J Biol Chem*, **254**(19):9747–9753, Oct 1979.

78. Wu, CC, MacCoss, MJ, Howell, KE, and Yates, JRr. A method for the comprehensive proteomic analysis of membrane proteins. *Nat Biotechnol*, **21**(5):532–538, May 2003.

79. Argos, P, Rao, JK, and Hargrave, PA. Structural prediction of membrane-bound proteins. *Eur J Biochem*, **128**(2-3):565–75, Nov 1982.

80. Kyte, J and Doolittle, RF. A simple method for displaying the hydropathic character of a protein. *J Mol Biol*, **157**(1):105–32, May 1982.

81. von Heijne, G. The distribution of positively charged residues in bacterial inner membrane proteins correlates with the trans-membrane topology. *EMBO J*, **5**(11):3021–27, Nov 1986.

82. von Heijne, G. Membrane protein structure prediction. Hydrophobicity analysis and the positive-inside rule. *J Mol Biol*, **225**(2):487–94, May 1992.

83. Mitaku, S, Hirokawa, T, and Tsuji, T. Amphiphilicity index of polar amino acids as an aid in the characterization of amino acid preference at membrane-water interfaces. *Bioinformatics*, **18**(4):608–616, Apr 2002.

84. Cserzo, M, Wallin, E, Simon, I, von Heijne, G, and Elofsson, A. Prediction of transmembrane alpha-helices in prokaryotic membrane proteins: the dense alignment surface method. *Protein Eng*, **10**(6):673–676, Jun 1997.

85. Rost, B, Casadio, R, Fariselli, P, and Sander, C. Transmembrane helices predicted at 95% accuracy. *Protein Sci*, **4**(3):521–33, Mar 1995.

86. Jones, DT, Taylor, WR, and Thornton, JM. A model recognition approach to the prediction of all-helical membrane protein structure and topology. *Biochemistry*, **33**(10):3038–49, Mar 1994.

87. Tusnady, GE and Simon, I. The HMMTOP transmembrane topology prediction server. *Bioinformatics*, **17**(9):849–850, Sep 2001.

88. Kahsay, RY, Gao, G, and Liao, L. An improved hidden Markov model for transmembrane protein detection and topology prediction and its applications to complete genomes. *Bioinformatics*, **21**(9):1853–1858, May 2005.

89. Zhai, Y and Saier, MHJ. The beta-barrel finder (BBF) program, allowing identification of outer membrane beta-barrel proteins encoded within prokaryotic genomes. *Protein Sci*, **11**(9):2196–2207, Sep 2002.

90. Persson, B and Argos, P. Prediction of membrane protein topology utilizing multiple sequence alignments. *J Protein Chem*, **16**(5):453–457, Jul 1997.

91. Eyrich, VA and Rost, B. META-PP: single interface to crucial prediction servers. *Nucleic Acids Res*, **31**(13):3308–3310, Jul 2003.

92. Sonnhammer, EL and Wootton, JC. Integrated graphical analysis of protein sequence features predicted from sequence composition. *Proteins*, **45**(3):262–273, Nov 2001.

93. Nilsson, J, Persson, B, and Von Heijne, G. Prediction of partial membrane protein topologies using a consensus approach. *Protein Sci*, **11**(12):2974–2980, Dec 2002.

94. Arai, M, Mitsuke, H, Ikeda, M, Xia, JX, Kikuchi, T, Satake, M, and Shimizu, T. ConPred II: a consensus prediction method for obtaining transmembrane topology models with high reliability. *Nucleic Acids Res*, **32**(Web Server issue):390–393, Jul 2004.

95. Jayasinghe, S, Hristova, K, and White, SH. MPtopo: A database of membrane protein topology. *Protein Sci*, **10**(2):455–458, Feb 2001.

96. Möller, S, Kriventseva, EV, and Apweiler, R. A collection of well characterised integral membrane proteins. *Bioinformatics*, **16**(12):1159–1160, Dec 2000.

97. Ikeda, M, Arai, M, Okuno, T, and Shimizu, T. TMPDB: a database of experimentally-characterized transmembrane topologies. *Nucleic Acids Res*, **31**(1):406–409, Jan 2003.

98. Möller, S, Croning, MD, and Apweiler, R. Evaluation of methods for the prediction of membrane spanning regions. *Bioinformatics*, **17**(7):646–653, Jul 2001. Evaluation Studies.

99. Ikeda, M, Arai, M, Lao, DM, and Shimizu, T. Transmembrane topology prediction methods: a re-assessment and improvement by a consensus method using a dataset of experimentally-characterized transmembrane topologies. *In Silico Biol*, **2**(1):19–33, 2002.

100. Drew, D, Sjostrand, D, Nilsson, J, Urbig, T, Chin, Cn, de Gier, JW, and von Heijne, G. Rapid topology mapping of Escherichia coli inner-membrane proteins by prediction and PhoA/GFP fusion analysis. *Proc Natl Acad Sci U S A*, **99**(5):2690–2695, Mar 2002.

101. Daley, DO, Rapp, M, Granseth, E, Melen, K, Drew, D, and von Heijne, G. Global topology analysis of the Escherichia coli inner membrane proteome. *Science*, **308**(5726):1321–1323, May 2005.

102. Melen, K, Krogh, A, and von Heijne, G. Reliability measures for membrane protein topology prediction algorithms. *J Mol Biol*, **327**(3):735–744, Mar 2003.

103. Granseth, E, Daley, DO, Rapp, M, Melen, K, and von Heijne, G. Experimentally constrained topology models for 51,208 bacterial inner membrane proteins. *J Mol Biol*, **352**(3):489–494, Sep 2005.

104. von Heijne, G. Patterns of amino acids near signal-sequence cleavage sites. *Eur J Biochem*, **133**(1):17–21, Jun 1983.

105. Perlman, D and Halvorson, HO. A putative signal peptidase recognition site and sequence in eukaryotic and prokaryotic signal peptides. *J Mol Biol*, **167**(2):391–409, Jun 1983.

106. Nielsen, H, Engelbrecht, J, Brunak, S, and von Heijne, G. A neural network method for identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Int J Neural Syst*, **8**(5-6):581–99, Oct 1997.

107. Nielsen, H, Engelbrecht, J, Brunak, S, and von Heijne, G. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Eng*, **10**(1):1–6, Jan 1997.

108. Klein, RD, Gu, Q, Goddard, A, and Rosenthal, A. Selection for genes encoding secreted proteins and receptors. *Proc Natl Acad Sci U S A*, **93**(14):7108–7113, Jul 1996.

109. Hoja, MR, Wahlestedt, C, and Hoog, C. A visual intracellular classification strategy for uncharacterized human proteins. *Exp Cell Res*, **259**(1):239–246, Aug 2000.

110. von Heijne, G. A new method for predicting signal sequence cleavage sites. *Nucleic Acids Res*, **14**(11):4683–4690, Jun 1986.

111. Hiller, K, Grote, A, Scheer, M, Munch, R, and Jahn, D. PrediSi: prediction of signal peptides and their cleavage positions. *Nucleic Acids Res*, **32**(Web Server issue):375–379, Jul 2004.

112. Bendtsen, JD, Nielsen, H, von Heijne, G, and Brunak, S. Improved prediction of signal peptides: SignalP 3.0. *J Mol Biol*, **340**(4):783–795, Jul 2004.

113. Zhang, Z and Wood, WI. A profile hidden Markov model for signal peptides generated by HMMER. *Bioinformatics*, **19**(2):307–308, Jan 2003.

114. Juncker, AS, Willenbrock, H, Von Heijne, G, Brunak, S, Nielsen, H, and Krogh, A. Prediction of lipoprotein signal peptides in Gram-negative bacteria. *Protein Sci*, **12**(8):1652–1662, Aug 2003.

115. Chou, KC. Prediction of protein signal sequences and their cleavage sites. *Proteins*, **42**(1):136–139, Jan 2001.

116. Vert, JP. Support vector machine prediction of signal peptide cleavage site using a new class of kernels for strings. *Pac Symp Biocomput*, pages 649–60, 2002.

117. Menne, KM, Hermjakob, H, and Apweiler, R. A comparison of signal sequence prediction methods using a test set of signal peptides. *Bioinformatics*, **16**(8):741–742, Aug 2000.

118. Zhang, Z and Henzel, WJ. Signal peptide prediction based on analysis of experimentally verified cleavage sites. *Protein Sci*, **13**(10):2819–2824, Oct 2004.

119. Klee, EW and Ellis, LBM. Evaluating eukaryotic secreted protein prediction. *BMC Bioinformatics*, **6**:256, Oct 2005.

120. Lao, DM, Arai, M, Ikeda, M, and Shimizu, T. The presence of signal peptide significantly affects transmembrane topology prediction. *Bioinformatics*, **18**(12):1562–1566, Dec 2002. Evaluation Studies.

121. Lao, DM, Okuno, T, and Shimizu, T. Evaluating transmembrane topology prediction methods for the effect of signal peptide in topology prediction. *In Silico Biol*, **2**(4):485–494, 2002.

122. Nielsen, H. *From sequence to sorting: prediction of signal peptides.* Ph.D. thesis, Stockholm University, 1999.

123. Oh, YS and Turner, RJ. Evidence that the COOH terminus of human presenilin 1 is located in extracytoplasmic space. *Am J Physiol Cell Physiol*, **289**(3):576–581, Sep 2005.

124. Laudon, H, Hansson, EM, Melen, K, Bergman, A, Farmery, MR, Winblad, B, Lendahl, U, von Heijne, G, and Naslund, J. A nine-transmembrane domain topology for presenilin 1. *J Biol Chem*, **280**(42):35352–35360, Oct 2005.

125. Baldi, P and Chauvin, Y. Hidden Markov Models of the G-protein-coupled receptor family. *J Comput Biol*, **1**(4):311–336, Winter 1994.

126. Benton, R, Sachse, S, Michnick, S, and Vosshall, L. Atypical Membrane Topology and Heteromeric Function of Drosophila Odorant Receptors In Vivo. *PLoS Biol*, **4**(2):e20, Jan 2006.

127. Schliep, A, Schonhuth, A, and Steinhoff, C. Using hidden Markov models to analyze gene expression time course data. *Bioinformatics*, **19 Suppl 1**:255–263, 2003.