BRICS

Basic Research in Computer Science

BRICS RS-99-6 Lyngsø et al.: Measures on Hidden Markov Models

Measures on Hidden Markov Models

Rune B. Lyngsø
Christian N. S. Pedersen
Henrik Nielsen

BRICS Report Series RS-99-6

ISSN 0909-0878 February 1999

See back inner page for a list of recent BRICS Report Series publications. Copies may be obtained by contacting:

BRICS
Department of Computer Science
University of Aarhus
Ny Munkegade, building 540
DK–8000 Aarhus C
Denmark

Telephone: +45 8942 3360
Telefax: +45 8942 3255
Internet: BRICS@brics.dk

BRICS publications are in general accessible through the World Wide Web and anonymous FTP through these URLs:

http://www.brics.dk
ftp://ftp.brics.dk
This document in subdirectory RS/99/6/

Measures on hidden Markov models

Rune B. Lyngsø[*] Christian N. S. Pedersen[†]
Henrik Nielsen[‡]

Abstract

Hidden Markov models were introduced in the beginning of the 1970's as a tool in speech recognition. During the last decade they have been found useful in addressing problems in computational biology such as characterising sequence families, gene finding, structure prediction and phylogenetic analysis. In this paper we propose several measures between hidden Markov models. We give an efficient algorithm that computes the measures for left-right models, e.g. profile hidden Markov models, and discuss how to extend the algorithm to other types of models. We present an experiment using the measures to compare hidden Markov models for three classes of signal peptides.

1 Introduction

A hidden Markov model describes a probability distribution over a potentially infinite set of sequences. It is convenient to think of a hidden Markov model as generating a sequence according to some probability distribution by following a first order Markov chain of states, called the

[*] Department of Computer Science, University of Aarhus, Denmark. E-mail: rlyngsoe@daimi.au.dk. Work done in part while visiting the Institute for Biomedical Computing at Washington University, St. Louis.

[†] Basic Research In Computer Science, Centre of the Danish National Research Foundation, University of Aarhus, Denmark. Supported by the ESPRIT Long Term Research Programme of the EU under project number 20244 (ALCOM-IT). E-mail: cstorm@brics.dk.

[‡] Center for Biological Sequence Analysis, Centre of the Danish National Research Foundation, Technical University of Denmark, Denmark. E-mail: hnielsen@cbs.dtu.dk.

path, from a specific start-state to a specific end-state and emitting a symbol according to some probability distribution each time a state is entered. One strength of hidden Markov models is the ability efficiently to compute the probability of a given sequence as well as the most probable path that generates a given sequence. Hidden Markov models were introduced in the beginning of the 1970's as a tool in speech recognition. In speech recognition the set of sequences might correspond to digitised sequences of human speech and the most likely path for a given sequence is the corresponding sequence of words. Rabiner [19] gives a good introduction to the theory of hidden Markov models and their applications to speech recognition.

Hidden Markov models were introduced in computational biology in 1989 by Churchill [5]. Durbin et al. [6] and Eddy [7, 8] are good overviews of the use of hidden Markov models in computational biology. One of the most popular applications is to use them to characterise sequence families by using so called profile hidden Markov models introduced by Krogh et al. [15]. For a profile hidden Markov model the probability of a given sequence indicates how likely it is that the sequence is a member of the modelled sequence family, and the most likely path for a given sequence corresponds to an alignment of the sequence against the modelled sequence family.

An important advance in the use of hidden Markov models in computational biology within the last two years, is the fact that several large libraries of profile hidden Markov models have become available [8]. These libraries not only make it possible to classify new sequences, but also open up the possibility of comparing sequence families by comparing the profiles of the families instead of comparing the individual members of the families, or of comparing entire sequence families instead of the individual members of the family to a hidden Markov model constructed to model a particular feature. To our knowledge little work has been published in this area, except for alignment of profiles [9].

In this paper we propose measures for hidden Markov models that can be used to address this problem. The measures are based on what we call the co-emission probability of two hidden Markov models. We present an efficient algorithm that computes the measures for profile hidden Markov models and observe that the left-right architecture is the only special property of profile hidden Markov models required by the algorithm. We describe how to extend the algorithm to broader classes of models and how to approximate the measures for general hidden Markov models. The method can easily be adapted to various special cases, e.g. if it is required that paths pass through certain states.

As the algorithm we present is not limited to profile hidden Markov models, we have chosen to emphasise this generality by presenting an application to a set of hidden Markov models for signal peptides. These models do not strictly follow the profile architecture and consequently cannot be compared using profile alignment [9].

The rest of the paper is organised as follows. In section 2 we discuss hidden Markov models in more detail. In section 3 we introduce the co-emission probability of two hidden Markov models and formulate an algorithm for computing this probability of two profile hidden Markov models. In section 4 we use the co-emission probability to formulate several measures between hidden Markov models. In section 5 we discuss extensions to more general models. In section 6 we present an experiment using the method to compare three classes of signal peptides. Finally in section 7 we briefly discuss how to compute relaxed versions of the co-emission probability.

2 Hidden Markov models

Let M be a hidden Markov model that generates sequences over some finite alphabet Σ with probability distribution P_M, i.e. $P_M(s)$ denotes the probability of $s \in \Sigma^*$ under model M. Like a classical Markov model, a hidden Markov model consists of a set of interconnected states. We use $P_q(q')$ to denote the probability of a transition from state q to state q'. These probabilities are usually called *state transition probabilities.* The transition structure of a hidden Markov model is often shown as a directed graph with a node for each state, and an edge between two nodes if the corresponding state transition probability is non-zero. Figure 1 shows an example of a transition structure. Unlike a classical Markov model, a state in a hidden Markov model can generate or emit a symbol according to a local probability distribution over all possible symbols. We use $P_q(\sigma)$ to denote the probability of generating or emitting symbol $\sigma \in \Sigma$ in state q. These probabilities are usually called *symbol emission probabilities.* If a state does not have symbol emission probabilities we say that the state is a silent state.

It is often convenient to think of a hidden Markov model as a generative model, in which a run generates or emits a sequence $s \in \Sigma^*$ with probability $P_M(s)$. A run of a hidden Markov model begins in a special start-state and continues from state to state according to the state transition probabilities until a special end-state is reached. Each time a non-silent state is entered, a symbol is emitted according to the symbol emission probabilities of the state. A run thus results in a Markovian sequence of states as well as a generated sequence of symbols. The name "hidden Markov model" comes from the fact that the Markovian sequence of states, also called the path, is hidden, while only the generated sequence of symbols is observable.

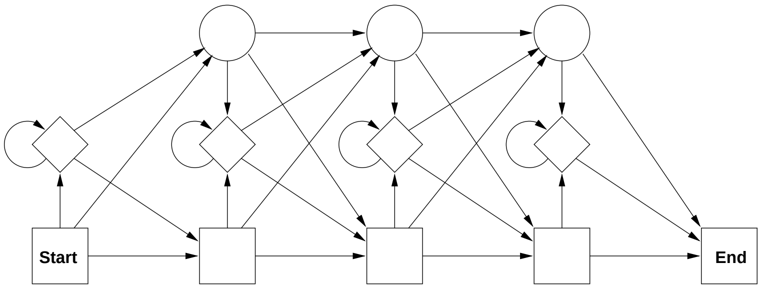


Figure 1: The transition structure of a profile hidden Markov model. The squares are the match-states, the diamonds are the insert-states and the circles are the silent delete-states.

Hidden Markov models have found applications in many areas of computational biology, e.g. gene finding [14] and protein structure prediction [20], but probably the most popular use is as *profiles* for sequence families. A profile is a position-dependent scoring scheme that captures the characteristics of a sequence family, in the sense that the score peaks around members of the family. Profiles are useful when searching for unknown members of a sequence family and several methods have been used to construct and use profiles [10, 16, 21]. Krogh et al. [15] realized that simple hidden Markov models, which they called profile hidden Markov models, were able to capture all other profile methods.

The states of a profile hidden Markov model are divided into match-, insert- and delete-states. Figure 1 illustrates the transition structure of a simple profile hidden Markov model. Note the highly repetitive transition structure. Each of the repeated elements consisting of a match-, insert- and delete-state models a position in the consensus sequence for the sequence family. The silent delete-state makes it possible to skip a position while the self-loop on the insert-state makes it possible to insert one or more symbols between two positions. Another distinctive feature of the structure of profile hidden Markov models is the absence of cycles, except for the self-loops on the insert-states. Hidden Markov models with this property are generally referred to as left-right [13] (or sometimes Bakis [1]) models, as they can be drawn such that all transitions go from left to right.

The state transition and symbol emission probabilities of a profile hidden Markov model (the parameters of the model) should be such that $P_M(s)$ is significant if s is a member of the sequence family. These probabilities can be derived from a multiple alignment of the sequence family, but more importantly, several methods exist to estimate them (or train the model) if a multiple alignment is not available [2, 6, 8].

3 Co-emission probability of two models

When using a profile hidden Markov model, it is sometimes sufficient just to focus on the most probable path through the model, e.g. when using a profile hidden Markov model to generate alignments. It is, however, well known that profile hidden Markov models possess a lot more information than the most probable paths, as they allow the generation of an infinity of sequences, each by a multitude of paths. Thus, when comparing two profile hidden Markov models, one should look at the entire spectrum of sequences and probabilities.

In this section we will describe how to compute the probability that two profile hidden Markov models independently generate the same sequence, that is for models M_1 and M_2 generating sequences over an alphabet Σ we compute

$$\sum_{s \in \Sigma^*} P_{M_1}(s) P_{M_2}(s). \tag{1}$$

We will call this the *co-emission probability* of the two models. The algorithm we present to compute the co-emission probability is a dynamic programming algorithm similar to the algorithm for computing the probability that a hidden Markov model will generate a specific sequence [6, Chapter 3]. We will describe how to handle the extra complications arising when exchanging the sequence with a profile hidden Markov model.

When computing the probability that a hidden Markov model M generates a sequence $s = s_1 \ldots s_n$, a table indexed by a state from M and an index from s is usually built. An entry (q, i) in this table holds the probability of being in the state q in M and having generated the prefix $s_1 \dots s_i$ of s. We will use a similar approach to compute the co-emission probability. Given two hidden Markov models M_1 and M_2, we will describe how to build a table A indexed by states from the two hidden Markov models, such that the entry $A(q, q')$ – where q is a state of M_1 and q' is a state of M_2 – holds the probability of being in state q in M_1 and q' in M_2 and having independently generated identical sequences on the paths to q and q'. The entry indexed by the two end-states will then hold the probability of being in the end-states and having generated identical sequences, that is the co-emission probability.

To build the table, A, we have to specify how to fill out all entries of A. For a specific entry $A(q, q')$ this depends on the types of states q and q'. As explained in the previous section, a profile hidden Markov model has three types of states (insert-, match- and delete-states) and two special states (start and end). We postpone the treatment of the special states until we have described how to handle the other types of states. For reasons of succinctness we will treat insert- and match-states as special cases of a more general type, which we will call a *generate*-state; this type encompasses all non-silent states of the profile hidden Markov models.

The generate-state will be a merging of match-states and insert-states, thus both allowing a transition to itself and having a transition from the previous insert-state; a match-state can be viewed as a generate-state with probability zero of choosing the transition to itself, and an insert-state can be viewed as a generate-state with probability zero of choosing the transition from the previous insert-state. Note that this merging of match- and insert-states is only conceptual; we do not physically merge any states, but just handle the two types of states in a uniform way. This leaves two types of states and thus four different pairs of types. This number can be reduced to three, by observing that the two cases of a generate/delete-pair are symmetric, and thus can be handled the same way.

The rationale behind the algorithm is to split paths up in the last transition(s)[1] and all that preceded this. We will thus need to be able to refer to the states with transitions to q and q'. In the following, m, i and d will refer to the match-, insert- and delete-state with a transition to q, and m', i' and d' to those with a transition to q'. Observe that if q (or q') is an insert-state, then i (or i') is the *previous* insert-state which,

[1] In some of the cases explained below, we will only extend the path in one of the models with an extra transition, hence the unspecificity.

by the generate-state generalisation, has a transition to q (or q') with probability zero.

delete/delete entry Assume that q and q' are both delete-states. As these states don't emit symbols, we just have to sum over all possible combinations of immediate predecessors of q and q', of the probability of being in these states and having independently generated identical sequences, multiplied by the joint probability of independently choosing the transitions to q and q'. For the calculation of $A(q, q')$ we thus get the equation

$$\begin{aligned}
&A(q, q') = \\
&\quad A(m, m')P_m(q)P_{m'}(q') + A(m, i')P_m(q)P_{i'}(q') + A(m, d')P_m(q)P_{d'}(q') \\
&\quad + A(i, m')P_i(q)P_{m'}(q') + A(i, i')P_i(q)P_{i'}(q') + A(i, d')P_i(q)P_{d'}(q') \\
&\quad + A(d, m')P_d(q)P_{m'}(q') + A(d, i')P_d(q)P_{i'}(q') + A(d, d')P_d(q)P_{d'}(q').
\end{aligned} \tag{2}$$

delete/generate entry Assume that q is a delete-state and q' is a generate-state. Envision paths leading to q and q' respectively while independently generating the same sequence. As q does not emit symbols while q' does, the path to q's immediate predecessor (that is, the path to q with the actual transition to q removed) must also have generated the same sequence as the path to q'. We thus have to sum over all immediate predecessors of q, of the probability of being in this state and in q' and having generated identical sequences, multiplied by the probability of choosing the transition to q. For the calculation of $A(q, q')$ in this case we thus get the following equation

$$A(q, q') = A(m, q')P_m(q) + A(i, q')P_i(q) + A(d, q')P_d(q). \tag{3}$$

generate/generate entry Assume that q and q' are both generate-states. The last character in sequences generated on the paths to q and q' are generated by q and q' respectively. We will denote the probability that these two states independently generate the same symbol by p, and it is an easy observation that

$$p = \sum_{\sigma \in \Sigma} P_q(\sigma)P_{q'}(\sigma). \tag{4}$$

The problem with generate/generate entries is that the last transitions on paths to q and q' might actually come from q and q' themselves, due to the self-loops of generate states. It thus seems that we need $A(q, q')$ to be able to compute $A(q, q')$!

So let us start out by assuming that at most one of the paths to q and q' has a self-loop transition as the last transition. Then we can easily compute the probability of being in q and q' and having independently generated the same sequence on the paths to q and q', by summing over all combinations of states with transitions to q and q' (including combinations with either q or q' but not both) the probabilities of these combinations, multiplied by p (for independently generating the same symbol at q and q') and the joint probability of independently choosing the transitions to q and q'. We denote this probability by $A_0(q, q')$, and by the above argument the equation for computing it is

$$\begin{aligned} A_0(q,q') = & p(A(m,m')P_m(q)P_{m'}(q') + A(m,i')P_m(q)P_{i'}(q') \\ & + A(m,d')P_m(q)P_{d'}(q') + A(m,q')P_m(q)P_{q'}(q') \\ & + A(i,m')P_i(q)P_{m'}(q') + A(i,i')P_i(q)P_{i'}(q') \\ & + A(i,d')P_i(q)P_{d'}(q') + A(i,q')P_i(q)P_{q'}(q') \\ & + A(d,m')P_d(q)P_{m'}(q') + A(d,i')P_d(q)P_{i'}(q') \\ & + A(d,d')P_d(q)P_{d'}(q') + A(d,q')P_d(q)P_{q'}(q') \\ & + A(q,m')P_q(q)P_{m'}(q') + A(q,i')P_q(q)P_{i'}(q') \\ & + A(q,d')P_q(q)P_{d'}(q')). \end{aligned} \tag{5}$$

Now let us cautiously proceed, by considering a pair of paths where one of the paths has exactly one self-loop transition in the end, and the other path has at least one self-loop transition in the end. The probability – that we surprisingly call $A_1(q, q')$ – of getting to q and q' along such paths while generating the same sequences is the probability of getting to q and q' along paths that do not both have a self-loop transition in the end, multiplied by the joint probability of independently choosing the self-loop transitions, and the probability of q and q' emitting the same symbols. But this is just

$$A_1(q, q') = rA_0(q, q'), \tag{6}$$

where

$$r = pP_q(q)P_{q'}(q') \tag{7}$$

> is the probability of independently choosing the self-loop transitions and emitting the same symbols in q and q'. Similarly we can define $A_k(q, q')$, and by induction it is easily proven that
>
> $$A_k(q, q') = rA_{k-1}(q, q') = r^k A_0(q, q'). \tag{8}$$
>
> As any finite path ending in q or q' must have a finite number of self-loop transitions in the end, we get
>
> $$\begin{aligned} A(q, q') &= \sum_{k=0}^{\infty} A_k(q, q') \\ &= \sum_{k=0}^{\infty} r^k A_0(q, q') \\ &= \frac{1}{1-r} A_0(q, q'). \end{aligned} \tag{9}$$
>
> Despite the fact that there is an infinite number of cases to consider, we observe that the sum over the probabilities of all these cases comes out as a geometric series that can easily be computed.

Each of the entries of A pertaining to match- insert- and delete-states can thus be computed in constant time using the above equations. As for the start-states (denoted by s and s') we initialise $A(s, s')$ to 1 (as we have not started generating anything and the empty sequence is identical to itself). Otherwise, even though they do not generate any symbols, we will treat the start-states as generate states; this allows for choosing an initial sequence of delete-states in one of the models. The start-states are the only possible immediate predecessors for the first insert-states, and together with the first insert-states the only immediate predecessors of the first match- and delete-states; the equations for the entries indexed by any of these states can trivially be modified according to this. The end-states (denoted by e and e') do not emit any symbols and are thus akin to delete-states, and can be treated the same way.

The co-emission probability of M_1 and M_2 is the probability of being in the states e and e' and having independently generated the same sequences. This probability can be found by looking up $A(e, e')$. In the rest of this paper we will use $A(M_1, M_2)$ to denote the co-emission probability of M_1 and M_2.

As all entries of A can be computed in constant time, we can compute the co-emission probability of M_1 and M_2 in time $\mathrm{O}(n_1 n_2)$ where

n_i denotes the number of states in M_i. The straightforward space requirement is also $\mathrm{O}(n_1 n_2)$ but can be reduced to $\mathrm{O}(n_1)$ by a standard trick [11, Chapter 11].

4 Measures on hidden Markov Models

Based on the co-emission probability we define two metrics that hopefully, to some extent, express how similar the families of sequences represented by two hidden Markov models are. A problem with the co-emission probability is that the models having the largest co-emission probability with a specific model, M, usually will not include M itself, as shown by the following proposition.

Proposition 1 *Let M be a hidden Markov model and $p = \max\{P_M(s) \mid s \in \Sigma^*\}$. The maximum co-emission probability with M attainable for any hidden Markov model is p. Furthermore, the hidden Markov models attaining this co-emission probability with M, are exactly those models, M', for which $P_{M'}(s) > 0 \Leftrightarrow P_M(s) = p$ for all $s \in \Sigma^*$.*

Proof. Let M' be a hidden Markov model with $P_{M'}(s) > 0 \Leftrightarrow P_M(s) = p$. Then

$$\sum_{s \in \Sigma^*, P_M(s) = p} P_{M'}(s) = 1 \tag{10}$$

and thus the co-emission probability of M and M' is

$$\sum_{s \in \Sigma^*} P_M(s) P_{M'}(s) \;=\; \sum_{s \in \Sigma^*, P_M(s) = p} P_M(s) P_{M'}(s) \;=\; p. \tag{11}$$

Now let M' be a hidden Markov model with $P_{M'}(s') = p' > 0$ for some $s' \in \Sigma^*$ with $P_M(s') = p'' < p$. Then the co-emission probability of M and M' is

$$\begin{aligned} \sum_{s \in \Sigma^*} P_M(s) P_{M'}(s) &= p'p'' + \sum_{s \in \Sigma^* \setminus \{s'\}} P_M(s) P_{M'}(s) \\ &\leq p'p'' + (1 - p')p \\ &< p. \end{aligned} \tag{12}$$

This proves that a hidden Markov model, M', has maximum co-emission probability, p, with M, if and only if the assertion of the proposition is fulfilled. □

Proposition 1 indicates that the co-emission probability of two models not only depends on how alike they are, but also on how 'self-confident' the models are, that is, to what extent the probabilities are concentrated to a small subset of all possible sequences.

Another way to explain this undesirable property of the co-emission probability, is to interpret hidden Markov models – or rather the probability distribution over finite sequences of hidden Markov models – as vectors in the infinite dimensional space spanned by all finite sequences over the alphabet. With this interpretation the co-emission probability, $A(M_1, M_2)$, of two hidden Markov models, M_1 and M_2, simply becomes the inner product,

$$\langle M_1, M_2 \rangle = |M_1||M_2| \cos v, \tag{13}$$

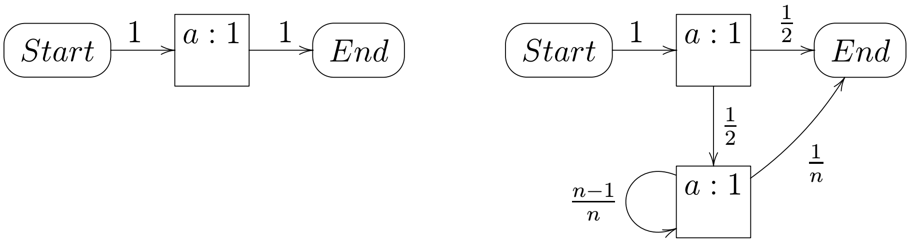
of the models. In the expression on the right hand side, v is the angle between the models – or vectors – and $|M_i| = \sqrt{\langle M_i, M_i \rangle}$ is the length of M_i. One observes the direct proportionality between the co-emission probability and the length (or 'self-confidence') of the models being compared. If the length is to be completely ignored, a good measure of the distance between two hidden Markov models would be the angle between them – two models are orthogonal, if and only if they can not generate identical sequences, and parallel (actually identical as the probabilities have to sum to 1) if they express the same probability distribution. This leads to the definition of our first metric on hidden Markov models.

Definition 1 *Let M_1 and M_2 be two hidden Markov models, and let $A(M, M')$ denote the co-emission probability of two hidden Markov models M and M'. We define the* angle *between M_1 and M_2 as*

$$D_{angle}(M_1, M_2) = \arccos\left(A(M_1, M_2) \Big/ \sqrt{A(M_1, M_1) A(M_2, M_2)} \right).$$

Having introduced the vector interpretation of hidden Markov models, another obvious metric to consider is the standard metric on vector spaces, that is, the (euclidian) norm of the difference between the two vectors

$$|M_1 - M_2| = \sqrt{\langle M_1 - M_2, M_1 - M_2 \rangle}. \tag{14}$$



(a) Hidden Markov model M_1 with $P_{M_1}(a) = 1$.

(b) Hidden Markov model M_2 with $P_{M_2}(a) = 1/2$ and $P_{M_2}(a^k) = \frac{1}{2n}(\frac{n-1}{n})^{k-2}$ for $k > 1$.

Figure 2: Two distinctly different models can have an arbitrarily small distance in the D_{angle} metric. It is easy to see that $A(M_1, M_1) = 1$, $A(M_1, M_2) = 1/2$ and $A(M_2, M_2) = 1/4 + 1/(8n - 4)$; for $n \to \infty$ one thus obtains $D_{\text{angle}}(M_1, M_2) \to 0$ but $D_{\text{diff}}(M_1, M_2) \to 1/2$.

Considering the square of this, we obtain

$$\begin{aligned} |M_1 - M_2|^2 &= \langle M_1 - M_2, M_1 - M_2 \rangle \\ &= \sum_{s \in \Sigma^*} (P_{M_1}(s) - P_{M_2}(s))^2 \\ &= \sum_{s \in \Sigma^*} \big(P_{M_1}(s)^2 + P_{M_2}(s)^2 - 2P_{M_1}(s)P_{M_2}(s)\big) \\ &= A(M_1, M_1) + A(M_2, M_2) - 2A(M_1, M_2). \end{aligned} \tag{15}$$

Thus this norm can be computed based on co-emission probabilities, and we propose it as a second choice for a metric on hidden Markov models.

Definition 2 *Let M_1 and M_2 be two hidden Markov models, and $A(M, M')$ be the co-emission probability of M and M'. We define the* difference *between M_1 and M_2 as*

$$D_{\mathit{diff}}(M_1, M_2) = \sqrt{A(M_1, M_1) + A(M_2, M_2) - 2A(M_1, M_2)}.$$

One problem with the D_{diff} metric is that $|M_1| - |M_2| \leq D_{\text{diff}}(M_1, M_2) \leq |M_1| + |M_2|$. If $|M_1| \gg |M_2|$ we therefore get that $D_{\text{diff}}(M_1, M_2) \approx |M_1|$, and we basically only get information about the length of M_1 from D_{diff}.

The metric D_{angle} is not prone to this weakness, as it ignores the length of the vectors and focuses on the sets of most probable sequences in the two models and their relative probabilities. But this metric can also lead to undesirable situations, as can be seen from figure 2 which shows that D_{angle} might not be able to discern two clearly different models. Choosing what metric to use, depends on what kind of differences one wants to highlight.

For some applications one might want a similarity measure instead of a distance measure. Based on the above metrics or the co-emission probability one can define a variety of similarity measures. We decided to examine the following two similarity measures.

Definition 3 *Let M_1 and M_2 be two hidden Markov models and $A(M, M')$ be the co-emission probability of M and M'. We define the* similarity *between M_1 and M_2 as*

$$\begin{aligned} S_1(M_1, M_2) &= \cos\left(D_{angle}(M_1, M_2)\right) \\ &= A(M_1, M_2) \Big/ \sqrt{A(M_1, M_1)A(M_2, M_2)} \end{aligned}$$

and

$$S_2(M_1, M_2) = 2A(M_1, M_2)\,/(A(M_1, M_1) + A(M_2, M_2))\ .$$

One can easily prove that these two similarity measures possess the following nice properties.

1. $0 \ \le\ S_i(M_1, M_2)\ \le\ 1$.
2. $S_i(M_1, M_2) = 1$ if and only if $\forall s \in \Sigma^* : P_{M_1}(s) = P_{M_2}(s)$.
3. $S_i(M_1, M_2) = 0$ if and only if $\forall s \in \Sigma^* : P_{M_i}(s) > 0 \Rightarrow P_{M_{3-i}}(s) = 0$, that is, there are no sequences that can be generated by both M_1 and M_2.

The only things that might not be immediately clear are that S_2 satisfies properties 1 and 2. This however follows from

$$A(M_1, M_1) + A(M_2, M_2) - 2A(M_1, M_2) = \sum_{s\in\Sigma^*} (P_{M_1}(s) - P_{M_2}(s))^2, \quad (16)$$

cf. equation 15, wherefore $2A(M_1, M_2) \le A(M_1, M_1) + A(M_2, M_2)$, and equality only holds if for all sequences their probabilities in the two models are equal.

5 Other types of hidden Markov models

Profile hidden Markov models are not by far the only type of hidden Markov models used in computational biology. Other types of hidden Markov models have been constructed for e.g. gene prediction [14] and recognition of trans-membrane proteins [20].We observe that the properties of the metrics and similarity measures introduced in the previous section do not depend on the structure of the underlying models, so once we can compute the co-emission probability of two models, we can also compute the distance between and similarity of the two models. The question thus is, can our method be extended to compute the co-emission probability for other types of hidden Markov models too?

The first thing one can observe, is that the only feature of the underlying structure of profile hidden Markov models we use, is that they are left-right models, i.e. we can number the states such that if there is a transition from state i to state j then $i \leq j$ (if the inequality is strict, that is $i < j$, then we do not even need the geometric sequence calculation, and the calculation of the co-emission probability reduces to a calculation similar to the forward/backward calculations [6, Chapter 3]). For all left-right hidden Markov models, e.g. profile hidden Markov models extended with free insertion modules [3, 12], we can thus use recursions similar to those specified in section 3 to compute the co-emission probability.

With some work the method can even be extended to all hidden Markov models where each state is part of at most one cycle, even if this cycle consists of more than the one state of the self-loop case. We will denote such models as hidden Markov models with only simple cycles. This extension can be useful when comparing models of coding DNA, that will often contain cycles with three states, or models describing a variable number of small domains. For general hidden Markov models we will have to resort to approximating the co-emission probability. In the rest of this section we will describe these two generalisations.

5.1 Hidden Markov models with only simple cycles

Assume that we can split M and M' into a number of disjoint cycles and single nodes, $\{C_i\}_{i \leq k}$ and $\{C'_i\}_{i \leq k'}$, such that $\{C_i\}$ and $\{C'_i\}$ are topologically sorted, i.e. for $p \in C_i$ ($p' \in C'_i$) and $q \in C_j$ ($q' \in C'_j$) and $i < j$ there is no path from q to p in M (from q' to p' in M'). To compute the co-emission probability of M and M', we will go from considering pairs of single nodes to considering pairs of cycles, i.e. we look at all nodes in a cycle at the same time.

Let C_i and $C'_{i'}$ be cycles[2] in M and M' respectively. Assume that we have already computed the co-emission probability, $A(q, q')$, for all pairs of nodes, q, q', where $q \in C_j$, $q' \in C'_{j'}$, $j \leq i$, $j' \leq i'$ and $(i, i') \neq (j, j')$. We will now describe how to compute the co-emission probability, $A(p, p')$, for all pairs of nodes, p, p', with $p \in C_i$ and $p' \in C'_{i'}$.

As with the profile hidden Markov models, cf. section 3, we will proceed in a step by step fashion. We start by restricting the types of paths we consider, to get some intermediate results; we then expand the types of paths allowed – using the intermediate results – until we have covered all possible paths.

The first types of paths we consider are paths, π and π', generating identical sequences that ends in p and p', but where the immediate predecessor of p on π is not in C_i, or the immediate predecessor of p' on π' is not in $C'_{i'}$. We will denote the co-emission probability at p, p' of paths of this type as $A_e(p, p')$, as it covers the co-emission probability of paths entering the pair of cycles, $C_i, C'_{i'}$, at p, p'; it can easily be computed as

$$A_e(q, q') = \sum_{\substack{r \to q, r' \to q' \\ (r, r') \notin C_i \times C'_{i'}}} P_r(q) P_{r'}(q') A(r, r') \sum_{\sigma \in \Sigma} P_q(\sigma) P_{q'}(\sigma), \qquad (17)$$

where $r \to q$ ($r' \to q'$) denotes that there is a transition from r to q in M (from r' to q' in M'). Here we assume that both q and q' are non-silent states; if both are silent, the sum over all symbols factor, $\sum_{\sigma \in \Sigma} P_q(\sigma) P_{q'}(\sigma)$ (the probability that q and q' generates identical symbols), should be omitted, and if one is silent and the other non-silent, the sum should furthermore only be over non-C_i (or non-$C'_{i'}$) predecessors of the silent state.

Before we proceed further, we will need some definitions that allow us to talk about successors of states and successors of pairs of states in $C_i, C'_{i'}$, and some related probabilities.

Definition 4 *Let $q \in C_i$ ($q' \in C'_{i'}$). The* successor *of q in C_i (q' in $C'_{i'}$) is the unique state $r \in C_i$ ($r' \in C'_{i'}$) for which there is a transition from q to r (from q' to r').*

The uniqueness of the successor follows from the requirement that the models only have simple cycles. For successors of pairs of states things

[2] If C_i or $C'_{i'}$ is not a cycle but a single node, the calculations of the co-emission probabilities pertaining to pairs of nodes from C_i and $C'_{i'}$ trivialises to calculations similar to equation 17 below.

are a little bit more complicated, as we want the successor of a pair to be the unique pair to which we can get to, generating the same number of symbols (zero or one) using one transition in one or both models. This is captured by definition 5.

Definition 5 *Let $q \in C_i$ and $q' \in C'_{i'}$. The* successor *of q, q' in $C_i, C'_{i'}$,* $\mathrm{suc}(q, q')$*, is the pair of states r, r' where*

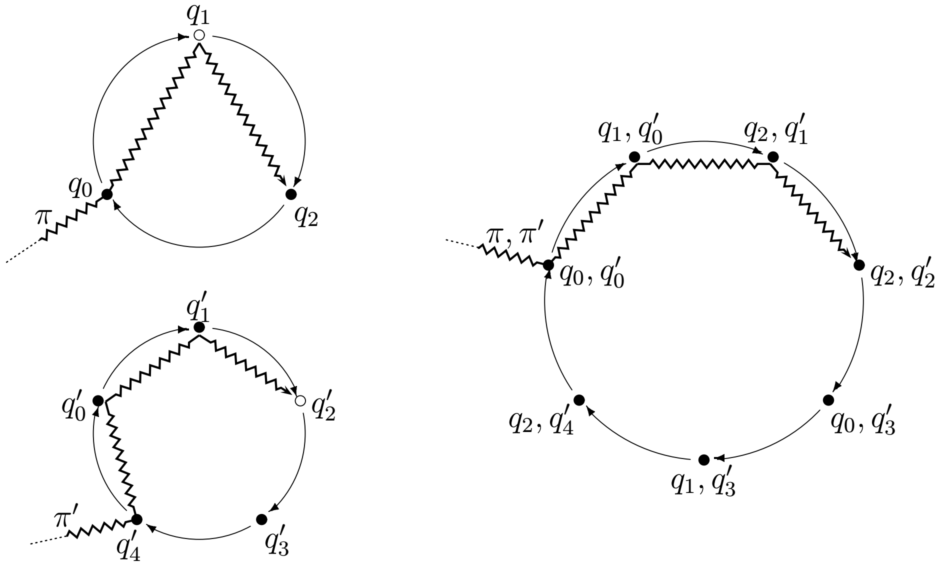
- *if the successor of q in C_i is silent or the successor of q' in $C'_{i'}$ is non-silent, then r is the successor of q; otherwise $r = q$.*
- *if the successor of q' in $C'_{i'}$ is silent or the successor of q in C_i is non-silent, then r' is the successor of q'; otherwise $r' = q'$.*

By this definition the successor of a pair of states, q, q', is the pair of successors of q and q' if both successors are silent or both successors are non-silent states. If the successor of q is a non-silent state and the successor of q' is a silent state then the successor of q, q' is the pair consisting of q and the successor of q'.

We will use $P_{q,q'}(\mathrm{suc}(q, q'))$ to denote the probability of getting from q, q' to $\mathrm{suc}(q, q')$ generating identical symbols. If $r, r' = \mathrm{suc}(q, q')$ are both non-silent, then $P_{q,q'}(r, r') = P_q(r)P_{q'}(r') \sum_{\sigma \in \Sigma} P_r(\sigma)P_{r'}(\sigma)$; if one or both are silent, the sum over all symbols factor, $\sum_{\sigma \in \Sigma} P_r(\sigma)P_{r'}(\sigma)$, should be omitted, and if only r (r') is silent, the $P_{q'}(r')$ factor ($P_q(r)$ factor) should furthermore be omitted as $q' = r'$ (as $q = r$).

More generally we will use $P_{q,q'}(r, r')$, where $q, r \in C_i$ and $q', r' \in C'_{i'}$, to denote the probability of getting from q, q' to r, r' generating identical sequences without cycling, i.e. by just starting in q, q' and going through successors until we reach r, r' the first time. We resolve the ambiguity of the meaning of $P_{q,q'}(q, q')$ by setting $P_{q,q'}(q, q') = 1$. To ease notation in the following, we furthermore define $P'_{q,q'}(r, r') = P_{q,q'}(\mathrm{suc}(q, q'))P_{\mathrm{suc}(q,q')}(r, r')$. The probability $P'_{q,q'}(q, q')$ is thus the probability of going through one full cycle of successors to q, q' until we are back at q, q'; if $r, r' \neq q, q'$ then $P'_{q,q'}(r, r') = P_{q,q'}(r, r')$.

One can observe that r, r' might not be anywhere in the sequence of successors starting at q, q'. If we can get to r, r' from q, q' going through consecutive successors, then there is a pair of paths from q to r and from q' to r', respectively, generating an equal number of symbols. Such a pair of paths does not necessarily exist. E.g. assume there is an even number of non-silent states in both C_i and $C'_{i'}$, and that the successor, r, of q in C_i is non-silent. Any path that starts in q and ends in r will generate



(a) Two example cycles, $C_i = \{q_0, q_1, q_2\}$ and $C'_{i'} = \{q'_0, q'_1, q'_2, q'_3, q'_4\}$. Hollow circles denote silent states and filled circles denote non-silent states.

(b) The cycle of the class of pairs in $C_i \times C'_{i'}$ containing q_0, q'_0.

Figure 3: An example of a pair of cycles in M and M' and one of the induced cycles of pairs. A path, π, ending in q_2 in C_i and a path, π', ending in q'_2 in $C'_{i'}$ are shown with zigzagged lines. If we assume that the two paths generate identical sequences, then the co-emission path, π, π', ends in q_2, q'_2 in $C_i, C'_{i'}$. Though π' enters $C'_{i'}$ at q'_4, the co-emission path, π, π', enters $C_i, C'_{i'}$ at q_0, q'_0, as the first symbol in the sequence generated by π and π' that is generated by states in both C_i and $C'_{i'}$, the second last symbol of the sequence, is generated by q_0 and q'_0 respectively.

an uneven number of symbols, while any path starting and ending in $q' \in C'_{i'}$ will generate an even number of symbols. It is thus impossible to get from q, q' to r, q' going through successors.

More formally, let $d_q(r)$ (resp. $d_{q'}(r')$) denote the number of non-silent states we go through going from q to r in C_i (from q' to r' in $C'_{i'}$), and let h (h') denote the total number of non-silent states in C_i (in $C'_{i'}$). Then by similar reasoning as in the above, we can get from q, q' to r, r' going through successors if and only if $d_q(r) \equiv d_{q'}(r') \mod \gcd(h, h')$. It is evident that we can always get back to q, q' when starting in q, q', and thus the pairs of states from $C_i, C'_{i'}$ can be partitioned into cycles of consecutive successors, cf. figure 3. If it is possible to get from q, q' to p, p' generating an equal number of symbols, i.e. q, q' and p, p' are in the same cycle of pairs, we will say that q, q' and p, p' belong to the same *class*, as the partition of pairs in this manner is actually a partition into equivalence classes.

We are now ready to compute the probability of getting simultaneously to p and p' having generated identical sequences, without having been simultaneously in p and p' previously on the paths. This is

$$A_0(p,p') = \sum_{\substack{q,q' \text{ belongs to the} \\ \text{same class as } p,p'}} A_e(q,q')P_{q,q'}(p,p') \tag{18}$$

as we sum over all possible pairs, q, q', where paths ending in p, p' can have entered $C_i, C'_{i'}$. It is similar to $A_0(p, p')$ for profile hidden Markov models in the sense, that it is the probability of reaching p and p' having generated identical sequences without having looped through p, p' previously.

To compute the A_0 entries efficiently for all pairs of states in a class, we exploit the fact that $P_{q,q'}(p, p') = P_{q,q}(\text{suc}(q, q'))P_{\text{suc}(q,q')}(p, p')$ (for $q, q' \neq p, p'$); we can thus compute $A_0(p, p')$ in an incremental way, starting at the successor of p, p' and going through the cycle of successors, adding the A_e values and multiplying by the probability of getting to the next successor. Furthermore, as

$$\begin{aligned} A_0(p,p')&P_{p,p'}(\text{suc}(p,p')) + A_e(\text{suc}(p,p')) \\ &= \sum_{\substack{q,q' \text{ belongs to the} \\ \text{same class as } p,p'}} A_e(q,q')P_{q,q'}(p,p')P_{p,p'}(\text{suc}(p,p')) \\ &\qquad\qquad + A_e(\text{suc}(p,p'))P_{\text{suc}(p,p')}(\text{suc}(p,p')) \\ &= A_0(\text{suc}(p,p')) + A_e(\text{suc}(p,p'))P'_{\text{suc}(p,p')}(\text{suc}(p,p')) \end{aligned} \tag{19}$$

we do not need to start from scratch when computing A_0 for the other pairs that belong to the same class as p, p' – which would require time proportional to the square of the number of pairs in the class – but can reuse $A_0(p, p')$ to compute $A_0(\text{suc}(p, p'))$ in constant time. Finally we observe that

$$A(p,p') = \sum_{i=0}^{\infty} P'_{p,p'}(p,p')^i A_0(p,p') = \frac{1}{1 - P'_{p,p'}(p,p')} A_0(p,p') \tag{20}$$

Algorithm 1 Computation of the co-emission probabilities at all pairs of states that are in the same class as p, p'.

```
$q, q' = p, p'$
$AccumulatedP = A_e(p, p')$
$r = 1$
while $\mathrm{suc}(q, q') \neq p, p'$ do
    $AccumulatedP = AccumulatedP \cdot P_{q,q'}(\mathrm{suc}(q, q')) + A_e(\mathrm{suc}(q, q'))$
    $r = r \cdot P_{q,q'}(\mathrm{suc}(q, q'))$
    $q, q' = \mathrm{suc}(q, q')$
end while
$r = r \cdot P_{q,q'}(\mathrm{suc}(q, q'))$
repeat /* $AccumulatedP = A_0(q, q')$ and $r = P'_{q,q'}(q, q')$ */
    $A(q, q') = AccumulatedP \cdot \frac{1}{1-r}$
    $AccumulatedP$
            $= AccumulatedP \cdot P_{q,q'}(\mathrm{suc}(q, q')) + (1 - r) \cdot A_e(\mathrm{suc}(q, q'))$
    $q, q' = \mathrm{suc}(q, q')$
until $\mathrm{suc}(q, q') = p, p'$
```

and

$$P'_{p,p'}(p, p') = P'_{q,q'}(q, q') \tag{21}$$

for all q, q' that belong to the same class as p, p'. This allows us to formulate algorithm 1 for computing the co-emission probability at all pairs in a cycle.

It is an easy observation that we run through all pairs of the class twice – once in the *while*-loop and once in the *repeat*-loop – thus using time proportional to the number of pairs in the class to compute the co-emission probabilities at each pair. Therefore, the overall time for handling the entries pertaining to the pair of cycles, $C_i, C'_{i'}$, is $\mathrm{O}(|C_i||C'_{i'}|)$ once we have computed the A_e entries; thus the time used to compute the co-emission probability of two hidden Markov models with only simple cycles is proportional to the product of the number of transitions in the two models. This is comparable to the complexity of $\mathrm{O}(n_1 n_2)$ for profile hidden Markov models, as this result relied on there only being a constant number of transitions to each state. In general we can compute the co-emission probability of two hidden Markov models, M_1 and M_2, with only simple cycles – including left-right hidden Markov models – in time $\mathrm{O}(m_1 m_2)$, where m_i denotes the number of transitions in M_i.

5.2 General hidden Markov models

For more complex hidden Markov models, let us examine what is obtained by iterating the calculations. Let $A'_i(q, q')$ be the value computed for entry (q, q') in the i'th iteration. If we assume that q and q' are either both silent or both non-silent states, then we can compute the new entry for (q, q') as

$$A'_{i+1}(q, q') = p \sum_{\substack{r \to q \\ r' \to q'}} A'_i(r, r') P_r(q) P_{r'}(q'), \tag{22}$$

where p is as defined in equation 4 if q and q' are non-silent states, and is 1 if q and q' are silent states. If q and q' are of different types, the summation should only be over the predecessors of the silent state as in equation 3. In each iteration we thus extend co-emission paths with one pair of states, and $A'_i(q, q')$ is the probability of getting to q, q' having generated identical sequences on a co-emission path of length i.

The resemblance of this iterated computation to the previous calculation of A_i is evident, but a well-known mathematical sequence is not easily recognisable in equation 22. Instead we observe that $A'_i(q, q')$ holds the probability of being in states q and q' and having generated identical prefixes in the two models after i iterations. If we assume that the only transitions from the end-states are self-loops with probability 1 (this makes the $A'_i(e, e')$ entry accumulate the probabilities of generating identical sequences after at most i iterations), then

$$A'_i(e, e') \;\le\; A(M_1, M_2) \;\le \sum_{q \in M_1,\, q' \in M_2} A'_i(q, q') \tag{23}$$

where $A(M_1, M_2)$ is the true co-emission probability of M_1 and M_2. This follows from the fact, that to generate identical sequences we must either already have done so, or at least have generated identical prefixes so far.

Now assume that for any two states, we can choose transitions to non-silent states (or the end-states) and emit different symbols with probability at least $1 - c$ where $c < 1$. Then the total weight with which $A'_i(q, q')$ contributes to the entries – not counting the special (e, e') entry – of A'_{i+1} is at most c. Thus

$$\sum_{\substack{q \in M_1,\, q' \in M_2 \\ (q,q') \neq (e,e')}} A'_{i+1}(q, q') \le c \sum_{\substack{q \in M_1,\, q' \in M_2 \\ (q,q') \neq (e,e')}} A'_i(q, q') \tag{24}$$

and by induction we get

$$\sum_{q \in M_1,\, q' \in M_2} A'_i(q,q') - A'_i(e,e') \;=\; \sum_{\substack{q \in M_1,\, q' \in M_2 \\ (q,q') \neq (e,e')}} A'_i(q,q') \;\leq\; c^i, \qquad (25)$$

which shows that the iteration method approximates the co-emission probability exponentially fast.

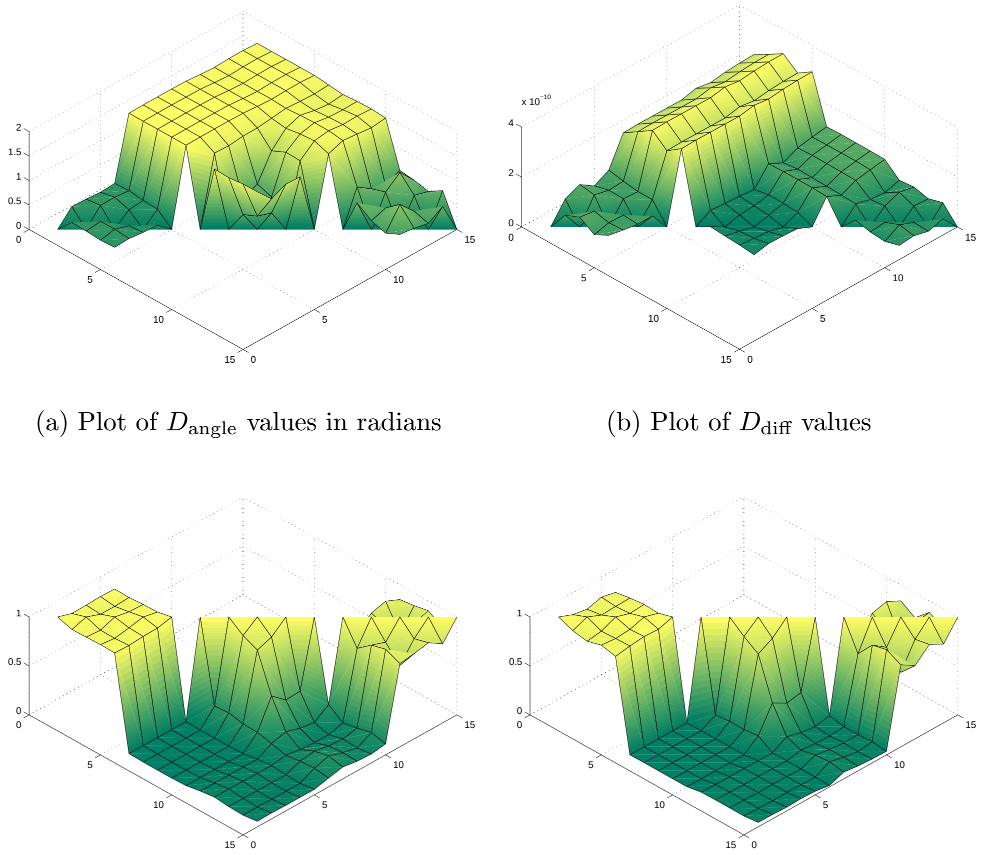
Though our assumption about the non-zero probability of choosing transitions and emissions such that we generate different symbols in the two models is valid for most, if not all, hidden Markov models used in practice, it is not even necessary. If d is the minimum number of paired transitions we have to follow from q and q' to get to the end-states[3] or states where we can emit different symbols after having generated identical prefixes, and c' is the probability of staying on this path and emit different symbols, we still get the exponential approximation of equation 25 with $c = (c')^{1/d}$. By these arguments we can approximate the co-emission probabilities and thus the metrics and similarity measures presented in section 4 of arbitrary hidden Markov models exponentially fast.

6 Results

We have implemented the method described in the previous sections for computing the co-emission probabilities of two left-right models. The program furthermore computes the derived measures and is currently available at `www.brics.dk/∼cstorm/hmmcomp`. The program was used to test the four measures in a comparison of hidden Markov models for three classes of secretory signal peptides – cleavable N-terminal sequences which target secretory proteins for translocation over a membrane.

Signal peptides do not have a well-defined consensus motif, but they do share a common structure: an N-terminal region with a positive charge, a stretch of hydrophobic residues, and a region of more polar regions containing the cleavage site, where two positions are partially conserved [22]. There are statistical differences between prokaryotic and eukaryotic signal peptides concerning the length and composition of these

[3] The end-states ensures that d exists – if we can not get to e and e', then we can not pass through q and q' and generate identical sequences. Therefore we may just as well ignore the (q,q') entry.



(a) Plot of D_{angle} values in radians

(b) Plot of D_{diff} values

(c) Plot of S_1 values

(d) Plot of S_2 values

Figure 4: Plots of the results obtained with the different measures. Models 1 through 5 are the models trained on eukaryotic sequences, models 6 through 10 are the models trained on Gram-positive bacterial sequences, and models 11 through 15 are the models trained on Gram-negative bacterial sequences. This gives 9 blocks, each of 25 entries, of different pairs of groups of organisms compared, but as all the measures are symmetric we have left out half the blocks showing comparisons between different groups of organisms. This should increase clarity, as no parts of the plots are hidden behind peaks.

regions [23, 17], but the distributions overlap, and in some cases, eukaryotic and prokaryotic signal peptides are found to be functionally interchangeable [4].

	Euk	G_{pos}	G_{neg}
Euk	0.231	1.56	1.52
G_{pos}		0.864	1.47
G_{neg}			0.461

(a) Table of D_{angle} values

	Euk	G_{pos}	G_{neg}
Euk	$6.77 \cdot 10^{-11}$	$2.56 \cdot 10^{-10}$	$2.67 \cdot 10^{-10}$
G_{pos}		$1.95 \cdot 10^{-11}$	$9.09 \cdot 10^{-11}$
G_{neg}			$4.43 \cdot 10^{-11}$

(b) Table of D_{diff} values

	Euk	G_{pos}	G_{neg}
Euk	0.967		
G_{pos}	$1.06 \cdot 10^{-2}$	0.547	
G_{neg}	$4.74 \cdot 10^{-2}$	0.102	0.866

(c) Table of S_1 values

	Euk	G_{pos}	G_{neg}
Euk	0.955		
G_{pos}	$1.78 \cdot 10^{-3}$	0.511	
G_{neg}	$2.93 \cdot 10^{-2}$	$4.78 \cdot 10^{-2}$	0.839

(d) Table of S_2 values

Figure 5: Tables of the average values of each block plotted in figure 4. The empty entries corresponds to the blocks left out in the plots.

The hidden Markov model used here is not a profile HMM, since signal peptides of different proteins are not necessarily related, and therefore do not constitute a sequence family that can be aligned in a meaningful way. Instead, the signal peptide model is composed of three region models, each having a characteristic amino acid composition and length distribution, plus seven states modelling the cleavage site – see Nielsen and Krogh [18] for a detailed description. A combined model with three branches was used to distinguish between signal peptides, signal anchors (a subset of transmembrane proteins), and non-secretory proteins; but only the part modelling the signal peptide plus the first few positions after the cleavage site has been used in the comparisons reported here.

The same architecture was used to train models of three different signal peptide data sets: eukaryotes, Gram-negative bacteria (with a double membrane), and Gram-positive bacteria (with a single membrane). For cross-validation of the predictive performance, each model was trained on five different training/test set partitions, with each training set comprising 80% of the data – i.e., any two training sets have 75% of the sequences in common.

The comparisons of the models are shown in figures 4 and 5. In general, models trained on cross-validation sets of the same group are more similar than models trained on data from different groups, and the two groups of bacteria are more similar to one another than to the eukaryotes.

However, there are some remarkable differences between the measures. According to D_{diff}, the two bacterial groups are almost as similar as the cross-validation sets, but according to D_{angle} and the similarity measures, they are almost as dissimilar as the bacterial/eukaryotic comparisons.

This difference actually reflects the problem with the D_{diff} measure discussed in section 4. The distribution of sequences for models trained on eukaryotic data are longer in the vector interpretation, i.e. the probabilities are more concentrated, than the distributions for models trained on bacterial data. What we mainly see in the D_{diff} values for bacterial/eukaryotic comparisons is thus the length of the eukaryotic models. This reflects two properties of eukaryotic signal peptides: they have a more biased amino acid composition in the hydrophobic region that comprises a large part of the signal peptide sequence; and they are actually *shorter* than their bacterial counterparts, thus raising the probability of the most probable sequences generated by this model.

D_{angle} also shows that the differences within groups are larger in the Gram-positive group than in the others. This may simply reflect the smaller sample size in this group (172 sequences vs. 356 for the Gram-negative bacteria and 1137 for the eukaryotes).

The values of D_{angle} in between-group comparisons are quite close to the maximal $\pi/2$. Thus the distributions over sequences for models of different groups are close to being orthogonal. This might seem surprising in the light of the reported examples of functionally interchangeable signal peptides; but it does not mean that no sequences can be generated by both eukaryotic and bacterial models, only that these sequences have low probabilities compared to those that are unique for one group. In other words: if a random sequence is generated from one of these models, it may with a high probability be identified which group of organisms it belongs to.

7 Discussion

Recall that the co-emission probability is defined as the probability that two hidden Markov models, M_1 and M_2, generate *completely identical* sequences, i.e. as $\sum_{s_1,s_2\in\Sigma} P_{M_1}(s_1)P_{M_2}(s_2)$ where $s_1 = s_2$. One problem with the co-emission probability – and measures based on it – is that it can be desirable to allow sequences to be slightly different. One might thus want to loosen the restriction of "$s_1 = s_2$" to, e.g., "s_1 is a substring (or subsequence) of s_2," or even "$|s_1| = |s_2|$" ignoring the symbols of the sequences and just comparing the length distributions of the two models.

Another approach is to take the view that the two hidden Markov models do not generate independent sequences, but instead generates alignments with two sequences. Inspecting the equations for computing the co-emission probability, one observes that we require that when one model emits a symbol the other model should emit an identical symbol. This corresponds to only allowing columns with identical symbols in the produced alignments. A less restrictive approach would be to allow other types of columns, i.e. columns with two different symbols or a symbol in only one of the sequences, and weighting a column according to the difference it expresses. The modifications proposed in the previous paragraph can actually be considered special cases of this approach. Our method for computing the co-emission probability can easily be modified to encompass these types of modifications.

Acknowledgements

This work was inspired by a talk by Xiaobing Shi on his work with David States on aligning profile hidden Markov models. The authors would like to thank Bjarne Knudsen and Jotun Hein for their valuable suggestions. Finally we would like to thank Anders Krogh for his help with the software used to train and parse the models. Christian N. S. Pedersen and Henrik Nielsen are supported by the Danish National Research Foundation.

References

[1] R. Bakis. Continuous speech word recognition via centisecond acoustic states. In *Proceedings of the ASA Meeting*, April 1976.

[2] P. Baldi, Y. Chauvin, T. Hunkapiller, and M. A. McClure. Hidden markov models of biological primary sequence information. In *Proceedings of the National Academy of Science, USA*, volume 91, pages 1059–1063, 1994.

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

[4] S. A. Benson, M. N. Hall, and T. J. Silhavy. Genetic analysis of protein export in *Escherichia coli* K12. *Annual Review of Biochemistry*, 54:101–134, 1985.

[5] G. A. Churchill. Stochastic models for heterogeneous DNA sequences. *Bulletin of Mathematical Biology*, 51:79–94, 1989.

[6] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probalistic Models of Proteins and Nucleic Acids.* Cambridge University Press, 1998.

[7] S. R. Eddy. Hidden markov models. *Current Opinion in Structurel Biology*, 6:361–365, 1996.

[8] S. R. Eddy. Profile hidden markov models. *Bioinformatics*, 14:755–763, 1998.

[9] O. Gotoh. Optimal alignment between groups of sequences and its application to multiple sequence alignment. *Computer Applications in the Biociences*, 9(3):361–370, 1993.

[10] M. Gribskov, A. D. McLachlan, and D. Eisenberg. Profile analysis: Detection of distantly related proteins. In *Proceedings of the National Academy of Science, USA*, volume 84, pages 4355–4358, 1987.

[11] D. Gusfield. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology.* Cambridge University Press, 1997.

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

[13] F. Jelinek. Continuous speech recognition by statistical methods. In *Proceedings of the IEEE*, volume 64, pages 532–536, April 1976.

[14] A. Krogh. Two methods for improving performance of an HMM and their application for gene finding. In *Proceedings of the 5th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 179–186, 1997.

[15] A. Krogh, M. Brown, I. S. Mian, K. Sjolander, and D. Haussler. Hidden markov models in computational biology: Applications to protein modeling. *Journal of Molecular Biology*, 235:1501–1531, 1994.

[16] R. Luthy, J. U. Bowie, and D. Eisenberg. Assessment of protein models with three-dimensional profiles. *Nature*, 356:83–85, 1992.

[17] H. Nielsen, S. Brunak, J. Engelbrecht, and G. von Heijne. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Engineering*, 10:1–6, 1997.

[18] H. Nielsen and A. Krogh. Prediction of signal peptides and signal anchors by a hidden Markov model. In *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 122–130, 1998.

[19] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 277–286, 1989.

[20] E. L. L. Sonnhammer, G. von Heijne, and A. Krogh. A hidden Markov model for predicting transmembrane helices in protein sequences. In *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 175–182, 1998.

[21] W. R. Taylor. Identification of protein sequence homology by consessus template alignment. *Journal of Molecular Biology*, 188:233–258, 1986.

[22] G. von Heijne. Signal sequences. The limits of variation. *Journal of Molecular Biology*, 184:99–105, 1985.

[23] G. von Heijne and L. Abrahmsén. Species-specific variation in signal peptide design. *FEBS Letter*, 244:439–446, 1989.

Recent BRICS Report Series Publications

RS-99-6 Rune B. Lyngsø, Christian N. S. Pedersen, and Henrik Nielsen. *Measures on Hidden Markov Models*. February 1999. 27 pp. To appear in *Seventh International Conference on Intelligent Systems for Molecular Biology*, ISMB '99 Proceedings, 1999.

RS-99-5 Julian C. Bradfield and Perdita Stevens. *Observational Mu-Calculus*. February 1999. 18 pp.

RS-99-4 Sibylle B. Fröschle and Thomas Troels Hildebrandt. *On Plain and Hereditary History-Preserving Bisimulation*. February 1999. 21 pp.

RS-99-3 Peter Bro Miltersen. *Two Notes on the Computational Complexity of One-Dimensional Sandpiles*. February 1999. 8 pp.

RS-99-2 Ivan B. Damgård. *An Error in the Mixed Adversary Protocol by Fitzi, Hirt and Maurer*. February 1999. 4 pp.

RS-99-1 Marcin Jurdziński and Mogens Nielsen. *Hereditary History Preserving Simulation is Undecidable*. January 1999. 15 pp.

RS-98-55 Andrew D. Gordon, Paul D. Hankin, and Søren B. Lassen. *Compilation and Equivalence of Imperative Objects (Revised Report)*. December 1998. iv+75 pp. This is a revision of Technical Report 429, University of Cambridge Computer Laboratory, June 1997, and the earlier BRICS report RS-97-19, July 1997. Appears in Ramesh and Sivakumar, editors, *Foundations of Software Technology and Theoretical Computer Science: 17th Conference*, FST&TCS '97 Proceedings, LNCS 1346, 1997, pages 74–87.

RS-98-54 Olivier Danvy and Ulrik P. Schultz. *Lambda-Dropping: Transforming Recursive Equations into Programs with Block Structure*. December 1998. 55 pp. To appear in *Theoretical Computer Science*.

RS-98-53 Julian C. Bradfield. *Fixpoint Alternation: Arithmetic, Transition Systems, and the Binary Tree*. December 1998. 20 pp.

RS-98-52 Josva Kleist and Davide Sangiorgi. *Imperative Objects and Mobile Processes*. December 1998. 22 pp. Appears in Gries and de Roever, editors, *IFIP Working Conference on Programming Concepts and Methods*, PROCOMET '98 Proceedings, 1998, pages 285–303.