Quasi-consensus-based comparison of profile hidden Markov models for protein sequences

Robel Y. Kahsay1, Guoli Wang2, Guang Gao1, Li Liao1,3,* and Roland Dunbrack2,∗

1Delaware Biotechnology Institute, Newark, DE 19715, USA, 2Fox Chase Cancer Center, Philadelphia, PA 19111, USA and 3Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, USA

Received on August 26, 2004; revised on February 14, 2005; accepted on March 2, 2005
Advance Access publication March 29, 2005

ABSTRACT
A simple approach for the sensitive detection of distant relationships among protein families and for sequence–structure alignment via comparison of hidden Markov models based on their quasi-consensus sequences is presented. Using a previously published benchmark dataset, the approach is demonstrated to give better homology detection and yield alignments with improved accuracy in comparison to an existing state-of-the-art dynamic programming profile–profile comparison method. This method also runs significantly faster and is therefore suitable for a server covering the rapidly increasing structure database. A server based on this method is available at http://liao.cis.udel.edu/website/servers/modmod

Contact: roland.dunbrack@fccc.edu; lliao@mail.eecis.udel.edu

INTRODUCTION
As the number of available protein sequences increases at a much faster pace than their structure and functions can be experimentally determined, the development of more sensitive and accurate sequence analysis algorithms remains important. Evolutionary relationships implied by sequence similarity remain a major basis for computationally predicting structure and function. Despite the tremendous number of methods devised, the task of detecting distant homologies, those that share sequence similarity lower than 30%, and predicting accurate sequence–structure alignments is still quite a challenge.

Detection ability and sequence–alignment accuracy can be improved by using multiple sequence information in the form of a position-specific scoring matrix or profile (Gribskov et al., 1987; Altschul et al., 1997). In contrast to methods that try to establish homology based on similarity between a pair of protein sequences, sequence–profile methods compare an individual protein sequence to columns in the other profile with an appropriate gap-penalty scoring matrices, obtained by transforming a multiple sequence alignment into log-odds or frequency matrices of the 20 amino acids at each position in the alignment (Gribskov et al., 1987). The second category consists of hidden Markov models (HMMs) (Krogh et al., 1994; Eddy, 1998), which have a more complex architecture designed to model the observed multiple sequence alignment, including insertion–deletion regions, probabilistically. In both categories, a sequence–profile comparison method can work in both directions: a profile query against a sequence database or a sequence query against a profile database.

To exploit the information contained in profiles, profile–profile comparison methods have been developed that compare a query profile to a database of profiles for enhanced detection sensitivity (Pietrokovski, 1996; Rychlewski et al., 2000; Yona and Levitt, 2002; Sadreyev and Grishin, 2003; Wang and Dunbrack Jr, 2004). Most of these profile–profile methods start with defining a scoring scheme between columns of the two profiles. The similarity between two profiles is then measured as the total score when aligning columns in one profile to columns in the other profile with an appropriate gap-penalty scheme, and such an alignment can be found by using the same dynamic programming (DP) technique as used in Smith–Waterman local alignment for a pair of sequences (Smith and Waterman, 1981). Several recent papers have compared many of the scoring functions previously proposed (Mittelman et al., 2003; Edgar and Sjolander, 2004; Wang and Dunbrack Jr, 2004).

Because of the favorable properties of HMMs in comparison with matrix profiles (Durbin et al., 1998), it is desirable to develop methods for comparing HMMs with one another. A DP scheme to align states of two HMMs for protein families was described several years ago by Shi and States (http://www.ornl.gov/sci/techresources/Human_Genome/publicat/99santa/108.html) and applied to cluster the models in the PFAM database (Bateman et al., 2004). Another approach for comparing two HMMs based on their ‘co-emission’ probability has also been presented and applied to HMMs for signal peptides (Lyngso et al., 1999). Recently, Madera and Chothia have made the PRC program available on the SUPERFAMILY site (http://supfam.org) that compares HMMs with each other, although the details of this method are not currently published.

In this paper, we propose a simple, alternative approach for comparing two profile HMMs using their quasi-consensus sequences. These sequences are those derived by taking the most...
probable sequence from the highest-probability path through the HMM. This method provides rapid remote homology detection and alignment prediction.

METHODS

Consensus-based comparison of profile HMMs

A profile HMM for a protein family can be considered as a probability distribution over an infinite space of all possible amino acid strings that assigns higher probability for the members of the family. Therefore, comparing two profile HMMs amounts to comparing two distributions of probability. A conceptual approach to comparing two HMMs $M_i$ and $M_j$ is to consider them as (row) vectors in the infinite space of all sequences and define a metric that measures the similarity between the vectors. If we choose the components of these infinite dimension vectors to be the probabilities or scores assigned to all sequences by the models, the models can be then represented as

\[ M_i = \{s_i(x_1), s_i(x_2), \ldots, s_i(x_{\infty})\} \]

\[ M_j = \{s_j(x_1), s_j(x_2), \ldots, s_j(x_{\infty})\} \]

where $\{x_1, x_2, x_3, \ldots, x_{\infty}\}$ is an infinite set of all sequences in an arbitrarily preset order. Although represented as vectors, the similarity measure between these vectors is not computable in practice due to the vectors’ infinite size. To circumvent this problem, we consider a plausible heuristic in which we focus on comparing the two vectors by their most significant components, where the corresponding sequence receives a maximal score. Let us assume that $M_i$ peaks at $s_i(C_i)$ and $M_j$ peaks at $s_j(C_j)$, with $C_i$ and $C_j$ defined as the consensus sequences for $M_i$ and $M_j$ respectively. We consider two models to be highly similar if their respective consensus sequences are very probable in both models. Specifically, we compare profile HMMs by defining a similarity metric given as

\[ S_{ij} = M_i \cdot n_{C_j} + M_j \cdot n_{C_i} \]

\[ = s_i(C_j) + s_j(C_i) \quad (1) \]

where $n_{C_j}$ and $n_{C_i}$ are the unit vectors in the directions of the consensus sequences $C_i$ and $C_j$ of models $M_i$ and $M_j$ respectively. However, this similarity metric is based on raw scores and does not take the significance of the scores $s_i(C_j)$ and $s_j(C_i)$ into consideration. One way to address the issue of score significance is to transform raw scores into z-scores, which measure how far the raw scores deviate from the distribution mean. Specifically, for a raw score $s_i(x)$, its z-score $z_i(x)$ is defined as

\[ z_i(x) = \frac{s_i(x) - \mu_i}{\sigma_i} \quad (2) \]

where $\mu_i$ is the mean and $\sigma_i$ is the standard deviation of the distribution $\{s_i(x_1), s_i(x_2), \ldots, s_i(x_{\infty})\}$ induced by model $M_i$. In practice, the mean and standard deviation of this distribution over an infinite set is estimated from a finite sampling, which in most cases is naturally obtained when searching a database of profile HMMs. In other words, in our case of searching a database of profile HMMs, the infinite dimensional vector representation of a model is reduced to a vector of a finite dimension $\{s_i(C_1), s_i(C_2), \ldots, s_i(C_N)\}$ where $N$ is equal to the database size and the $C_i$ are the consensus sequences of the models in the database. Thus, $s_i(C_j)$ is the kth component of $M_i$, with $C_i$ being the consensus sequence of the kth model in the database, and can be transformed into a z-score $z_i(C_j)$ as defined above. Our z-score-based similarity measure can then be taken as

\[ Z_{ij} = z_i(C_j) + z_j(C_i) \quad (3) \]

where $z_i(C_j)$ and $z_j(C_i)$ are the respective z-scores. In order to avoid the effect of the ‘self component’ $s_j(C_j)$ on the above metric, $C_i$ is excluded before calculating the mean and standard deviation.

One problem with using consensus sequences to compare profile HMMs is that the task of finding the exact consensus sequence for a given HMM is an NP-hard problem (Lyngso et al., 1999). As a substitute for the exact consensus sequence, in this work we use a quasi-consensus sequence computed by a heuristic approach using the hmmemit program from the HMMER package (Eddy, 1998) with the `-c` option. This is found by traversing the nodes in the profile HMM and calculating the state occupancy of the match and insert states in each node. If the match state in a node is used with probability $\geq 0.5$, then this match state is ‘consensus’ and the maximally likely residue in the state is chosen. On the other hand, if the state occupancy says that the insert state is used with probability $\geq 0.5$, this insert state is ‘consensus’ and is used to generate the degenerate symbol (‘X’) as many times as the expectation value of the insertion length.

Sequence alignments

To search a database of HMMs associated with known structures, we first construct an HMM for the query sequence from a multiple sequence alignment of similar proteins, for instance those generated by a PSI-BLAST search. Our goal is to find an alignment between the query sequence (as seed sequence to its own HMM) and the seed sequences for HMMs in the database. Alignments between the seed sequences of two profile HMMs being compared can be found by aligning quasi-consensus sequences to the HMMs and using simple transitivity. To illustrate this, let us assume that there are two HMMs, $M_i$ and $M_j$, generated using seed sequences $S_i$ and $S_j$, and let $C_i$ and $C_j$ be quasi-consensus sequences for $M_i$ and $M_j$ respectively. Our goal is to find an alignment between the seed sequences $S_i$ and $S_j$. To get such an alignment, we can use either of the quasi-consensus sequences as an intermediate sequence between $S_i$ and $S_j$. That is, we can align $C_i$ to models $M_i$ and $M_j$ using a standard sequence–model alignment procedure, such as the posterior decoding or Viterbi algorithm (Durbin et al., 1998). We use the align2model program in the SAM package for this purpose (Krogh et al., 1994). The resulting $S_i$–$C_i$ and $C_i$–$S_j$ alignments can then be used to determine the $S_i$–$S_j$ alignment using transitivity. Similarly, the other consensus sequence $C_j$ can also be used. In a later section on alignment accuracy, we describe how to combine these two alignments to reach a single final alignment between $S_i$ and $S_j$.

Benchmark datasets

In a previous paper on comparison of different profile–profile alignment scoring functions (Wang and Dunbrack Jr, 2004), we built two datasets for benchmarking alignment accuracy and homology detection based on consensus DALI (Holm and Sander, 1995) and CE (Shindyalov and Bourne, 1998) structural alignments of domains defined in SCOP (Murzin et al., 1995). These were defined as a ‘Set A’ of 3441 pairs of sequences for experiments on sequence alignment accuracy and a ‘Set S’ of 665 sequences for assessing sequence search sensitivity and specificity. In this work, we reduced ‘Set S’ to 569 sequences. We used multiple sequence alignments of proteins homologous to those in Sets A and S from our previous work (Wang and Dunbrack Jr, 2004). Profile HMMs were constructed from these multiple sequence alignments using the SAM package (Karplus et al., 1998) with the w0.5 script. Then, for each of the constructed models, quasi-consensus sequences were generated using the hmmemit program from the HMMER package (Eddy, 1998).

For accuracy, following Wang and Dunbrack Jr (2004), about one-third of those sequence pairs in Set A were selected randomly to be used as a training set (1136 pairs), with the rest serving as a testing set (2305 pairs). The testing set was further split into eight groups, each corresponding to a range of sequence percent identity. The distribution of sequence pairs over the eight ranges is shown in Table 1.

We compare our method with the COMPASS program (Sadreyev and Grishin, 2003). COMPASS was used as described in its documentation. The same multiple sequence alignments used to construct HMMs were provided to COMPASS for profile–profile alignments. For COMPASS, all versus all comparisons were made by comparing each of the 569 multiple sequence alignments (as queries) to a database of the 569 profiles (created from the 569 multiple sequence alignments using the COMPASS package program mk_compass_db) using the compass_vs_db program.
As shown in Table 2 and Figure 1, the results for COMPASS show that using the symmetric $Z$ measure as given in Equation (3) does better than using the $E$-value measure in terms of search specificity and are comparable in terms of search sensitivity. However, we see that the symmetric $Z$ measure is much better than the asymmetric $Z$ measure. In both the consensus and seed-based comparisons, the symmetric $Z$ measure is much better than the asymmetric one. As shown in the last plot of Figure 1 and Table 2, the consensus-based comparison of HMMs achieves better performance than the other two approaches—COMPASS and seed sequence-based HMM comparison. The symmetric ROC$_{100}$ score for the quasi-consensus method is 0.915 while COMPASS achieves a value of 0.883. More importantly, the quasi-consensus method is much faster and is shown to produce more accurate alignments, as discussed below.

### Evaluation of alignment accuracy

The accuracy of the predicted alignments is quantified by three parameters $Q_{\text{modeler}}$, $Q_{\text{developer}}$ and $Q_{\text{combined}}$ as proposed in earlier work (Sauder et al., 2000; Yona and Levitt, 2002). The accuracy parameter from the modeler’s point of view, $Q_{\text{modeler}}$, is the ratio of the number of correctly aligned pairs of residues to the total number of aligned positions in the alignment being evaluated. The parameter from the developer’s point of view, $Q_{\text{developer}}$, is the ratio of the number of correctly aligned pairs to the number of aligned positions in the structural alignment. The combined accuracy parameter, $Q_{\text{combined}}$, is the total number of correct matches divided by the total number of positions that are aligned in either the structural alignment or the alignment being evaluated. Aligned pairs that are in both the sequence and the structure alignment are counted only once in the denominator of $Q_{\text{combined}}$. In this experiment, for every pair in the testing part of Set A (2305 pairs), we created two alignments by using each of the two consensus sequences as a hinge as described earlier. We then combined these two alignments into a final alignment according to the following two schemes. The first scheme (‘MAX’) simply picks one alignment, as a whole, out of the two alignments. The alignment that is picked has the higher score when the quasi-consensus sequence was aligned to the HMM. The second scheme (called ‘AND’ selection) is to take segments that are common in both alignments. The final alignment thus obtained misses out segments where the two alignments disagree. Three variations to this AND-selection are suggested on which alignment of these uncommon segments shall be picked to patch the final alignment. In the first variation (‘AND1’ selection), for a region where the two alignments disagree, we choose the alignment that contains fewer gap positions. As a refinement of AND1 selection, the second variation (‘AND2’ selection) calculates a BLOSUM score for each of the two alignments at an uncommon segment and picks the alignment that has a larger score. The third variation (‘AND3’ selection) chooses for a non-common segment the alignment that agrees better with the structural alignment for that segment. However, the AND3 selection is not usable when the correct ‘answer’, i.e. the structural alignment, is not available. The AND3 was the best alignment one could obtain with a method that was always able to tell which alignment was better.

The quality of the final alignments obtained by using these selection schemes and by using COMPASS was evaluated, and the accuracy parameters were averaged in the eight regions of sequence identity range as listed in Table 1. The results are reported in Figure 2. We observe that regions in the two alignments that agree with one
Fig. 1. ROC curves. The legend sym/asym indicates whether the symmetric (both terms) or asymmetric (only the first term) is used in Equation (3). QUASI is when quasi-consensus sequences are used for HMM–HMM comparison and SEED is when seed sequences are used.

Table 2. ROC100 (normalized areas under the curves in Figure 1) values for the three approaches: COMPASS, seed sequence-based HMM comparison and quasi-consensus sequence-based HMM comparison

<table>
<thead>
<tr>
<th></th>
<th>COMPASS Sym</th>
<th>Seed Sym</th>
<th>Consensus Sym</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetric</td>
<td>0.883450</td>
<td>0.858050</td>
<td>0.914950</td>
</tr>
<tr>
<td>Asymmetric</td>
<td>0.839538</td>
<td>0.761250</td>
<td>0.866337</td>
</tr>
</tbody>
</table>

The areas are divided by the area of the rectangle enclosed by lines $T_p = 800$ and $F_p = 100$.

another (AND) are more accurate per residue of the alignment ($Q_{\text{Mode}l}$) than those that include regions where the alignments disagree (AND1, AND2 and AND3). However, the latter alignments are also considerably longer and do provide some correct information, resulting in higher $Q_{\text{Developer}}$ scores. We see that, with the exception of the first identity range, both the MAX selection and AND2 selection outperform COMPASS in terms of $Q_{\text{Combined}}$.

We repeated the same experiment by using seed sequences instead of the quasi-consensus sequences to compare the HMMs. In this case, a seed sequence of one model is aligned to the other model, and vice versa. From the two resulting alignments, a final alignment was generated by using the selection schemes discussed above. The accuracy for these seed-based alignments is shown in Figure 3. The results show that the seed-sequence method does not behave well at very low sequence identities, but it does better than COMPASS at higher sequence identities.

The results of these experiments as shown in Figures 2 and 3 indicate that it is advisable to use consensus-based alignments in the lower sequence identity regions and seed-based alignments in the higher sequence identity regions. Because of this, we need to decide whether to choose a seed-based or a consensus-based alignment depending on the symmetric $z$-score between the two HMMs as given in Equation (3). Using the training part of Set A (1136 pairs), we found that a value of −22 resulted in the highest alignment accuracy. Below this cutoff (i.e. more negative scores) we used the seed-based alignment and above this cutoff we used the consensus-based. The results of this mixed scheme on the testing set of Set A (2305) pairs are shown in Figure 4 along with the COMPASS results.

Statistical significance

To provide an estimation of the significance of hits produced by the server, we calculated the probability of a false positive at each value of the symmetric $z$-score [Equation (3)]. This was performed on a set of 7924 profiles in the SUPERFAMILY database.
Comparing profile hidden Markov models

Values for averaged sequence alignment accuracy parameters $Q_{\text{Modeler}}$, $Q_{\text{Developer}}$ and $Q_{\text{Combined}}$. In each plot, averaged parameters for COMPASS and alignments from five selection schemes for consensus-based HMM–HMM alignments are shown.

Values for averaged sequence alignment accuracy parameters $Q_{\text{Modeler}}$, $Q_{\text{Developer}}$ and $Q_{\text{Combined}}$. In each plot, averaged parameters for COMPASS and alignments from five selection schemes for seed-based HMM–HMM alignments are shown.

All unique HMM–HMM comparisons for this set were performed using the quasi-consensus method, and the results were sorted by symmetric $z$-scores. For a particular value of the symmetric $z$-score, the proportion of false positives with a score equal to or greater than that score was calculated. The results are plotted in Figure 5. In this figure, $F(z_{\text{sym}}) = -\log_{10}(P(z_{\text{sym}}))$ is plotted as a function of the symmetric $z$-score. Because it is not possible to estimate the $P$-value until the first false positive is reached (as $z_{\text{sym}}$ decreases from infinity), the values in the figure represent an upper bound on the value of $P$ and a lower bound on $F$. The points are fitted to a polynomial of degree 8, so that $F(z_{\text{sym}}) = \sum_{k=0}^{8} a_k z_{\text{sym}}^k$ with a correlation coefficient of 0.98. This function is used only to calculate the statistical significance for any value of $z_{\text{sym}}$. No meaning is attached to the parameter values.
Fig. 4. Values for averaged sequence alignment accuracy parameters $Q_{\text{Modeler}}$, $Q_{\text{Developer}}$ and $Q_{\text{Combined}}$. In each plot, averaged parameters for COMPASS and alignments from five selection schemes for optimal mixture of seed-based and consensus-based HMM–HMM alignments are shown. The mixing approach is discussed in the text.

Algorithmic complexity

Like most profile–profile methods, COMPASS uses a DP technique to align two files of size $n$ and $m$ columns where a column is represented by a distribution over amino acids. Consider two profiles $A = [a_1, a_2, \ldots, a_m]$ and $B = [b_1, b_2, \ldots, b_n]$ where $a_i$ is the $i$th column in profile $A$ and $b_j$ is the $j$th column in profile $B$. To align $A$ and $B$ with affine gap scores, we need to fill three $n$-by-$m$ DP matrices as given by

$$M_{i,j} = \max\{M_{i-1,j-1} + s(a_i, b_j), IA_{i-1,j-1} + s(a_i, b_j)\}$$
$$IA_{i,j} = \max\{M_{i-1,j-1} - d, IA_{i-1,j-1} - e\}$$
$$IB_{i,j} = \max\{M_{i,j-1} - e, IB_{i,j-1} - e\}$$

where $s(a_i, b_j)$ is the substitution score for columns $a_i$ and $b_j$, and $d$ and $e$ are the gap opening and gap extension penalties respectively. This substitution score $s(a_i, b_j)$ for each pair of columns $(a_i, b_j)$ needs to be computed on the fly and this is an expensive computation. Similarly, in our consensus-based approach, to align a sequence of length $n$ to a profile HMM with $m$ nodes using the align2model program in the SAM package with the adpstyle 5 option (posterior decoding), we need to fill six $n$-by-$m$ DP matrices (three forward, three backward) as given in Durbin et al. (1998). However, in this case, there is no need to compute the substitution score between the $i$th residue in the sequence and the $j$th match/insert state in the HMM since this is simply given as the emission probability/score of the residue in the state. Thus, even though both profile–profile alignment and sequence–HMM alignment have the same algorithmic complexity of $O(nm)$, the need to compute the substitution matrix makes profile–profile methods computationally expensive.

As described above, generating a quasi-consensus sequence requires just a “walk” through the chain of states in the model and the time complexity for this step is $O(n)$, where $n$ is the number of match states in the model. Moreover, in the task of searching a database of profile HMMs, quasi-consensus sequences for all the models in the database.

![Fig. 5. Probability of a false positive as a function of the symmetric z-score from an all-against-all comparison of 7924 models from the SUPERFAMILY database. The curve is a polynomial fit to the data and has no explicit meaning otherwise.](image-url)
Comparing profile hidden Markov models

database can be generated and stored. In our experiment running on a system with a Pentium 1.13 GHz CPU and 376 Mb of RAM, it took COMPASS about 38 h to compute the all-versus-all comparison of 569 protein families, but it took our approach only about 1 h to do the same.

DISCUSSION

In this work, we propose a method for comparing profile HMMs via quasi-consensus sequences. While finding a true consensus sequence for a given profile HMM is NP-hard, quasi-consensus sequences seem to be a good surrogate to the true consensus sequences. We compare two profile HMMs by using quasi-consensus sequences from one model and score it against the other model and vice versa, and the similarity between the models is measured by a metric defined based on such comparisons. We also proposed aligning seed sequences of these profile HMMs using quasi-consensus sequences as hinges. To test the utility and effectiveness of our method, we used a benchmark dataset from our earlier published work. Our experiments demonstrate that our method outperforms a state-of-the-art profile–profile method, COMPASS, at identifying protein families with similar structures as defined at the SCOP fold level. The alignments generated by our method are more accurate than those generated by COMPASS, as measured against three established benchmark metrics. Moreover, unlike COMPASS, our approach is basically a sequence–profile comparison and does not require the computation of a substitution matrix and hence runs much faster.

With a slightly better accuracy and a considerably faster speed, our approach should prove to be a good alternative to current profile–profile methods for use in database searches against proteins of known structure. As the size of the Protein Data Bank increases, this advantage is significant.

A server based on this method is available at http://liao.cis.udel.edu/website/servers/modmod

ACKNOWLEDGEMENTS

This publication was made possible by NIH Grant R01-HG02302 to R.L.D., and funds from the Pennsylvania Tobacco Settlement (R.L.D. and G.W.).

REFERENCES