PROTEINS: Structure, Function, and Genetics 52:446–453 (2003)

Detecting Distant Homologs Using Phylogenetic Tree-Based HMMs

Bin Qian[1] **and Richard A. Goldstein**[1,2*]
[1]*Biophysics Research Division, University of Michigan, Ann Arbor, Michigan*
[2]*Department of Chemistry, University of Michigan, Ann Arbor, Michigan*

ABSTRACT **It is often desired to identify further homologs of a family of biological sequences from the ever-growing sequence databases. Profile hidden Markov models excel at capturing the common statistical features of a group of biological sequences. With these common features, we can search the biological database and find new homologous sequences. Most general profile hidden Markov model methods, however, treat the evolutionary relationships between the sequences in a homologous group in an ad-hoc manner. We hereby introduce a method to incorporate phylogenetic information directly into hidden Markov models, and demonstrate that the resulting model performs better than most of the current multiple sequence-based methods for finding distant homologs. Proteins 2003;52:446–453.** © 2003 Wiley-Liss, Inc.

Key words: homology search; hidden Markov model; protein evolution; Tree-HMM

INTRODUCTION

Despite much experimental and computational work, we still have little information about a large fraction of all newly discovered proteins. One of the most important tools in generating information about a particular target protein of interest is through sequence comparisons, trying to identify homologs about which more is known. Homology searches can be accomplished simply by comparing the target sequence in a pairwise manner against all of the sequences in the database. Many well-developed methods based on this principle have been developed, including such popular tools as BLAST[1] and FASTA.[2] The statistically significant hits, that is, those unlikely to have resulted by chance, are classified as likely homologs. Although these methods have proven their ability to detect closely related sequences (those with sequence identity larger than 30%), their performance decreases quickly when one tries to find more distant homologs.[3]

To overcome this limitation, researchers have developed various homology search methods based on the common features of a group of related sequences. These methods include profiles (PROBE,[4] PROSITE,[5] PSI-BLAST[6]), hidden Markov models (HMMER,[7] SAM[8]), and family pairwise search methods.[9] Of these, hidden Markov models (HMMs) have proven to be an especially sensitive method for finding remote homologs.[10] In addition to remote homology detection, HMMs have been applied to a wide variety of other bioinformatics problems such as multiple sequence alignments and motif searches.[11,12]

HMMs are based on the concept that observed sequences represent probabilistic realizations of an underlying statistical model, that the model can be extracted (with some accuracy) from the observed sequences, and that it is better to compare new sequences directly to the model than to the individual realizations. The most simple way to construct an HMM is to assume that the observed sequences represent random and uncorrelated examples of the set of possible sequences, weighted by the probability that they would result from the model. This assumption is, however, far from true. Our current sets of sequences are heavily weighted towards specific taxonomic groupings, from kingdoms to species. As a result, one needs a way of weighting the sequences properly so to minimize the influence of multiple closely related sequences. In the absence of a theoretically derived weighting scheme, most HMM methods use one of a set of ad-hoc empirical procedures. This may be even more problematic for representing the insertions and deletions that occur during the evolutionary history. These weighting schemes also tend to maximize the effect of non-homologs incorrectly included with the initial set of model proteins. Additionally, the extraction of a *single* probabilistic model may be overly restrictive. The set of "homologs" might not form a homogeneous group, but rather have a hierarchical structure based on the underlying evolutionary patterns of speciation and gene duplication. More information might be provided by constructing a set of such models, each representing some part of this substructure. This is difficult to do if the underlying nature of the substructure is ignored.

Explicit modeling of the underlying phylogenetic relationships represents the most natural approach to resolving these issues. Little work has been performed in this direction. Holmes and Bruno[13] developed a software package (*Handel*) that generates multiple sequence alignments based on the phylogeny of the target sequences. While the underlying model (the "links" model, developed by Thorne

Grant sponsor: NIH; Grant number: LM0577; Grant sponsor: NSF; Grant number: BIR9512955.

*Correspondence to: Richard A. Goldstein. E-mail: richardg@umich.edu

Received 28 September 2002; Accepted 26 November 2002

and collaborators[14]) includes a clever treatment of indels, insertions and deletions are considered to occur one residue at a time with constant probability over the sequence. This approach, therefore, ignores the dependence of the indel probabilities upon the context of the location, and cannot deal well with long indels. Rehmsmeier and Vingron[15] described a procedure using "tree augmentation," where a possible target protein is inserted into a preexisting tree based on an evolutionary distance algorithm. While the phylogenetic relationships are considered in reconciling the various distances into a consistent phylogenetic tree, the distances themselves are computed in a pairwise manner, neglecting the evolutionary process of substitutions, insertions, and deletions. Even so, they demonstrated improved performance compared with other standard methods.

In this study, we use a different approach to utilize phylogenetic information in a homology search. With general phylogenetic theory, one can easily calculate the posterior probabilities of amino acids at each node of the phylogenetic tree.[16] These posterior probabilities can be viewed as an amino-acid profile of the protein family at that node. Profiles at different nodes can be viewed as common features of the protein family at different evolution stages. All combined, they can serve as a more accurate description of the subset of the protein family descended from that node. In particular, we can construct a profile HMM at each node of the phylogenetic tree where the emission probabilities are given by the posterior probabilities resulting from the phylogenetic reconstruction.

We also need to construct the HMM so that the insertions and deletions are treated as specific evolutionary events in the context of the phylogenetic relationships. There have been several attempts to incorporate indels into phylogenetic model, including treating indels as the 21st (for protein data) or 5th (for DNA/RNA data) character, Hein's empirical model,[17] Thorne's links model,[14] and the Tree-HMM scheme (T-HMM) of Mitchison and Durbin.[18,19] Among these, the Tree-HMM scheme has the most natural association with HMMs, as well as allowing both heterogeneity of insertion and deletion probabilities and a more natural representation of long indels. Just as we can represent amino acid substitutions in the evolutionary process as changes in the emitted amino acid in the HMM, the Tree-HMM approach models insertion and deletion events as changes in the *path* of the sequence through the HMM. And just as we can use the probabilistic reconstruction of ancestral nodes as the emission probabilities for the HMM corresponding to this particular node, we can use a probabilistic reconstruction of the path of the ancestral sequence through the HMM to calculate the transition probability between various states of the HMM. This allows us to deal with indels and amino acid substitutions in a consistent way. While the T-HMM method lacks the ability of modelling the event of insertion followed by deletion, this is of minimum disadvantage in practical use.[19]

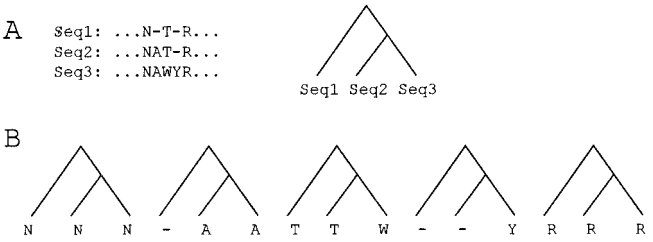

Fig. 1. **A:** Section of the alignment of 3 sequences and the corresponding phylogenetic tree. **B:** The amino acids in each column of the alignment follows the same phylogenetic relationship.

Using the T-HMM approach, we can calculate an HMM for every node in the phylogenetic tree corresponding to a set of homologous proteins. As we do not know where the putative homologous sequence will be on the existing tree, we compare each target sequence with the HMM representing every node, and use the highest score as the homology search score. We describe this approach more fully below. We then describe its performance in identifying distant homologies, demonstrating superior performance compared with standard HMMs (HMMER) as well as other methods such as PSI-BLAST, Family-BLAST, and PROSITE.

METHODS

Theory

Phylogenetic tree construction

There are three major approaches for inferring a phylogenetic tree from protein or DNA sequence data. One is the maximum parsimony method,[20] which is intuitive and fast but lacks a rigorous theoretical basis. The second is through various distance methods, where pairwise evolutionary distances are calculated and then reconciled into a consistent phylogenetic tree. This method, the basis of the homology search method of Rehmsmeier and Vingron,[15] neglects correlations induced by the shared evolutionary history. The third approach is the more statistically-sound maximum likelihood method.[21,22] Mitchison and Durbin's T-HMM scheme was created in a maximum likelihood framework.

Consider three sequences related by a simple phylogenetic tree as shown in Figure 1(A). Each currently observed protein sequence is attached at a so-called "leaf" node, that is, a node directly connected with only one other node. Because of the assumption that different columns of a sequence alignment are evolutionarily independent, we can treat the phylogeny of the amino acids corresponding to each alignment column separately. As shown in Figure 1(B), every alignment column follows the same phylogeny as the whole sequences. A combination of these events will give the overall phylogenetic description of the sequence family.

In maximum likelihood methods, we are interested in the model that maximizes the conditional probability that the observed sequences would result if the model were correct. The model can represent any combination of the substitution rates, the branch lengths, and the tree topol-

ogy. By assuming independence of the various locations in the multiple alignment, the overall likelihood L can be calculated easily. If the likelihood L_c for a particular location c in the sequence alignment is defined as equal to the probability $p(D_c|T, \theta)$ of the observed data D_c at this location arising given the tree T and model parameters θ

$$L_c \equiv p(D_c|T,\theta) \tag{1}$$

then the likelihood L of the sequence family is

$$L = \prod_c p(D_c|T,\theta) \tag{2}$$

The likelihood values $p(D_c|T, \theta)$ can be easily computed in a manner linear with the number of currently observed sequences, as described by Felsenstein.[22] In general, we calculate the log likelihood $\log(L) = \Sigma_c \log(L_c)$.

In order to calculate the likelihoods, we need to have a model for the rate of substitutions. The 20 $\times$ 20 score matrix $\mathbf{P}(t)$ representing the probability of amino acid i being replaced by amino acid j during evolutionary time t is calculated by

$$\mathbf{P}(t) = e^{Qt} \tag{3}$$

where $\mathbf{Q}$ is the 20 $\times$ 20 instantaneous rate matrix. Diagonal elements Q_{ii} are set so that the sum of each row is equal to 0.

For *reversible* models, the probability of observing a substitution of amino acid i for j and of j for i are identical. If π_i represent the amino acid equilibrium frequencies, reversibility can be written as

$$Q_{ij}\pi_i = Q_{ji}\pi_j \tag{4}$$

In this case, we can represent the $\mathbf{Q}$ matrix as the product of a symmetric rate matrix $\mathbf{R}$ times the equilibrium probabilities:

$$Q_{ij} = R_{ij}*\pi_j \tag{5}$$

Reversible models are convenient in that the likelihood calculation does not depend upon an accurate placement of the root of the tree, that is, the most recent common ancestor to the set of sequences.

As mentioned above, in a standard phylogenetic tree, different sites in a sequence are considered as independent probabilistic events. Deletion and insertion events are generally not modeled, as they are obviously not independent.

T-HMM

As shown in Figure 2(A), given a set of homologous proteins with the corresponding sequence alignment, we can construct each sequence's path through the HMM and build a profile HMM. Our profile HMM has the same structure as a traditional one, with Match (M) states, Delete (D) states, and Insert (I) states, and transitions between these states.[11,12] The amino acids in the sequence represent the sequence of emissions from Match and Insert states. Specifically, if an amino acid is in an alignment column where more than 50% of the sequences have gap characters, then this amino acid is assigned to a Insert state. Otherwise, it is assigned to a Match state.

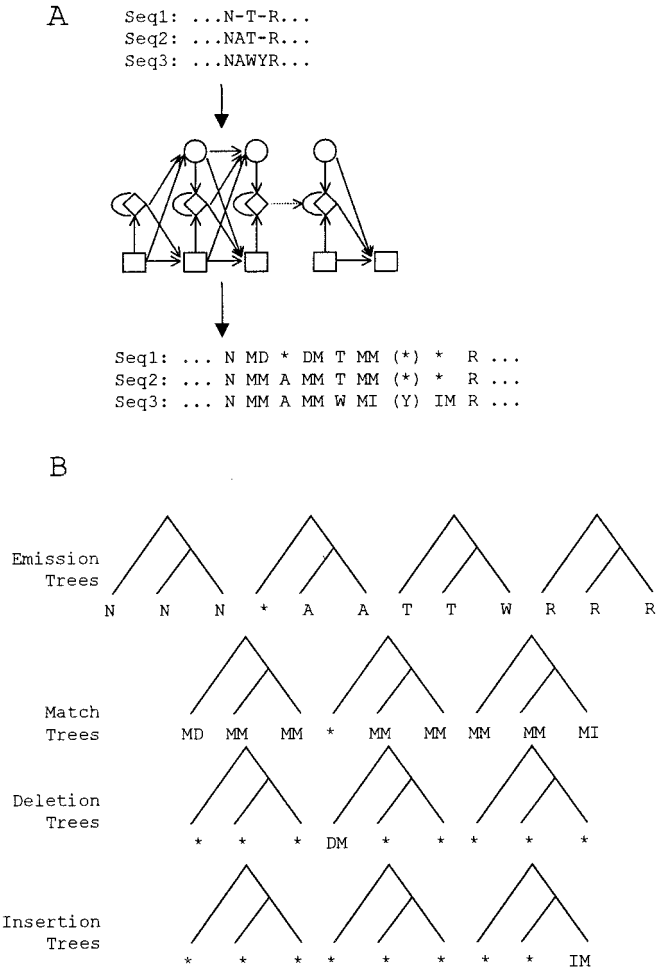

Fig. 2. **A:** Construction of a profile-HMM from a multiple alignment. **B:** The amino-acids and state-transition characters follow the same phylogeny as the sequences. We use 9 state-transition characters: MM (match state to match state), MD (match state to deletion state), MI (match state to insertion state), DD (deletion state to deletion state), DM (deletion state to match state), DI (deletion state to insertion state), II (insertion state to insertion state), IM (insertion state to match state), and ID (insertion state to deletion state).

In the T-HMM method of Mitchison and Durbin, the path of the amino acid sequence through the HMM is represented as a sequence of state-transition characters.[18,19] For example, if we represent a transition from a Match to a Match state as an MM, Match to Delete as an MD, and Match to Insert as MI, etc., a path through the HMM involving a chain of Match states with one insertion of length two could be represented as the sequence MM MM MI II IM MM MM. While amino acid substitutions that occur during evolution involve changes in the emission of a given match or insert state, insertions and deletions involve changes in the path, and thus changes in the state-transition characters. For instance, a deletion event might change (MM MM MM MM MM) to (MM MD . . . DM MM). In this way, we can model the insertions and deletions that occur during the evolutionary process

as changes in these state-transition characters. As shown in Figure 2(B), the state-transition characters follow the same phylogenetic relationship as the amino acids.

A change in the state-transition character can be represented as a rotation of the corresponding transition arrow in the HMM around its origin so that the state-transition characters starting from the same state (e.g., MM, MD, and MI) can evolve into each other, but state-transition characters starting from different states (e.g., MM and IM) cannot. For this reason, we need to have separate phylogenetic relationships for state-transition characters that start as Match, Delete, and Insert, which are assumed to evolve independently. There are, therefore, three independent instantaneous rate matrices corresponding to the three different possible beginning states, each with parameters representing the rate of substitutions from one state-transition character to another. In this scheme, there are both undetermined emission characters and state-transition characters, which are indicated by "*"s. For example, in Figure 2(A), Seq 1 has HMM path " . . . N MD * DM . . . " where a deletion follows amino acid emission "N." As there is no amino acid emitted from this deletion, in Figure 2(B) there is a "*" on the "Seq 1" leaf of the second Emission tree. Likewise, since a "MD" character occupies the "Seq 1" leaf node of the first Match transition tree, there are no Insert or Delete state-transition characters for Seq 1 at the first location. Thus, there are "*"s on the "Seq 1" leaf nodes of the first Insertion and Deletion transition trees. Since all these undetermined characters are actually not observed in our data, they are modeled in the likelihood calculation with flat priors, which essentially eliminate the nodes occupied by "*"s from the likelihood calculation.

As in a general phylogenetic model, we assume that changes in the state-transition characters are independent probabilistic events. Thus, we can calculate the log likelihood of a given set of HMM paths on a particular tree by calculating the log likelihood of each site independently, and then summing over all three state-transition character trees at each site.[22,19] This is not to say that insertions and deletions are treated as multiple occurrences of insertions and deletions of single amino acids. Rather, a deletion of a set of amino acids is considered a transition from an MM to an MD state-transition character, followed by a sequence of DD state-transition characters, terminated by a DM. Representing the probability of a DD state-transition character as a constant is equivalent to modeling the distribution of gap lengths as an exponential. In this case, calculating the log likelihood would effectively represent the deletion with a standard affine gap penalty.

Implementation

Optimization of the substitution rate matrix

In order to perform the likelihood analysis, we need the matrices that define both substitutions between amino acids as well as between the various state-transition characters. For the amino acid substitution matrices, we use the WAG matrix by Whelan and Goldman.[23] Here the same substitution matrix is used for every HMM location. One could, however, use a more complicated scheme such as RIND program, which uses a different equilibrium frequency distribution for each HMM location,[24] or assign HMM locations to different classes with different substitution matrices.[25] While straightforward, these modifications were not made for simplicity.

Unlike the substitution matrix of amino acids, the substitution rate matrices for state-transition characters are not readily available. We can derive the matrices, however, by optimizing them for a set of well-defined multiple sequence alignments and phylogenetic trees. We adopt a likelihood-based optimization scheme to achieve our substitution rate matrices for these state-transition characters. This is done calculating the log likelihood of a number of sets of observed sequences as a function of the values of these substitution rates, and adjusting these rates in order to maximize this likelihood.

In order to perform this optimization, we need sets of proteins containing sufficient insertion and deletion events with reliable sequence alignments. Structural alignments are considered good quality and generally reflect the evolutionary relationship between structurally similar homologous sequences. We choose the CE (Combinatorial Extension) database[26] as our source of structurally aligned proteins. For each of the first 10 representative sequences in the CE database, we picked the 29 structural neighbors with the least sequence similarity with the representative sequence. These 10 sets of structurally aligned sequences served as our training set for the adjustment of the substitution rate matrices for state-transition characters.

For each set, we inferred a phylogenetic tree. First MOLPHY (MOLecular PHYlogenetics)[27] was used to generate the 15 most possible candidate tree topologies for each data set, followed by PAML[28] to find the most likely tree topology from the 15 candidates and calculate the branch lengths. The likelihood of the tree was calculated according to the "pruning" method described by Felsenstein.[22] A linear simplex optimization method[29] was used to adjust the state-transition character substitution matrix to optimize the likelihood of the observed sequences given the tree.

In a profile HMM, a sequence of consecutive amino acid insertion events is modeled by one HMM location.[11,12] This means that a location might have multiple II (Insert state to Insert state) state-transition characters. Thus, a thorough bookkeeping may be needed to keep track of the number of IIs at each HMM location. Furthermore, unlike Match or Delete state-transition characters where one state-transition character can only evolve to another state-transition character, IIs can change in numbers at each HMM location. All these will significantly increase the computational complexity if we were to model the evolutionary relationship of IIs. Mitchison and Durbin handled this by using only Delete and Match states in their model. We take another approach, including Insert states in our model, but use a more general way to infer the II probability for each location. Specifically, we count $N_{\mathrm{II}c} + N_{\mathrm{IM}c} + N_{\mathrm{ID}c}$, the number of II, IM, and ID at location c in the profile HMM constructed from the original alignment, and

TABLE I. IDs of the SCOP Superfamilies Used in Our Test[†]

a.1.1	a.39.1	b.10.1	b.43.4	c.1.2	c.37.1	c.66.1	d.131.1	d.2.1	g.3.6
a.118.1	a.4.1	b.18.1	b.47.1	c.1.8	c.47.1	c.67.1	d.142.1	d.58.1	g.3.7
a.26.1	a.4.5	b.29.1	b.55.1	c.2.1	c.52.1	c.68.1	d.153.1	d.92.1	g.39.1
a.3.1	b.1.1	b.40.4	b.6.1	c.3.1	c.55.3	c.69.1	d.16.1	f.2.1	

[†]A total of 39 superfamilies were used. For each superfamily, sequences in the first family in that superfamily were used as the test set of true homologs, and the other sequences in that superfamily were used to build the profiles.

calculate the probability of II at this location using the following formula:

$$\mathrm{Prob}_c(\mathrm{II}) = \frac{N_{\mathrm{II}c} + 1}{N_{\mathrm{II}c} + N_{\mathrm{IM}c} + N_{\mathrm{ID}c} + 3} \tag{6}$$

where a uniform pseudo-count is used as a prior. We then keep the II transition probability fixed at this value during later calculation. IM and ID probablities are still treated the same way as Match and Delete state-transition characters. Although we omit the phylogenetics of the II state-transition characters, we still model the phylogenetics of IM and ID. As a result, the rate matrix for state-transitions starting with Insert states only encodes IM to ID transitions, and is thus a 2×2 matrix instead of 3×3. The normalization ($\mathrm{Prob}_c(\mathrm{II}) + \mathrm{Prob}_c(\mathrm{ID}) + \mathrm{Prob}_c(\mathrm{IM}) = 1$) is performed afterwards.

Constructing the T-HMM

We generally observe the currently existent sequences at the leaves of a phylogenetic tree. It is possible, however, to perform a probabilistic reconstruction of the internal nodes based on this data.[16] Consider Ω_c, the amino acid at location c in ancestral node Ω. According to a likelihood analysis, the probability $p(\Omega_c = A_x)$ that amino acid A_x occupied this position in this ancestral sequence can be modeled as

$$p(\Omega_c = A_x|\mathrm{T},\theta,\mathrm{D_c}) = \frac{p(D_c|T,\theta,\Omega_c = A_x)\,\pi(A_x)}{\mathrm{p}(\mathrm{D_c}|\mathrm{T},\theta)} \tag{7}$$

where $\pi(A_x)$ represents the a priori probability of amino acid A_x at that location irrespective of the observed data, and $p(D_c|T, \theta, \Omega_c = A_x)$ represents the conditional probability of the observed sequence data at that location given $\Omega_c = A_x$. (This relationship is properly normalized, as $p(D_c|T, \theta) = \Sigma_{A_x} p(D_c|T, \theta, \Omega_c = A_x)\,\pi(A_x).)\,p(\Omega_c = A_x)$ then represents the relative probabilities of each amino acid at this node. Such a reconstruction can be considered a profile for that particular node, with the probabilistic model serving as emission rates. Similarly, if we have phylogenetic trees representing changes in state-transition characters in current sequences, we can perform a probabilistic model for the various state-transition characters at each of the internal nodes, representing the probabilities of the various types of state-transitions. These probabilities can be combined with the probabilities of the various amino acids to create a profile HMM at each node. Although the leaf nodes represent observed data, we can calculate probabilistic models for these nodes as well by deleting the sequence at this location and then performing a probabilistic reconstruction based on the other observed nodes.

We can then use these created HMMs to perform a search for homologs by aligning all of the proteins in the database against every profile HMM for each of the various nodes using the Viterbi algorithm.[30] The highest score is used as the score between the sequence and the model. In order to normalize the scores, we use the reverse HMM null model by Karplus et al.[8] In a reverse HMM null model, the reversal of a HMM serves as the null model of that HMM. The ratio of the probabilities that the sequence aligned with model and null model is used as the score. That is

$$\mathrm{Score} = \log\left[\frac{P(\mathrm{Sequence}|\mathrm{HMM})}{P(\mathrm{Sequence}|\mathrm{reverse\ HMM})}\right] \tag{8}$$

Thus, the score is normalized by the query sequence length. We can choose to use a local alignment or a global alignment depending on the type of analysis. In this research, we used a global alignment method to compute the comparison scores.

Testing the Method

Once the matrix parameters were determined, we moved to testing the ability of the T-HMM approach to identify new homologies. The structural classification of proteins (SCOP)[31] database has been used as a standard database to test homology detection methods.[10,15] The SCOP database contains detailed structural and evolutionary relationships between all proteins whose structure is known. This database is hierarchically organized by class, fold, superfamily, family, domain. Domains in a same family are considered as having a close evolutionary relationship. Domains belong to the same superfamily but different families are considered as having a distant evolutionary relationship. We used the ASTRAL SCOP version 1.59 PDB40[32] (May 15, 2002 release), in which all sequences have less than 40% identity to each other. Thus, the database has much less redundancy than the whole SCOP database.

From ASTRAL PDB40, we chose those superfamilies that contain at least 2 families. In each superfamily, the first family was used as test family, and all members of the other families were used to build the model that represents this superfamily. Furthermore, to ensure adequate performance from alternative methods such as standard HMMs and profiles, only those superfamilies that have at least 10 sequences in their model-building group were used. We ended up with 39 superfamilies in our test set. The

superfamily IDs we used in our test are listed in Table I. All sequences in our test set had less than 40% sequence identity with any protein in the corresponding training set.

T-Coffee was used to build the multiple alignment of the training set.[33] A phylogenetic tree was then constructed using Molphy and PAML, as described above, and a T-HMM constructed. Finally, the resulting T-HMMs were used to search PDB40 database.

Comparison With Other Methods

In order to test this method versus other standard approaches, we compared with HMMER, PSI-BLAST, PROSITE, and family BLAST, as described below. In each case, the performance was evaluated with the default parameters.

Profile HMMs

Profile hidden Markov models (profile HMMs)[34] have proven to be a sensitive method for detecting distant sequence similarities. We used HMMER 2.2,[7] an implementation of profile HMM for protein sequence analysis. Specifically, we used hmmbuild in the HMMER package to build HMM models, and calibrated the statistical parameters of the model by using hmmcalibrate in the HMMER package. Finally, we searched our test protein database by using hmmsearch.

Family-BLAST

The Family-BLAST method is the BLAST version of the family pairwise search method.[9] The query sequence is compared with every sequence in the model homologous family, and the best score is taken as the query versus model comparison score. We used the E-values from blastp as the scores.

PSI-BLAST

One of the popular profile-based methods is PSI-BLAST.[6] PSI-BLAST uses site-specific amino-acid profiles to search protein database and find homologs. As PSI-BLAST generally performs several iterations during one search, it uses intermediate sequences that may not be in the model-building groups for other methods. In principle, almost all of the techniques described here, including the T-HMM approach, can be implemented in an iterative fashion, achieving potentially superior results when intermediate sequences are available. As our investigation concerns the basic ability of the various methods to generate profiles for detection of distant homologs, in order to provide a better comparison, we started the PSI-BLAST search from the same model-building groups, and constrained it to perform only one iteration.

PROSITE

A generalized profile from protein multiple sequence alignment can be used as a very sensitive method for the detection of distant sequence relationships.[35] One of the popular protein profile databases is PROSITE.[5,36] PROSITE is built by software set pftools 2.0.[37] This program package contains programs for generalized profile applications including pfmake, which builds profile from multiple sequence alignment; pfsearch, which searches as a protein database using a profile; pfscale, which calibrates the database search scores.

RESULTS

Optimized Rate Matrix

The optimized state-transition character symmetrical rate matrices $\mathbf{R}$ and equilibrium frequencies π_i are

	MM	MD	MI
π	0.96	0.022	0.02
MM			
MD	0.97		
MI	20.5	0	

	DD	DM	DI
π	0.47	0.52	0.011
DD			
DM	0.046		
DI	24.1	0	

	IM	ID
π	0.97	0.028
IM		
ID	1	

resulting in the following $\mathbf{Q}$ matrices:

	MM	MD	MI
MM	−0.44	0.021	0.41
MD	0.93	−0.93	0
MI	19.64	0	−19.64

	DD	DM	DI
DD	0.30	0.024	0.28
DM	0.021	−0.021	0
DI	11.26	0	−11.26

	IM	ID
IM	−0.028	0.028
ID	0.97	−0.97

Homology Detection Performance

We use an ROC (Receiver Operating Characteristic) plot to compare the homology search results by different methods. For a given superfamily model, each sequence in the PDB40 database has a score that measures the similarity between the sequence and the model. We combine all the results from every superfamily model search, and sort the list by scores. For a perfect method, all the true homologs will be at the top of the list, followed by the non-homologs. This cannot be achieved by any currently available method. In a practical case, we have homologs and non-homologs mixed in the sorted list. By counting the number of true homologs detected when a certain number of non-homologs are included in our results, we can measure the comparative homology-detection ability of the methods.

As shown in Figure 3, the number of true positives (homologs) is plotted versus the number of false positives (non-homologs) when we move the threshold to include different numbers of false positives. Since PDB40 database contains 4,383 protein sequences and we have 39 superfamilies, the total number of comparisons made is 170,937 (4,383 × 39). This number minus the total number of sequences used in profile construction (1,063) and the total number of true positives (215) is the total number of

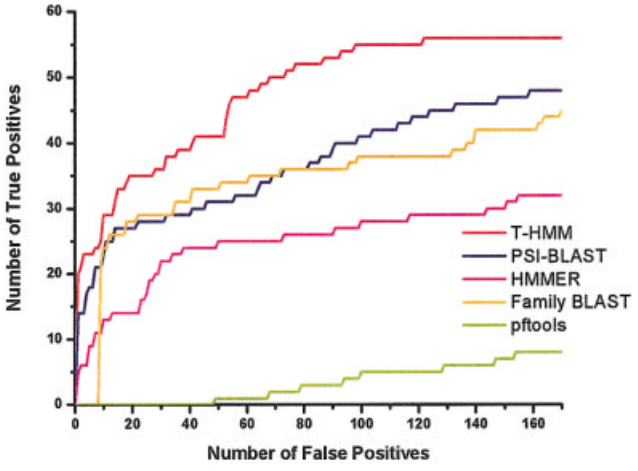

Fig. 3. ROC plot with universal threshold.

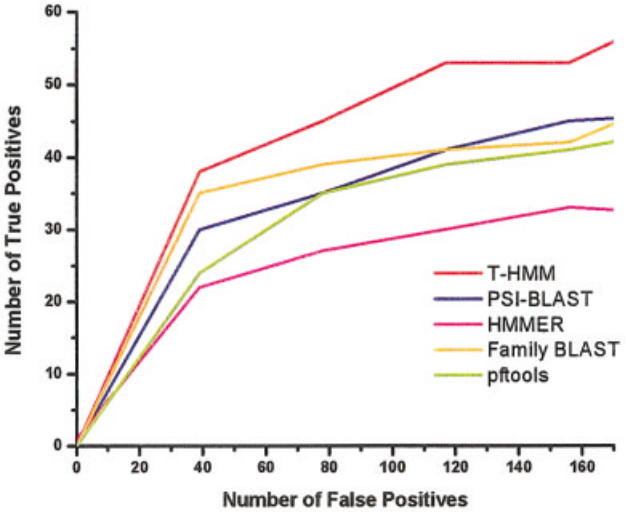

Fig. 4. ROC plot with individual threshold adjusted for equal numbers of false positives.

true negatives (168,294). Thus, the false-positive rate is 0.1% when there are 168 false positives. Among the 4,383 database sequences, there are 215 homologs to the 39 superfamilies. As we can see from Figure 3, when the false-positive rate is 0.1%, T-HMM method can detect 26% of the homologs, whereas PSI-BLAST, Family-BLAST, HMMER, and pftools can detect 22, 20, 14, and 3% of the homologs, respectively.

Note the above analysis requires the scores to be scaled so that the results are normalized for model length, sequence length, sequence composition, etc. Although each method we use has its own mechanism of normalizing the scores, it is still possible that the way a method normalizes the scores may affect the rank of search results. In addition, it may be possible that the score for false positives for easier targets might exceed the score for the most difficult true positives, even if all of the true positives had higher scores than all of the false positives observed with every target. Thus, we also make a coverage plot where each superfamily search is considered independently. First, we search the PDB40 database with each of the 39 superfamily models. Each superfamily model gives us a list of 4,383 scores for the 4,383 proteins in the PDB40 database. We then sort each list by the scores. Now we can set individual thresholds for each list so that exactly one false positive is observed in each list, and count the number of homologs detected in each list. The numbers are summed over to give us the total number of true positives when we have 39 false positives. Then we can move the threshold so that each list gives 2 false positives (78 total false positives), and so on. Note in this way, the total number of false positives recorded is always integer folds of 39. The result is shown in Figure 4. As we can see, Figure 4 gives us qualitatively the same result as Figure 3, except the curve for pftools. In Figure 4, pftools shows better performance than it shows in Figure 3, although still inferior to the other methods. That suggests the score normalization method in pftools does not give us a good context-independent measurement of the score. On the contrary, the scoring normalization schemes for T-HMM, PSI-BLAST, HMMER, and Family-BLAST seem to be doing well.

DISCUSSION

It has long been recognized that a set of homologous proteins provides much more information than a single instance, including increasing our ability to recognize further homologs. There have been a number of different methods developed that try to encode the patterns of conservation and variation in such an aligned set in order to detect distant homologs. Interestingly, we find that most of the attempts to generate a statistical profile based on these sequences, including PSI-BLAST, PROSITE, and HMMER, do not do significantly better than the more straightforward Family-BLAST approach of comparing each protein in the homologous set against the target protein and taking the maximum score. This conclusion is also supported in work by Rehmsmeier and Vingron.[15] The one exception is the T-HMM method described in this study. In contrast to these other approaches, the T-HMM method explicitly uses the phylogenetic relationships between the various homologous proteins to build the models. The results are significantly improved: specifically, at an error rate of 1/1,000, the T-HMM method finds almost twice as many homologs as the general HMM method HMMER, and significantly more than the PSI-BLAST, PROSITE, and Family-BLAST methods.

As described above, PSI-BLAST works in an iterative manner, using the created model to identify new homologs that can be used to further refine the model.[6] This can cause a significant improvement in performance when intermediate sequences exist. There are, however, two potential limitations to such an approach. Firstly, it is important to avoid contamination of the model by false positives, as even a small number can result in the incorporation of further false positives into the model. For this reason, the stringency requirement for including homologs in the intermediate steps must be quite strict. In

addition, as further sequences are incorporated into the model, the model becomes increasingly more general, and may lose resolution as it tries to represent an increasing number of different subtypes in one statistical model.

Presumably, all of the different forms of statistical models, T-HMM included, can be run in an iterative manner. All of these approaches, iterative or not, require the construction of a probabilistic model from the current set of homologs. It was this ability that we assayed in our tests. In fact, of the various statistical model building approaches, the T-HMM approach might be the most appropriate approach to be converted to an iterative form. For one thing, the explicit representation of the evolutionary relationships of the proteins would allow the immediate identification of distant (and possibly suspect) relationships between clusters of closely related sequences. The confidence in the distant relationships could then be evaluated independently. In addition, the evolutionary distance between the clusters would prevent one cluster from overly influencing the HMMs representing the other clusters. In this way, contamination of the model would be localized to the corresponding cluster. Finally, the T-HMM method does not attempt to create a universal model for the entire set of sequences, but rather creates a cluster of submodels. In this way, the fidelity of the models to various subtypes is not compromised.

ACKNOWLEDGMENTS

We thank Sarah Ingalls, Feng Gao, Matt Dimmic, David States, and Tom Blackwell for helpful discussions, and Todd Raeker and Michael Kitson for computational assistance. Financial support was provided by NIH grant LM0577 and NSF equipment grant BIR9512955.

REFERENCES

1. Altschul SF, Gish W, Miller W, Myers E, Lipman D. Basic local alignment tool. J Mol Biol 1990;215:403–410.
2. Pearson WR, Lipman DJ. Improved tools for biological sequence analysis. Proc Nat Acad Sci USA 1988;85:2444–2448.
3. Brenner SE, Chothia C, Hubbard TJP. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. Proc Nat Acad Sci USA 1998;95:6073–6078.
4. Neuwald A, Liu J, Lipman D, Lawrence C. Extracting protein alignment models from the sequence database. Nucleic Acids Res 1997;25:1665–1667.
5. Falquet L, Pagni M, Bucher P, Hulo N, Sigrist CJ, Hofmann K, Bairoch A. The PROSITE database, its status in 2002. Nucleic Acids Res 2002;30(1):235–238.
6. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DL. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25:3389–3402.
7. Eddy S. Hmmer: Profile hidden Markov models for biological sequence analysis. http://hmmer.wustl.edu, 2001.
8. Karplus K, Barrett C, Hughey R. Hidden Markov models for detecting remote protein homolo-gies. Bioinformatics 1998;14:846–856.
9. Grundy WN. Homology detection via family pairwise search. J Comput Biol 1998;5:479–491.
10. Park J, Karplus K, Barrett C, Hughey R, Haussler D, Hubbard T, Chothia C. Sequence comparsions using multiple sequences detect three times as many remote homologues as pairwise methods. J Mol Biol 1998;284:1201–1210.
11. Krogh A, Brown M, Mian IS, Sjölander K, Haussler D. Hidden Markov models in computational biology. J Mol Biol 1994;235:1501–1531.
12. Eddy SR. Hidden Markov models. Curr Opin Struct Biol 1996;6:361–365.
13. Holmes I, Bruno WJ. Evolutionary HMMs: a Bayesian approach to multiple alignment. Bioinformatics 2001;17:803–820.
14. Thorne JL, Kishino H, Felsenstein J. An evolutionary model for maximum likelihood alignment of DNA sequences. J Mol Evol 1991;33:114–124.
15. Rehmsmeier M, Vingron M. Phylogenetic information improves homology detection. Proteins 2001;45:360–371.
16. Koshi JM, Goldstein RA. Probabilistic reconstruction of ancestral protein sequences. J Mol Evol 1996;42:413–420.
17. Hein J. A unified approach to phylogenies and alignments. Methods Enzymol 1990;183:625–644.
18. Mitchison G, Durbin R. Tree-based maximum likelihood substitution matrices and hidden Markov models. J Mol Evol 1995;41:1139–1151.
19. Mitchison G. A probabilistic treatment of phylogeny and sequence alignment. J Mol Evol 1999;49:11–22.
20. Fitch WM. On the problem of discovering the most parsimonious tree. Am Nat 1977;111:223–257.
21. Cavalli-Sforza LL, Edwards AWF. Phylogenetic analysis: Models and estimation procedures. Am J Hum Genet 1967;19:233–257.
22. Felsenstein J. Evolutionary trees from DNA sequences: A maximum likelihood approach. J Mol Evol 1981;17:368–376.
23. Whelan S, Goldman N. A general empirical model of protein evolution derived from multi-ple protein families using a maximum-likelihood approach. Mol Biol Evol 2001;18:691–699.
24. Bruno WJ. Modeling residue usage in aligned protein sequences via maximum likelihood. Mol Biol Evol 1996;13:1368–1374.
25. Dimmic M, Mindell DP, Goldstein RA. Modeling evolution at the protein level using an adjustable amino acid fitness model. Pacific Symp Biocomput 2000:18–29.
26. Shindyalov IN, Bourne PE. Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. Prot Eng 1998;11:739–747.
27. Adachi J, Hasegawa M. MOLPHY version 2.3: Programs for molecular phylogenetics based on maximum likelihood Comput Sci Monogr 1996;28:1–150.
28. Yang Z. Phylogenetic analysis by maximum likelihood (paml), version 3.12. http://abacus.gene.ucl.ac.uk/software/paml.html, March 2002.
29. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Numerical recipes in C. Cambridge: Cambridge University Press; 1992.
30. Durbin R, Eddy SR, Krogh A, Mitchison GJ. Biological sequence analysis: probabilistic models of proteins and amino acids. Cambridge: Cambridge University Press; 1998. p 109–110.
31. Murzin AG, Brenner SE, Hubbard TJP, Chothia C. SCOP: A structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–540.
32. Chandonia J, Walker N, Conte LL, Koehl P, Levitt M, Brenner S. ASTRAL compendium enhancements. Nucleic Acids Res 2002;30:260–263.
33. Notredame C, Higgins D, Heringa J. T-Coffee: A novel method for multiple sequence alignments. J Mol Biol 2000;302:205–217.
34. Eddy SR, Mitchison G, Durbin R. Hidden Markov models. Curr Opin Struct Biol 1995;6:361–365.
35. Gribskov M, McLachlan AD, Eisenberg D. Profile analysis: detection of distantly related proteins. Proc Nat Acad Sci USA 1987;84:4355–4358.
36. Bairoch A. Prosite: a dictionary of sites and patterns in proteins. Nucleic Acids Res 1991;19:2241–2245.
37. Bucher P, Karplus K, Moeri N, Hofmann K. A flexible search technique based on generalized profiles. Comput Chem 1996;20:3–24.