To appear in *Computer Applications in the Biosciences*, 1997.

Meta-MEME: Motif-based Hidden Markov Models of Protein Families

William N. Grundy* bgrundy@cs.ucsd.edu
Department of Computer Science and Engineering
University of California, San Diego (619) 453-4364
La Jolla, California 92093-0114 FAX (619) 534-7029

Timothy L. Bailey tbailey@sdsc.edu
San Diego Supercomputer Center
P.O. Box 85608 (619) 534-8350
San Diego, California 92186-9784 FAX (619) 534-5127

Charles P. Elkan elkan@cs.ucsd.edu
Department of Computer Science and Engineering
University of California, San Diego (619) 534-8897
La Jolla, California 92093-0114 FAX (619) 534-7029

Michael E. Baker mbaker@ucsd.edu
Department of Medicine
University of California, San Diego (619) 534-4164
La Jolla, CA 92093-0623 FAX (619) 534-1424

*Corresponding author.

Abstract

Motivation: Modeling families of related biological sequences using Hidden Markov models (HMMs), although increasingly widespread, faces at least one major problem: because of the complexity of these mathematical models, they require a relatively large training set in order to accurately characterize a given family. For families in which there are few known sequences, a standard linear HMM contains too many parameters to be trained adequately.

Results: This work attempts to solve that problem by generating smaller HMMs which precisely model only the conserved regions of the family. These HMMs are constructed from motif models generated by the EM algorithm using the MEME software. Because motif-based HMMs have relatively few parameters, they can be trained using smaller data sets. Studies of short-chain alcohol dehydrogenases and 4Fe-4S ferredoxins support the claim that motif-based HMMs exhibit increased sensitivity and selectivity in database searches, especially when training sets contain few sequences.

Availability: `http://www.sdsc.edu/MEME`

Contact: bgrundy@cs.ucsd.edu

1 Introduction

A hidden Markov model describes a series of observations by a "hidden" stochastic process. Although introduced relatively recently to computational molecular biology [Churchill, 1989], HMMs have been in use for speech recognition for many years [Baker, 1975]. In speech recognition, the series of observations being modeled is a spoken utterance; in computational biology, the series of observations is a biological sequence. One immediately apparent difference between these two domains is the amount of available training data. Training sets for state-of-the-art speech recognition systems can contain many gigabytes of recorded speech; in contrast, families of related biological sequences usually consist of kilobytes or even hundreds of bytes of characters. Even for speech recognition systems, for which the training set size is relatively large, researchers attempt to simplify their models in order to reduce the number of trainable parameters [Woodland *et al.*, 1994]. When modeling biological sequences, the need for smaller models is even more pronounced. This paper addresses that need by developing hidden Markov models which precisely model only the highly conserved regions of a family of sequences.

These motif-based HMMs consist primarily of motif models generated by MEME (Multiple EM for Motif Elicitation) [Bailey and Elkan, 1995a; Bailey and Elkan, 1995b]. Meta-MEME is a software tool for combining MEME motif models within a standard linear HMM framework. Because Meta-MEME operates in an automated fashion, it is particularly useful for analyzing the increasingly large sequence databases becoming available.

In addition to being trainable from smaller data sets, motif-based HMMs are well suited for recognizing distant homologies. By modeling the spacer regions between motifs in a very simple way, these models selectively discard information from the training set about the contents of spacer regions. This discarding of information is beneficial for distantly related sequences, because distant homologs typically show conservation only in functionally or structurally important portions of their sequences. Meta-MEME focuses on these regions and does not attempt to model the less-conserved, intermediate regions in detail.

In many ways, Meta-MEME resembles the BLOCKS method for protein family classification [Henikoff and Henikoff, 1994b; Henikoff and Henikoff, 1996]. The BLOCKMAKER program discovers highly conserved regions of protein families by combining motifs found by either the MOTIF algorithm [Smith *et al.*, 1990] or the Gibbs sampling algorithm [Lawrence *et al.*, 1993]. Individual blocks may be represented as ungapped position-specific scoring matrices, similar to the motif models created by MEME. However, MEME is more likely than BLOCKMAKER to split a motif in two if any of the sequences contain an insertion or deletion, so MEME motifs tend to be shorter than BLOCKMAKER blocks. Since motifs (and blocks) are supposed to model ungapped regions, MEME generally produces more accurate models. The BLOCKS database [BLOCKS, 1996] contains, for each known protein family, an ordered set of blocks along with the minimum and maximum observed spacings between the blocks in the training set. The BLIMPS program [Henikoff *et al.*, 1995] searches this database using a single sequence as a query, thus taking into account the order and spacing of blocks. Clearly, Meta-MEME and the BLOCKS method share many features. In general, however, a hidden Markov model approach is more attractive because of its well-founded underlying probabilistic theory.

Hidden Markov models

A hidden Markov model is a mathematical framework which models a series of observations based upon a hypothesized, underlying but hidden process. The model consists of a set of states and transitions between these states. Each state emits a signal based upon a set of emission probabilities and then stochastically transitions to some other state, based upon a set of transition probabilities. These two probability distributions, when combined with the initial state distribution, completely characterize an HMM.

A useful HMM tutorial was written by Rabiner [Rabiner, 1995], and more detailed information is available in [Rabiner and Juang, 1993]. The tutorial describes three basic problems for HMMs: given an observation sequence and a model, how do we (1) efficiently compute the probability of the observation sequence, given the model, (2) choose a corresponding state sequence which is optimal in some meaningful sense (i.e., best "explains" the observations), and (3) adjust the parameters of the model to maximize the probability of the sequence, given the model? In computational biology, an HMM models a family of related sequences. Thus, Rabiner's three problems correspond to (1) determining whether a given sequence belongs to the modeled family, (2) finding an alignment of the given sequence to the rest of the family, and (3) training the model based upon known members of the family.

Standard HMMs for molecular biology

Hidden Markov models were first applied to problems in molecular biology by [Churchill, 1989]. [Krogh *et al.*, 1994] applied HMMs to protein modeling and brought widespread recognition to the approach. We refer to the linear HMMs described in that paper as "standard HMMs". The structure of these HMMs attempts to reflect the process of evolution.

The core of the standard model is a sequence of states, called "match states," which represent the canonical sequence for this family. Each match state corresponds to one position in the canonical sequence. This series of states is similar to a profile [Gribskov *et al.*, 1990], since each state contains a frequency distribution across the entire alphabet. The probabilities that a given state emits each possible base are taken from this frequency distribution and are called the "emission probabilities" for that state.

To model the process of evolution, two additional types of states — insert and delete states — are included in the HMM. One delete state lies in parallel with each match state and allows the match state to be skipped. Since delete state do not emit characters, aligning a sequence to a delete state corresponds to the sequence having a deletion at that position. Insert states with self-loops are juxtaposed between match states, allowing one or more bases to be inserted between two match states. These three series of states are connected as shown in Figure 1. The topology of the model is linear: once a state has been traversed, it cannot be entered a second time. Although this type of model may fail to accurately model genetic copying events, the enforced linearity allows for efficient training of the models.

Standard HMMs have been most successfully applied to the task of characterizing families of proteins containing a relatively large number of known sequences [Krogh *et al.*, 1994; Baldi *et al.*, 1994; Eddy, 1995].

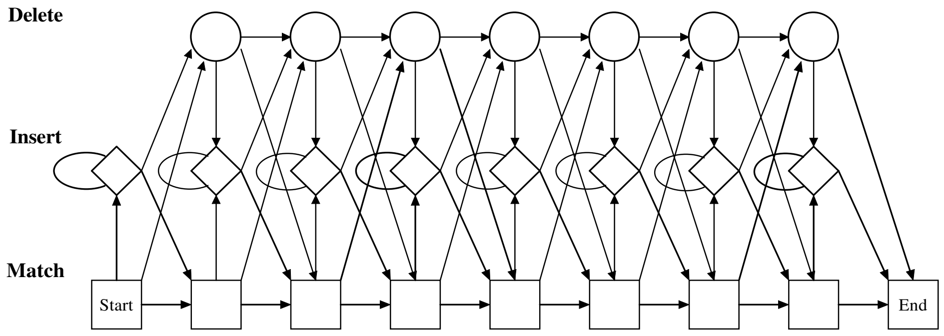


Figure 1: Outline of the topology of a standard linear HMM. Emission probability distributions for match and insert states are not shown.

For families in which fewer sequences are known, a standard HMM contains too many parameters to be trained to precision. A standard HMM of length n using an alphabet of size 20 contains 6 transition probabilities and 19 match state emission probabilities for each of n positions, as well as 19 insert state emission probabilities, yielding a total of $25n + 19$ trainable parameters. For a short sequence of length 100, such a model contains 2519 parameters. Many small families of biological sequences contain less than this number of characters in all known family members combined.

Small families such as these cannot effectively train a standard linear HMM because reliable training requires that the number of samples greatly exceeds the number of free parameters. For example, [Krogh *et al.*, 1994] mention a lower limit of approximately 70 carefully selected training sequences in order to adequately model the globin family. A model based upon a smaller data set may overfit the data, modelling details specific to the training set but not to the larger protein family. In order to avoid overfitting, standard HMMs often rely upon a set of Bayesian prior probabilities [Brown *et al.*, 1995; Sjolander *et al.*, 1996]. In this case, however, with a small training set and a large model, the trained model may depend upon the prior probabilities more than it reflects the training sequences. The only effective means of ensuring that the trained model reflects the characteristics of a particular protein family is to keep the number of model parameters small.

Searching using HMMs

Having constructed an HMM, the model can be applied to the task of recognizing a family of biological sequences in a sequence database. An ideal HMM would pick out all and only the members of the family from the rest of the database. This database search can be carried out using existing software. Two standard HMM packages are freely available, SAM [Hughey and Krogh, 1996; SAM, 1996] and HMMER [Eddy, 1995; HMMER, 1996]. Although the SAM package allows for slightly more complicated models, HMMER is more appropriate for our needs because it includes a variety of searching algorithms.

The results of HMM searches may be compared using a modified form of the receiver operating characteristic (ROC), which we describe in more detail below. We have performed a series of such searches on two different families, using varying training set sizes. The data from these searches show that, for the data sets we investigated, motif-based HMMs perform as well as standard HMMs for large training set sizes and significantly outperform standard HMMs for smaller training sets.

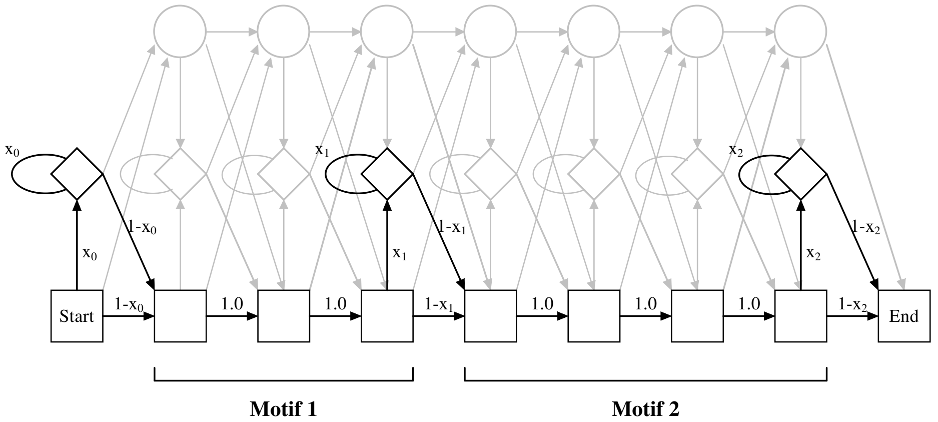


Figure 2: A small motif-based HMM. Only the darker nodes and transitions are used in the model; the gray background nodes would appear in a standard HMM but are unreachable in this HMM. Note that this is a simplified example; real motifs generated by MEME are longer.

2 Algorithm

Overview of the algorithm

Meta-MEME is a software tool for creating hidden Markov models which focus on highly conserved regions, called motifs. Because of their relatively small size, these motif-based HMMs address the problems caused by insufficient training data.

Meta-MEME currently uses motif models as generated by MEME, a tool which uses expectation-maximization to discover motifs in sets of DNA or protein sequences. Given such a set of sequences, MEME outputs one or more probabilistic models of motifs found in the data. The models consist of a frequency matrix and are therefore similar to a gapless profile. A parallelized version of MEME running on a supercomputer is available on the World-Wide Web [Grundy *et al.*, 1996; ParaMEME, 1996].

MEME motifs provide reliable indicators of family membership. If trained on a set of related sequences, MEME will build motif models of the most highly conserved regions in that data set. For related sequences, these highly conserved regions represent evidence of the sequences' shared evolutionary history. A candidate sequence which closely matches the other members of the family in motif regions is much more likely to be homologous than a candidate for which the match lies in a region of lower conservation. The motifs therefore provide a concise signature for the family. Because MEME can find such signatures, it is a powerful tool for recognizing families of proteins. Hidden Markov models provide a framework for combining MEME motifs into an even more accurate and precise recognition tool.

Meta-MEME extends the MEME software to build sequence-length models, rather than models of single motifs. Meta-MEME generates models by first finding a set of motif models and then combining these models within a linear HMM framework. The MAST software, as described below, is used to search a database, finding a schema representing the canonical order and spacing of motifs within the family.

The motif-based hidden Markov models constructed by Meta-MEME are a simplified form of the standard HMM (see Figure 2). The motifs themselves allow neither gaps nor insertions; thus, each motif is modeled by a sequence of match states, with transition probabilities of 1.0 between adjacent states.

The regions between motifs are not modeled very precisely, since the contents of these spacer regions are not highly conserved. Each spacer region is modeled using a single insert state. The transition probabilities into this state and on the state's self-loop are calculated such that the expected length of the emission from this state equals the length of the corresponding spacer region in the canonical motif occurrence schema. The insert state's emission probability distribution is set to a uniform distribution, but this distribution is ignored by the HMMER search tools described below. In effect, then, each spacer region is modeled by a single length parameter. A model of length n containing m motifs therefore contains $19n$ match state emission probabilities and $m + 1$ transition probabilities, for a total of $19n + m + 1$ trainable parameters. In practice, this number will be much smaller than the corresponding number for standard HMMs, since motif-based HMMs contain far fewer match states.

The length of the spacer region is not highly constrained by the model. An insert state gives an exponentially decaying distribution of spacer lengths. For spacers of any appreciable length, that distribution is very flat. Thus, the model should be fairly resilient to insertions or deletions within the spacer regions.

MEME parameters

One of Meta-MEME's primary goals is to operate in a completely unsupervised fashion. While it might be possible and even desirable in many cases to build expert human knowledge into the model of a particular family, the increasing quantity of sequence data available precludes such an approach in general. We have therefore run MEME using its default parameters, as specified on the ParaMEME web site. Specifically, we use the ZOOPS motif occurrence model, which stands for "zero or one occurrence per sequence." Note that, although the resulting model is tuned to find motifs which appear no more than once in each sequence, it may still find repeated motifs. We use Dirichlet mixtures for prior probabilities, modified by the megaprior heuristic [Bailey and Gribskov, 1996]. The minimum width of a motif is specified as 12 (although the motifs returned may be shorter than this, due to a shortening heuristic in MEME), and the maximum width is 55.

Selecting motifs: A majority heuristic

In order for Meta-MEME to build multi-motif models from MEME output in an unsupervised way, the program must decide automatically how many motifs to use. To do so, Meta-MEME uses a simple heuristic. As MEME generates successive motifs for a data set, it first finds the highly significant motifs and then begins to model motifs which are conserved in only a subset of the given sequences. In effect, MEME finds motifs representing subfamilies of the given family. Since such subfamily motifs are not useful for characterizing the entire family, they should not be included in the Meta-MEME model. Models generated by Meta-MEME, therefore, only incorporate those motifs for which the motif occurs in the majority of the training sequences, up to a maximum of six motifs.

Finding the canonical motif occurrence schema

Once the motif models have been generated by MEME and selected according to the majority occurrence heuristic, they must be combined into a single model. In order to use the standard HMM framework, the motifs must be arranged in a linear fashion. Ideally, the order and spacing of motifs should reflect the canonical order and spacing of motifs in the family. The Motif Annotation and Search Tool (MAST) [Bailey and Gribskov, 1997] is part of the MEME software distribution [MEME, 1996]. MAST searches a database for motif occurrences and assigns a score to each sequence based upon the sequence's most likely match to each of the given motifs. The sequences from the database with statistically significant matches to the given set of motifs are returned as part of the MAST output. For each such sequence, MAST produces a motif occurrence schema which shows the motif occurrences with p-values less than 0.0001, as well as the lengths of the spaces between occurrences. Meta-MEME searches this output for the highest-scoring sequence containing significant matches to each of the motifs selected for use in the HMM. The motif occurrence schema associated with this sequence is then used as the canonical schema.

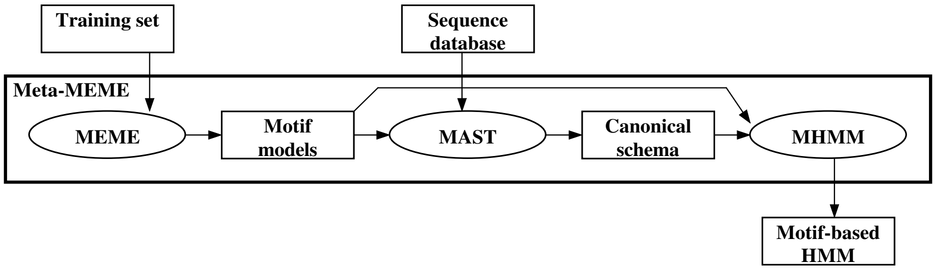


Figure 3: A schematic diagram of Meta-MEME. The primary inputs are a set of sequences and a sequence database. The program produces a linear HMM of the given family in ASCII HMMER format.

Calculating spacer state transition probabilities

The transition probabilities for insert states between motifs must be calculated such that the expected spacer lengths correspond to the values in the canonical motif occurrence schema. Consider an HMM state for which the incoming transition probability is x, the outgoing transition probability is $1-x$, and the probability of a self-loop is x. Let n be the number of times the node is visited. Then the expected number of visits, μ, to such a node is, by definition,

$$\mu = \sum_{n=0}^{\infty} n(1-x)x^n \tag{1}$$

At first there are two possibilities: visit the node with probability x, or skip it with probability $1-x$. Skipping the node gives a spacer of length 0, while visiting it gives a spacer length 1 plus the expected remaining path length, ν. So we have

$$\mu = (1-x)0 + x(1+\nu) \tag{2}$$

Because of the Markov property, regardless of the path length so far, if we reach this node again then the expected path length from it is simply μ. So we have

$$\mu = x(1+\mu) \tag{3}$$

Solving for x yields

$$x = \mu/(1+\mu) \tag{4}$$

This equation is used to calculate transition probabilities for spacer states.

A schematic diagram of Meta-MEME is shown in Figure 3. Given a set of motif models and the canonical sequences, the program *mhmm* calculates the appropriate spacer state transition probabilities and writes out a linear, motif-based HMM in HMMER format.

3 Results

Data sets

We first applied Meta-MEME to a group of dehydrogenases that includes mammalian 11β-hydroxysteroid and 17β-hydroxysteroid dehydrogenase and their homologs in the short chain alcohol dehydrogenase family. We chose this data set because it is large and phylogenetically diverse [Persson *et al.*, 1991; Baker, 1994;

Baker, 1996], providing a good test of the sensitivity and selectivity of Meta-MEME on a protein family of biological interest.

The thirty-eight sequences used in the training set are listed in Appendix A. Pairwise alignments of almost all of these sequences are less than 30% identical after using gaps and insertions to maximize identities. Many sequences are less than 20% identical after use of gaps and insertions. These thirty-eight sequences represent a small portion of the approximately 650 known dehydrogenases in genpept release 95 [GenBank, 1996].

We also applied Meta-MEME to a set of 4Fe-4S ferredoxins. The family members are listed in Appendix B. These 159 sequences comprise all known 4Fe-4S ferredoxins in SWISSPROT release 33 [Bairoch, 1994]. Family members were selected using PROSITE 13.1 [Bairoch, 1992]. Ten additional members were added to the family, based upon ROC analysis and sequence comparisons. The SWISSPROT identifiers for all 159 sequences, as well as the justifications for including the ten additional sequences, are given in Appendix B. Nested training sets were selected at random from all 159 sequences, without regard to sequence similarity.

Creating standard linear HMMs

The standard linear HMMs used for comparison with Meta-MEME were constructed using the default settings of the HMMER program *hmmt*, version 1.8. The training algorithm begins with a uniform model with length equal to the average length of sequences in the training set. The model is trained via expectation-maximization, using a simulated annealing protocol to avoid local optima. The initial Boltzmann temperature is 5.0, with a temperature decrease of 5% at each iteration.

Smith/Waterman search

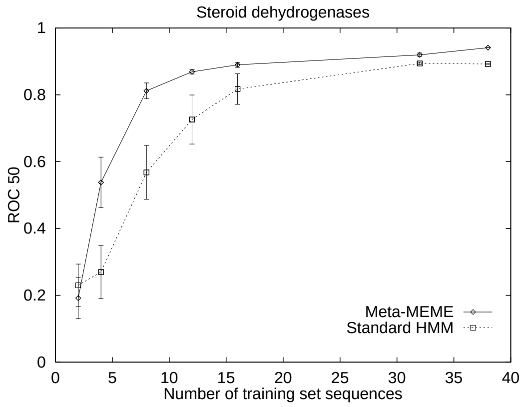
Numerous algorithms exist for searching a database using a hidden Markov model. HMMER offers four such programs, which vary in the way they match sequences against models. The first, *hmmsw*, performs a local Smith/Waterman search for matches of a partial sequence to a partial model; *hmms* matches a complete model against complete sequences; *hmmls* matches a complete model against one or more partial sequences; and *hmmfs* matches fragments of a model to multiple non-overlapping partial sequences. Informal experiments with these programs yielded consistently better results using *hmmsw*.

In the best case, a database search with an HMM would return sequence scores which ranked all of the family members above all of the non-family members. However, all of the HMMER programs suffered from intermediate-scoring sequence fragments. When a sequence fragment exists in the database, it will match only a portion of the model, giving a relatively low score. Then, even though the fragment is a member of the family, it may be ranked among the non-family members.

Because sequence fragments are a deficiency of the database rather than of the search method, and because many fragments are redundant with the whole sequences included in the database, we opted to filter such fragments from the database. Rather than use a fixed threshold for all models, we calculated from the canonical motif signature the minimum length of a sequence containing two motifs and two spacers. All sequences in the database shorter than this value are filtered out. The filtered database is then used for both the Meta-MEME search and the standard HMM search.

Comparing search results: ROC_{50}

We compare search results using a modified form of the receiver operating characteristic. The ROC curve plots true positives as a function of true negatives using a continuously varying decision threshold. The area under this curve, the ROC value, combines measures of a search's selectivity and sensitivity



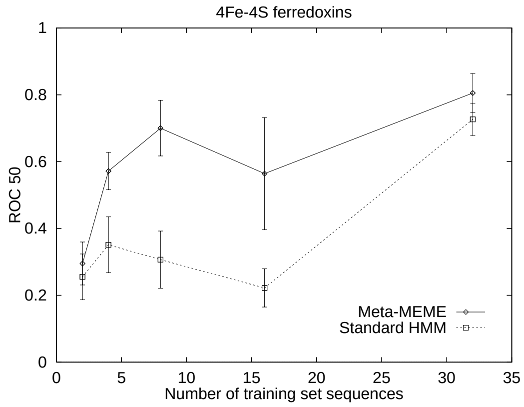


Figure 4: Comparison of Meta-MEME and standard linear HMMs in recognizing (a) short-chain alochol dehydrogenases and (b) 4Fe-4S ferredoxins. Each point represents an average of ten separate runs, except for the ferredoxin runs using 16-sequence training sets, for which only three runs completed (see the discussion below). Error bars represent standard error.

into a single value. Unfortunately, for large database searches, the number of negatives far exceeds the number of positives, so ROC values must be computed to a high degree of precision. A similar statistic, ROC_{50} [Gribskov and Robinson, 1996], provides a wider spread of values. ROC_{50} is the area under the ROC curve plotted until 50 false positives are found. This value has the advantages of being easier to compute, of requiring less storage space, and of corresponding to the typical biologist's willingness to sift through only approximately fifty false positives. ROC_{50} scores are normalized to range from 0.0 to 1.0, with 1.0 corresponding to the most sensitive and selective search.

Short-chain alcohol dehydrogenases

Figure 4(a) shows that Meta-MEME outperforms standard linear HMMs for most subsets of the dehydrogenase training set, with the most striking difference between the two methods appearing for smaller data sets. Each series in the figure represents the average of ten successions of training and testing runs, using randomly selected, nested subsets of the 38-sequence training set. Error bars represent standard error. For each subset of sequences, a standard and a motif-based HMM were built and were used to search genpept 95. Not only does Meta-MEME consistently score better than the standard linear HMMs, the motif-based HMMs appear to be more robust across different random subsets, as evidenced by the relative smoothness of the Meta-MEME curve.

Figure 5 shows an "alignment" of four different motif-based HMMs, built from nested subsets of the dehydrogenase training set. These motifs illustrate the biological basis for the sensitivity of Meta-MEME. Motifs 1 and 2 are part of the nucleotide cofactor binding site [Branden and Tooze, 1991; Wierenga *et al.*, 1985; Wierenga *et al.*, 1986]; motif 3 is part of the catalytic site. A protein sequence that had, for example, motifs 1 and 3 interchanged would not have the same 3D structure and could not function as a steroid dehydrogenase. By scoring protein similarity and dissimilarity on the basis of motif order and spacing, Meta-MEME effectively models spatial information in the 3D structure of the canonical dehydrogenase. This information differentiates homologs from unrelated proteins which contain isolated fragments resembling sequences in the training set. Comparison of protein 3D structures is the most sensitive method for determining homology [Chothia and Lesk, 1986]. This explains Meta-MEME's excellent ability to recognize alcohol dehydrogenase homologs as seen in Figure 4(a).

The motifs discovered using smaller training sets correspond strongly to the original motifs found using

38 sequences: 9-[2]-64-[1]-12-[6]-17-[4]-9-[3]-73

```
---------LVTGAASGIG----------------------------------------------------------
------VDVLVNNAG*------------EDWDRVIxVNLTGVF*-----------------GRIVNVSSVAG-----
----YSASKAAVxGLTRSLALELAPxGIRVNVVAPG-----------------------------------------
--------------------------------
```

16 sequences: 5-[2]-61-[1]-42-[4]-12-[3a]-5-[3b]-33-[5]-13

```
****-----LVTGASRGIG****------------------------------------------------------
-------DVLVNNAG****------------------------------------------GRIVNVSS--------
----YSASKAALxGLTRSLALE-----IRVNAVAPGFVxTDM---------------------------------FL
ASDEASYIT-------------**********
```

8 sequences: 11-[2]-65-[1]-64-[3a]-22-[3b]-26-[7]-28

```
-----------TGASSGIG----------------------------------------------------------
-------DVLVNNAG**------------------------------------------------------------
----YAASKAAL----------------------PGxIxTDM--------------------------IPIGRMGQP
EEIA---------------------------*
```

4 sequences: 13-[1]-18-[6]-37-[3a]-22-[3b]-41

```
***********************************************************************------
-------DALINNAG------------------VFHINVVGPIR---------------------------------
----YxMSKAAL----------------------PGWVxTDM-----------------------------------
------**************************
```

Figure 5: Comparison of four motif-based HMMs built from a nested series of random subsets of the 38-sequence dehydrogenase training set. The canonical schema for each model is shown, with the lengths of spacers alternating with motif numbers in brackets. In the models, motifs are represented by their consensus sequence. Hyphens ("–") represent the expected length of spacers generated by insert nodes, and asterisks ("*") are gaps inserted into this diagram in order to align the models.

the largest training set. In the figure, motifs are numbered consecutively according to the order in which they were discovered. Any motif from one training set which overlaps with a motif from a previous training set is assigned the same number as the first. Using the largest training set, MEME finds five motifs which appear in more than half of the training set. The third of these motifs, however, is very long (32 residues); in subsequent analyses using smaller data sets, motif 3 gets split into two halves (marked 3a and 3b). Furthermore, motif 5, which was discarded because of the majority occurrence heuristic in the 38-sequence analysis, is found and included in the HMM based upon sixteen sequences. Motif 6 is lost when the training set is reduced from thirty-eight to sixteen sequences but is recovered when the training set size reaches 4 sequences. Motifs 4 and 5 are lost between sixteen and eight sequences, and motif 2 is lost when four sequences are used. Only one new motif (marked 7) is introduced in the smaller training sets; other candidates are discarded because of the majority occurrence heuristic.

The order and spacing of the motifs within the different models is also conserved. In all four models, the order of motifs is identical. Furthermore, spaces between motifs are consistent across the four models. In the figure, hyphens represent spacer states in the model, whereas asterisks represent "gaps," which were inserted into the figure in order to align the motifs. Very few asterisks were required in order to generate a perfect alignment. Only the last model, based upon four training sequences, contains a significant missing portion.

The motif-based HMMs are considerably smaller than their standard HMM counterparts. For the dehydrogenase family, the average model length from Meta-MEME contains 58 states; the standard models average 264 states. Assuming six motifs per model, the average Meta-MEME model therefore contains $(19 * 58) + 6 + 1 = 1109$ trainable parameters. The standard HMM, by contrast, averages $25 * 264 = 6600$ parameters. The standard model is therefore 6.0 times as large as the motif-based model.

4Fe-4S ferredoxins

A similar set of experiments was conducted using the 4Fe-4S ferredoxin data set. In addition to using a different, considerably smaller family, the ferredoxin searches were carried out on a different database, SWISSPROT 33 instead of genpept 95. Nonetheless, Meta-MEME again consistently outperforms the standard HMMs, as shown in Figure 4(b). The degree of separation between the two series is even greater than for the dehydrogenases. The standard HMMs of the ferredoxin family are on average 5.1 times as large as the average motif-based HMM.

Although Meta-MEME outperforms standard HMMs, both methods perform more poorly for ferredoxin data sets of size 16 than for smaller, 8- or 4-sequence data sets. This anomaly results from the interaction of two of the heuristics described above. For many of the 16-sequence data sets, the majority occurence heuristic selected a relatively large number of motifs. Unfortunately, it was often impossible for MAST to locate a single sequence containing all of these motifs. Consequently, a canonical motif occurence schema was found for only three of the runs. As a result, neither Meta-MEME nor HMMER completed the other runs, since the filtering of the database depends upon the canonical schema. This adverse interaction of heuristics only occurred with the ferredoxin data set and only with training sets of size 16. A variant of our heuristics would overcome this problem; however, our emphasis in this work is to demonstrate the general utility of motif-based HMMs. Rather than fine-tuning heuristics, future work will replace these heuristics by, for example, completely connecting the motifs and learning the occurrence schema from the given data.

4 Discussion

Results from Meta-MEME are encouraging. As expected, motif-based HMMs discriminate better than their standard linear counterparts for the two protein families we investigated, yet due to their small size, motif-based HMMs require fewer training sequences in order to be trained to precision. Furthermore, since HMM search algorithms are generally linear in the size of the model, motif-based HMMs can search a database 5-6 times faster than a standard model. By focusing its models on highly conserved regions of the training set, Meta-MEME effectively ignores noisy portions of the data, thereby allowing the software to recognize distant homologs. Finally, because Meta-MEME operates in an unsupervised fashion, the software is appropriate for the analysis of large databases, where domain-specific expert knowledge may not be available for every family.

Meta-MEME's performance may be affected by biases in the training set. In the experiments reported here, the dehydrogenase training set was hand-selected so as to fairly uniformly represent a particular protein family. However, in the ferredoxin experiments, randomly selected training sets containing several closely related sequences may have biased some of the trained ferredoxin models. These biases would explain the relatively large standard error bars in Figure 4(b). Such biases could have been reduced by first removing highly similar sequences using a program such as PURGE [Neuwald and Green, 1994]. In addition to reducing training set bias, this approach reduces the amount of computation required during training. Several researchers have shown that weighting schemes, which attempt to compensate for bias in the training set by assigning weights to individual sequences, may significantly improve the performance of database searching algorithms [Henikoff and Henikoff, 1994a; Altschul *et al.*, 1989; Sibbald and Argos, 1990; Thompson *et al.*, 1994]. [Eddy *et al.*, 1995] have developed a maximum discrimination training algorithm for hidden Markov models which addresses the same problem. Use of such methods may also provide a means of improving Meta-MEME's performance.

We hope to improve Meta-MEME's models in several ways. First, we will use them as initialization for standard HMM training. This method will allow the motif-based HMMs to be tuned more precisely to the training set. Second, we plan to improve the modeling of spacer regions. A standard HMM insert state gives an exponential distribution of gap lengths, which is not biologically realistic. In order to model spacer lengths more realistically, we will include at each insert state an explicit probability distribution for its output length. In addition, we will investigate improved methods for choosing the number of motifs to include in each model.

Eventually, we hope that motif-based HMMs can address another problem faced by linear HMMs: their inability to adequately model sequence families containing large-scale copying of domains. The linearity of motif-based HMMs may be removed if the motif models are completely connected to one another. Because the total number of motifs is small, such a model may still be trained effectively. This generalized HMM will allow a sequence to possess occurrences of the motifs in any order. For each pair of motifs, the HMM will learn the probability of the second motif following the first motif directly. If, as is typical, one ordering of the motifs is most common, the trained HMM will assign a higher probability to a sequence that has the motifs in this order.

Acknowledgments

The authors would like to thank Michael Gribskov for providing a carefully curated 4Fe-4S ferredoxin data set, and the reviewers for useful comments and suggestions. William Grundy is funded by the National Defense Science and Engineering Grant Fellowship Program. Charles Elkan is funded by a Hellman Faculty Fellowship from UCSD. Timothy L. Bailey was supported by the National Biomedical Computation Resource, an NIH/NCRR funded research resource (P41 RR-08605), and the NSF through cooperative agreement ASC-02827. Paragon time was made available through a grant to NBCR (NIH P41 RR08605).

References

[Altschul *et al.*, 1989] S. F. Altschul, R. J. Carroll, and D. J. Lipman. Weights for data related by a tree. *Journal of Molecular Biology*, 207(4):647–53, 1989.

[Bailey and Elkan, 1995a] T. L. Bailey and C. P. Elkan. Unsupervised learning of multiple motifs in biopolymers using EM. *Machine Learning*, 21(1–2):51–80, October 1995.

[Bailey and Elkan, 1995b] T. L. Bailey and C. P. Elkan. The value of prior knowledge in discovering motifs with MEME. In C. Rawlings et al., editor, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 21–29. AAAI Press, 1995.

[Bailey and Gribskov, 1996] T. L. Bailey and M. Gribskov. The megaprior heuristic for discovering protein sequence patterns. In D. J. States, P. Agarwal, T. Gaasterland, L. Hunter, and R. Smith, editors, *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*, pages 15–24. AAAI Press, 1996.

[Bailey and Gribskov, 1997] T. L. Bailey and M. Gribskov. MAST — motif alignment and search tool. In preparation, 1997.

[Bairoch, 1992] A. Bairoch. PROSITE: a dictionary of sites and patterns in proteins. *Nucleic Acids Research*, 20:2013–2018, 1992.

[Bairoch, 1994] A. Bairoch. The SWISS-PROT protein sequence data bank: current status. *Nucleic Acids Research*, 22(17):3578–3580, September 1994.

[Baker, 1975] J. K. Baker. The dragon system — an overview. *IEEE Trans. Acoust. Speech Signal Processing*, ASSP-23(1):24–29, February 1975.

[Baker, 1994] M. E. Baker. Sequence analysis of steroid and prostaglandin metabolizing enzymes: application to understanding catalysis. *Steroids*, 59:248–258, 1994.

[Baker, 1996] M. E. Baker. Unusual evolution of mammalian 11β- and 17β-hydroxysteroid and retinol dehydrogenases. *Bioessays*, 18:63–70, 1996.

[Baldi *et al.*, 1994] P. Baldi, Y. Chauvin, T. Hunkapiller, and M. A. McClure. Hidden Markov models of biological primary sequence information. *Proceedings of the National Academy of Sciences of the United States of America*, 91(3):1059–1063, 1994.

[BLOCKS, 1996] Blocks WWW server. `http://www.blocks.fhcrc.org/`, 1996.

[Branden and Tooze, 1991] C. Branden and J. Tooze. *Introduction to protein structure*. Garland, 1991.

[Brown *et al.*, 1995] M. Brown, R. Hughey, A. Krogh, I. Mian, K. Sjolander, and D. Haussler. Using dirichlet mixture priors to derive hidden Markov models for protein families. In C. Rawlings et al., editor, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 47–55. AAAI Press, 1995.

[Chothia and Lesk, 1986] C. Chothia and A. M. Lesk. The relation between the divergence of sequence and structure in proteins. *EMBO Journal*, 5:823–826, 1986.

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

[Eddy *et al.*, 1995] S. R. Eddy, G. Mitchison, and R. Durbin. Maximum discrimination hidden Markov models of sequence consensus. *Journal of Computational Biology*, 2:9–23, 1995.

[Eddy, 1995] S. R. Eddy. Multiple alignment using hidden Markov models. In C. Rawlings et al., editor, *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*, pages 114–120. AAAI Press, 1995.

[GenBank, 1996] GenBank overview. `http://www.ncbi.nlm.nih.gov/Web/Genbank/index.html`, 1996.

[Gribskov and Robinson, 1996] M. Gribskov and N. L. Robinson. Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. *Computers and Chemistry*, 20(1):25–33, 1996.

[Gribskov *et al.*, 1990] M. Gribskov, R. Lüthy, and D. Eisenberg. Profile analysis. *Methods in Enzymology*, 183:146–159, 1990.

[Grundy *et al.*, 1996] W. N. Grundy, T. L. Bailey, and C. P. Elkan. ParaMEME: A parallel implementation and a web interface for a DNA and protein motif discovery tool. *CABIOS*, 12(4):303–310, 1996.

[Henikoff and Henikoff, 1994a] S. Henikoff and J. G. Henikoff. Position-based sequence weights. *Journal of Molecular Biology*, 243:574–578, 1994.

[Henikoff and Henikoff, 1994b] S. Henikoff and J. G. Henikoff. Protein family classification based on searching a database of blocks. *Genomics*, 19:97–107, 1994.

[Henikoff and Henikoff, 1996] J. G. Henikoff and S. Henikoff. Blocks database and its applications. *Methods in Enzymology*, 266, 1996.

[Henikoff *et al.*, 1995] S. Henikoff, J. G. Henikoff, W. J. Alford, and S. Pietrokovski. Automated construction and graphical presentation of protein blocks from unaligned sequences. *Gene-COMBIS, Gene*, 163(GC):17–26, 1995.

[HMMER, 1996] S. R. Eddy group, Dept. of Genetics, Washington University. `http://genome.wustl.edu/eddy/hmm.html`, 1996.

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

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

[Lawrence *et al.*, 1993] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton. Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science*, 262(5131):208–214, 1993.

[MEME, 1996] MEME ANSI C source code. `ftp://cs.ucsd.edu/pub/tbailey/meme`, 1996.

[Neuwald and Green, 1994] A. F. Neuwald and P. Green. Detecting patterns in protein sequences. *Journal of Molecular Biology*, 239(5):698–712, 1994.

[ParaMEME, 1996] MEME - multiple EM for motif elicitation. `http://www.sdsc.edu/MEME`, 1996.

[Persson *et al.*, 1991] B. Persson, M. Krook, and H. Jornvall. Characteristics of short chain alcohol dehydrogenases and related enzymes. *European Journal of Biochemistry*, 200(53):7–543, 1991.

[Rabiner and Juang, 1993] L. R. Rabiner and B. Juang. *Fundamentals of speech recognition*. Prentice Hall, 1993.

[Rabiner, 1995] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1995.

[SAM, 1996] SAM: Sequence alignment and modeling system. `http://www.cse.ucsc.edu/research/compbio/sam.html`, 1996.

[Sibbald and Argos, 1990] P. R. Sibbald and P. Argos. Weighting aligned protein or nucleic acid sequences to correct for unequal representation. *Journal of Molecular Biology*, 216(4):813–8, 1990.

[Sjolander *et al.*, 1996] K. Sjolander, K. Karplus, M. Brown, R. Hughey, A. Krogh, I. S. Mian, and D. Haussler. Dirichlet mixtures: A method for improving detection of weak but significant protein sequence homology. *CABIOS*, 1996.

[Smith *et al.*, 1990] H. O. Smith, T. M. Annau, and S. Chandrasegaran. Finding sequence motifs in groups of functionally related proteins. *Proceedings of the National Academy of Sciences of the United States of America*, 87:826–830, 1990.

[Thompson *et al.*, 1994] J. D. Thompson, D. G. Higgins, and T. J. Gibson. Improved sensitivity of profile searches through the use of sequence weights and gap excision. *CABIOS*, 10(1):19–29, 1994.

[Wierenga *et al.*, 1985] R. K. Wierenga, M. C. De Maeyer, and W. G. J. Hol. Interaction of pyrophosphate moieties with α-helices in dinucleotide binding proteins. *Biochemistry*, 24:1346–1357, 1985.

[Wierenga *et al.*, 1986] R. K. Wierenga, P. P. Terpstra, and W. G. J. Hol. Prediction of the occurrence of the ADP-binding β-α-β-fold in proteins using an amino acid sequence fingerprint. *Journal of Molecular Biology*, 187:101–107, 1986.

[Woodland *et al.*, 1994] P. C. Woodland, J. J. Odell, V. Valtchev, and S. J. Young. Large vocabulary continuous speech recognition using HTK. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 125–128. IEEE, 1994.

A Short chain alcohol dehydrogenases

2BHD_STREX	20-Beta-Hydroxysteroid Dehydrogenase
3BHD_COMTE	3-Beta-Hydroxysteroid Dehydrogenase
ACT3_STRCO	Putative Ketoacyl Reductase
ADH_DROME	Alcohol Dehydrogenase
AP27_MOUSE	Adipocyte P27 Protein (AP27).
BA72_EUBSP	7-Alpha-Hydroxysteroid Dehydrogenase
BDH_HUMAN	D-Beta-Hydroxybutyrate Dehydrogenase Precursor
BEND_ACICA	Cis-1,2-Dihydroxy-3,4-Cyclohexadiene-1-Carboxylate Dehydrogenase
BPHB_PSEPS	Biphenyl-2,3-Dihydro-2,3-Diol Dehydrogenase
BUDC_KLETE	Acetoin(Diacetyl) Reductase
CSGA_MYXXA	C-Factor.
DHB2_HUMAN	Estradiol 17 Beta-Dehydrogenase 2
DHB3_HUMAN	Estradiol 17 Beta-Dehydrogenase 3
DHCA_HUMAN	Carbonyl Reductase (NADPH)
DHES_HUMAN	Estradiol 17 Beta-Dehydrogenase
DHGB_BACME	Glucose 1-Dehydrogenase B
DHII_HUMAN	Corticosteroid 11-Beta-Dehydrogenase
DHMA_FLAS1	N-Acylmannosamine 1-Dehydrogenase
ENTA_ECOLI	2,3-Dihydro-2,3-Dihydroxybenzoate Dehydrogenase
FABG_ECOLI	3-Oxoacyl-[Acyl-Carrier Protein] Reductase
FABI_ECOLI	Enoyl-[Acyl-Carrier-Protein] Reductase (NADH)
FIXR_BRAJA	FixR Protein.
FVT1_HUMAN	Follicular Variant Translocation Protein 1 Precursor (FVT-1).
GUTD_ECOLI	Sorbitol-6-Phosphate 2-Dehydrogenase
HDE_CANTR	Hydratase-Dehydrogenase-Epimerase (HDE).
HDHA_ECOLI	7-Alpha-Hydroxysteroid Dehydrogenase
HMTR_LEIMA	H Region Methotrexate Resistance Protein
LIGD_PSEPA	C Alpha-Dehydrogenase
MAS1_AGRRA	Agropine Synthesis Reductase.
NODG_RHIME	Nodulation Protein G (Host-Specificity Of Nodulation Protein C).
PCR_PEA	Protochorophyllide Reductase Precursor
PGDH_HUMAN	15-Hydroxyprostaglandin Dehydrogenase (NAD(+))
PHBB_ZOORA	Acetoacetyl-Coa Reductase
RFBB_NEIGO	dTDP-Glucose 4,6-Dehydratase
RIDH_KLEAE	Ribitol 2-Dehydrogenase
YINL_LISMO	Hypothetical 26.8 Kd Protein In Inla 5'region (ORFA).
YRTP_BACSU	Hypothetical 25.3 Kd Protein In Rtp 5'region (ORF238)
YURA_MYXXA	Hypothetical Protein In Uraa 5'region (Fragment).

SWISSPROT identifiers and descriptions for the 38 dehydrogenase training set.

B 4Fe-4S ferredoxins

FER1_AZOVI	FER2_RHOCA	FER2_RHORU	FER_MYCSM	FER_SACER
FER_STRGR	FER_PSEPU	FER_PSEST	FER_THETH	FER_CLOAC
FER_CLOBU	FER_CLOPA	FER_CLOPE	FER_CLOSP	FER_CLOST
FER_CLOTM	FER_CLOTS	FER_MEGEL	FER_PEPAS	FER1_RHORU
FER_BUTME	FER_CHLLT	FER1_CHLLI	FER2_CHLLI	FER_CHRVI
FER_METBA	FER_METTL	FER_THEAC	FER2_DESDN	FER3_DESAF
FER1_DESVM	FER_ENTHI	FERX_ANASP	FERN_AZOCH	FERV_AZOVI
FDXN_RHILT	FERN_RHIME	FERN_BRAJA	FER1_RHOCA	FER_ALIAC
FER_SULAC	FER1_RHOPA	FERN_AZOVI	FER3_ANAVA	FER3_PLEBO
FER3_RHOCA	FER_CLOTH	FER_DESGI	FER1_DESDN	FER2_DESVM
FER_THELI	FER_THEMA	FIXX_RHILP	FIXX_RHILE	FIXX_RHIME
FIXX_RHILT	PSAC_ANTSP	PSAC_CHLRE	PSAC_CUCSA	PSAC_EUGGR
PSAC_MAIZE	PSAC_MARPO	PSAC_PEA	PSAC_PINTH	PSAC_SPIOL
PSAC_TOBAC	PSAC_WHEAT	PSAC_CYAPA	PSAC_ANASP	PSAC_ANAVA
PSAC_FREDI	PSAC_SYNEN	PSAC_SYNP2	PSAC_SYNP6	PSAC_SYNY3
PSAX_SYNY3	DHSB_BACSU	DHSB_ECOLI	FRDB_ECOLI	FRDB_HAEIN
FRDB_PROVU	YFRA_PROVU	FRDB_WOLSU	FDHB_METFO	FRHG_METTH
FIXG_RHIME	RDXA_RHOSH	PHFL_DESVH	PHFL_DESVO	COOF_RHORU
DMSB_ECOLI	DMSB_HAEIN	YFFE_ECOLI	FDNH_ECOLI	FDOH_ECOLI
FDXH_HAEIN	FDHB_WOLSU	HMC2_DESVH	HMC6_DESVH	ASRA_SALTY
GLPC_ECOLI	GLPC_HAEIN	HYCB_ECOLI	HYCF_ECOLI	HYDN_ECOLI
PHSB_SALTY	PSRB_WOLSU	NRFC_ECOLI	NRFC_HAEIN	NAPF_ECOLI
NAPF_HAEIN	NAPG_ECOLI	NAPG_HAEIN	NAPH_ECOLI	NAPH_HAEIN
YGL5_BACST	YJES_ECOLI	YA43_HAEIN	DHSB_USTMA	DHSB_YEAST
DHSB_SCHPO	DHSB_HUMAN	DHSB_RAT	DHSB_DROME	DHSB_ARATH
MBHT_ECOLI	PHF1_CLOPA	ASRC_SALTY	NUIC_MAIZE	NUIC_MARPO
NUIC_ORYSA	NUIC_TOBAC	NUIC_WHEAT	NUIC_PLEBO	NUIC_SYNY3
NUIM_BOVIN	NUIM_RHOCA	NQO9_PARDE	NUOI_ECOLI	DCMA_METSO
YJJW_ECOLI	FER1_DESAF	FIXX_AZOCA	FIXX_BRAJA	ISP1_TRYBB
NARH_ECOLI	NARY_ECOLI	NIFJ_ANASP	NIFJ_KLEPN	YAAT_ECOLI
FER_METTE	PSAC_ODOSI	YEIA_ECOLI	FER_BACTH	FER_BACST
DHSB_CHOCR	DHSB_CYACA	NARH_BACSU	YWJF_BACSU	

SWISSPROT numbers for the 159 4Fe-4S ferredoxins.

Ten of the sequences above are not included in the PROSITE 13.1 listing for this family. DHSB_CHOCR, DHSB_CYACA, FER_METTE, and PSAC_ODOSI are included here based on homology to PROSITE annotated families in this group, and ROC analysis. ISP1_TRYBB, excluded from this group by PROSITE, appears to be closely related to NADH oxidoreductases in this group as shown by ROC and sequence comparisons (NQQ9, NUIM, NUOI, HYCF, NUIC). NARH_BACSU, NARH_ECOLI and NARY_ECOLI, while showing lower ROC, have excellent 4Fe-4S sequences highly similar to those in DMSB, PHSB, FDNH, HYCB, etc. YEIA_ECOLI is a possible type III ferredoxin and has a very strong ROC. YWJF_BACSU is included in the positives because of high ROC, significant similarity to glycerol-3-phosphate dehydrogenase subunits (GLPC) which are ferredoxins, and clear presence of two appropriate 4Fe-4S binding sequences.