Optimizing the size of the sequence profiles to increase the accuracy of protein sequence alignments generated by profile–profile algorithms

Aleksandar Poleksic* and Mark Fienup
Department of Computer Science, University of Northern Iowa, Cedar Falls, IA, 50614, USA

1 INTRODUCTION

Despite the obvious progress in the structural genomics efforts, protein and DNA remote homology detection and protein three-dimensional structure prediction are still some of the most challenging problems in computational molecular biology. One of the limiting factors for accurate modelling of a protein’s structure from its primary sequence is the quality of the alignment of a query protein to the primary template (John and Sali, 2003). A number of approaches to overcome this limitation have been proposed including the profile-based methods for sequence alignment.

PSI-BLAST (Altschul et al., 1997) builds the so-called ‘position-specific scoring matrix’ (PSSM) by iteratively collecting database sequences related to the query. After the first iteration, the algorithm uses PSSM in place of the query sequence to search for new homologues. The procedure is terminated when no new database sequences are found.

Hidden Markov model (HMM)-based algorithms, such as HMMER (Eddy, 1998) and SAMT02 (Karplus et al., 2001), use an analytical model in place of the PSSM. This explicit probabilistic model allows HMM-based methods to accurately compute the likelihood that an arbitrary database sequence is generated from the model.

Profile–profile methods utilize information about aligned residues in the multiple alignments corresponding to both query and the template sequences. The sequence profiles are typically built with iterative profile-sequence algorithms, such as PSI-BLAST. The score assigned for aligning a pair of profile positions reflects the similarity between the distributions of amino-acid letters in multiple alignments at those positions.

Profile–profile methods are often able to recognize similarity between sequences sharing <20% of identical residues, as evidenced by the large-scale benchmarking experiments such as CASP (Tress et al., 2005; Vincent et al., 2005; Wang et al., 2005) CAFASP (Fischer et al., 2003), and LiveBench (Rychlewski and Fischer, 2005). But, despite their success in detecting distant sequence relationships, many aspects of current profile–profile techniques need further improvements. These include:

(1) The procedures for constructing the sequence profiles.
(2) The design of the alignment scoring function.
(3) The statistical methods for assessing the significance of the raw alignment scores.

In this article we present a new approach to constructing, aligning and scoring alignment profiles. We demonstrate a correlation between the size of the profiles (also called ‘profile thickness’) and the sensitivity of a profile–profile method. More specifically, we show that fixing and optimizing the size of the profiles, by fine tuning the average number of different residue types at various profile positions, can improve the inference of remote relationships between protein sequences. In addition, we prove that utilizing a profile dependent scoring scheme, such as profile derived background frequencies, further improves the detection of similarities between distantly related sequences.
The results of our study are incorporated into a profile–profile alignment method called UNI-FOLD. The performance of UNI-FOLD is tested in the Lindahl and SCOP benchmarks for fold recognition and in the SALIGN benchmark for the alignment accuracy.

2 METHODS

2.1 Scoring function

2.1.1 Preliminaries The difference between sequence–sequence and profile–profile methods is in the scoring scheme. In a profile–profile method, the score for aligning profile positions \(i\) and \(j\) reflects the similarity between the amino-acid probability distributions \(q'\) and \(q''\) at the positions \(i\) and \(j\). The simplest such measure is the dot product, which represents the summation of the products of the amino-acid frequencies:

\[
\text{Dot}(q', q'') = \sum_{k=1}^{20} q'_k q''_k
\]

(1)

Dot product is implemented in a number of profile–profile methods, including FFAS03 (Jaroszewski et al., 2005) and ORFeus (Ginalski et al., 2003).1

More complicated scoring functions rely on the background model when scoring pairs of profile positions (Debe et al. 2006; Sadreyev and Grishin, 2003; Yona and Levitt, 2002). The scoring function in PROF_SIM (Yona and Levitt, 2002) uses the information theory measure called Jensen–Shannon (JS) divergence. The JS divergence between two probability distributions \(q'\) and \(q''\) is given by:

\[
D_{\text{JS}}(q', q'') = \frac{1}{2} D_{\text{KL}}(q', r) + \frac{1}{2} D_{\text{KL}}(q'', r)
\]

(2)

where \(D_{\text{KL}}\) represents Kullback–Leibler divergence (relative entropy):

\[
D_{\text{KL}}(q', q'') = \sum_{k=1}^{20} q'_k \log \frac{q'_k}{q''_k}
\]

(3)

and \(r\) is the ‘most likely’ common source distribution for \(q'\) and \(q''\):

\[
r = \frac{1}{2} q' + \frac{1}{2} q''
\]

(4)

The score for aligning two profile positions in PROF_SIM is defined as:

\[
\text{Score}(q', q'') = \frac{1}{2}(1 - J)(1 + S)
\]

(5)

where \(J = D_{\text{JS}}(q', q'')\), \(S = D_{\text{BS}}(r, b)\), and \(b\) is the ‘overall’ (background) distribution of 20 amino-acid letters.

It is easy to see that the score assigned for matching a pair of amino-acid distributions \(q'\) and \(q''\) in PROF_SIM depends not only on the similarity between \(q'\) and \(q''\), but also on the similarity between their most likely source distribution \(r\) and the background distribution \(b\). In other words, even the score between two similar distributions \(q'\) and \(q''\) (\(J\) close to 0) will not be very high unless both of them are far apart from the background distribution (\(S\) close to 1).

The COMPASS method (Sadreyev and Grishin, 2003) uses the effective amino-acid counts when scoring a pair of profile positions. COMPASS scoring function is given by:

\[
\text{Score}(i, j) = c_1 \sum_{k=1}^{20} n'_k \ln \frac{n'_k}{b_k} + c_2 \sum_{k=1}^{20} n''_k \ln \frac{n''_k}{b_k}
\]

(6)

where \(n'_k\) and \(n''_k\) are the ‘effective counts’ for the amino acid \(k\) at the positions \(i\) and \(j\), respectively, \(b_k\) is the background probability of \(k\), and \(c_1\) and \(c_2\) are the weighting parameters. The weights \(c_1\) and \(c_2\) are used to balance the contribution of the two terms in formula (6). The values \(c_1\) and \(c_2\) are given by:

\[
c_1 = \frac{\sum_{k=1}^{20} n'_k - 1}{\sum_{k=1}^{20} n'_k + \sum_{k=1}^{20} n''_k - 2}
\]

(7)

\[
c_2 = \frac{\sum_{k=1}^{20} n''_k - 1}{\sum_{k=1}^{20} n'_k + \sum_{k=1}^{20} n''_k - 2}
\]

(8)

One advantage of function (6) is that it separates similar profile columns (positive scores) from non-similar columns (negative scores). It can also be shown that (6) reduces to PSI-BLAST scores if one profile consists of a single sequence.

2.1.2 UNI-FOLD scoring scheme UNI-FOLD scoring scheme is a twofold generalization of (6). UNI-FOLD uses position specific (as opposed to fixed) background frequencies. In addition, UNI-FOLD scoring function takes into account the similarity between the predicted secondary structure states of the residues in two proteins.

Our scoring function can be described as a multinomial trials process, generalized for the non-integer setting. Aside from being intuitive, this description of the scoring function reveals our motivation for incorporating position-specific background frequencies into the alignment process.

In UNI-FOLD, the score for matching the profile positions \(i\) and \(j\) is the sum of the sequence and the structure term:

\[
\text{Score}(i, j) = \text{seq}(i, j) + \text{struct}(i, j)
\]

(9)

The sequence term is given by:

\[
\text{seq}(i, j) = s(i, j) + s(j, i)
\]

(10)

where

\[
s(i, j) = \log \frac{p_q(n)}{p_q(n)}
\]

(11)

\[
p_q(n) = N_j \prod_{k=1}^{20} \left( \frac{n''_k}{N''_k} \right)^{n''_k}
\]

(12)

and

\[
N_j = \sum_{k=1}^{20} n''_k
\]

(14)

In (11)–(14), \(n''_k\) are the effective amino-acid counts at position \(j\) and \(b''_k\) are the background amino-acid frequencies at position \(i\).

The effective amino-acid letters in column \(j\) can be viewed as the outcomes of a multinomial process in which the total number of trials is \(N_j\) and the probabilities of 20 different outcomes are \(q''_k\), \(\ldots\), \(q''_{20}\). The ( multinomial) probability of observing exactly \(n''_k\) outcomes of each amino-acid \(k\) in the column \(j\) is given by (12). In other words, if the amino-acid distribution \(q'' = [q''_k]_{k=1}^{20}\) is viewed as a multinomial model, then (12) represents the probability that the residue counts in column \(j\) are generated from the model \(q'' = [q''_k]_{k=1}^{20}\). Thus, the score \(s(i, j)\) for
aligning positions \(i\) and \(j\) given by (11) represents the logarithm of the ratio of the probability that the residue counts \(n^i = (n^i_{k=1})_{k=1}^{20}\) are generated by the model \(q^i = (q^i_k)_{k=1}^{20}\) and the probability that the same counts are generated by the background model \(b^i = (b^i_k)_{k=1}^{20}\).

Using some basic properties of logarithms, it can be shown that \(\text{seq}(i,j)\) has the form (6) without the weighting terms \(c_1\) and \(c_2\). We note that using non-balanced terms in (6) makes sense, because the number of effective amino-acid counts at a profile position is an indicator of the ‘confidence’ in the amino-acid probability distribution at that position.

UNI-FOLD scoring function further generalizes (6) by incorporating secondary structure matching and position-specific background frequencies. Using the secondary structure information is known to improve the sensitivity of a fold recognition method (Ginalska et al., 2003; Heringa, 2000; Söding, 2005). The secondary structure score in ORFeus is defined as a shifted dot product between the secondary structure probability vectors. In HHsearch the log-odd-based secondary structure scores are weighted and then added to each amino-acid column-column score.

In UNI-FOLD, the secondary structure contribution to the overall score, \(\text{struct}(i,j)\), is computed using the same set of formulas (10)–(14) (the only difference is that \(k \in \{1, 2, 3\}\) Therefore the secondary structure scoring in UNI-FOLD can also be viewed as a multinomial process with three outcomes, whose probabilities are the frequencies of the three secondary structure states: \(q^i_k\) (helix), \(q^i\) (strand) and \(q^i_{c}\) (coil).^2 However, while the number of effective residues at the position \(j\) in \(\text{seq}(i,j)\) can be derived from the alignments of the amino-acid sequences, for \(\text{struct}(i,j)\) there is no notion of the effective secondary structure counts \(n^i_k\) (helix), \(n^i\) (strand) and \(n^i_{c}\) (coil). In lieu of the observed secondary structure counts, we set each \(n^i_k\), \(n^i\) and \(n^i_{c}\) to a fixed positive number \(c\). We tested several values for \(c\) and found \(c=5\) to work the best in identifying remote relationships between the sequences in a small test set (data not shown).

### 2.2 Background residue probabilities

The sequence score (11) for aligning a pair of positions \(i\) and \(j\) relates the probability that the effective counts \(n^i = (n^i_k)_{k=1}^{20}\) at the position \(j\) are generated from the model \(q^i\) to the probability that \(n^i\) are generated from the background model \(b^i = (b^i_k)_{k=1}^{20}\). In the simplest case, the background frequencies are set to the overall probabilities of 20 amino-acid types (Robinson and Robinson, 1991). However, scoring functions that implement fixed background frequencies are often unable to distinguish true sequence similarities from those that are product of a compositional bias in two profiles. For example, an alanine-rich profile will typically score high against any other alanine-rich profile.

To compensate for compositional bias in protein families, UNI-FOLD uses profile-derived background frequencies. The background frequencies \(b^i = (b^i_k)_{k=1}^{20}\) at the profile position \(i\) are computed by averaging the observed amino-acid frequencies over the remaining positions in the profile, i.e.

\[
b^i_k = \frac{N^i_k}{\sum_{k=1}^{20} N^i_k} \quad (15)
\]

where

\[
N^i_k = \sum_{n^i > 0} n^i_k \quad (16)
\]

In addition to using position-specific amino-acid background frequencies in \(\text{seq}(i,j)\), UNI-FOLD incorporates protein-specific background probabilities when calculating \(\text{struct}(i,j)\). In other words, the secondary structure background probabilities in UNI-FOLD are set to the overall probabilities of the three secondary structure states in the protein for which the profile is built.

We note that the idea of using the profile-dependent background frequencies is not new in sequence alignment methods. The profile-specific background models have been successfully used in HMM algorithms (see, for example, Barrett et al., 1997). However, to the best of our knowledge, profile-dependent background frequencies have never been used in profile–profile settings before.

As explained in Barrett et al. (1997), there are other ways to specify the background model when aligning two profiles. For example, the background frequencies can be different for every template (database) profile, but same for each query profile. We have not tested the template-specific background model, but, according to Barrett et al. it is not likely to work very well.

A head-to-head comparison in the performance between the scoring function that uses Robinson and Robinson background frequencies and the function that uses profile-derived background frequencies is given in the ‘Results’ section.

### 2.3 Alignment score significance

UNI-FOLD raw alignment scores are normalized by computing the \(P\)-values from the Gumbel type distributions of the background alignment scores. The Gumbel parameters \(\lambda\) and \(\mu\) are computed independently for each profile by aligning the profile to a representative set of 2500 sequence profiles computed for the FSSP family representatives (Holm et al., 1992).

The significance of a raw alignment score between the query and the template (database) profile is calculated from the Gumbel distribution specified by the template’s parameters \(\lambda\) and \(\mu\). Hence, the \(P\)-value of an alignment score \(S\) between the query and the template in UNI-FOLD represents the probability of getting the score of at least \(S\) in the comparison of the template profile against the profiles built for FSSP family representatives. We emphasize that pre-computing the distribution parameters for the database profiles is a standard technique used in many programs, such as HMMER (Eddy, 1998) and HHsearch (Söding, 2005).

### 2.4 Profile construction

In order for a profile–profile algorithm to recognize a weak similarity between two sequences (such as, for example, the relationship between the sequences that belong to the same SCOP fold), the sequence profiles must include as many diverse family members as possible. However, the same strategy might not be optimal when aligning two highly similar sequences. The problem is that the diverse profiles, i.e. those containing distantly related family members, often include a high percentage of false positives. The presence of false positives can compromise an otherwise accurate alignment between highly similar sequences (Sadreyev and Grishin, 2004).

The profiles in UNI-FOLD are computed using PSI-BLAST, but the number of PSI-BLAST iterations used to build the profiles is varied from one profile to another. More specifically, instead of building the profiles with a fixed number of PSI-BLAST iterations, we require that each profile has the same (or similar) size, as defined below.

We modified the PSI-BLAST program to output the effective number of different residue types observed in columns of the PSI-BLAST multiple alignments. In the PSI-BLAST paper (Altschul et al., 1997), this parameter is denoted by \(N_C\) and is used to weight the contribution of the observed amino-acid counts when computing the score for matching a sequence residue to the profile column \(C\). Since \(N_C\) represents the effective number of different amino-acid letters (including the gap character), the value of \(N_C\) cannot exceed 21. The variability of \(N_C\) in an example profile is illustrated in the Figure 1.
We define the profile size $T$ as the average value of $N_C/C_0$ over all positions predicted to be in regular secondary structure regions of the protein for which the profile is constructed. More precisely, the profile size is defined as

$$T = \frac{1}{n} \sum_{C \neq \text{loop}} (N_C - 1)$$

where $n$ is the number of residues that are not part of a protein’s loop region. We take the average value of $N_C - 1$ over the regular secondary structure regions only, because of the significant variability of the amino-acid types in the protein’s loop structures.

Figure 2 shows the distribution of the profile size ($T$) for profiles constructed for 300 randomly chosen protein sequences by running PSI-BLAST with two, four and eight iterations.

We define the profile size $T$ as the average value of $N_C - 1$ over all positions predicted to be in regular secondary structure regions of the protein for which the profile is constructed. More precisely, the profile size is defined as

$$T = \frac{1}{n} \sum_{C \neq \text{loop}} (N_C - 1)$$

where $n$ is the number of residues that are not part of a protein's loop region. We take the average value of $N_C - 1$ over the regular secondary structure regions only, because of the significant variability of the amino-acid types in the protein’s loop structures.

Figure 2 shows the distribution of the profile size ($T$) for profiles constructed for a set of 300 randomly chosen protein sequences.

Thus, the question arises whether using the profiles that contain fixed (and ‘optimum’) amount of sequence information, as measured by the profile-size parameter $T$, can improve the inference of distant relationships between protein sequences.

We analysed the sensitivity of UNI-FOLD on different collections of profiles built for the set of 312 training sequences chosen at random from the scop_1.71_pdb_40 database (Chandonia et al., 2004). The description of the training set is given in Table 1.3

Each collection of profiles is computed using a different profile size cutoff $T$. More specifically, for each value of $T$, we generated a set of profiles for the training sequences by running up to 10 PSI-BLAST iterations on each sequence, until the size of the profile reaches $T$. If $T$ cannot be reached with the default PSI-BLAST parameters, the PSI-BLAST $h$-value for the inclusion of the database sequences into the profile is relaxed (from 0.001 to 0.005) and the procedure is repeated once again.

The accuracy of UNI-FOLD on each collection of profiles is measured as the percentage of the relationships between the training sequences identified at different SCOP levels (Murzin et al., 1995).

Figure 3 shows the ability of UNI-FOLD to recognize pairs of training sequences that belong to the same SCOP family, superfamily and fold. As seen in this figure, the sensitivity of UNI-FOLD in detecting SCOP family relationships is almost independent of the profile size. For proteins in the same SCOP superfamily, the sensitivity graph resembles the graph of a negative quadratic function, reaching its maximum at $T = 11$. At the fold level, the accuracy of UNI-FOLD is monotone increasing function of $T$.

We excluded from the training set any sequence that belongs to the Lindahl test set. The Lindahl test set is used to assess the performance of UNI-FOLD (see the ‘Results’ section). A full list of training sequences can be found at http://blackhawk.cs.uni.edu/training_set.txt

**Table 1. The training set**

<table>
<thead>
<tr>
<th>Description</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of sequences</td>
<td>312</td>
</tr>
<tr>
<td>Different families</td>
<td>256</td>
</tr>
<tr>
<td>Different superfamilies</td>
<td>226</td>
</tr>
<tr>
<td>Different folds</td>
<td>181</td>
</tr>
<tr>
<td>Possible hits on family level</td>
<td>82</td>
</tr>
<tr>
<td>Possible hits on superfamily level</td>
<td>82</td>
</tr>
<tr>
<td>Possible hits on fold level</td>
<td>95</td>
</tr>
<tr>
<td>Possible false hits</td>
<td>96,360</td>
</tr>
</tbody>
</table>

Fig. 2. The distribution of the profile size parameter $T$ for profiles constructed for 300 randomly chosen protein sequences by running PSI-BLAST with two, four and eight iterations.

Fig. 3. The percentage of SCOP relationships between the training sequences detected by UNI-FOLD as a function of the profile size ($T$).
Table 2. Lindahl benchmark for fold recognition sensitivity

<table>
<thead>
<tr>
<th>Method</th>
<th>Family</th>
<th>Superfamily</th>
<th>Fold</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top 1</td>
<td>Top 5</td>
<td>Top 1</td>
</tr>
<tr>
<td>PSI-BLAST</td>
<td>71.2</td>
<td>72.3</td>
<td>27.4</td>
</tr>
<tr>
<td>HMMER</td>
<td>67.7</td>
<td>73.5</td>
<td>20.7</td>
</tr>
<tr>
<td>SAM-T98</td>
<td>70.1</td>
<td>75.4</td>
<td>28.3</td>
</tr>
<tr>
<td>BLASTLINK</td>
<td>74.6</td>
<td>78.9</td>
<td>29.3</td>
</tr>
<tr>
<td>SSEARCH</td>
<td>68.6</td>
<td>75.5</td>
<td>20.7</td>
</tr>
<tr>
<td>SSHMM</td>
<td>63.1</td>
<td>71.7</td>
<td>18.4</td>
</tr>
<tr>
<td>THREADER</td>
<td>49.2</td>
<td>58.9</td>
<td>10.8</td>
</tr>
<tr>
<td>FUGUE</td>
<td>82.2</td>
<td>85.8</td>
<td>41.9</td>
</tr>
<tr>
<td>RAPTOR</td>
<td>75.2</td>
<td>77.8</td>
<td>39.3</td>
</tr>
<tr>
<td>PROSPECT II</td>
<td>84.1</td>
<td>88.2</td>
<td>52.6</td>
</tr>
<tr>
<td>SPARKS</td>
<td>81.6</td>
<td>88.1</td>
<td>52.5</td>
</tr>
<tr>
<td>SP3</td>
<td>81.6</td>
<td>86.8</td>
<td>55.3</td>
</tr>
<tr>
<td>FOLDpro</td>
<td>85.0</td>
<td>89.9</td>
<td>55.5</td>
</tr>
<tr>
<td>UNI-FOLD</td>
<td>86.8*</td>
<td>92.6*</td>
<td>69.1*</td>
</tr>
</tbody>
</table>

The results for PSI-BLAST, HMMER, SAM-T98, BLASTLINK, SSEARCH and SSHMM are taken from Lindahl and Elofsson (2000); for FUGUE, and THREADER from Shi et al. (2001); for RAPTOR from Xu et al. (2003); for PROSPECT II from Kim et al. (2003); for SPARKS and SP3 from Zhou and Zhou (2005), and for FOLDpro from Cheng and Baldi (2006). The best results are denoted by asterisk.

Table 3. Sensitivity of five versions of UNI-FOLD in the Lindahl benchmark

<table>
<thead>
<tr>
<th>Method</th>
<th>Family</th>
<th>Superfamily</th>
<th>Fold</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top 1</td>
<td>Top 5</td>
<td>Top 1</td>
</tr>
<tr>
<td>UNI-FOLD</td>
<td>83.1</td>
<td>91.2</td>
<td>53.2</td>
</tr>
<tr>
<td>UNI-FOLD'</td>
<td>83.8</td>
<td>90.6</td>
<td>62.2</td>
</tr>
<tr>
<td>UNI-FOLD&quot;</td>
<td>85.0</td>
<td>91.9</td>
<td>68.0</td>
</tr>
<tr>
<td>UNI-FOLD*</td>
<td>86.8*</td>
<td>92.6*</td>
<td>69.1*</td>
</tr>
<tr>
<td>UNI-FOLD**</td>
<td>86.5</td>
<td>91.4</td>
<td>70.0*</td>
</tr>
</tbody>
</table>

The results presented in Tables 2 and 3 represent the fraction of true hits in first and top five ranks identified by different fold recognition methods: PSI-BLAST, HMMER (Eddy, 1998), SAM-T98 (Karplus et al., 1998), BLASTLINK, SSEARCH, SSHMM (Hargbo and Elofsson, 1999), THREADER (Jones et al., 1992), FUGUE (Shi et al., 2001), RAPTOR (Xu et al., 2003), PROSPECT II (Kim et al., 2003), SPARKS (Zhou and Zhou, 2004), SP3 (Zhou and Zhou, 2005) and FOLDpro (Cheng and Baldi, 2006).

UNI-FOLD outperforms the second best method in the Lindahl benchmark by ~2%, 14% and 4%, at the family, superfamily and the fold level, respectively. Our method performs especially well at the SCOP superfamily level, recognizing 69% of related pairs in the top rank and 82% in the top five rank. In addition, 42% of the superfamily relationships identified by UNI-FOLD are not detectable by the popular PSI-BLAST method.

Table 3 shows the contribution of various modifications of our algorithm to the overall sensitivity in the Lindahl test. As seen in Table 3, switching to profiles of fixed (optimal) size improves the sensitivity of UNI-FOLD at all SCOP levels. The benefit of this modification is largest at the SCOP superfamily level (53.2% versus 62.2%).

Our analysis also indicates that using profile derived, as opposed to fixed background frequencies (UNI-FOLD versus UNI-FOLD*) results in an average increase in sensitivity of about 2%.

On the other hand, there is no significant difference in the performance between UNI-FOLD and UNI-FOLD*. The percentage of correct UNI-FOLD matches in the top rank is higher than that of UNI-FOLD* at the family level by 0.3% and at the fold level by 0.6%. On the other hand, UNI-FOLD* detects more superfamily pairs (70.0% versus 69.1%). Although these results sound surprising, a similar finding that downplays the importance of large databases for profile building has been reported before (Lindahl and Elofsson, 2000, p. 620).

The results presented in Tables 2 and 3 represent the fraction of true hits in first and top five ranks identified by different fold recognition methods: PSI-BLAST, HMMER (Eddy, 1998), SAM-T98 (Karplus et al., 1998), BLASTLINK, SSEARCH, SSHMM (Hargbo and Elofsson, 1999), THREADER (Jones et al., 1992), FUGUE (Shi et al., 2001), RAPTOR (Xu et al., 2003), PROSPECT II (Kim et al., 2003), SPARKS (Zhou and Zhou, 2004), SP3 (Zhou and Zhou, 2005) and FOLDpro (Cheng and Baldi, 2006).
recognition methods, but they do not provide any insight into the reliability of those methods. In theory, even the most sensitive method can assign high scores to false hits and low scores to true matches, as long as a true match is always placed at the top of the ranked list.

3.1.2 Reliability One way to assess the reliability of a fold recognition method is to compute specificity–sensitivity plots (Lindahl and Elofsson, 2000). The specificity is defined as the percentage of all predicted positives that are true positives:

$$Spec(s) = \frac{true\_pos(s)}{true\_pos(s) + false\_pos(s)}$$ (18)

The sensitivity is the percentage of true hits that are predicted as positives:

$$Sens(s) = \frac{true\_pos(s)}{true\_pos(s) + false\_neg(s)}$$ (19)

In formulas (18) and (19), $true\_pos(s)$ denotes the number of correct hits with score greater than $s$, $false\_pos(s)$ is the number of false hits with score greater than $s$ and $false\_neg(s)$ represents the number of correct hits with score below $s$.

The specificity–sensitivity curves for FOLDpro and UNI-FOLD are shown in Figures 4-6. Both programs have much better reliability than the profile-sequence methods tested in the Lindahl benchmark (data not shown). For reference, we also include plots for the popular HMMER program.

As seen in Figure 4, the sensitivity of UNI-FOLD at the family level is similar to that of FOLDpro for specificity values between 0 and 0.8, and better for specificity values close to 1. This indicates that, unlike FOLDpro, UNI-FOLD rarely assigns high scores to false matches. At the superfamily level UNI-FOLD has better sensitivity for all specificity values (Fig. 5). Finally, at the fold level (Fig. 6) both UNI-FOLD and FOLDpro have much better tradeoff between the sensitivity and specificity than HMMER. However, no method in our benchmark is able to reliably detect the fold level similarities.

3.2 SCOP benchmark

We emphasize that Lindahl benchmark (Table 2) does not include results for some of the best fold recognition methods available today, such as, for example, HHsearch. HHsearch is a homology detection and alignment method based on the pairwise comparison of profile HMMs. An earlier version of HHsearch (v1.2) performed very well in the last rounds of CASP and LiveBench experiments.

One way to compare HHsearch and UNI-FOLD is to compute a set of HMMs for sequences in the Lindahl set and run all-against-all HHsearch on this set. However, because the Lindahl sequences come from different databases (Lindahl and Elofsson, 2000), constructing HMMs for Lindahl test set (and in particular including DSSP information into the HMMs) requires string matching, which is an involved and error prone procedure. To avoid this, we compared the performance of UNI-FOLD and HHsearch on a different and much larger set of sequences from the latest release of the SCOP database.
Table 4. SCOP benchmark for fold recognition

<table>
<thead>
<tr>
<th>Method</th>
<th>Famil</th>
<th>Superfami</th>
<th>Fold</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top 1</td>
<td>Top 5</td>
<td>Top 1</td>
</tr>
<tr>
<td>HHsearch 1.5.0</td>
<td>80.2</td>
<td>89.0</td>
<td>69.8*</td>
</tr>
<tr>
<td>UNI-FOLD</td>
<td>80.3*</td>
<td>90.4*</td>
<td>68.9</td>
</tr>
</tbody>
</table>

The results of all-against-all comparison between 3950 sequences from SCOP 1.73 (=15.6 million comparisons). The sequences in each pair share less than 20% sequence identity. The best results are denoted by asterisk.

To construct the test set, we first downloaded the models for representative SCOP1.73 sequences from the HHsearch website. An all-against-all comparison between 12 155 sequences in this set results in almost 150 million pairs, with many sequences sharing high percentage of sequence identity (up to 70%). To obtain a more representative and easier to manage test set, we identified a subset of models for which the seed sequences are members of SCOP1.73_20 (obtained by clustering SCOP 1.73 at 20% sequence identity). This set contains 3950 sequences (HMMs), which corresponds to about 15.6 millions pairs in an all-against-all comparison.

Table 4 shows the ability of HHsearch and UNI-FOLD to identify pairs of sequences from the test set that belong to the same SCOP group. As seen in Table 4, UNI-FOLD performs slightly better at the family level, recognizing 80.3% (1746 out of 2175) relationships in the Top1 rank and 90.4% (1967 out of 2175) in the Top5 rank. HHsearch is able to recognize 0.9% more superfamilies in Top1 rank but 1.4% less in Top5 rank. The largest difference between the two methods is seen in the Fold Top5 category: 1010 pairs detected by UNI-FOLD versus 905 detected by HHsearch.

We also note that the parameters of HHSearch were kept at their default values. In this mode HHSearch uses DSSP secondary structure assignment stored in the template’s HMM when aligning two SCOP models. In contrast, UNI-FOLD uses only predicted secondary structure for both sequences. According to Söding (2005), using DSSP (as opposed to predicted) secondary structure results in about 1.2% increase in the sensitivity of the HHsearch program.

3.3 SALIGN benchmark

We compared the accuracy of UNI-FOLD to the accuracy of five methods in the SALIGN benchmark for alignment quality. The SALIGN test set consists of 200 pairs of structurally similar proteins aligned with the structure–structure alignment programme CE. The pairs are selected so that each CE alignment matches at least 50% of residues in both chains, with at least 100 residues aligned in each sequence. In addition, at least 90% of residues of one chain are covered by CE alignment. There are on average 65% of structurally equivalent Cα with an RMSD of 3.5 Å.

The percent CE overlap for UNI-FOLD in SALIGN benchmark is 57.4% (Table 5). This is higher than the accuracy of SP3 (56.6%) and SALIGN (Marti-Renom et al., 2004) (56.4%), despite the fact that both of these programs are optimized for alignment accuracy (SP3 is trained in the ProSup benchmark and SALIGN is trained on a set of CE alignments), while UNI-FOLD is optimized for the fold recognition. For reference, the BLAST program is able to correctly match only 26.1% of all aligned pairs.

The accuracy of UNI-FOLD drops to 56.8% when the fixed background frequencies are used in the scoring scheme. An additional drop of 0.6% is seen when the secondary structure scoring in UNI-FOLD is turned off.

We also note that the length of UNI-FOLD alignments is similar to the alignments generated by CE. The number of residue pairs matched by UNI-FOLD exceeds the number of pairs aligned by CE by 1.8% (48 514 UNI-FOLD pairs versus 47 667 CE pairs).

3.4 Speed

A typical UNI-FOLD search with a query sequence of ~250 residues against the PDB90 database consisting of ~15000 sequences takes about 70 min on a 2.8 GHz Pentium 4 computer. Thus, the running time of UNI-FOLD is better than both the mean (661 min) and the median (287 min) response time for CASP7 servers. However, several well-established servers have better response time than UNI-FOLD, such as HHpred3, shub (Fischer, 2003) and nFOLD (Jones et al., 2005).

4 CONCLUSION

We study the accuracy of a profile–profile alignment method as a function of the size of the alignment profiles. The profile size is defined as the average number of different residue types observed in the columns of the multiple alignment used for profile construction. The relationship between the algorithm's
accuracy and the profile size is used to determine the optimal profile size parameter for the detection of protein sequence relationships in each of the three SCOP groups: family, superfamily and fold.

Our study finds that the sensitivity of a profile-profile method in detecting SCOP family relationships is independent on the profile size. On the other hand, most superfamily relationships can be identified by using the profiles of size of \(~11\). The similarity detection rate at the fold level is directly proportional to the size of the sequence profiles. In other words, most fold level relationships can be identified with rich profiles, built from the multiple alignments consisting of many remotely related protein sequences.

The techniques described in this article are implemented in a profile-profile alignment algorithm UNI-FOLD. UNI-FOLD uses a probabilistic scoring scheme and a profile-specific statistics for estimating the \(P\)-values of the alignment scores. To compensate for the compositional bias in the protein families, the background frequencies in UNI-FOLD are derived from the profile by averaging the observed amino-acid frequencies over the profile's positions.

We tested the sensitivity and reliability of UNI-FOLD utilizing the Lindahl and SCOP benchmarks for fold recognition and SALLIGN benchmark for the alignment accuracy. To address a possible performance bias due to the increasing protein sequence databases, we also measured the sensitivity of a version of our algorithm that uses an older protein sequence database for profile construction and protein secondary structure prediction.

UNI-FOLD outperforms all other methods tested in the Lindahl benchmark in every SCOP category. However, the difference in the performance between the top fold recognition methods at the SCOP family and fold level is only 2-4%. At the SCOP superfamily level, UNI-FOLD is capable of recognizing 69% of all related pairs, while the second best method can detect only 56%. For comparison, the popular PSI-BLAST program is able to recognize only 27% of all superfamily relationships between the sequences in the Lindahl test set.

Our study finds that up to 2% of the improvement in sensitivity of UNI-FOLD is due to utilizing profile-specific background frequencies. Although we have not investigated the impact of the background model on the fold recognition sensitivity outside of the setting described in this article, other authors report their methods do not benefit from using protein-specific background model. The increase in UNI-FOLD sensitivity when switching to profile-specific background frequencies is due to the fact that the compositional bias in protein families in UNI-FOLD is treated in the alignment step. In other words, the added value of using the profile-specific background frequencies might not be as high for methods that eliminate the compositional bias in the profile-construction step, e.g. by filtering the multiple alignments used for profile construction.

UNI-FOLD can be easily incorporated into a consensus method for additional boost in sensitivity and specificity for protein structure prediction. The applicability of our method extends beyond the comparative protein structure modelling, since the algorithm does not use any protein structure information. Thus, we hope that UNI-FOLD will become a valuable tool in protein structure and functional annotation studies.

ACKNOWLEDGEMENTS

This work is partially supported by a Battelle Grant from the Board of Regents, State of Iowa. We gratefully thank Professor Jianlin Cheng for providing us with the benchmark of FOLDpro method.

Conflict of Interest: none declared.

REFERENCES


