Multiple Sequence Alignment Based on Profile Alignment of Intermediate Sequences

Yue Lu[1] and Sing-Hoi Sze[1,2]

[1] Department of Biochemistry & Biophysics
[2] Department of Computer Science,
Texas A&M University, College Station, TX 77843, USA

Abstract. Despite considerable efforts, it remains difficult to obtain accurate multiple sequence alignments. By using additional hits from database search of the input sequences, a few strategies have been proposed to significantly improve alignment accuracy, including the construction of profiles from the hits while performing profile alignment, the inclusion of high scoring hits into the input sequences, the use of intermediate sequence search to link distant homologs, and the use of secondary structure information. We develop an algorithm that integrates these strategies to further improve alignment accuracy by modifying the pair-HMM approach in ProbCons to incorporate profiles of intermediate sequences from database search and utilize secondary structure predictions as in SPEM. We test our algorithm on a few sets of benchmark multiple alignments, including BAliBASE, HOMSTRAD, PREFAB and SABmark, and show that it significantly outperforms MAFFT and ProbCons, which are among the best multiple alignment algorithms that do not utilize additional information, and SPEM, which is among the best multiple alignment algorithms that utilize additional hits from database search. The improvement in accuracy over SPEM can be as much as 5 to 10% when aligning divergent sequences. A software program that implements this approach (ISPAlign) is at http://faculty.cs.tamu.edu/shsze/ispalign.

1 Introduction

Although many algorithms have been proposed for multiple sequence alignment (Thompson et al. 1994; Morgenstern et al. 1996; Stoye 1998; Notredame et al. 2000; Lee et al. 2002; Edgar 2004; Van Walle et al. 2004; Do et al. 2005; Katoh et al. 2005; Lassmann and Sonnhammer 2005; Pei and Grishin 2006; Roshan and Livesay 2006; Yamada et al. 2006), it remains difficult to obtain accurate alignments. Common techniques to improve alignment accuracy include performing iterative refinements after the initial alignment is constructed (Gotoh 1996; Edgar 2004; Do et al. 2005; Roshan and Livesay 2006; Yamada et al. 2006), using consistency-based pairwise alignments in progressive approaches (Notredame et al. 2000; Do et al. 2005; Pei and Grishin 2006; Roshan and Livesay 2006), and incorporating structural alignments (O'Sullivan et al. 2004; Van Walle et al. 2004). A few other strategies combine alignments from existing algorithms to obtain an improved alignment (Bucka-Lassen et al. 1999; Wallace et al. 2006).

T. Speed and H. Huang (Eds.): RECOMB 2007, LNBI 4453, pp. 283–295, 2007.

With the rapidly increasing number of sequences in biological databases, it has been observed that the use of additional sequences from database search can significantly improve alignment accuracy. Among the most successful approaches that use this strategy are profile alignment algorithms that use database search to find related sequences for each input sequence, construct a profile from the hits, and then align the profiles instead of the sequences, including algorithms that start from two sequences (Marti-Renom et al. 2004), and algorithms that start from multiple sequences (Simossis et al. 2005; Zhou and Zhou 2005). Alternatively, Heger et al. (2004) identified clusters of residues to form columns of a multiple alignment by linking distant homologs through the hits.

We observe that instead of constructing a profile for each input sequence from the hits, which only compares each hit to the input sequence that generates it, it may be more accurate to perform a more extensive multiple alignment of the hits together with the input sequences, which allows comparisons among all the sequences involved. The usefulness of such a strategy has been demonstrated during the construction of the PREFAB database (Edgar 2004), in which the incorporation of additional hits from database search into the input sequences significantly improves accuracy as opposed to aligning the input sequences alone. One drawback of this approach is that the inclusion of hits that are not intermediate between the input sequences can introduce noise, since these hits do not contribute to defining a better alignment between them. We will show that a careful definition of intermediate sequences from database search in addition to the computation of profiles for these sequences will significantly improve alignment accuracy.

By defining an intermediate sequence as a common hit from database search that links two input sequences, an intermediate sequence search technique has been used successfully to establish distant homologs (Park et al. 1997; Gerstein 1998). The strategy was later generalized to multiple intermediate sequence search (Salamov et al. 1999; Li et al. 2000), in which chains of intermediate sequences found through iterative database search are used to link very distant homologs. Bolten et al. (2001) used such transitive homologies to cluster protein sequences for structure predictions. Heger et al. (2004) used a graph-theoretic approach to link intermediate sequences through transitive homologies to detect short active site motifs, while Margelevičius and Venclovas (2005) used the intermediate sequence search strategy to distinguish between reliable and unreliable regions in alignments. Instead of defining intermediate sequences as common hits, we will develop a more relaxed definition to maximize the amount of information that can be extracted from the hits.

Since the number of hits that are also intermediate sequences can be very large, it is not practical to simply add them to the input sequences and perform a multiple alignment on the combined sequence set. Motivated by the fact that similar sequences are likely to contain redundant information, our algorithm uses a greedy strategy to choose a small subset of intermediate sequences that are far away from each other, which, together with the original sequences form a combined set of input sequences. Instead of aligning these sequences directly, we construct a profile for each sequence in the combined set by incorporating information from other intermediate sequences and aligning the profiles by modifying the pair-HMM approach (Durbin et al. 1998) in ProbCons (Do et al. 2005). This is in contrast with the strategy used in Simossis et al. (2005) and Zhou and Zhou (2005) which constructs a profile from the hits of an input sequence. We will show that our strategy of constructing profiles from intermediate sequences instead of from the hits helps to prevent the introduction of excessive noise when aligning closely related sequences. To further improve alignment accuracy, we obtain a secondary structure prediction for each sequence in the combined set and incorporate these predictions into the pair-HMM alignment. While this strategy of using secondary structure predictions is similar to the one employed in Zhou and Zhou (2005), it is different from the technique used in Pei and Grishin (2006) which employs secondary structure information during HMM training without explicitly using secondary structure predictions in alignments.

We compare the performance of our algorithm to MAFFT (Katoh et al. 2005) and ProbCons (Do et al. 2005), which are among the best multiple alignment algorithms that do not utilize additional information, and SPEM (Zhou and Zhou 2005), which is among the best multiple alignment algorithms that utilize additional hits from database search, on benchmark multiple alignments from BAliBASE (Thompson et al. 2005), HOMSTRAD (Mizuguchi et al. 1998), PREFAB (Edgar 2004), and SABmark (Van Walle et al. 2004). We will show that our algorithm outperforms MAFFT, ProbCons and SPEM in almost all situations, with very significant improvements when aligning divergent sequences. Before presenting the algorithm in detail, we first describe the general strategies employed in each stage in the next few sections.

2 Finding Intermediate Sequences

Although most intermediate sequence search strategies define an intermediate sequence either as a common hit from database search that links two input sequences (Park et al. 1997; Gerstein 1998), or as hits that form a chain linking two input sequences (Salamov et al. 1999; Li et al. 2000), such a requirement is very stringent since it may not be possible to link very divergent sequences together even if the database search is performed iteratively. We consider the following relaxed definition of an intermediate sequence which only requires that it is intermediate between the two input sequences.

Definition 1. Given two sequences s_1 and s_2, and a distance score $d(s_1, s_2)$ between them, a sequence r is intermediate between s_1 and s_2 if $d(r, s_1) < d(s_1, s_2)$ and $d(r, s_2) < d(s_1, s_2)$.

The problem of finding intermediate sequences between multiple input sequences is defined as follows.

Definition 2. Given n input sequences $s_1, \ldots, s_n$, and m hits $r_1, \ldots, r_m$ from database search of these sequences, find all hits r_k that are intermediate between some pair of input sequences s_i and s_j.

Similar to previous approaches, our goal is to find an appropriate subset of sequences that contain useful information between the input sequences $s_1, \dots, s_n$. We do not require that these intermediate sequences have a phylogenetic interpretation or have an appropriate evolutionary relationship to the input sequences. Also, since any hit that is intermediate between some pair of input sequences is potentially useful, it is included in the definition. Note that there is no need to compute pairwise distances between the potentially very large number of hits. The number of pairwise distance score computations that are needed to identify the intermediate sequences from among the hits is $O(mn+n^2)$, while the number of score comparisons is $O(mn^2)$.

3 Choosing Intermediate Sequences

The next problem of choosing a small subset of intermediate sequences to add to the input sequences is defined as follows. Our goal is to identify a combined set of sequences that are as divergent as possible.

Definition 3. Given n input sequences $s_1, \dots, s_n$, m intermediate sequences $r_1, \dots, r_m$, add k intermediate sequences from among $r_1, \dots, r_m$, denoted by $s_{n+1}, \dots, s_{n+k}$, so that the minimum distance between sequences in the combined set $s_1, \dots, s_{n+k}$ is the largest possible when distances between the input sequences $s_1, \dots, s_n$ are ignored.

Figure 1 shows a greedy algorithm that iteratively adds an intermediate sequence s_{n+j} that is farthest away from the current sequence set $s_1, \dots, s_{n+j-1}$, in which the minimum distance between s_{n+j} and $s_1, \dots, s_{n+j-1}$ is the largest possible. Although the greedy strategy does not guarantee optimum divergence of the sequences $s_1, \dots, s_{n+k}$, they should be reasonably far away from each other. The total number of pairwise distance score computations needed is $O(m(n+k))$, and there is no need to compute distances between all pairs of the potentially very large number of intermediate sequences.

```
Input: $n$ input sequences $s_1, \dots, s_n$, $m$ intermediate sequences $r_1, \dots, r_m$,
        distance score $d(r, s)$ between two sequences $r$ and $s$.
Output: $k$ intermediate sequences $s_{n+1}, \dots, s_{n+k}$ added to $s_1, \dots, s_n$.

$R \leftarrow \{r_1, \dots, r_m\}$;
for each $r_i$ in $R$ do { $d_i \leftarrow \min_{1 \le j \le n} d(r_i, s_j)$; }
for $j \leftarrow 1$ to $k$ do {
   $s_{n+j} \leftarrow r_i$ with the maximum $d_i$; remove $r_i$ from $R$;
   for each $r_i$ in $R$ do { $d_i \leftarrow \min(d_i, d(r_i, s_{n+j}))$; } }
```

Fig. 1. Greedy algorithm to choose a small subset of intermediate sequences to add to the input sequences

4 Constructing Sequence Profiles

Instead of aligning the sequences $s_1, \ldots, s_{n+k}$ directly, a profile is constructed for each of these sequences as follows: for each intermediate sequence r_i from among $r_1, \ldots, r_m$, assign it to the s_j from among $s_1, \ldots, s_{n+k}$ that is most similar to r_i. For each sequence s_j with assigned sequences $r_{i_1}, \ldots, r_{i_t}$, we combine all the pairwise alignments between s_j and each r_{i_p} into a star alignment with s_j as the center (Gusfield 1993). For each column in the star alignment that contains a residue of s_j, the relative frequency of each residue within the column is then used to construct a profile as a probability distribution of residues (gap characters are ignored). Here the choice of scoring functions for the profile is not very important since Edgar and Sjölander (2004) showed that most scoring functions do not have significant performance differences. One caution is that we need to make sure that the number of very closely related sequences assigned to each s_j is not excessively large to avoid over-contribution of these sequences to the profile. This can be achieved by removing sequences from the original set of intermediate sequences so that none of the remaining sequences are very similar to each other before choosing the subset of intermediate sequences. In difference from the approach in Simossis et al. (2005) and Zhou and Zhou (2005), hits that are not intermediate sequences are not used to avoid noise from these hits.

5 Alignment Via Modified Pair-HMM

We modify the pair-HMM approach in Durbin et al. (1998) to incorporate profiles and secondary structure predictions. The original model consists of three states: M emits an aligned pair of residues (x, y) with probability $e(x, y)$, X emits a residue x in the first sequence that is aligned to a gap with probability $e(x)$, while Y emits a residue y in the second sequence that is aligned to a gap with probability $e(y)$ (Fig. 2). In addition to the original residue, each position is now associated with a probability distribution of residues. Let $p_1(x, i)$ be the probability of finding the residue x at position i in the first sequence and let $p_2(y, j)$ be the probability of finding the residue y at position j in the second sequence. We modify the model to incorporate profiles as follows: define the emission probability of state M as $e'(i, j) = \sum_x \sum_y p_1(x, i) p_2(y, j) e(x, y)$ if the emission is at position i in the first sequence and at position j in the second sequence, the emission probability of state X as $e'(i) = \sum_x p_1(x, i) e(x)$ if the emission is at position i in the first sequence, and the emission probability of state Y as $e'(j) = \sum_y p_2(y, j) e(y)$ if the emission is at position j in the second sequence. These changes replace the original emission probabilities of the single residues by the average emission probabilities over a distribution of residues so that in the degenerate case when the profiles represent simple sequences, the effect is the same as before.

We incorporate secondary structure predictions into the pair-HMM model as follows: in state M, we introduce an additional parameter α and subdivide the emission probability $e'(i, j)$ into two cases to obtain a modified state $M(\alpha)$ with

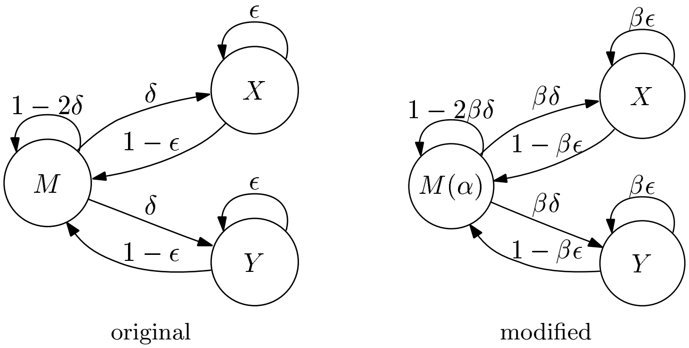

Fig. 2. The original and the modified pair-HMM models. In the original model, state M emits an aligned pair of residues, states X and Y emit a residue in the first and the second sequences respectively that is aligned to a gap, δ is the gap opening probability, and ϵ is the gap extension probability (Durbin et al. 1998). In the modified model, the state $M(\alpha)$ is obtained from M with emission probability $e(x, y)$ by defining the emission probability to be $\alpha e(x, y)$ if the paired residues (x, y) have the same secondary structure type and $(1 - \alpha)e(x, y)$ otherwise. The factor β is applied to δ and ϵ to compensate for the change. To incorporate profiles, the residue emission probabilities are replaced by the average emission probabilities over a distribution of residues.

emission probability $\alpha e'(i, j)$ if the original paired residues (x, y) at position i in the first sequence and at position j in the second sequence have the same secondary structure type, and with emission probability $(1 - \alpha)e'(i, j)$ otherwise. Since this decrease in emission probability will tend to allow more gaps than before in the ideal case in which every aligned residue pair has the same secondary structure type, we apply the factor β to the gap opening and extension probabilities to compensate for it while keeping the ratio between the two probabilities unchanged to preserve the affine gap model (Fig. 2). This modified pair-HMM can then be utilized within a progressive alignment strategy to obtain a multiple alignment (Do et al. 2005).

6 Detailed Algorithm

We now describe a procedure and the associated parameters that give very good results for our algorithm. Note that this is only among one of the many possible ways to implement the algorithm.

Following SPEM (Zhou and Zhou 2005), for each input sequence, we use PSI-BLAST (Altschul et al. 1997) to perform database search on a filtered version of the non-redundant protein database (NR) that excludes low complexity regions, transmembrane regions and likely coiled-coil regions (Jones 1999), and retain hits that have less than 98% identity to the input sequence and have e-value less than 0.001. One advantage of using PSI-BLAST is that it performs iterative database search automatically to look for distant homologs. Instead of keeping the entire sequence of a hit, only the regions within a PSI-BLAST local alignment are retained to avoid the introduction of noise from unrelated regions. Note that if there are more than one PSI-BLAST local alignment that satisfy the above condition within a hit, they are considered to be separate hits.

We then extract intermediate sequences from among these hits according to Definitions 1 and 2. To obtain an accurate distance score $d(s_1, s_2)$ between two sequence s_1 and s_2, we use SSEARCH (Smith and Waterman 1981) to obtain an optimal alignment between s_1 and s_2 and define $d(s_1, s_2)$ as the e-value of the alignment. Note that the use of e-values here does not pose any problems since no addition operations are performed.

To avoid over-contribution of very similar intermediate sequences in the later profile construction step, we use CD-HIT (Li et al. 2002) to remove some of the closely related sequences so that the identity between the remaining intermediate sequences is less than 85%. We then use Definition 3 and the algorithm in Fig. 1 with $k = 5$ to add at most five intermediate sequences to the input sequences to obtain $s_1, \dots, s_{n+k}$. We choose $k = 5$ so that the final multiple alignment step will not become much slower than simply aligning the original input sequences. The identity of a pairwise alignment from SSEARCH is used to obtain an accurate distance score $d(s_1, s_2)$ between two sequences s_1 and s_2 by defining $d(s_1, s_2)$ as $1 -$ identity (note that this distance is different from what we use above). Note that CD-HIT cannot be used for this purpose since it initially uses counts of short tuples to estimate pairwise similarity, which is inaccurate when the identity level between the sequences $s_1, \dots, s_{n+k}$ is low.

We then construct profiles according to the algorithm in Section 4 in which an intermediate sequence r_i is assigned to the sequence from among $s_1, \dots, s_{n+k}$ that has the best SSEARCH alignment to r_i. To obtain a secondary structure prediction for each of the sequences $s_1, \dots, s_{n+k}$, we follow SPEM (Zhou and Zhou 2005) and use PSIPRED (Jones 1999) to assign one of the three possible types (helix, strand or coil) to each residue.

With the profiles and secondary structure predictions, we modify ProbCons (Do et al. 2005) by changing its pair-HMM model according to Section 5. The parameters in Fig. 2 are as follows: the original residue emission probabilities and the transition probabilities δ and ϵ are from ProbCons. The parameter α that modifies the emission probabilities is 0.65, while the parameter β that modifies the transition probabilities is 0.75. These two parameters are determined by testing a few combinations and choosing one that gives satisfactory performance in PREFAB (Edgar 2004). We use the default setting in ProbCons that utilizes two sets of gap states with the same modifying parameter β for both sets. There is no change in the later progressive alignment or the iterative refinement steps and the alignment on the original input sequences is returned.

7 Performance on Benchmark Sets

We test our algorithm (ISPAlign) on benchmark multiple alignments from BAliBASE 3.0 (Thompson et al. 2005), HOMSTRAD (Mizuguchi et al. 1998), PREFAB 4.0 (Edgar 2004), and SABmark 1.65 (Van Walle et al. 2004). We

Table 1. Average SPS and CS scores (in %) on the full length sequence set in BAliBASE 3.0. Reference 1 is further subdivided into two subsets: 1V1 ($<$ 25% identity), and 1V2 (20–40% identity). The number in braces denotes the number of alignments in each subset. Within each subset, the best accuracy value is in bold. The values in parentheses denote the p-values, with — indicating insignificant differences. Since most of the subsets are very small, p-values are computed only for reference 1 and the entire set. Twenty-two cases are omitted due to unavailability of results from SPEM.

	SPS				CS			
	MAFFT	ProbCons	SPEM	ISPAlign	MAFFT	ProbCons	SPEM	ISPAlign
1V1 {38}	64.8	64.5	73.1	**76.0**	44.6	40.4	51.6	**56.9**
1V2 {42}	92.8	93.4	92.1	**93.5**	83.9	85.6	82.6	**85.8**
1 (V1–V2) {80}	79.5	79.7	83.1	**85.2**	65.2	64.2	67.9	**72.1**
(vs MAFFT)			(4e–5)	(5e–8)			(0.01)	(2e–7)
(vs ProbCons)			(7e–4)	(2e–6)			(0.01)	(2e–5)
(vs SPEM)				(0.002)				(9e–5)
2 {37}	91.8	89.7	88.0	**91.9**	46.0	40.8	47.1	**53.8**
3 {29}	81.4	78.8	82.8	**83.5**	56.8	54.3	51.4	**59.9**
4 {36}	89.2	86.8	87.5	**90.3**	**67.9**	60.9	55.4	63.3
5 {14}	88.2	87.5	87.0	**90.3**	57.6	59.4	55.9	**63.9**
All (1–5) {196}	84.5	83.3	85.0	**87.5**	60.3	57.3	58.3	**64.6**
(vs MAFFT)			(0.005)	(2e–11)			(—)	(2e–10)
(vs ProbCons)			(5e–4)	(2e–13)			(—)	(4e–10)
(vs SPEM)				(3e–7)				(5e–11)

compare our performance to MAFFT 5.8 (using the most accurate linsi strategy, Katoh et al. 2005), ProbCons 1.10 (Do et al. 2005) and SPEM (Zhou and Zhou 2005).

For BAliBASE, two score measures are used to perform accuracy assessment of each multiple alignment on the original input sequences: the sum-of-pairs score (SPS) evaluates the percentage of residue pairs that an algorithm can align correctly in the reference alignment, while the column score (CS) evaluates the percentage of entire columns that an algorithm can align correctly (Thompson et al. 1999). For PREFAB, evaluations are made on the original pairs of input sequences using the Q score defined in Edgar (2004), which has the same meaning as the SPS score. For BAliBASE and PREFAB, evaluations are made only on the core regions that are assigned to the reference alignments. While we test MAFFT and ProbCons both on the original pairs in PREFAB and on the full set of sequences that includes random hits from database search, we test SPEM and ISPAlign only on the original pairs since these algorithms utilize hits from database search automatically. For SABmark, reference sequences are specified in pairs and evaluations are based on the f_{D} and the f_{M} scores in Van Walle et al. (2004), in which f_{D} has the same meaning as SPS and f_{M} evaluates the percentage of correctly aligned residue pairs in the test alignment. We define the f_{D} score and the f_{M} score for each alignment as the average f_{D} score and the average f_{M} score respectively over all these pairs. For each test set, we use the Wilcoxon matched-pairs signed-ranks test (Wilcoxon 1947) over large enough subsets with 0.05 as the p-value cutoff for significance.

Table 1 shows performance comparisons on the full length sequence set in BAliBASE 3.0. For both reference 1 and the entire set, ISPAlign improved over MAFFT, ProbCons and SPEM very significantly, with the biggest improvements

Table 2. Average SPS and CS scores (in %) on HOMSTRAD. Each subset includes all alignments with average pairwise identity within the specified range, with * indicating worse performance in *p*-value. Since ProbCons consistently performs better than MAFFT, comparisons are made only between ProbCons, SPEM and ISPAlign. Only the *p*-values for the CS scores are shown.

	SPS			CS			SPEM	ISPAlign	ISPAlign
	ProbCons	SPEM	ISPAlign	ProbCons	SPEM	ISPAlign	(vs ProbCons)	(vs ProbCons)	(vs SPEM)
0–20% {156}	49.7	67.2	**68.5**	43.1	61.0	**62.7**	(4e–23)	(5e–24)	(4e–5)
20–40% {459}	80.5	85.6	**86.8**	74.7	80.4	**81.9**	(2e–29)	(2e–53)	(7e–7)
40–70% {348}	94.8	94.9	**95.5**	92.2	92.3	**93.2**	(0.03)	(2e–9)	(0.003)
70–100% {69}	**99.1**	98.5	99.0	**99.1**	98.4	98.9	(0.007*)	(—)	(—)
All {1032}	81.9	86.8	**87.8**	77.4	82.7	**84.0**	(2e–46)	(8e–87)	(1e–12)

Table 3. Average Q scores (in %) on PREFAB 4.0. Each subset includes all structure pairs with identity within the specified range, with * indicating worse performance in *p*-value. Comparisons are made between MAFFT and ProbCons using two sequences (MAFFT^2, ProbCons^2) and using all (at most 50) sequences (MAFFT^{50}, ProbCons^{50}), SP^2 (which is a specialized version of SPEM for two sequences), and ISPAlign^2 (ISPAlign starting from two sequences). Since MAFFT^{50} has the best accuracy among MAFFT and ProbCons, *p*-value comparisons are made only against MAFFT^{50}.

							SP^2	ISPAlign^2	ISPAlign^2
	MAFFT^2	ProbCons^2	MAFFT^{50}	ProbCons^{50}	SP^2	ISPAlign^2	(vs MAFFT^{50})	(vs MAFFT^{50})	(vs SP^2)
0–20% {887}	36.2	38.9	56.7	55.6	64.6	**64.8**	(3e–36)	(5e–46)	(0.03)
20–40% {588}	81.0	82.8	87.1	87.2	89.7	**90.1**	(2e–16)	(6e–28)	(0.01)
40–70% {112}	96.2	96.4	96.0	95.4	95.3	**97.6**	(0.02*)	(—)	(—)
70–100% {95}	97.9	97.8	**98.0**	97.3	97.2	**98.0**	(6e–4*)	(—)	(0.005)
All {1682}	59.4	61.4	72.3	71.7	77.3	**77.7**	(1e–46)	(7e–69)	(2e–4)

in the 1V1 subset when identity is very low (improvement in the CS score was over 5%). SPEM improved over MAFFT and ProbCons very significantly for the SPS score. For the CS score, SPEM significantly improved over MAFFT and ProbCons for reference 1, but the overall improvement was not significant for the entire set.

Table 2 shows performance comparisons on HOMSTRAD. Except for 70 to 100% identity, all the *p*-values of ISPAlign over SPEM, ISPAlign over ProbCons, and SPEM over ProbCons were highly significant. For 70 to 100% identity, SPEM performed significantly worse than ProbCons, while the differences between ISPAlign and ProbCons or SPEM were not significant. In general, as identity increases, less improvements were observed for both SPEM and ISPAlign.

Table 3 shows performance comparisons on PREFAB 4.0 using two versions of MAFFT and ProbCons: MAFFT^2 and ProbCons^2 use the original input pair, while MAFFT^{50} and ProbCons^{50} use the full sequence set that includes random hits from database search and has at most 50 sequences. For 0 to 20% identity and 20 to 40% identity, the improvements of SPEM or ISPAlign over MAFFT^{50} were highly significant, while the improvements of ISPAlign over SPEM were significant but not as much. For 40 to 70% identity, SPEM performed significantly worse than MAFFT^{50}, while the differences between ISPAlign and MAFFT^{50}

Table 4. Average f_D and f_M scores (in %) on the Twilight and Superfamily subsets of SABmark 1.65. Four cases are omitted in the Twilight subset and three cases are omitted in the Superfamily subset since no reference alignments of sufficiently good quality are available. None of these subsets include false positive sequences. Since ProbCons consistently performs better than MAFFT, comparisons are made only between ProbCons, SPEM and ISPAlign.

	f_D			f_M		
	ProbCons	SPEM	ISPAlign	ProbCons	SPEM	ISPAlign
Twilight {205}	29.3	44.2	**46.1**	21.0	30.8	**32.0**
(vs ProbCons)		(2e–26)	(6e–29)		(1e–27)	(3e-29)
(vs SPEM)			(0.01)			(0.005)
Superfamily {422}	57.1	68.3	**69.0**	43.6	50.9	**51.6**
(vs ProbCons)		(4e–49)	(1e–51)		(1e–48)	(1e–51)
(vs SPEM)			(0.02)			(7e–4)

Table 5. Average CS scores (in %) on HOMSTRAD and average Q scores (in %) on PREFAB 4.0 using a few methods that are of increasing levels of complexity. Method 1 constructs a profile from the hits of each input sequence and performs profile alignment using the modified HMM model that incorporates profiles but not secondary structure predictions. Method 2 removes the hits that are not intermediate sequences before performing profile alignment. Method 3 adds intermediate sequences to the input sequences, constructs profiles based on the intermediate sequences and performs profile alignment on the combined sequence set. Method 4 is the full ISPAlign algorithm that also utilizes secondary structure predictions. For PREFAB, ProbCons uses the original input pair while all the methods start from this input pair. The p-value comparisons are made against the previous method to the left, with * indicating worse performance.

	HOMSTRAD CS					PREFAB Q				
	ProbCons	Method1	Method2	Method3	Method4	ProbCons	Method1	Method2	Method3	Method4
0–20%	43.1	59.1	59.2	59.4	**62.7**	38.9	58.2	58.6	61.3	**64.8**
(vs previous)		(3e–22)	(—)	(0.04)	(6e–8)		(2e–103)	(—)	(6e–12)	(7e–29)
20–40%	74.7	79.1	79.6	81.4	**81.9**	82.8	88.7	89.0	89.7	**90.1**
(vs previous)		(2e–24)	(0.003)	(7e–14)	(0.005)		(9e–45)	(—)	(2e–4)	(0.004)
40–70%	92.2	92.1	92.5	93.1	**93.2**	96.4	94.4	96.6	**97.8**	97.6
(vs previous)		(—)	(8e–4)	(0.001)	(—)		(—)	(0.002)	(—)	(0.008*)
70–100%	99.1	98.2	99.1	**99.2**	98.9	97.8	97.0	96.9	**98.1**	98.0
(vs previous)		(6e–4*)	(1e–4)	(—)	(0.003*)		(0.04*)	(0.02)	(—)	(—)
All	77.4	81.7	82.2	83.2	**84.0**	61.4	73.5	73.9	75.7	**77.7**
(vs previous)		(5e–38)	(1e–6)	(1e–14)	(1e–6)		(7e–146)	(—)	(2e–15)	(4e–28)

or SPEM were not significant. For 70 to 100% identity, ISPAlign performed significantly better than SPEM but did not improve over MAFFT^{50}, while SPEM performed significantly worse than MAFFT^{50}. For the entire set, all the p-values of ISPAlign over MAFFT^{50}, ISPAlign over MAFFT^{50} and SPEM over MAFFT^{50} were highly significant.

Table 4 shows performance comparisons on the Twilight and Superfamily subsets of SABmark 1.65. While the improvements of SPEM or ISPAlign over ProbCons for both subsets were highly significant, the improvements of ISPAlign over SPEM were significant but not as much.

In all the subsets that we have assessed, ISPAlign always performs at least as well as ProbCons and SPEM and is much better in many cases, especially when the input sequences are divergent in which the improvements are always significant and in many cases highly significant. Also, the improvements in the CS scores are sometimes more significant than the improvements in the SPS scores. In general, the contribution from utilizing additional sequences from database search decreases as the input sequences become more closely related. When the input sequences become very similar, while SPEM has significant accuracy decreases in many cases, ISPAlign still always performs at least as well. Since not many intermediate sequences are added to the input sequences before performing the profile alignment step, ISPAlign is efficient enough to perform an individual multiple alignment of moderate size in a reasonable time. In most cases, ISPAlign is only slightly slower than SPEM, with at most about a two times slowdown in some cases.

To evaluate contributions from various components of the algorithm to the alignment accuracy under different identity levels, we compare the performance of a few methods that are of increasing levels of complexity on HOMSTRAD and PREFAB 4.0 (Table 5). When the identity is low, the biggest improvements were from the use of profiles, while significant improvements were obtained from the addition of intermediate sequences to the input sequences and from the use of secondary structure predictions. When the identity is high, improvements were mainly from the removal of hits that are not intermediate sequences.

8 Discussion

While we have described a procedure for ISPAlign that gives very good performance, there are still many opportunities to further improve its accuracy. Instead of adding a fixed number of intermediate sequences to the input sequences, it may be better to add more sequences as the number of input sequences increases. Alternatively, intermediate sequences can be added until all the minimum distances between each of the remaining intermediate sequences and the current set of sequences fall below a threshold. Also, instead of modifying the parameters used by ProbCons by applying the factors α and β, it may be better to re-train the pair-HMM using a set of confirmed secondary structures. This can be done in a framework suggested by Do et al. (2006). It is also possible to use other multiple alignment algorithms to perform the profile alignment step as long as profiles and secondary structure predictions can be incorporated, which can lead to further improvements as better multiple alignment algorithms become available. It may also be beneficial to utilize three-dimensional structures when they are available.

Acknowledgments. This work was supported by NSF grants CCR-0311590 and DBI-0421815.

References

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., Lipman, D.J.: Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. **25** (1997) 3389–3402

Bolten, E., Schliep, A., Schneckener, S., Schomburg, D., Schrader, R.: Clustering protein sequences — structure prediction by transitive homology. Bioinformatics **17** (2001) 935–941

Bucka-Lassen, K., Caprani, O., Hein, J.: Combining many multiple alignments in one improved alignment. Bioinformatics **15** (1999) 122–130

Do, C.B., Gross, S.S., Batzoglou, S.: CONTRAlign: discriminative training for protein sequence alignment. Lect. Notes Bioinformatics **3909** (2006) 160–174

Do, C.B., Mahabhashyam, M.S., Brudno, M., Batzoglou, S.: ProbCons: probabilistic consistency-based multiple sequence alignment. Genome Res. **15** (2005) 330–340

Durbin, R., Eddy, S., Krogh, A., Mitchison, G.: Biological sequence analysis. Cambridge University Press (1998)

Edgar, R.C.: MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res. **32** (2004) 1792–1797

Edgar, R.C., Sjölander, K.: A comparison of scoring functions for protein sequence profile alignment. Bioinformatics **20** (2004) 1301–1308

Gerstein, M.: Measurement of the effectiveness of transitive sequence comparison, through a third 'intermediate' sequence. Bioinformatics **14** (1998) 707–714

Gotoh, O.: Significant improvement in accuracy of multiple protein sequence alignments by iterative refinement as assessed by reference to structural alignments. J. Mol. Biol. **264** (1996) 823–838

Gusfield, D.: Efficient methods for multiple sequence alignment with guaranteed error bounds. Bull. Math. Biol. **55** (1993) 141–154

Heger, A., Lappe, M., Holm, L.: Accurate detection of very sparse sequence motifs. J. Comp. Biol. **11** (2004) 843–857

Jones, D.T.: Protein secondary structure prediction based on position-specific scoring matrices. J. Mol. Biol. **292** (1999) 195–202

Katoh, K., Kuma, K., Toh, H., Miyata, T.: MAFFT version 5: improvement in accuracy of multiple sequence alignment. Nucleic Acids Res. **33** (2005) 511–518

Lassmann, T., Sonnhammer, E.L.L.: Kalign — an accurate and fast multiple sequence alignment algorithm. BMC Bioinformatics **6** (2005) 298

Lee, C., Grasso, C., Sharlow, M.F.: Multiple sequence alignment using partial order graphs. Bioinformatics **18** (2002) 452–464

Li, W., Jaroszewski, L., Godzik, A.: Tolerating some redundancy significantly speeds up clustering of large protein databases. Bioinformatics **18** (2002) 77–82

Li, W., Pio, F., Pawlowski, K., Godzik, A.: Saturated BLAST: an automated multiple intermediate sequence search used to detect distant homology. Bioinformatics **16** (2000) 1105–1110

Margelevičius, M., Venclovas, Č.: PSI-BLAST-ISS: an intermediate sequence search tool for estimation of the position-specific alignment reliability. BMC Bioinformatics **6** (2005) 185

Marti-Renom, M.A., Madhusudhan, M.S., Sali, A.: Alignment of protein sequences by their profiles. Protein Sci. **13** (2004) 1071–1087

Mizuguchi, K., Deane, C.M., Blundell, T.L., Overington, J.P.: HOMSTRAD: a database of protein structure alignments for homologous families. Protein Sci. **7** (1998) 2469–2471

Morgenstern, B., Dress, A., Werner, T.: Multiple DNA and protein sequence alignment based on segment-to-segment comparison. Proc. Natl. Acad. Sci. USA **93** (1996) 12098–12103

Notredame, C., Higgins, D.G., Heringa, J.: T-Coffee: a novel method for fast and accurate multiple sequence alignment. J. Mol. Biol. **302** (2000) 205–217

O'Sullivan, O., Suhre, K., Abergel, C., Higgins, D.G., Notredame, C.: 3DCoffee: combining protein sequences and structures within multiple sequence alignments. J. Mol. Biol. **340** (2004) 385–395

Park, J., Teichmann, S.A., Hubbard, T., Chothia, C.: Intermediate sequences increase the detection of homology between sequences. J. Mol. Biol. **273** (1997) 349–354

Pei, J., Grishin, N.V.: MUMMALS: multiple sequence alignment improved by using hidden Markov models with local structural information. Nucleic Acids Res. **34** (2006) 4364–4374

Roshan, U., Livesay, D.R.: Probalign: multiple sequence alignment using partition function posterior probabilities. Bioinformatics **22** (2006) 2715–2721

Salamov, A.A., Suwa, M., Orengo, C.A., Swindells, M.B.: Combining sensitive database searches with multiple intermediates to detect distant homologues. Protein Eng. **12** (1999) 95–100

Simossis, V.A., Kleinjung, J., Heringa, J.: Homology-extended sequence alignment. Nucleic Acids Res. **33** (2005) 816–824

Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. J. Mol. Biol. **147** (1981) 195–197

Stoye, J.: Multiple sequence alignment with the divide-and-conquer method. Gene **211** (1998) GC45–56

Thompson, J.D., Higgins, D.G., Gibson, T.J.: CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position specific gap penalties and weight matrix choice. Nucleic Acids Res. **22** (1994) 4673–4680

Thompson, J.D., Koehl, P., Ripp, R., Poch, O.: BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. Proteins **61** (2005) 127–136

Thompson, J.D., Plewniak, F., Poch, O.: A comprehensive comparison of multiple sequence alignment programs. Nucleic Acids Res. **27** (1999) 2682–2690

Van Walle, I., Lasters, I., Wyns, L.: Align-m — a new algorithm for multiple alignment of highly divergent sequences. Bioinformatics **20** (2004) 1428–1435

Wallace, I.M., O'Sullivan, O., Higgins, D.G., Notredame, C.: M-Coffee: combining multiple sequence alignment methods with T-Coffee. Nucleic Acids Res. **34** (2006) 1692–1699

Wilcoxon, F.: Probability tables for individual comparisons by ranking methods. Biometrics **3** (1947) 119–122

Yamada, S., Gotoh, O., Yamana, H.: Improvement in accuracy of multiple sequence alignment using novel group-to-group sequence alignment algorithm with piecewise linear gap cost. BMC Bioinformatics **7** (2006) 524

Zhou, H., Zhou, Y.: SPEM: improving multiple sequence alignment with sequence profiles and predicted secondary structures. Bioinformatics **21** (2005) 3615–3621