A segment alignment approach to protein comparison
Yuzhen Ye, Lukasz Jaroszewski, Weizhong Li and Adam Godzik∗
The Burnham Institute, La Jolla, CA 92037, USA

ABSTRACT

Motivation: Local structure segments (LSSs) are small structural units shared by unrelated proteins. They are extensively used in protein structure comparison, and predicted LSSs (PLSSs) are used very successfully in ab initio folding simulations. However, predicted or real LSSs are rarely exploited by protein sequence comparison programs that are based on position-by-position alignments.

Results: We developed a SEgment Alignment algorithm (SEA) to compare proteins described as a collection of predicted local structure segments (PLSSs), which is equivalent to an unweighted graph (network). Any specific structure, real or predicted corresponds to a specific path in this network. SEA then uses a network matching approach to find two most similar paths in networks representing two proteins. SEA explores the uncertainty and diversity of predicted local structure information to search for a globally optimal solution. It simultaneously solves two related problems: the alignment of two proteins and the local structure prediction for each of them. On a benchmark of protein pairs with low sequence similarity, we show that application of the SEA algorithm improves alignment quality as compared to FFAS profile-profile alignment, and in some cases SEA alignments can match the structural alignments, a feat previously impossible for any sequence based alignment methods.

Availability: SEA is freely available for academic users on a web server http://ffas.ljcrf.edu/sea.

Supplementary information: http://ffas.ljcrf.edu/sea

Contact: adam@burnham.org

INTRODUCTION

With increasing evolutionary distance the similarities between homologous proteins become less and less evident on the sequence level, until the only remaining relationship is a general fold similarity (Rost, 1997). Proteins with similar folds can be described as having a similar spatial arrangement of small structural units, most conspicuous being alpha helices and beta strands. Such units are often shared by proteins with different folds. We define local structure segments (LSSs) as maximal structural units that are shared by proteins with different folds. Such segments can be predicted by nearest-neighbor methods which typically produce a list of Predicted Local Structure Segments (PLSSs) for a given protein (Fig. 1, Rychlewski and Godzik, 1997; Yi and Lander, 1993; Bystroff and Baker, 1998). Generally, such predictions are ambiguous—multiple and often contradictory PLSSs are predicted along a sequence. This ambiguity can be viewed either as a result of prediction defects, or as a fundamental feature of local structure preferences. In the latter interpretation, PLSSs can be viewed as a list of potential local segments, some of which are later eliminated by other factors, such as non-local interactions in the final structure.

Although nearest-neighbor algorithms initially consider the ambiguity in local structure, most do not carry these ideas further. Instead, they use only single position secondary structures averaged over the segments (Rychlewski and Godzik, 1997; Yi and Lander, 1993). The notable exception is Baker and colleagues (Bystroff and Baker, 1998) who further combined the predicted segments for a compact tertiary structure in their de novo protein structure prediction program ROSETTA (Simons et al., 1999).

Meanwhile, most protein comparison methods are firmly based on the concept of residue-level alignments (Waterman, 1995), including programs that recognize distantly homologous or even non-homologous, but structurally similar proteins (Sternberg et al., 1999). One can ask whether the language of position-by-position alignments adequately describes distant evolutionary relationships, or whether this is an oversimplifying assumption that discards otherwise important information. Analysis of structural similarities between distantly related proteins suggest the latter, but so far, lack of
adequate algorithms has made this a purely rhetorical question.

Here we present a SEgment Alignment (SEA) algorithm, a segment-based protein comparison algorithm that operates in the entire space of predicted local structure segments. SEA finds the best match between all possible paths through PLSS networks representing two proteins, and therefore can use segments different from the locally best to contribute to the globally optimal alignment.

SEGMENT ALIGNMENT (SEA) FORMULATION

Given two protein sequences $S_1[1 \ldots M]$ and $S_2[1 \ldots N]$ whose local structure segments (LSSs) are predicted, SEA’s goal is to find an optimal alignment between these two proteins by comparing all possible combination of segments. Even if some combinations can be excluded (see Fig. 1 for explanation), it would be prohibitively expensive to enumerate all of them. By representing the set of PLSSs for each protein as a network (unweighted graph), protein alignment is transformed into the problem of getting a path from source to sink vertex in each network with the optimal similarity score to a path in another network. This is a well-known network matching problem that can be solved by dynamic programming in polynomial time. A similar problem, called the spliced sequence alignment, has been proposed and solved for assembling genes from alternative exons (Gelfand et al., 1996; Novichkov et al., 2001).

Figure 1 shows how to construct a network of PLSSs for a given protein. First, each residue is described as a vertex in the graph, and two artificial vertices are added to the very beginning (source vertex) and the end (sink vertex) of this protein. Then, for each PLSS $\alpha$, we add an edge between the vertices of its first (denoted as first($\alpha$)) and last (denoted as last($\alpha$)) positions. We say that the segment $\alpha$ covers position $i$ if first($\alpha$) $\leq$ $i$ and last($\alpha$) $\geq$ $i$, and specify $i$ position as $i_a$. The set of PLSSs covering position $i$ is denoted as $E(i)$. In practice, some parts of the protein may not be covered by any segments due to poor predictions. For the sake of continuity in any potential path, virtual edges (i.e. edges that do not correspond to a predicted segment) are added to all residues to form a complete network. An assembly of connected PLSSs corresponds to a path in this network.

The task of SEA is then to compare two networks of PLSSs by dynamic programming (Fig. 2). For any pair of positions, $i$ and $j$, their covering segments are considered in a combinatorial way (total $|E(i)||E(j)|$ combinations) and are compared to get the optimal similarity score; it makes SEA different from sequence pair-wise alignments where only residues at positions $i$ and $j$ are compared. We define $V(i, j)$ as the maximum similarity score for transforming $S_1[1 \ldots i]$ to $S_2[1 \ldots j]$, calculated by

\[
V(i, j) = \max_{\text{all}(\alpha, \beta) \text{combinations}, \alpha \in E(i), \beta \in E(j)} V(i_{\alpha}, j_{\beta})
\]

(1)

where $V(i_{\alpha}, j_{\beta})$ is the maximum similarity score for transforming $S_1[1 \ldots i_{\alpha}]$ to $S_2[1 \ldots j_{\beta}]$. It is computed recursively by the following formula, where $S(i_{\alpha}, j_{\beta})$, $D(i_{\alpha}, j_{\beta})$ and $I(i_{\alpha}, j_{\beta})$ are the maximum similarity scores for transforming $S_1[1 \ldots i_{\alpha}]$ to $S_2[1 \ldots j_{\beta}]$ ending with
substitution, deletion and insertion at \((i_a, j_p)\), respectively.

\[
\begin{align*}
V(i_a, j_p) &= \max \left\{ S(i_a, j_p), D(i_a, j_p), I(i_a, j_p), 0 \right\} \\
S(i_a, j_p) &= \max \gamma, \delta \left[ V \left( (i-1)_\gamma, (j-1)_\delta \right) + \Delta(i_a, j_p) \right] \\
D(i_a, j_p) &= \max \gamma [ \max [V \left( (i-1)_\gamma, j_p \right) - g], \left[ D \left( (i-1)_\gamma, j_p \right) - h \right] ] \\
I(i_a, j_p) &= \max \delta [ \max [V \left( i_a, (j-1)_\delta \right) - g], \left[ I \left( i_a, (j-1)_\delta \right) - h \right] ]
\end{align*}
\]

where

\[
\begin{align*}
\gamma &= \alpha & \text{if } (\text{first}(\alpha) \neq i) \\
\gamma &\in E(i-1) \text{ & } \text{last}(\gamma) = i-1 & \text{else} \\
\delta &= \beta & \text{if } (\text{first}(\beta) \neq j) \\
\delta &\in E(j-1) \text{ & } \text{last}(\beta) = j-1 & \text{else}
\end{align*}
\]

Equations (3) and (4) define the important compatibility requirement for the continuation of segments. The affine gap function is applied, with \(g\) and \(h\) standing for the gap initiating penalty and gap extension penalty, respectively (Gusfield, 1999). The similarity score of aligned positions \(i\) and \(j\) is \(\Delta(i_a, j_p)\) (see Implementation), and in principle could be any measure of similarity between segments.

As SEA considers all segment combinations at each pair of positions, its computational complexity is about \(O(\text{NMC}_1 \text{C}_2)\), where \(\text{C}_1\) and \(\text{C}_2\) are the average numbers of segments that cover a position in each protein (the segment coverage).

**Implementation**

We implemented the SEA algorithm in C++ on a Linux platform. The running time of SEA on a 1 GHz PII with 1 Gb of RAM varies from several seconds to minutes, depending on the length of the query proteins and their PLSSs distributions. In this section, we address some of the practical issues in implementing the SEA approach. It is important to note that all the solutions discussed below are specific for this particular implementation of the SEA algorithm. In particular, non-local segment similarity measures can be used instead of a formula from Equation (5).

**The prediction and representation of local structures**

We used the HMMSTR/Rosetta server (http://honduras.bio.rpi.edu/~sites/hmmstr/server.html) with its default parameters to predict the local structure segments (PLSSs) and the one-dimensional local structures (1D). Correspondingly, we adopted its 11 symbols for local structures [HGEeBbDLlxc], each having different backbone dihedral \((\phi\) and \(\psi\) regions (Bystroff and Baker, 1998), and simply described each PLSS as a short string of local-structure symbols. Several variants of the SEA algorithm were introduced, including SEA_all (using the entire set of PLSSs), SEA_cn (n is the maximum segment coverage, e.g. SEA_c30 and SEA_c5), and two special cases in which the network representing a protein is simplified to a single path: SEA_1D using the 1D prediction and SEA_true using segments derived from the actual 3D structure.

**Scoring scheme**

The similarity score between two aligned positions in the current implementation of SEA is formulated as,

\[
\Delta(i_a, j_p) = W_a \times \Delta(\text{Aa}_i, \text{Aa}_j) + W_s \times \Delta(\alpha, \beta) \tag{5}
\]

where \(W_a\) and \(W_s\) are the relative weights of sequence similarity and local structure similarity, satisfying \(W_a + W_s = 1\). \(\Delta(\text{Aa}_i, \text{Aa}_j)\) is the sequence similarity defined by Blosum62 similarity matrix (Henikoff and Henikoff, 1992), and \(\Delta(\alpha, \beta)\) is the similarity between local structures defined in the next paragraph. \(W_s\) is set to 0.5, and the gap opening and elongation parameters are set to \(-5\) and \(-1\). This set of parameters was partly optimized on a small set of SEA_true alignments.

We extracted a local structure similarity matrix from the subset of the HOMSTRAD database (SUB177) containing 706 proteins in 177 structural families (Shi et al., 2001). The proteins’ local structures were calculated from their 3D structures. The local structure similarities were then calculated from the log-odds of the probability of matching a pair of local structures in this database relative to a random one (Henikoff and Henikoff, 1992).
A segment alignment approach to protein comparison

Fig. 3. Comparison of the alignments between λ-repressor from *E.coli* (1lliA) and 434 repressor (1r69) by CE (top) and SEA (bottom). See text for details.

The measures of alignment accuracy

We evaluated different alignment algorithms by the quality of the resulting alignments, measured by the root mean square deviation (RMSD) of Cα positions after optimal superposition and the shift score (Cline et al., 2002). The CE (Shindyalov and Bourne, 1998) structural alignments were used as reference alignments.

The benchmark for SEA validation

We tested SEA’s performance with a database of 409 family-level and 225 superfamily-level pairs (Jaroszewski et al., 2001), which was selected from the SCOP database clustered at the 40% sequence identity threshold (Murzin et al., 1995).

RESULTS AND DISCUSSION

The general performance of SEA on the benchmark

SEA’s alignment results for all 409 family-level similar pairs at different inputs are listed in Table 1. For comparison, we also listed the alignment results of these proteins by BLAST (Altschul et al., 1990), ALIGN (Myers and Miller, 1988), FFAS (Rychlewski et al., 2000) and CE. Briefly, BLAST produced the worst alignments (shift average 0.44), whereas SEA_c30 and SEA_c10 produced most accurate alignments. All SEA variants performed better than other programs tested here, except that SEA_1D performed on the level similar to that of FFAS.

For pairs in the same superfamily, the alignment quality was generally much worse, reflecting the fact that similarity between these pairs on both structure and sequence levels is generally lower than for family-level pairs. Nevertheless, the general trends were very similar.

The SEA_true results are the upper quality limit for the SEA approach, since it assumes that the segment prediction for the protein of unknown structure is perfect. The remaining large gap between this computation and the structural comparison (CE) is due to the lack of the packing information for the local structure elements; this may be alleviated when a better segment similarity measurement is introduced.

As RMSD is dependent on alignment length and the shift score is dependent on the reference alignment, both measures are less than perfect in comparing alignments. To better understand the SEA performance, we list detailed descriptions for some alignment pairs in Table 2, some of which we discuss further below to show the advantages of SEA. For these proteins, the alignments by SEA using maximally diverse local segments are all closer to 3D-structural comparison results than all other alignments, including the profile–profile alignment algorithm (FFAS), which was shown to produce better alignments than PSI-BLAST and other popular alignment programs (Rychlewski et al., 2000).

Prediction ambiguity improves alignment quality

Incorporating ambiguity into SEA compensates for poor local structure predictions and on average improves alignment quality. This could be illustrated by the alignment between λ-repressor from *E.coli* (PDB code 1lli, chain A) and 434 repressor from phage 434 (PDB code 1r69). The structural alignment of these two proteins is in the top of Figure 3 along with the true local structures (next to the alignment) and the predicted local structures. In each protein there is one major prediction mistake (marked by dark lines in Figure 3). As a result, SEA_1D matches these incorrectly predicted segments and the resulting alignment is incorrect (RMSD is 9.23). In contrast, SEA using a diverse set of local structures (SEA_c30) achieved better alignment.
(bottom of Figure 3, the local structures confirmed by SEA from a set of possible PLSSs are shown beside the bottom alignment) than the SEA_1D and pure sequence-based alignments (Table 2). More importantly, the segments contributing to the final alignment are not locally optimal, but are closer to the true structures.

Another example is the comparison of *E. coli* bacterioferritin (PDB code 1bci, chain A) and amphibian red-cell L ferritin (PDB code 1rcd). These two proteins have very low sequence identity but high structural similarity (RMSD is 1.75). BLAST was not able to align these two proteins whereas the SEA_c30 alignment (RMSD is 2.77) is similar to the CE based 3D comparison with only minor differences in the gap regions. This is a big success compared to the results of SEA_c50, SEA_c30 and SEA_c10 achieved better alignments in the unstable regions and thus produced better complete evolutionary relationship between the two repeated domains in Myb and mating-type protein-2: it shows that the evolutionary relationship between mating-type protein-2 with the R2 domain is closer than with the R3 domain.

### Alignment quality versus local structure prediction ambiguity

Accounting for the ambiguity of the local structure prediction is necessary for a good comparison; however, using too diverse PLSS set will increase the probability of a random match and slow down the calculation. It is important to keep proper prediction diversity in SEA calculation.

Table 2 lists the alignment results for different segment coverage (all, 50, 30, 10 and 5). Although there is no simple relationship between segment coverage and the alignment quality, the results show that proper diversities (e.g. c30) are superior to low diversities (e.g. c5) and to single predicted local structure (1D). Interestingly, very high diversities do always produce optimal results, as in the alignments between staphylococcal enterotoxin A (PDB code 1esf, chain A) and toxic shock syndrome toxin-1 from *Staphylococcus aureus* (PDB code 2tss, chain A).

Y. Ye et al.

Table 1. General performance of SEA incorporating different local structure diversities

<table>
<thead>
<tr>
<th>Subset</th>
<th>Measures</th>
<th>CE</th>
<th>SEA_true</th>
<th>SEA_c30</th>
<th>SEA_c10</th>
<th>SEA_c5</th>
<th>SEA_1D</th>
<th>BLAST</th>
<th>ALIGN</th>
<th>FFAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Family (409 pairs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average-shift</td>
<td>0.61</td>
<td>0.56</td>
<td>0.56</td>
<td>0.54</td>
<td>0.49</td>
<td>0.44</td>
<td>0.48</td>
<td>0.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift &gt; 0.9</td>
<td>73</td>
<td>69</td>
<td>63</td>
<td>56</td>
<td>47</td>
<td>51</td>
<td>60</td>
<td>43</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift &gt; 0.7</td>
<td>207</td>
<td>199</td>
<td>192</td>
<td>183</td>
<td>152</td>
<td>146</td>
<td>165</td>
<td>161</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift &gt; 0.5</td>
<td>282</td>
<td>260</td>
<td>259</td>
<td>251</td>
<td>215</td>
<td>197</td>
<td>228</td>
<td>227</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMSD ≤ 3.0</td>
<td>257</td>
<td>95</td>
<td>82</td>
<td>76</td>
<td>63</td>
<td>77</td>
<td>54</td>
<td>40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMSD ≤ 5.0</td>
<td>397</td>
<td>237</td>
<td>184</td>
<td>171</td>
<td>177</td>
<td>147</td>
<td>157</td>
<td>118</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMSD ≤ 8.0</td>
<td>408</td>
<td>294</td>
<td>248</td>
<td>249</td>
<td>249</td>
<td>231</td>
<td>196</td>
<td>206</td>
<td>194</td>
<td></td>
</tr>
<tr>
<td>All</td>
<td>409</td>
<td>345</td>
<td>404</td>
<td>398</td>
<td>368</td>
<td>366</td>
<td>364</td>
<td>372</td>
<td>409</td>
<td></td>
</tr>
<tr>
<td>Superfamily (225 pairs)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average-shift</td>
<td>0.27</td>
<td>0.12</td>
<td>0.12</td>
<td>0.12</td>
<td>0.08</td>
<td>0.09</td>
<td>0.06</td>
<td>0.07</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift &gt; 0.9</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift &gt; 0.7</td>
<td>17</td>
<td>8</td>
<td>9</td>
<td>7</td>
<td>4</td>
<td>10</td>
<td>9</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift &gt; 0.5</td>
<td>54</td>
<td>26</td>
<td>23</td>
<td>21</td>
<td>17</td>
<td>18</td>
<td>18</td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMSD ≤ 3.0</td>
<td>55</td>
<td>12</td>
<td>6</td>
<td>7</td>
<td>6</td>
<td>8</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMSD ≤ 5.0</td>
<td>160</td>
<td>44</td>
<td>16</td>
<td>18</td>
<td>18</td>
<td>11</td>
<td>18</td>
<td>11</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>RMSD ≤ 8.0</td>
<td>163</td>
<td>69</td>
<td>37</td>
<td>41</td>
<td>28</td>
<td>23</td>
<td>22</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>All</td>
<td>166</td>
<td>128</td>
<td>217</td>
<td>204</td>
<td>181</td>
<td>177</td>
<td>41</td>
<td>149</td>
<td>225</td>
<td></td>
</tr>
</tbody>
</table>

Two subsets of the benchmark (family-level subset and superfamily-level subset) are compared with SEA variants and other programs: CE, BLAST, ALIGN and FFAS (see text for the descriptions of these programs). The average-shift row lists the shift score averaged over all the alignments of each subset by different programs. The number of alignments with shift score >0.9, >0.7 and >0.5, RMSD ≤ 3.0, ≤5.0 and ≤8.0 in each subset are also listed in rows. For the alignment to be counted in RMSD evaluation, its length must be at least half of its corresponding structural alignment. As the CE structural alignments are used as reference alignments, its evaluation by shift score is meaningless and the corresponding numbers are blanked out.
A segment alignment approach to protein comparison

Table 2. A more detailed comparison of the performances of SEA and other alignment programs

<table>
<thead>
<tr>
<th>Pro</th>
<th>Len</th>
<th>Ide</th>
<th>SEA_all</th>
<th>SEA_c50</th>
<th>SEA_c30</th>
<th>SEA_c10</th>
<th>SEA_c5</th>
<th>SEA_1D</th>
<th>SEA_true</th>
<th>FFAS</th>
<th>BLAST</th>
<th>CE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1lliA</td>
<td>89</td>
<td>33</td>
<td>2.65</td>
<td>2.02</td>
<td>2.03</td>
<td>2.03</td>
<td>2.20</td>
<td>9.23</td>
<td>2.03</td>
<td>4.99</td>
<td>1.74</td>
<td>1.95</td>
</tr>
<tr>
<td>1reiA</td>
<td>63</td>
<td></td>
<td>0.92</td>
<td>0.96</td>
<td>0.94</td>
<td>0.94</td>
<td>0.91</td>
<td>−0.10</td>
<td>0.94</td>
<td>0.78</td>
<td>0.68</td>
<td>1.00</td>
</tr>
<tr>
<td>1bcfA</td>
<td>158</td>
<td>18</td>
<td>3.98</td>
<td>2.99</td>
<td>2.77</td>
<td>2.50</td>
<td>12.14</td>
<td>7.88</td>
<td>2.35</td>
<td>13.61</td>
<td>−</td>
<td>1.75</td>
</tr>
<tr>
<td>1r69</td>
<td>63</td>
<td></td>
<td>0.67</td>
<td>0.82</td>
<td>0.83</td>
<td>0.84</td>
<td>0.15</td>
<td>0.30</td>
<td>0.90</td>
<td>0.29</td>
<td>−</td>
<td>1.00</td>
</tr>
<tr>
<td>1lshA</td>
<td>147</td>
<td>10</td>
<td>3.80</td>
<td>4.36</td>
<td>4.38</td>
<td>3.54</td>
<td>3.70</td>
<td>4.06</td>
<td>3.02</td>
<td>4.16</td>
<td>−</td>
<td>2.31</td>
</tr>
<tr>
<td>1ihA</td>
<td>141</td>
<td></td>
<td>0.75</td>
<td>0.62</td>
<td>0.63</td>
<td>0.76</td>
<td>0.76</td>
<td>0.37</td>
<td>0.81</td>
<td>0.70</td>
<td>−</td>
<td>1.00</td>
</tr>
<tr>
<td>1hbgA</td>
<td>147</td>
<td>19</td>
<td>2.97</td>
<td>4.71</td>
<td>2.82</td>
<td>5.74</td>
<td>3.11</td>
<td>6.22</td>
<td>3.08</td>
<td>5.53</td>
<td>−</td>
<td>2.11</td>
</tr>
<tr>
<td>1mbdA</td>
<td>153</td>
<td></td>
<td>0.89</td>
<td>0.76</td>
<td>0.91</td>
<td>0.70</td>
<td>0.82</td>
<td>0.49</td>
<td>0.80</td>
<td>0.70</td>
<td>−</td>
<td>1.00</td>
</tr>
<tr>
<td>1iyuA</td>
<td>79</td>
<td>17</td>
<td>2.85</td>
<td>2.85</td>
<td>2.85</td>
<td>2.84</td>
<td>2.85</td>
<td>7.55</td>
<td>2.24</td>
<td>12.04</td>
<td>4.63</td>
<td>2.57</td>
</tr>
<tr>
<td>1fycA</td>
<td>106</td>
<td></td>
<td>0.91</td>
<td>0.91</td>
<td>0.91</td>
<td>0.90</td>
<td>0.91</td>
<td>−0.09</td>
<td>−0.09</td>
<td>0.64</td>
<td>−0.09</td>
<td>1.00</td>
</tr>
<tr>
<td>1lsaA</td>
<td>103</td>
<td></td>
<td>0.69</td>
<td>0.69</td>
<td>0.69</td>
<td>0.35</td>
<td>−0.11</td>
<td>−0.07</td>
<td>0.69</td>
<td>0.58</td>
<td>−</td>
<td>1.00</td>
</tr>
<tr>
<td>1wiuA</td>
<td>93</td>
<td>10</td>
<td>3.41</td>
<td>3.41</td>
<td>3.41</td>
<td>3.45</td>
<td>3.45</td>
<td>15.44</td>
<td>3.54</td>
<td>4.08</td>
<td>12.24</td>
<td>2.15</td>
</tr>
<tr>
<td>1tuaA</td>
<td>89</td>
<td></td>
<td>0.82</td>
<td>0.82</td>
<td>0.82</td>
<td>0.80</td>
<td>0.80</td>
<td>0.31</td>
<td>0.65</td>
<td>0.68</td>
<td>0.35</td>
<td>1.00</td>
</tr>
<tr>
<td>1mjcA</td>
<td>69</td>
<td>19</td>
<td>3.78</td>
<td>6.36</td>
<td>3.14</td>
<td>6.90</td>
<td>6.74</td>
<td>3.46</td>
<td>−</td>
<td>7.00</td>
<td>4.59</td>
<td>2.44</td>
</tr>
<tr>
<td>1sroA</td>
<td>76</td>
<td></td>
<td>0.65</td>
<td>0.56</td>
<td>0.72</td>
<td>0.49</td>
<td>0.56</td>
<td>0.44</td>
<td>−</td>
<td>0.57</td>
<td>0.33</td>
<td>1.00</td>
</tr>
</tbody>
</table>

The Pro column lists the pdb code of each protein. The Len column lists the length of each protein. The Ide column lists the identity percentage of each protein pair calculated from the structural alignments by CE. The Ali row lists the alignment length of each pair. The low RMSD values caused by short alignments are shown in italic fonts.

alignments between these two proteins than SEA_c5, SEA_1D, FFAS and BLAST (see Table 2). This case shows that local structure information is crucial for improving alignments, especially in the less conserved regions.

CONCLUSION

In this paper, we introduced the SEGment Alignment (SEA) approach for comparing proteins described as collections of segments. SEA utilizes the full extent of predicted local structure information by exploring all possible PLSSs, some of which are correctly predicted while others are not. Such prediction uncertainty is unavoidable, due to limitations of individual prediction approaches but also due to the cooperative effect of the whole structure, as evident in frequent cases of identical sequence segments adopting different local structure in different proteins (Mezei, 1998). SEA solves the chicken and egg dilemma of how to accurately predict protein local (secondary) structure which requires accurate protein alignment, while accurate alignment requires accurate structure prediction, by doing both simultaneously.

The preliminary version of SEA performed very well on a limited set of test cases. It is worth noting that many of the choices made in this exploratory effort were not fully optimized. Systematic study of the effects of LSS prediction in SEA is necessary. It would also be interesting and vital to compare the LSS prediction by the I-site method; (Bystroff and Baker, 1998) and other approaches (e.g. nearest-neighbor method Yi and Lander, 1993). We expect better results as we improve
segment definitions, similarity measurements (especially those segment-based such as RMSD between segments), and significance evaluations. We are now exploring the local structure prediction using FFAS and testing the influence of length of segments on the alignment quality.

SEA has many potential applications, such as fold-recognition, distant homology detections and refining protein alignments for better structure modeling. As SEA also predicts the local structures, it can also be used for local structure predictions where many possibilities of local structures are given. We are currently developing and testing applications of SEA to protein modeling and fold recognition, which will be described in separate publications.

ACKNOWLEDGEMENTS

We appreciate valuable comments about the algorithm development from Dr Haixu Tang. We are indebted to Dr Chris Bystroff for his kind and immediate responses about the I-site related services and their prediction server. We thank Dr Melissa Cline for kindly providing us their shift score calculation program, Dr Ilya N. Shindyalov for kindly providing us his structural comparison program CE and Drs Jason Hoffman and Iddo Friedberg for help in editing. This research was supported by a grant NIH GM63208.

REFERENCES


A segment alignment approach to protein comparison


