Three aspects are usually considered when developing an MSA. The alignment of multiple sequences is essential for many tasks, such as integral transmembrane (TM) proteins, where residues are more conserved between major secondary structure elements than within. Also, gaps are much more frequent in disordered regions of proteins. For example, ESPSgp, calculated from predicted secondary structures makes steady improvement over Kalign2 when tested on Balibase 3.0 as well as on a dataset derived from Pfam-A seed alignments.

Availability and Implementation: KalignP is freely available at http://kalign.cbr.su.se. The source code of KalignP is available under the GNU General Public License, Version 2 or later from the same website.

Contact: arne@bioinfo.se

Supplementary Information: Supplementary data are available at Bioinformatics online.

Received on March 2, 2011; revised on March 25, 2011; accepted on April 8, 2011

1 INTRODUCTION

The alignment of multiple sequences is essential for many tasks in sequence analysis, e.g. protein structure and function prediction, similarity search and phylogenetic analysis (Pei, 2008). Due to the importance of multiple sequence alignment (MSA) in a variety of biological applications, a number of programs have been developed. Three aspects are usually considered when developing an MSA program, the alignment accuracy, the computational speed and the memory usage. Kalign2 (Lassmann et al., 2009) is one of the fastest MSA methods, and still the alignment accuracy is comparable to many of the slower methods. Moreover, it produces high quality alignments consistently when aligning large number of sequences, i.e. Kalign2 is well suited for MSA tasks in large-scale genome analysis.

However, in contrast to other methods such as ClustalW (Thompson et al., 1994), Kalign2 does not allow externally supplied position specific gap penalties (ESPSGP). Thompson et al. (1994) showed that gaps occur far more often between major secondary structure elements than within. Also, gaps are much more frequent in disordered regions of proteins. For integral transmembrane (TM) proteins, residues are more conserved at TM regions and few gaps are allowed within (Kauko et al., 2008).

Further, the inclusion of ESPSGP allows some inclusion of expert knowledge into the alignment procedure. Here, we introduce KalignP as a modified version of Kalign2 that accepts ESPSGP. Unlike ClustalW, where ESPSGP is only applicable to profile alignments, KalignP incorporates ESPSGP into sequences and forward them to the full progressive alignment.

2 METHODS

2.1 Implementation of KalignP

KalignP inherits the syntax of Kalign2 but extends its functionality for protein sequence alignment. The ESPSGP can be supplied in the Enhanced Fasta format (see Supplementary Material) so that gap penalties are set individually for each position. The default gap penalties will be replaced by ESPSGP when aligning two single sequences. Gap penalties of profiles (consensus of sequences) are calculated in the same way as Kalign2 but ESPSGP will be used instead of the constant gap penalty. KalignP also inherits one of the best merits of Kalign2: low memory consumption. Since KalignP uses the same progressive method as Kalign2, it consumes almost the same amount of memory as Kalign2. KalignP takes about 50% more CPU time than Kalign2 for the alignment itself. The KalignP package, including automatic secondary structure prediction and ESPSGP calculation from secondary structures predicted by PSIPRED, single (Jones, 1999) (version 3.2), is about 15 times slower than Kalign2, but this speed is still comparable to many other MSA programs such as ClustalW and Muscle (Edgar, 2004), see Table 1 and Supplementary Material.

2.2 Estimation of ESPSGP from predicted secondary structure

The principle to estimate ESPSGP from the predicted secondary structure is simple: increase the gap penalty at helices and sheets while lowering the gap penalty at coils. ESPSGP for position $i$ is set as

$$ESPSGP(i) = GP_{default} \times (1.0 + m(i))$$

where $m(i)$ is a variable to adjust the gap penalty at residue $i$ based on the predicted structural information, see Supplementary Material for details.

3 RESULTS

We benchmarked KalignP using gap penalties from predicted secondary structure with other methods on Balibase 3.0. KalignP outperformed Kalign2 on all five reference-sets (Table 1). The results were evaluated by SP (Sum-of-Pairs) score and TC (Total Columns) score (Thompson et al., 2005). SP score is the sum of scores of the projected pairwise alignments and TC score is the sum of scores of each column for a multiple alignment. The $P$-values of paired $t$-test of all 386 alignments in Balibase 3.0 are
Table 1. Performance of multiple sequence alignment methods on Balibase 3.0 (using core blocks) in SP score. All SP scores are multiplied with 100 for easy reading.

<table>
<thead>
<tr>
<th>Method</th>
<th>CPU^a (s)</th>
<th>RV13</th>
<th>RV12^b</th>
<th>RV20</th>
<th>RV30</th>
<th>RV40</th>
<th>RV50^a</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kalign2</td>
<td>57</td>
<td>64.4</td>
<td>91.3</td>
<td>91.9</td>
<td>82.5</td>
<td>88.4</td>
<td>81.9</td>
<td>83.6</td>
</tr>
<tr>
<td>KalignP 9 (94 b)</td>
<td>65.6</td>
<td>91.8</td>
<td>92.4</td>
<td>83.4</td>
<td>89.8</td>
<td>83.9</td>
<td>84.6</td>
<td></td>
</tr>
<tr>
<td>Mafft</td>
<td>223</td>
<td>56.0</td>
<td>89.6</td>
<td>91.1</td>
<td>84.0</td>
<td>87.8</td>
<td>84.9</td>
<td>81.8</td>
</tr>
<tr>
<td>ClustalW</td>
<td>1281</td>
<td>58.2</td>
<td>88.4</td>
<td>88.8</td>
<td>77.1</td>
<td>78.9</td>
<td>76.9</td>
<td>78.6</td>
</tr>
<tr>
<td>Muscle</td>
<td>1281</td>
<td>65.7</td>
<td>92.3</td>
<td>91.5</td>
<td>84.2</td>
<td>86.5</td>
<td>85.3</td>
<td>84.4</td>
</tr>
<tr>
<td>T_coffee</td>
<td>25 293</td>
<td>73.0</td>
<td>94.4</td>
<td>93.4</td>
<td>87.1</td>
<td>89.2</td>
<td>90.2</td>
<td>87.8</td>
</tr>
<tr>
<td>Probcons</td>
<td>26 085</td>
<td>74.0</td>
<td>94.6</td>
<td>93.7</td>
<td>87.5</td>
<td>90.3</td>
<td>90.1</td>
<td>88.3</td>
</tr>
</tbody>
</table>

^aThe CPU time refers to single threaded process running on a Linux box with Intel quad-core 2.50 GHz CPU and 4 GB memory.
^bReference 1: equi-distant sequences with ≤20% identity.
^cReference 2: families aligned with highly divergent 'orphan' sequences.
^dReference 3: subgroups with ≤25% residue identity between groups.
^eReference 4: sequences with N/C-terminal extensions.
^fReference 5: sequences with internal insertions.
^gReference 6: sequences with internal insertions. (Note: 5/8 values for paired t-test are 2.75e-4 and 3.85e-3 for SP and TC scores, respectively.)

Table 2. Benchmark on re7 (transmembrane proteins) of Balibase 3.0 (left) and PfamA-mem using 51 alignments of transmembrane proteins derived from Pfam-A seed alignments (right)

<table>
<thead>
<tr>
<th>Method</th>
<th>CPU^a (s)</th>
<th>SP</th>
<th>TC</th>
<th>CPU^a (s)</th>
<th>SP</th>
<th>TC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kalign2</td>
<td>5</td>
<td>89.2</td>
<td>43.6</td>
<td>3</td>
<td>84.6</td>
<td>59.3</td>
</tr>
<tr>
<td>KalignP 9 (974 h)</td>
<td>90.8</td>
<td>45.4</td>
<td>4 (88 b)</td>
<td>86.2</td>
<td>62.6</td>
<td></td>
</tr>
<tr>
<td>Mafft</td>
<td>12</td>
<td>87.0</td>
<td>43.0</td>
<td>20</td>
<td>83.6</td>
<td>56.6</td>
</tr>
<tr>
<td>ClustalW</td>
<td>155</td>
<td>79.2</td>
<td>40.4</td>
<td>20</td>
<td>84.0</td>
<td>59.3</td>
</tr>
<tr>
<td>Muscle</td>
<td>93</td>
<td>88.7</td>
<td>43.1</td>
<td>99</td>
<td>85.7</td>
<td>61.3</td>
</tr>
<tr>
<td>T_coffee</td>
<td>2946</td>
<td>90.8</td>
<td>50.2</td>
<td>3093</td>
<td>88.4</td>
<td>66.8</td>
</tr>
<tr>
<td>Probcons</td>
<td>3217</td>
<td>92.4</td>
<td>53.0</td>
<td>1441</td>
<td>88.6</td>
<td>67.1</td>
</tr>
</tbody>
</table>

^aThe CPU time refers to single threaded process running on a Linux box with Intel quad-core 2.50 GHz CPU and 4 GB memory.
^bReference 2: families aligned with highly divergent 'orphan' sequences.
^cReference 3: subgroups with ≤25% residue identity between groups.
^dReference 4: sequences with N/C-terminal extensions.
^eReference 5: sequences with internal insertions.
^fReference 6: sequences with internal insertions. (Note: 5/8 values for paired t-test are 2.75e-4 and 3.85e-3 for SP and TC scores, respectively.)

4 CONCLUSIONS

Here, we present KalignP, a method that allows externally supplied position specific gap penalty, and maintains the high speed and accuracy of Kalign2. Using gap-penalties from predicted secondary structures a steady improvement is seen. Further, KalignP is more flexible than Kalign2, as researchers can modify the behavior of alignment using other knowledge.

REFERENCES