BIOINFORMATICS APPLICATIONS NOTE

Vol. 21 no. 7 2005, pages 1271–1273
doi:10.1093/bioinformatics/bti142

Sequence analysis

Multiple sequence alignment with user-defined constraints at GOBICS

Burkhard Morgenstern[1,*], Nadine Werner[1], Sonja J. Prohaska[2], Rasmus Steinkamp[1], Isabelle Schneider[1], Amarendran R. Subramanian[3], Peter F. Stadler[2] and Jan Weyer-Menkhoff[1]

[1]Institut für Mikrobiologie und Genetik, Universität Göttingen, Abteilung für Bioinformatik, Goldschmidtstraße 1, D-37077 Göttingen, Germany, [2]Institut für Informatik und Interdisziplinäres Zentrum für Bioinformatik, Universität Leipzig, Germany and [3]Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Sand 13, D-72076 Tübingen, Germany

Received on August 17, 2004; revised on November 4, 2004; accepted on November 5, 2004
Advance Access publication November 16, 2004

ABSTRACT

Summary: Most multi-alignment methods are fully automated, i.e. they are based on a fixed set of mathematical rules. For various reasons, such methods may fail to produce biologically meaningful alignments. Herein, we describe a semi-automatic approach to multiple sequence alignment where biological expert knowledge can be used to influence the alignment procedure. The user can specify parts of the sequences that are biologically related to each other; our software program uses these sites as anchor points and creates a multiple alignment respecting these user-defined constraints. By using known functionally, structurally or evolutionarily related positions of the input sequences as anchor points, our method can produce alignments that reflect the true biological relationships among the input sequences more accurately than fully automated procedures can do.

Availability: Our software is available online at GÖttingen BIoinformatics Compute Server (GOBICS), http://dialign.gobics.de/anchor/index.php

Contact: burkhard@gobics.de

A large number of multi-alignment programs have been developed during the last 20 years, (for recent reviews see Notredame, 2002; Simossis and Heringa, 2004); the performance of these tools has been studied extensively (Lassmann and Sonnhammer, 2002; Pollard *et al*., 2004, http://www.biomedcentral.com/1471–2105/5/6). Practically all state-of-the-art alignment methods are fully automated. They construct alignments following a fixed set of algorithmical rules where only a limited number of parameters can be adjusted by the user. Automatic alignment methods are clearly necessary in situations where no expert knowledge about the input sequences is available or if large amounts of data are to be processed. However, if a researcher is already familiar with a specific sequence family under study, he or she may know certain regions in the sequences that are functionally or phylogenetically related and should therefore be aligned to each other. Here, it is useful to have an alignment method that can incorporate such user-defined homology information and then create an alignment respecting these constraints.

Multiple alignment under constraints has been proposed by Myers *et al*. (1996) and, more recently, by Sammeth *et al*. (2003, http://www.biomedcentral.com/1471–2105/4/66) and Brown and Hudek (2004). The multi-alignment program DIALIGN (Morgenstern, 2004; Morgenstern *et al*., 1996) has an option to calculate alignments under pre-defined constraints. Initially, this program feature has been implemented to reduce the alignment search space and program running time for large genomic sequences (Brudno *et al*., 2003; Morgenstern *et al*., 2002; Schmollinger *et al*., 2004). However, user-defined constraints—or anchor points, as we call them—can also be used to improve the biological quality of multiple alignments. To this end, known homologies can be specified by the user. A semi-automatic alignment procedure is then carried out where the user-specified homologies are aligned wherever possible; the remainder of the sequences is then automatically aligned by DIALIGN according to these user-defined constraints. A detailed description of this algorithm is given elsewhere (Morgenstern *et al*., 2004); this previous paper also describes applications of our approach to genomic sequences around the *Hox* gene cluster.

To make our anchored multi-alignment tool easily available to the research community, we developed a World Wide Web (WWW) interface at GOBICS (GÖttingen BIoinformatics Compute Server). The user can specify an arbitrary number of anchor points that are taken into account for the alignment. Each of these anchor point corresponds to a pair of equal-length segments of two of the input sequences, as shown in Figure 1. An anchor point is therefore characterized by five coordinates: the two sequences involved, the starting positions in the respective sequences and the length of the anchored segments. As a sixth parameter, our method requires a score that determines the priority of anchor points. The latter parameter is necessary, since it is in general not possible to use all proposed anchors simultaneously, so the algorithm may need to select a suitable subset of them. Here, our method uses the same greedy procedure that is used in the original DIALIGN approach to select consistent sets of local pairwise alignments for multiple alignment (Morgenstern, 1999).

Our anchoring procedure works as follows: if a position x in one of the input sequences is assigned to a position y in another input sequence through one of the selected anchor points, this does not

[*]To whom correspondence should be addressed.

Published by Oxford University Press 2004.

```
WKKNADAPKRAMTSFMKAAY
WNLDTNSPEEKQAYIQLAKDDRIRYD
WRMDSNQKNPDSNNPKAAYNKGDANAPK
```

(A) conserved motif

```
WKKNAD-----APKRamtsfmKAAY------------
WNLDTN-----SPEE------KQAYiqlaKDDriryd
WRMDSNqknpdSNNP------KAAYn---KGDsnapk
```

(B) default DIALIGN alignment without anchoring

```
WKKNADAPKRAMTSFMKAAY
WNLDTNSPEEKQAYIQLAKDDRIRYD
WRMDSNQKNPDSNNPKAAYNKGDANAPK
```

(C) anchor point #1

```
WKKNADAPKRAMTSFMKAAY
WNLDTNSPEEKQAYIQLAKDDRIRYD
WRMDSNQKNPDSNNPKAAYNKGDANAPK
```

(D) anchor point #2

```
1  2  4   6  6  4.5
2  3  6  11  8  1.3
```

(E) file with coordinates for anchor points

```
WKk-------NADAPKRAMTSFMKAa---Y-
WNLDT-----NSPEEKQAYIQLAKDDrirYd
WRMDSnqknpDSNNPKAAYn---KGDsnapk
```

(F) anchored alignment

Fig. 1. Multiple alignment with user-defined anchor points. **(A)** A (fictitious) set of protein sequences that contains a conserved motif (blue); we assume that this motif represents some real homology that is known to the user, e.g. a functional site. **(B)** The default version of DIALIGN is not able to recognize this homology; the motif is only partially aligned. To enforce alignment of the known motif, we define two anchor points **(C)** and **(D)**, shown in red. Each anchor point corresponds to a pair of equal-length segments of two of the input sequences. **(E)** A file containing the coordinates of the two anchor points is created. Each line corresponds to one of the anchor points, the numbers in a line denote the two sequences involved, the two respective starting points, and the length of the anchor point. In addition, a score is is to be specified as a sixth parameter for each anchor point to prioritize anchor points in case they are mutually inconsistent, i.e. if they exclude each other and cannot be used simultaneously in one alignment. The first line in our file corresponds to anchor point #1. It involves sequences 1 and 2, the starting points are 4 and 6, and the length of the anchor is 6. The score 4.5 is irrelevant in our example since the two anchor points are consistent with each other, so both can be used simultaneously to anchor the alignment. **(F)** The alignment created with these anchor points correctly aligns the motif (blue) and aligns the remainder of the sequences given the constraints imposed by the two anchor points.

necessarily mean that x and y will be aligned to each other in the output alignment. Rather, it means that y is the only position from the second sequence that can be aligned to x. However, whether or not x will actually be aligned to y depends on the degree of local sequence similarity among the sequences around positions x and y. If no statistically significant similarity can be detected, x and y may remain unaligned. In any case—even if x and y are not aligned—an anchor point connecting positions x and y ensures that positions to the left of x can be aligned only to positions to the left of y and vice versa. This relation is transitive (for details see Morgenstern *et al*., 2004). This way, anchor points reduce the alignment search space and speed up the program running time.

The scoring and greedy selection of anchor points makes it possible for the user to prioritize potential anchor points according to arbitrary criteria. In the *Hox* gene example described by Morgenstern *et al*. (2004), for example, biologically verified gene boundaries were used as anchor points to enforce correct alignment of duplicated genes. For large datasets, one may want to use local sequence similarities identified by some local alignment program as additional anchor points in order to speed up the alignment procedure as outlined by Brudno *et al*. (2003). In such a situation, known homologies such as gene boundaries are, of course, more reliable than automatically detected sequence similarities. Thus, it makes sense to first accept those homologies as anchors and then to define local alignments as additional anchors, under the condition that they are consistent with those known homologies. Hence in this case, one would give high scores to the biologically verified anchor points to ensure that they are given higher priority in case of possible inconsistencies, whereas automatically created anchor points would receive lower scores.

At our WWW server, anchor points for multiple alignment can be uploaded in a simple anchor file; the corresponding file format is explained in Figure 1 and in the online manual. Alternatively, if only a few anchor points are used, their coordinates can be entered through an online form on the submission page. In addition, several other program parameters are available. The output alignment is returned in several different formats together with the original list of anchor points that has been provided by the user. A second list of anchors is returned where all inconsistent anchor points are labelled, so that the user knows which of the proposed anchor points have actually been used for the alignment procedure. A detailed online manual explains all relevant program features such as input and output format and gives an example to demonstrate how anchor points affect the alignment procedure.

ACKNOWLEDGEMENTS

The work was supported by DFG grant MO 1048/1-1 to B.M., I.S. and J.W.-M. and by DFG Bioinformatics Initiative BIZ-6/1-2 to S.J.P. and P.F.S.

REFERENCES

Brown,D.G. and Hudek,A.K. (2004) New algorithms for multiple DNA sequence alignment. In Jonassen,I. and Kim,J. (eds), *Algorithms in Bioinformatics*, Vol. 3240 *Lecture Notes in Bioinformatics*, pp. 314–325.

Brudno,M., Chapman,M., Göttgens,B., Batzoglou,S. and Morgenstern,B. (2003) Fast and sensitive multiple alignment of large genomic sequences. *BMC Bioinformatics*, **4**, 66.

Lassmann,T. and Sonnhammer,E.L. (2002) Quality assessment of multiple alignment programs. *FEBS Lett.*, **529**, 126–130.

Morgenstern,B. (1999) DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, **15**, 211–218.

Morgenstern,B. (2004) DIALIGN: Multiple DNA and protein sequence alignment at BiBiServ. *Nucleic Acids Res.*, **32**, W33–W36.

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

Morgenstern,B., Prohaska,S.J., Werner,N., Weyer-Menkhoff,J., Schneider,I., Subramanian,A.R. and Stadler,P.F. (2004) Multiple sequence alignment with user-defined constraints. In *Proceedings of GCB'04*, Vol. **P-53** of *Lecture Notes in Informatics*, pp. 25–36.

Morgenstern,B., Rinner,O., Abdeddaïm,S., Haase,D., Mayer,K., Dress,A. and Mewes,H.-W. (2002) Exon discovery by genomic sequence alignment. *Bioinformatics*, **18**, 777–787.

Myers,G., Selznick,S., Zhang,Z. and Miller,W. (1996) Progressive multiple alignment with constraints. *J. Comput. Biol.*, **3**, 563–572.

Notredame,C. (2002) Recent progress in multiple sequence alignment: a survey. *Pharmacogenomics*, **3**, 131–144.

Pollard,D.A., Bergman,C.M., Stoye,J., Celniker,S.E. and Eisen,M.B. (2004) Benchmarking tools for the alignment of functional noncoding DNA. *BMC Bioinformatics*, **5**, 6.

Sammeth,M., Morgenstern,B. and Stoye,J. (2003) Divide-and-conquer alignment with segment-based constraints. *Bioinformatics*, **19**(Suppl. 2), ii189–ii195.

Schmollinger,M., Nieselt,K., Kaufmann,M. and Morgenstern,B. (2004) DIALIGN P: fast pair-wise and multiple sequence alignment using parallel processors. *BMC Bioinformatics*, **5**, 128.

Simossis,V. and Heringa,J. (2004) Integrating protein secondary structure prediction and multiple sequence alignment. *Curr. Protein Pept. Sci.*, **5**, 249–266.