Published online 24 April 2013 *Nucleic Acids Research, 2013, Vol. 41, Web Server issue* ***W3–W7***
doi:10.1093/nar/gkt283

DIALIGN at GOBICS—multiple sequence alignment using various sources of external information

Layal Al Ait[1,*], Zaher Yamak[2] and Burkhard Morgenstern[1,*]

[1]Department of Bioinformatics, University of Göttingen, Institute of Microbiology and Genetics, Goldschmidtstr. 1, 37077 Göttingen, Germany and [2]Bioinformatics group, Azm Center for Research in Biotechnology and its Applications, Ecole Doctorale des Sciences et de Technologie, Beirut, Lebanon

Received February 9, 2013; Revised March 30, 2013; Accepted March 31, 2013

ABSTRACT

DIALIGN is an established tool for multiple sequence alignment that is particularly useful to detect local homologies in sequences with low overall similarity. In recent years, various versions of the program have been developed, some of which are fully automated, whereas others are able to accept user-specified external information. In this article, we review some versions of the program that are available through 'Göttingen Bioinformatics Compute Server'. In addition to previously described implementations, we present a new release of DIALIGN called 'DIALIGN-PFAM', which uses hits to the PFAM database for improved protein alignment. Our software is available through http://dialign.gobics.de/.

INTRODUCTION

'DIALIGN' is a tool for pairwise and multiple alignment of nucleic acid or protein sequences (1). The program combines global and local alignment features, its main strength is its ability to discover local homologies among sequences without detectable global homology. This makes the program particularly useful to analyse remotely related protein families or genomic sequences where functional regions are typically conserved at the primary-sequence level, whereas non-functional parts of the sequences are less conserved. In many studies, DIALIGN has been successfully used to analyse protein families or genomic sequences, see e.g. (2,3).

Many versions of DIALIGN have been developed since the program was first introduced in 1996. The standard version of the program performs alignments without human intervention and is based on primary-sequence information alone. Later versions of DIALIGN can use additional sources of information or expert knowledge to produce more accurate alignments. The most recent addition is an option for protein alignment where the input sequences are searched against the Pfam database of protein domains (4). Positions of the sequences matching the same position in some Pfam domain are then preferably aligned (5). This latest program version is outlined in the present article.

During the first years, the main development work on DIALIGN was carried out at 'University of Bielefeld'. The 'Bielefeld Bioinformatics Server' (BiBiServ) still offers various program versions for online usage and for download. Later, the work on DIALIGN was continued at 'University of Göttingen', and more recent versions of the program are offered via 'Göttingen Bioinformatics Compute Server' (GOBICS) at www.gobics.de.

PREVIOUS VERSIONS OF DIALIGN

DIALIGN 2.2

To calculate a multiple sequence alignment (MSA), the standard version of the program, 'DIALIGN 2.2', first calculates all pairwise alignments of the input sequences as described in (6). That is, a 'sparse dynamic programming' algorithm is used to find an optimal alignment in the sense of a segment-based 'objective function' (7). MSAs are then calculated based on these pairwise alignments using a time-efficient greedy algorithm described in (8). No human intervention is necessary or possible. This version of the program is available through *BiBiServ* at http://bibiserv.techfak.uni-bielefeld.de/.

DIALIGN-TX

Greedy algorithms are fast but may be error prone. In DIALIGN, the greedy algorithm may select spurious random similarities among the input sequences that prevent the program from aligning biologically

*To whom correspondence should be addressed. Tel: +49 551 39 13957, Fax: +49 551 39 14929; Email: layal@gobics.de
Correspondence may also be addressed to Burkhard Morgenstern. Tel: +49 551 39 14966, Fax: +49 551 39 14929; Email: burkhard@gobics.de

meaningful homologies. Thus, a more recent development, 'DIALIGN-TX' (9), uses various heuristics to reduce the influence of isolated random similarities on the resulting MSA. Among other approaches, it uses a mixture of the 'greedy' algorithm used in the original 'DIALIGN' implementation with a more classical 'progressive' approach. 'DIALIGN-TX' is available online through 'GOBICS' at http://dialign-tx.gobics.de/; the source code is freely available from the same URL.

Anchored DIALIGN

Most MSA programs are fully automated. That is, except for parameter tuning, they do not allow nor require any human intervention during the alignment procedure. This is adequate, of course, if no further information is available, or if large amounts of data have to be analysed automatically. Often, however, the user of an MSA program has already some expert knowledge about the sequences to be aligned, e.g. he/she may know some homologies among the input sequences that should be aligned. In such cases, it would be desirable to have an MSA program that uses this expert information and aligns the remainder of the sequences automatically.

The 'anchored alignment' option in 'DIALIGN' is doing this (10). Here, the user can specify segments of the input sequences that should be aligned with each other, so-called 'anchor points' for the alignment. The remainder of the sequences is then aligned automatically, respecting the constraints given by the user-selected 'anchor points'. Technically, an 'anchor point' is a pair of equal-length segments from two distinct sequences. As it may not be possible to include all user-defined anchor points in one single output MSA, the program has to prioritize the proposed anchor points. To this end, 'scores' can be given to the selected anchor points to define their priority.

Aligning long DNA sequences with DIALIGN, CHAOS and ABC

The run time of most pairwise alignment methods is proportional to the product of the sequence length. Thus, if long genomic sequences are aligned, program run time becomes an issue. To overcome this problem, methods for genomic sequence alignment usually start with a fast search for strong local similarities. In a second step, sequences between those similarities are aligned with a slower, but more sensitive, method. On our web server, we use the program 'CHAOS' (11) to quickly identify local alignments of genomic sequences; we then align the remainder of the sequences with 'DIALIGN'. Finally, the results are visualized with the software 'ABC' (12), see (13) for more details. This approach is available on our server at http://dialign.gobics.de/ chaos-dialign-submission.

DIALIGN USING PFAM MATCHES

Recently, the developers of *Clustal Ω* proposed an approach to MSA that they called 'External Profile Alignment' (14). Here, the user can provide a pre-calculated 'profile HMM' (15) of a protein domain that he/she thinks may be present in the input sequences. Matching sequences are then locally aligned to this 'external profile' and thereby, indirectly, aligned to each other. In the latest version of 'DIALIGN', we apply this approach systematically. In short, we search all input sequences against the Pfam database of protein domains. Segments of the sequences matching to the same positions in some Pfam domain are then preferentially aligned in the final output MSA. We called this new approach 'DIALIGN-PFAM'; a first version of this approach is described in a conference paper (5). The algorithm described later in the text is slightly different from this original version; Figure 1 shows a flowchart for our algorithm.

Algorithm

Each protein family in Pfam is represented by a model consisting of one or several MSAs of domains and

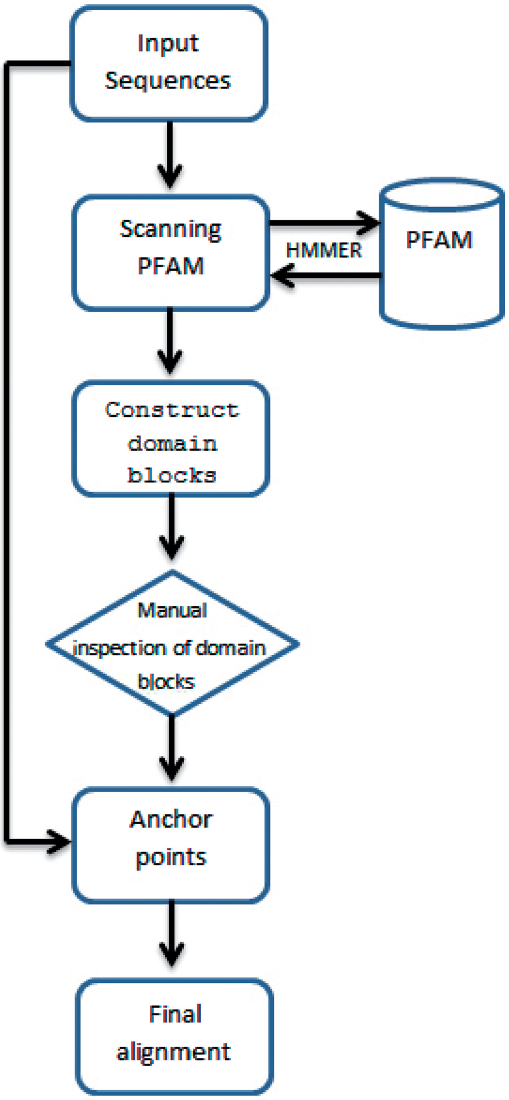


Figure 1. Flowchart of DIALIGN-PFAM.

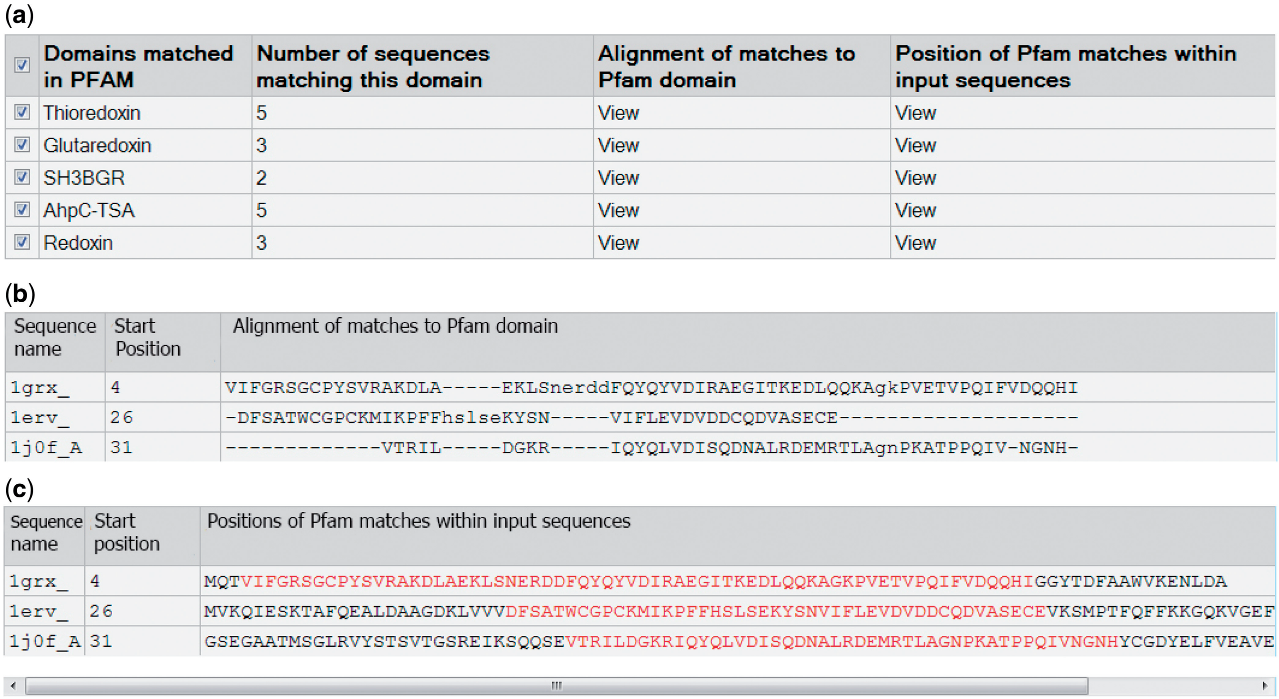
‘profile Hidden Markov Models’ (pHMM) derived from these alignments. The first step in our approach is to scan the input sequences against Pfam using ‘HMMER’ (16).

‘HMMER’ assigns quality scores to matches between a query protein sequence and models of protein domains in a database. To control which ‘HMMER’ hits are used by our algorithm, we use two threshold values for the *E*-values of these hits. Our first threshold parameter, E_m, applies to full models in Pfam and ensures that only models with an *E*-value less than E_m are taken into consideration. The second threshold, E_d, applies to single domains such that profiles, which satisfy the first threshold condition, are further filtered with this one. As default values, we use 5×10^{-3} for E_m and 10^{-4} for E_d.

After ‘HMMER’ matches to Pfam are obtained and filtered with our threshold parameters, the next step is to construct so-called ‘domain blocks’, which are the basis of our alignment approach. A ‘domain block’ consists of two or more segments of the input sequences that are matched, possibly with gaps, to the same Pfam domain. This way, segments from one ‘domain block’ are, indirectly, aligned to each other, i.e. two positions from the input sequences are aligned if they are matched to the same position in some Pfam domain.

In a third step, the user can manually inspect the aligned ‘domain blocks’ obtained in this way and select or deselect them for the final multiple alignment step.

Finally, the selected ‘domain blocks’ are used by ‘DIALIGN’ as ‘anchors’ to calculate a multiple alignment of the input sequences. Technically, pairs of segments of the input sequences aligned to the same segment in some Pfam domain are defined as ‘anchor points’. For a single Pfam domain, it is usually possible to integrate all derived ‘anchor points’ into one output MSA. In ‘DIALIGN’ terminology, these anchor points are generally ‘consistent’ with each other. It may not be possible, however, to integrate anchor points from all the selected ‘domain blocks’ into one single output MSA. Because of such possible ‘inconsistencies’, we have to determine the priority of the selected blocks. To this end, we define for each ‘domain block’ a ‘score’, as the sum of the scores of all involved ‘HMMER’ hits to Pfam. The priority of an anchor point is then defined according to this score; anchor points derived from our domain blocks are considered in the order of decreasing scores. That means, our program first



(a)

☑	Domains matched in PFAM	Number of sequences matching this domain	Alignment of matches to Pfam domain	Position of Pfam matches within input sequences
☑	Thioredoxin	5	View	View
☑	Glutaredoxin	3	View	View
☑	SH3BGR	2	View	View
☑	AhpC-TSA	5	View	View
☑	Redoxin	3	View	View

(b)

Sequence name	Start Position	Alignment of matches to Pfam domain
1grx_	4	VIFGRSGCPYSVRAKDLA-----EKLSnerddFQYQYVDIRAEGITKEDLQQKAgkPVETVPQIFVDQQHI
1erv_	26	-DFSATWCGPCKMIKPFFhslseKYSN-----VIFLEVDVDDCQDVASECE--------------------
1j0f_A	31	-------------VTRIL-----DGKR-----IQYQLVDISQDNALRDEMRTLAgnPKATPPQIV-NGNH-

(c)

Sequence name	Start position	Positions of Pfam matches within input sequences
1grx_	4	MQTVIFGRSGCPYSVRAKDLAEKLSNERDDFQYQYVDIRAEGITKEDLQQKAGKPVETVPQIFVDQQHIGGYTDFAAWVKENLDA
1erv_	26	MVKQIESKTAFQEALDAAGDKLVVVDFSATWCGPCKMIKPFFHSLSEKYSNVIFLEVDVDDCQDVASECEVKSMPTFQFFKKGQKVGEF
1j0f_A	31	GSEGAATMSGLRVYSTSVTGSREIKSQQSEVTRILDGKRIQYQLVDISQDNALRDEMRTLAGNPKATPPQIVNGNHYCGDYELFVEAVE

Figure 2. Example program run with DIALIGN-PFAM. An input file with seven protein sequences was uploaded to our server. Our program used HMMER to search each of the seven input sequences against Pfam. Overall, matches to five different Pfam domains were found by HMMER. (**a**) Each line in the first table corresponds to one of the matched Pfam domains, e.g. the first line corresponds to the *Thioredoxin* domain. The second column indicates how many of the input sequences matched to the respective domain (e.g. five of our seven input sequences matched to the *Thioredoxin* domain). By clicking ‘View’ in the third and fourth column, respectively, the user can look at ‘alignments’ of the Pfam matches and at their positions within the input sequences. The checkboxes on the left-hand side can be used to select/deselect matches to Pfam domains as anchor points for the final MSA calculated by our program. By default, all matches are selected. (**b**) The second table is obtained by clicking ‘View’ in the third column of table (a). It shows a multiple alignment of segments of the input sequences matching to the same Pfam domain (so-called ‘local view’). In our example (b), three input sequences (1grx_, 1erv_, 1j0f_A) were matched to the same Pfam domain. The alignment in (b) was constructed by our program by aligning those sequence positions to each other that were matched by HMMER to the same position in the corresponding Pfam domain. (**c**) The third table is obtained by clicking ‘View’ in the fourth column of the table in (a). It shows the ‘global view’, i.e. the positions of the matching segments in the respective input sequences. Segments matched by HMMER to the corresponding Pfam domain are shown in red.

accepts all anchor points from the 'domain block' with the highest score, then the anchor points from the block with the second highest score—as long as they are consistent with the already accepted anchor points—and so forth.

Interactive selection of blocks

After all 'domain blocks' have been calculated as described earlier in the text, the user has the option to view these blocks in two different ways. A 'local view' shows the local MSA, possibly containing gaps, that has been derived from all matches to one specific Pfam domain. In addition, a 'global view' of a given block is provided showing the non-aligned full input sequences with the segments from the block highlighted. By default, all the constructed blocks are included in the multiple alignment process, but the user can decide to discard an arbitrary number of blocks.

In our original conference paper (5), we reported benchmark results on 'BAliBASE' (17) and 'SABmark' (18) for a previous version of our algorithm. In short, 'DIALIGN' using Pfam hits performed consistently better than the standard version of the program that uses primary-sequence information alone. The modified algorithm outlined in the present article produces slightly better results than the original version described in (5), but is considerably faster. We are planning to give a detailed comparison of these two algorithms in an extended journal version of our conference paper.

Input/Output

'DIALIGN-PFAM' takes as an input a file in 'FASTA' format containing a set of protein sequences. The user can adjust the threshold parameters E_m and E_d for the Pfam search; default values are provided. As scanning Pfam with 'HMMER' may take a while, the user is given a URL where he/she can retrieve the results of the HMMER search later, to continue with the next step of the program. Figure 2 shows the local and global view on a simple 'domain block' involving three sequences identified from an input set of seven protein sequences. As the final alignment process by DIALIGN may also take some time, the user is given another URL to retrieve the final MSA later. The result of a program run will be stored and are downloadable from our server for 1 week. 'DIALIGN-PFAM' is available online at http://dialign-pfam.gobics.de/ SequenceAlignment/.

Example

Figure 2 shows an example of how 'domain blocks' are shown to the user by DIALIGN-PFAM. Here, we ran the program on a set of seven protein sequences. Matches to five different Pfam domains were found by 'HMMER'. As shown in Figure 2a, five of the input sequences had matches to the *Thioredoxin* domain, three sequences had matches to the *Glutaredoxin* domain, two sequences had matches to the *SH3BGR* domain, five sequences had matches to *AhpC-TSA* domain and three sequences had matches to the *Redoxin* domain. In Figure 2b, the 'local' view of the *Glutaredoxin* domain block is shown. Figure 2c shows the 'global' view of this domain block within the input sequences; here, matches to the *Glutaredoxin* domain are shown in red.

ACKNOWLEDGEMENTS

The authors thank Dr. Eduardo Corel and Dr. Thomas Lingner for helpful discussions and for exchanging interesting ideas and thoughts, and Dr. Kifah Tout for general support.

FUNDING

Funding for open access charge: Department fund.

Conflict of interest statement. None declared.

REFERENCES

1. 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.
2. Göttgens,G., Barton,L., Gilbert,J., Bench,A., Sanchez,M., Bahn,S., Mistry,S., Grafham,D., McMurray,A., Vaudin,M. *et al.* (2000) Analysis of vertebrate SCL loci identifies conserved enhancers. *Nat. Biotechnol.*, **18**, 181–186.
3. Stanke,M., Tzvetkova,A. and Morgenstern,B. (2006) AUGUSTUS+ at EGASP: using EST, protein and genomic alignments for improved gene prediction in the human genome. *Genome Biol.*, **7**, S11.
4. Finn,R., Tate,J., Mistry,J., Coggill,P., Sammut,J., Hotz,H., Ceric,G., Forslund,K., Eddy,S., Sonnhammer,E. *et al.* (2008) The Pfam protein families database. *Nucleic Acids Res.*, **36**, D281–D288.
5. Ait,L.A., Corel,E. and Morgenstern,B. (2012) Using protein-domain information for multiple sequence alignment. In: *Proceedings of the IEEE 12th International Conference on BioInformatics and BioEngineering (BIBE 12)*. LArnaca, Cyprus, pp. 163–168.
6. Morgenstern,B. (2002) A simple and space-efficient fragment-chaining algorithm for alignment of DNA and protein sequences. *Appl. Math. Lett.*, **15**, 11–16.
7. Morgenstern,B. (1999) DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment. *Bioinformatics*, **15**, 211–218.
8. Abdeddaïm,S. and Morgenstern,B. (2000) Speeding up the DIALIGN multiple alignment program by using the 'greedy alignment of biological sequences library' (GABIOS-LIB). In: Caraux,G., Gascuel,O. and Sagot,M.F. (eds), In: *Proceedings of the Journées Ouvertes: Biologie, Informatique et Mathématiques (JOBIM)*. Montpellier, pp. 1–8.
9. Subramanian,A.R., Kaufmann,M. and Morgenstern,B. (2008) DIALIGN-TX: greedy and progressive approaches for the segment-based multiple sequence alignment. *Algorithms Mol. Biol.*, **3**, 6.
10. Morgenstern,B., Prohaska,S.J., Pöhler,D. and Stadler,P.F. (2006) Multiple sequence alignment with user-defined anchor points. *Algorithms Mol. Biol.*, **1**, 6.
11. 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, http://www.biomedcentral.com/1471-2105/4/66.
12. Cooper,G.M., Singaravelu,S.A. and Sidow,A. (2004) ABC: software for interactive browsing of genomic multiple sequence alignment data. *BMC Bioinformatics*, **5**, 192.
13. Brudno,M., Steinkamp,R. and Morgenstern,B. (2004) The CHAOS/DIALIGN WWW server for multiple alignment of genomic sequences. *Nucleic Acids Res.*, **32**, W41–W44.

14. Sievers,F., Wilm,A., Dineen,D., Gibson,T.J., Karplus,K., Li,W., Lopez,R., McWilliam,H., Remmert,M., Sding,J. *et al*. (2011) Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol. Syst. Biol.*, **7**, 539.
15. Eddy,S. (1998) Profile hidden Markov models. *Bioinformatics*, **14**, 755–763.
16. Finn,R., Clements,J. and Eddy,S. (2011) HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res.*, **39**, W29–W37.
17. Thompson,J.D., Koehl,P., Ripp,R. and Poch,O. (2005) BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, **61**, 127–136.
18. Walle,I.V., Lasters,I. and Wyns,L. (2005) SABmark - a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics*, **21**, 1267–1268.