Rapid Automatic Detection and Alignment of Repeats in Protein Sequences

Andreas Heger* **and Liisa Holm**
European Bioinformatics Institute, Cambridge, United Kingdom

ABSTRACT **Many large proteins have evolved by internal duplication and many internal sequence repeats correspond to functional and structural units. We have developed an automatic algorithm, RADAR, for segmenting a query sequence into repeats. The segmentation procedure has three steps: (i) repeat length is determined by the spacing between suboptimal self-alignment traces; (ii) repeat borders are optimized to yield a maximal integer number of repeats, and (iii) distant repeats are validated by iterative profile alignment. The method identifies short composition biased as well as gapped approximate repeats and complex repeat architectures involving many different types of repeats in the query sequence. No manual intervention and no prior assumptions on the number and length of repeats are required. Comparison to the Pfam-A database indicates good coverage, accurate alignments, and reasonable repeat borders. Screening the Swissprot database revealed 3,000 repeats not annotated in existing domain databases. A number of these repeats had been described in the literature but most were novel. This illustrates how in times when curated databases grapple with ever increasing backlogs, automatic (re)analysis of sequences provides an efficient way to capture this important information. Proteins 2000;41:224–237.**

Key words: multiple alignment; repeat; trace matrix; dot plot; profile; Smith-Waterman

INTRODUCTION

Evolution modifies and recombines existing building blocks instead of inventing everything from scratch. In the protein world, these building blocks have been termed “domains”[1,2] and the identification and characterisation of new domains is a major goal of protein science. Automated methods exist that systematically try to find shared building blocks between proteins. The most sensitive methods employ exhaustive structural comparisons whereas the more complete methods in terms of protein space coverage use extensive sequence comparisons.[3,4] A fast route to discover new domains and motifs is to look for internal duplications or multiplications in one protein sequence (i.e., internal repeats). In this article, we present a sensitive and fast method to detect and align repeats in a protein sequence.

A variety of repeats can be observed in biological sequences.[5] Very short repeats of one or a few amino acid lengths are usually regarded as composition bias. Small repeats like zinc fingers or leucine rich repeats often correspond to distinct structural modules. Multiplication of complete globular domains, like immunoglobulin domains, has been used by evolution for building large proteins, for example, titin.[6] In proteins, we typically observe approximate repeats. Due to deletions, insertions, and mutations, repeat units have diverged, and often they are separated by large gaps. The task of finding the correct periodicity and repeat boundaries is not trivial, and methods using a variety of different approaches have been proposed.[7–13]

One class of algorithms uses dynamic programming to detect repeats by aligning a sequence to itself and analysing the collection of suboptimal alignments. A sequence can be aligned to another one using the Smith-Waterman algorithm.[14] This algorithm finds the locally optimal alignment between two sequences given substitution scores for matching similar residues and gap penalties for introducing insertions or deletions. Suboptimal alignments can be retrieved using a modified version of the Smith-Waterman algorithm.[15] The presence of suboptimal alignments indicates that the sequence might contain a repeat. The statistical significance of suboptimal alignments can be estimated by sequence shuffling.[13] Having a model for one repeat unit, a slightly modified version of the Smith-Waterman algorithm[16] can then align this repeat to the sequence, where the alignment is allowed to wrap around past the end of the repeat to the beginning of the next repeat unit.

The collection of suboptimal alignments can be visualized in a trace-plot (e.g., PLFASTA[17]). This is a $N \times N$ matrix for a sequence of length N, where a dot is displayed in row i and column j, if the entry in the matrix for the two residues i and j is part of a suboptimal alignment.

Here, we present a fast and sensitive algorithm for finding repeats in proteins based on the analysis of the trace matrix. The algorithm takes in a single query sequence and returns an explicit multiple alignment of repeats found in the sequence. No assumptions about the expected length and number of repeat units are made and no restrictions are imposed on sequence length and sequence separation between repeat units. Specifically, we incorporated the following ideas into the algorithm: (1)

*Correspondence to: Andreas Heger, European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, United Kingdom. E-mail: heger@ebi.ac.uk

Received 7 April 2000; Accepted 28 June 2000

TABLE I. SWISS-PROT Accession Numbers for Protein Sequences Mentioned in this Article

Identifier	Accession number
APOH_CANFA	P33703
EXG1_COCCA	P49426
FINC_BOVIN	P07589
KSC5_ECOLI	P42217
LIPA_NEIME	Q05013
MUC2_HUMAN	Q02817
PGBM_HUMAN	P98160
SPCA_DROME	P13395
TF3B_CANAL	P43072
YPX2_CAEEL	Q20256
YQ12_CAEEL	Q09449
ZN85_HUMAN	Q03923
ZYX_CHICK	Q04584

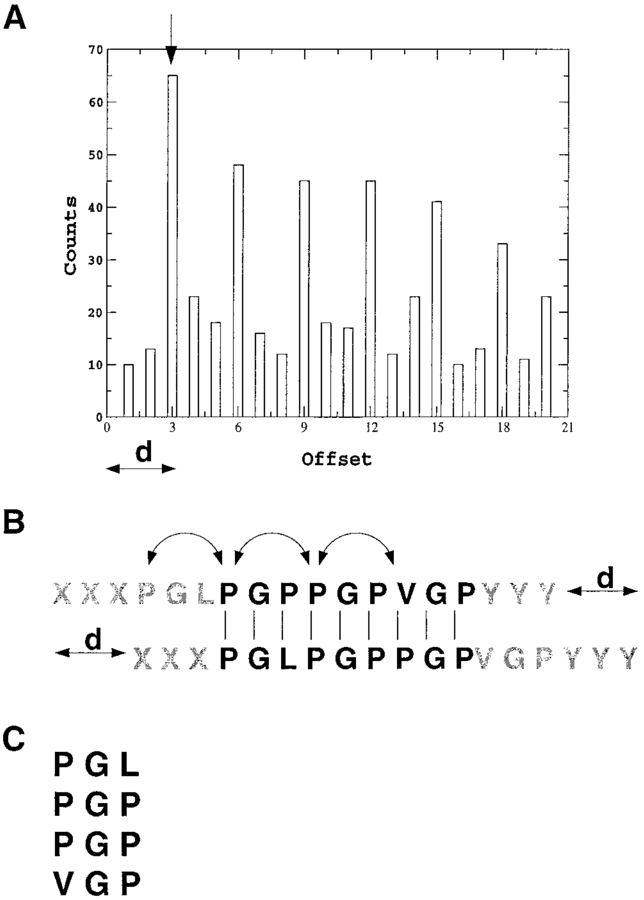


Fig. 1. Locating strict tandem repeats. (**A**) A histogram of offsets of identical words of length 3 in the query sequence (here: the collagen CA1C_CHICK). The offset *d* with maximum counts corresponds to the length of one repeat unit. (**B**) The sequence is aligned with offset *d* to itself. Ends with negative scores are trimmed (grey residues). The multiple alignment (**C**) is created by mapping residue *i* in the aligned region to residue *i* + *d.*

Information about residue conservation is used to increase the sensitivity of detection. (2) The search space is restricted to the set of residues aligned in the trace matrix (this rules out the majority of unlikely alignments and thereby increases speed). (3) The algorithm works recursively to find the shortest, nonreducible repeat unit. (4) The algorithm works iteratively to find different types of repeats.

We assess the sensitivity and accuracy of the algorithm using as reference alignments created by Pfam,[18] which employs one the most sensitive multiple sequence alignment methods available. Finally, the algorithm is applied to a representative subset of SWISS-PROT[19] in order to find novel repeats.

MATERIALS AND METHODS

Datasets

NRDB90 and PairsDB

NRDB90[20] is a filtered subset of the union of several sequence databases (SWISS-PROT, SWISS-NEW, TREMBL, TREMBLNEW, Genbank, PIR, Wormpep, and PDB) where no two sequences are more than 90% identical. A database of precomputed multiple alignments was created by performing an all-vs.-all database search with BLAST[21] among the sequences in NRDB90. This database (PairsDB) was used for fast look-up of multiple alignments needed for the calculation of a sequence profile for the query sequence.

Pfam

Pfam[18] was used as a reference to assess the quality of repeat detection. All sequences of Pfam-A, Release 4.4 with multiple hits of at least one family hidden Markov model were selected. Sequences not present in NRDB90 and alignments with less than ten residues overlap were excluded. Collagen-repeats and bacterial transferase hexapeptide repeats (PF00132) were eliminated from the dataset, since the Pfam domain is built up of more than one repeat unit. 11-S sulfur clusters were not considered because of their low complexity. The final dataset contained 3057 repeat types in 2800 sequences. Subsequently, the repeats were partitioned into two sets according to average sequence identity using a threshold of 30%. The average sequence identity in the multiple alignment was calculated columnwise. For every column x in the multiple alignment the number of occurrences $n_{i,x}$ of each amino acid type was determined. The average sequence identity c_x of column x was then calculated by the following formula:

$$c_x = \frac{\sum_{i=1,n_{i,x}>1}^{20} n_{i,x}}{\sum_{i=1}^{20} n_{i,x}} * \frac{1}{\sum_{i=1,n_{i,x}>1}^{20} 1} \qquad (1)$$

If there were no identical amino acids at all in column x, c_x was set to 0. To obtain the average sequence identity for the whole multiple alignment, the c_x were averaged over all columns. For example, if an alignment of four sequences contains the column $(AACC)^T$, normalization per amino acid classes according to Eq. (1) gives 50% average

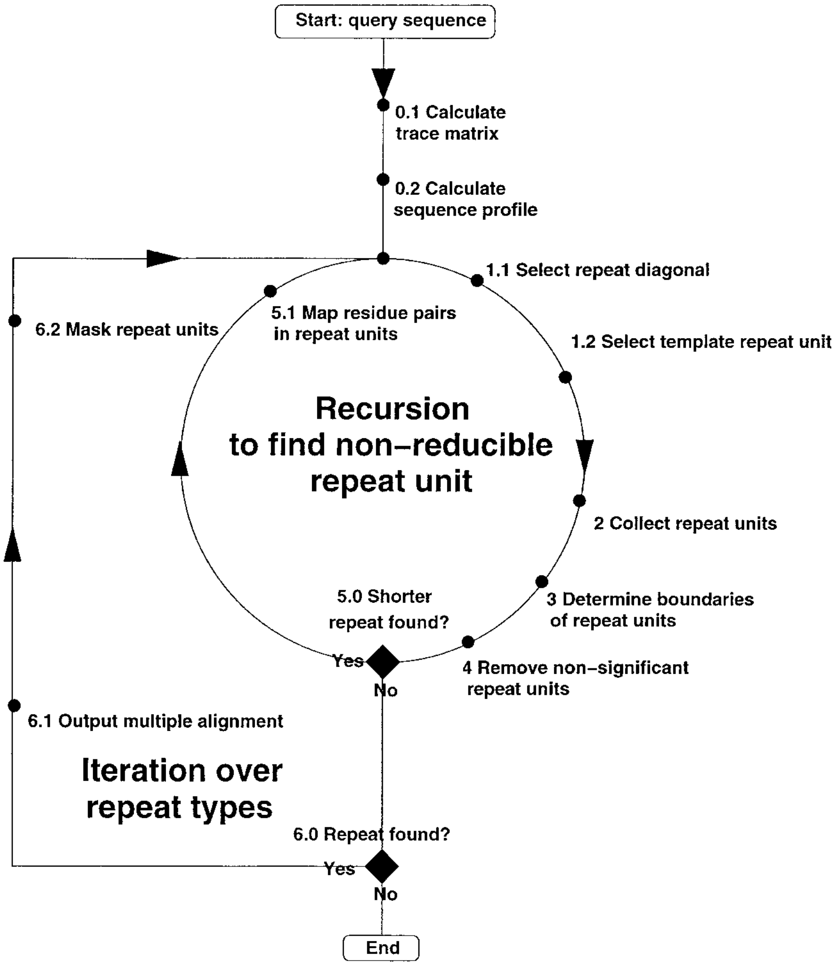

Start: query sequence
0.1 Calculate trace matrix
0.2 Calculate sequence profile
1.1 Select repeat diagonal
1.2 Select template repeat unit
2 Collect repeat units
3 Determine boundaries of repeat units
4 Remove non-significant repeat units
5.0 Shorter repeat found?
Yes
No
5.1 Map residue pairs in repeat units
Recursion to find non-reducible repeat unit
6.2 Mask repeat units
6.1 Output multiple alignment
Iteration over repeat types
6.0 Repeat found?
Yes
No
End

Fig. 2. Overview of the sequence of steps in the algorithm for finding long divergent repeats. The algorithm accepts a single sequence as input and retrieves a multiple alignment from a precomputed database of BLAST-alignments. The inner loop is recursive and is used to reduce the repeat unit length to the shortest, nonreducible length. The outer loop iterates over different repeat types present in the sequence.

identity. This is deemed more intuitive than simply averaging over all six possible residue pairs which would yield an average sequence identity of $(2*100\% + 4*0\%)/6 = 33\%$.

SWISS-PROT

SWISS-PROT[19] is a manually curated and extensively annotated database of protein sequences. We used all entries in SWISS-PROT (Release 38) which were present in nrdb90. The resulting dataset contained 44217 sequences. In regions with composition bias, often noninformative repeats are detected. To avoid those, and to reduce CPU time, regions with composition bias were masked (Casari et al., unpublished).

Novel repeats detected by RADAR were identified by checking the annotation of SWISS-PROT entries as follows: only those sequences were considered, which did not contain the keyword “repeat,” had no multiply occurring entries in the feature table labelled as “domain,” and had at most one hit to every Pfam hidden Markov model. The complete and filtered results are available via ftp at ftp.ebi.ac.uk/pub/contrib/heger/radar. Throughout the text, protein sequences are labelled with their SWISS-PROT identifier. The corresponding stable accession numbers are listed in Table I.

Eliminating Ungapped Short Repeats

To reduce the complexity of the search for large approximate repeats, we first use a fast procedure for finding and masking regions with runs of short conserved repeats that can be aligned without gaps. Examples of this type of repeats are collagens, which are made up of a large number of triplet-repeats.

The query sequence is divided into words of length (ktuple) three, and a histogram with the sequence separation of identical words is created, similar to FASTA[17] (Fig. 1A). The offset with the highest number of counts is taken as the length d of one repeat unit. We limit the range of d to between 1 and 20 residues. In the next step, the sequence is aligned to itself using offset d (Fig. 1B). The alignment is then trimmed at the ends to achieve the optimum positive alignment score, where the scores for residue matches and mismatches are given by the BLOSUM50 substitution matrix.[22]

A multiple alignment is generated from the pairwise alignment by a gapless wraparound procedure: every residue i in the alignment is mapped to the same column as residue $i + d$, where d is the length of one repeat unit (Fig. 1C). Because the alignment is gapless, this is

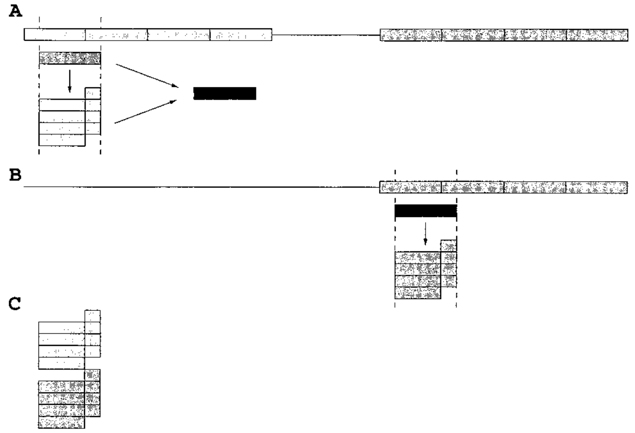


Fig. 3. Iteratively collecting repeat units. (**A**) A repeat is shown with a large gap in between the fourth and fifth repeat units (shaded boxes). The template repeat (grey box) is aligned to the sequence using wraparound dynamic programming. The first four repeat units are collected. The amino acid counts in the template repeat are updated with the counts from the four repeat units and a new profile for the template repeat is calculated. (**B**) Another alignment is performed using the new template repeat and the remaining four repeat units are collected. (**C**) The final multiple alignment of all repeat units as a result of two iterations.

consistent. Only strict tandem repeats are detected this way.

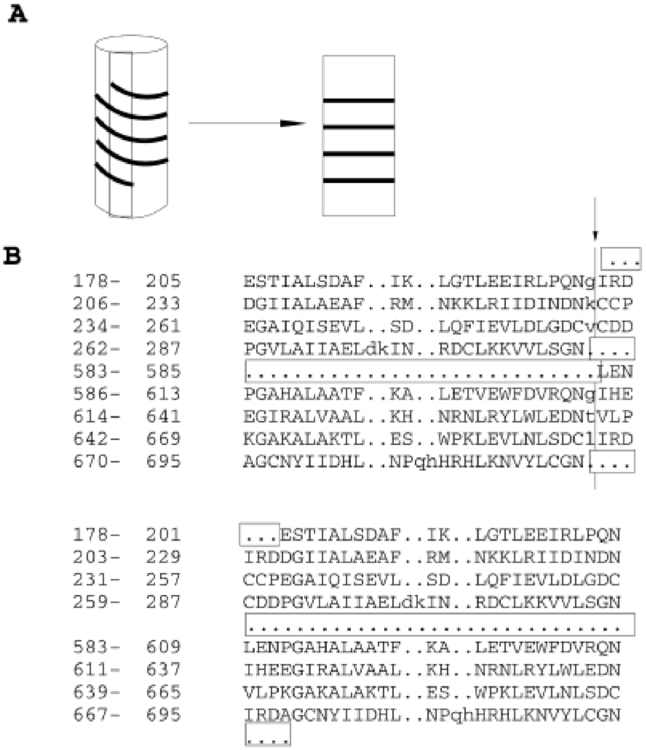
The threshold for accepting an alignment was set to 60. If an alignment has a score greater than the threshold, then all residues involved are masked and will not be considered for any subsequent alignment. The procedure is repeated, until no strict tandem repeats greater than the threshold can be found.

Algorithm for Detecting and Aligning Repeats

The central object in the algorithm is the trace matrix, the collection of suboptimal self–self alignments. More precisely, a trace matrix for a sequence of length N is an $N \times N$ symmetric matrix. It contains the substitution score in row i and column j, if residues i and j are aligned in a legitimate suboptimal alignment of the sequence with itself. All other entries are never visited; conceptually, they are set to minus infinite. The trace matrix is usually quite sparse.

Our algorithm tries to find the highest scoring path in the trace matrix. The search proceeds recursively to find the smallest nonreducible repeat unit and iteratively to collect all different repeat types present in the sequence. Figure 2 outlines the sequence of steps described in detail in the following sections. During its way through the recursion cycle, the putative repeat has to pass several checkpoints. Whenever it falls below one of the thresholds, the recursion is stopped and the algorithm continues to step 6 (see below).

0.1 Calculation of Suboptimal Alignments and Collection of Residue Pairs. The first step is the creation of the initial trace matrix. Input is the sequence from the previous filtering step with all residues being masked that were already identified as part of a strict tandem repeat. Suboptimal alignments of the query sequence are obtained from the program lfasta from the FASTA package. The lfasta program was run twice with two different scoring matrices (BLOSUM50 and PAM250[23]) using default gap penalties and a ktup-value of 1. All pairs of matched residues (i, j) in the reported alignments are entered into the trace matrix. These "dots" (i, j) are rescored using a profile in the next step.



```
                                          ...
178-  205     ESTIALSDAF..IK..LGTLEEIRLPQNgIRD
206-  233     DGIIALAEAF..RM..NKKLRIIDINDNkCCP
234-  261     EGAIQISEVL..SD..LQFIEVLDLGDCvCDD
262-  287     PGVLAIIAELdkIN..RDCLKKVVLSGN....
583-  585     .............................LEN
586-  613     PGAHALAATF..KA..LETVEWFDVRQNgIHE
614-  641     EGIRALVAAL..KH..NRNLRYLWLEDNtVLP
642-  669     KGAKALAKTL..ES..WPKLEVLNLSDClIRD
670-  695     AGCNYIIDHL..NPqhHRHLKNVYLCGN....

178-  201     ...ESTIALSDAF..IK..LGTLEEIRLPQN
203-  229     IRDDGIIALAEAF..RM..NKKLRIIDINDN
231-  257     CCPEGAIQISEVL..SD..LQFIEVLDLGDC
259-  287     CDDPGVLAIIAELdkIN..RDCLKKVVLSGN
              ...............................
583-  609     LENPGAHALAATF..KA..LETVEWFDVRQN
611-  637     IHEEGIRALVAAL..KH..NRNLRYLWLEDN
639-  665     VLPKGAKALAKTL..ES..WPKLEVLNLSDC
667-  695     IRDAGCNYIIDHL..NPqhHRHLKNVYLCGN
              ....
```

Fig. 4. Finding repeat boundaries. (**A**) Cutting an alignment wrapped around a cylinder. Depending on the size of the overlap of the terminal repeat units (shaded region), a cut is placed within or without the overlap region. (**B**) An example (YK63_CAEEL). The overlap between the first and last repeat is large, the cut is therefore placed in the nonoverlapping region (arrow). As a result, the internal gap has moved in between repeat units.

0.2 Profile Calculation. Even in the most diverged repeats, at least some key residues are usually conserved. This is crucial information and we use it in our algorithm. Information about residue conservation can be obtained by using an alignment of the query sequence to related sequences, from which a sequence profile[24] (position-specific scoring matrix) can be generated. Multiple alignments for the query sequence were retrieved from a database of precomputed multiple alignments (PairsDB, see datasets). A sequence profile (position-specific scoring matrix) was derived using a nine-component Dirichlet-mixture regulariser.[25] Having obtained the profile, the scores for all residue pairs in the trace matrix from the previous step are recalculated, but this time using information about residue conservation.

1 Creating a Model of the Repeat Unit. To collect and align repeat units in the query sequence, we need a model of one repeat unit. We do this by selecting a template repeat unit from the query sequence. Because the

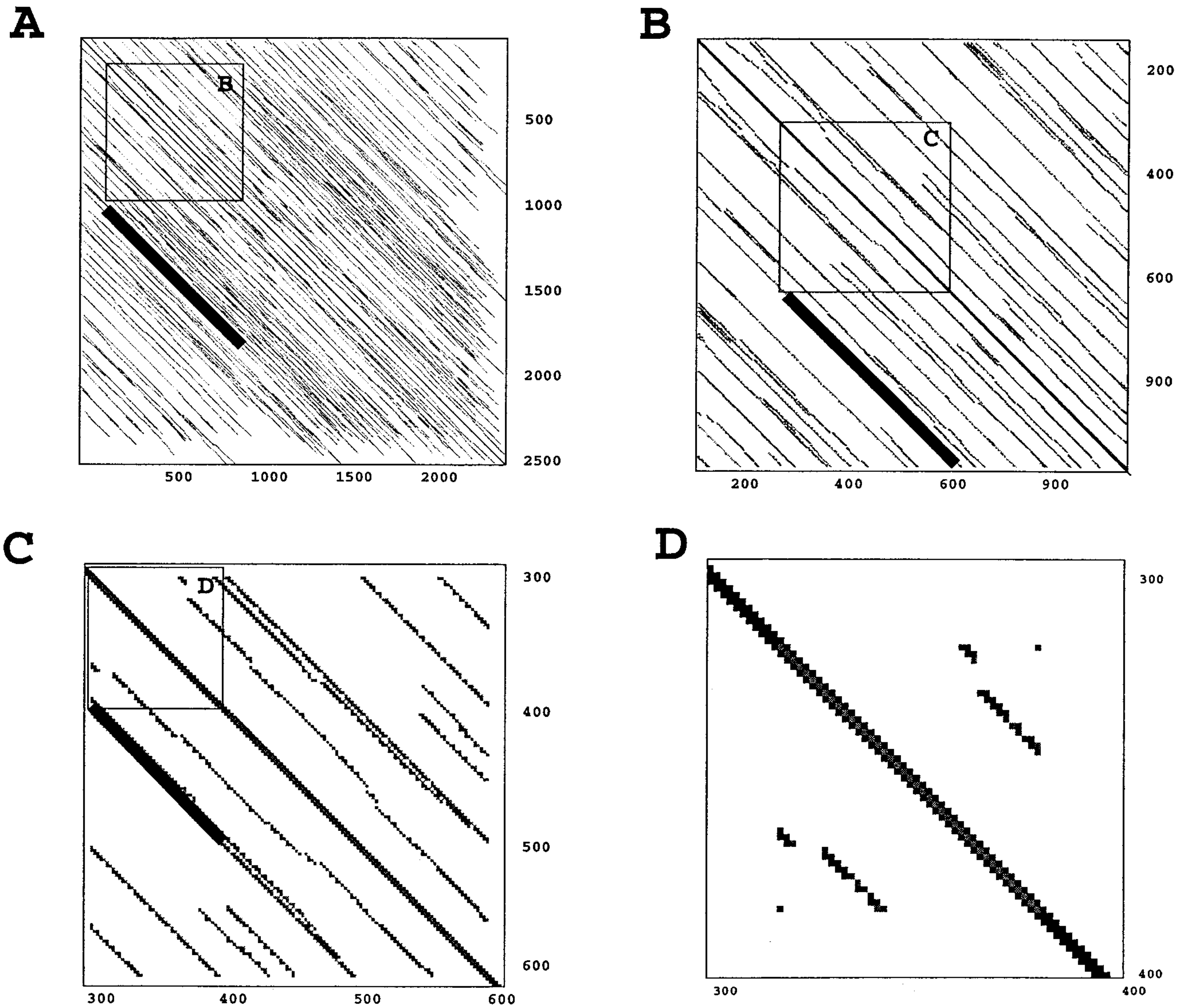


Fig. 5. Finding the shortest, nonreducible repeat unit. An example is shown for spectrin (SPCA_DROME). Spectrin contains 21 repeat units of length 100 separated into two blocks by an SH3 domain. (**A**) The initial trace matrix. The offset of the first repeat diagonal (thick black line) is 900 residues, which corresponds roughly to the separation of the two blocks. The box is drawn around the template repeat. (**B**) Input for the first recursion is the area around the template repeat unit from the previous step. Alignment traces from other parts of the trace-matrix have been mapped to this region. The template repeat chosen now has a length of roughly 300, corresponding to three repeat units. (**C**) Input for the second recursion. The template repeat chosen now has the correct length of 100. (**D**) Input for the third recursion. No significant alignments were found. Therefore, the result of C is output.

template repeat has a profile associated with it, this is already a generalised model of a repeat unit. The selection of the template repeat proceeds in two steps. First, we estimate the length of one repeat unit by identifying the repeat diagonal d (see below). The template repeat unit is then selected by finding the highest scoring local alignment of length d on this diagonal, where the alignment trace is allowed to deviate from the diagonal within limits.

1.1 Selection of Repeat Diagonal. The repeat diagonal is selected by adding up the residue pair scores on each diagonal, skipping entries which are minus infinite. The scores of the diagonals are then averaged over a window size of three and the highest scoring one is selected. In long proteins, it might happen that high scoring segments along one diagonal are masked by low scoring fragments somewhere else on the same diagonal. Therefore, if no diagonal has been found with a positive score in the first step, the same calculation is performed using only positive residue pair scores. The distance d from the highest scoring diagonal to the main diagonal gives the initial estimate of the length of the template repeat. If no repeat diagonal can be found, the recursion exits (go to step 6).

1.2 Select Template Repeat Unit. An alignment is performed, where the search space is restricted to a region centered around the repeat diagonal d. The width of the region is set heuristically to 20% of d with a minimum width of seven residues. From this alignment, the highest scoring local alignment with a maximum length of d is determined. This corresponds to choosing the highest scoring alignment of two consecutive repeat units. The first one of these repeat units is arbitrarily selected as the template repeat. If no alignment is found with a score of more than 10 and a length of more than ten residues, the algorithm exits the recursion (go to step 6).

2 Collect Repeat Units. For the collection of repeat units, the query sequence is aligned to the template repeat unit using wraparound dynamic programming (Fig. 3). This algorithm can deal well with short gaps between repeat units. However, when there are long intervening regions between repeats, the gap penalties grow too large and the alignment ends. Therefore, we iteratively collect repeat units by repeatedly performing the alignment; a similar procedure was used by Heringa and Argos.[8] The iteration is stopped, when the alignment score falls below a threshold of 10. During each iteration, the scores for all residue pairs in the trace matrix are updated by mapping the "column counts" of the newly found repeat units onto the template repeat unit. "Column counts" are the numbers of amino acids observed in a given column of the multiple alignment. As repeats are stacked, columns in the repeat model (profile) inherit counts from several positions in the multiple alignment based on the BLAST search with the query sequence against a large sequence database. Our repeat model is thereby updated on the fly, and generalized as more repeat units are found in the protein sequence.

3 Determining the Boundaries of Repeat Units. Because the boundaries of the template repeat unit need not coincide with the locations of the true repeat units, the correct boundaries have to be determined in a separate step. A multiple alignment of repeat units can be visualised as being wrapped around a cylinder, where the circumference of the cylinder corresponds to the length of one repeat unit (Fig. 4A). Setting boundaries for a repeat corresponds to vertically cutting the cylinder at the optimal position. If the alignment length is an integer multiple of the repeat unit length, the choice of where to cut is straightforward. However, due to noise and fragmentary repeats, this is often enough not the case. Therefore, a heuristic decision rule is applied: (1) If the overlap between the first and the last repeat (shaded region in Fig. 4A) is more than 25% of the repeat length, then it is assumed that there exists an incomplete repeat unit. The cut is then placed in the nonoverlapping region at the position that minimises the cost of gaps in the final alignment. (2) When the overlap between the first and last repeat is less than 25%, we regard the terminal residues of the repeat region as noise. The cut is then placed in the overlapping region. Again, the cost of gaps in the final alignment is minimized, but additionally a penalty is paid for residues extending into the noisy ends. Gaps between repeat units are not penalized in either case, so that large gaps tend to end up between repeat units and not in the middle of one repeat unit.

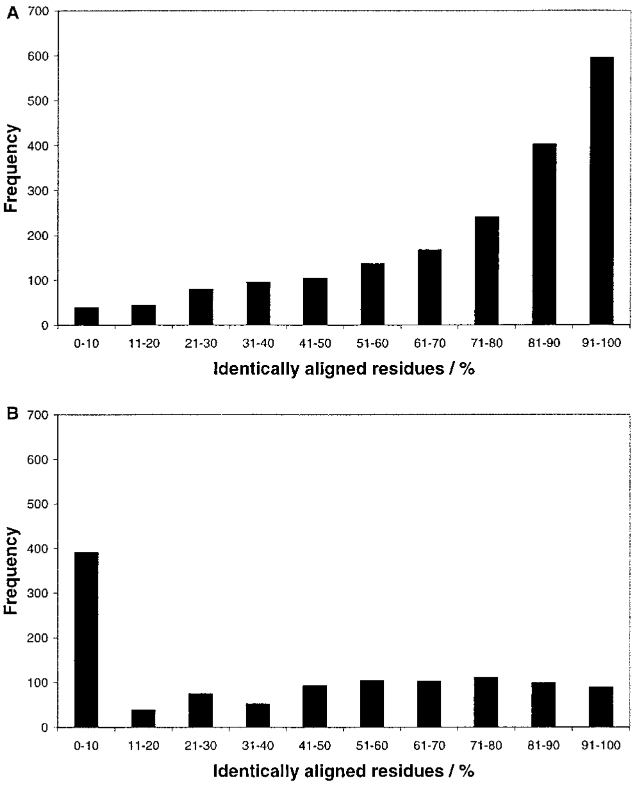


Fig. 6. Comparison with Pfam. 2,800 sequences with at least two matches of the same hidden Markov model were analysed using our algorithm. The alignments were then compared to the Pfam alignments, which had been classified according to their average sequence identity. (**A**) Percentage of residue pairs identically aligned. Only repeats with more than 30% average sequence identity are shown. (**B**) Same as A, but only distant repeats with less than 30% average sequence identity are shown.

4 Removal of Nonsignificant Repeat Units. Finally, spurious hits to the repeat model (profile) are eliminated by calculating a Z-score[8,9] for each repeat unit. A profile is derived from the multiple alignment of repeat units and the score of each unit is determined without taking into account end-gaps (i.e., only internal gaps are penalized). A background distribution of alignment scores is determined by aligning the profile against 100 shuffled versions of the complete protein sequence using the Smith-Waterman-algorithm. The Z-score of a repeat unit is the number of standard deviations of the repeat unit score above the mean. Repeat units with a Z-score below six are discarded. This threshold was chosen empirically. Columns at the ends of the multiple alignment are removed when they are occupied by only one repeat unit, that is the repeat unit can shrink with respect to d.

5 Recursion to Find Nonreducible Repeat Unit. The highest scoring diagonal selected in step 1.1 need not correspond to the shortest period of the repeat pattern. Therefore, we recursively repeat the algorithm to find the shortest nonreducible repeat unit (Fig. 5).

All residue pairs with one residue being part of the multiple alignment are mapped to the region around the template repeat (5.1). Pairs outside this region are then

TABLE II. Results Achieved for Some Well-Known Repeat Types Using Alignments From Pfam as a Reference

		Repeats above 30% identity[a]		Repeats below 30% identity[a]	
Repeat-type	Average percent identity of repeats in Pfam [%]	Found[b]/total	Identically aligned residue pairs [%][c]	Found[b]/total	Identically aligned residues pairs [%][c]
Immunoglobulin	36	98/99	62	28/51	31
Annexin	47	17/17	85	−/−	–
Lim	32	14/14	79	28/28	70
Fibronectin	36	55/55	76	42/52	35
EGF-like	39	55/55	75	19/22	46
Zinc	52	267/267	84	10/26	25
Armadillo	40	16/16	53	−/−	–
Leucine	39	57/60	55	10/15	40
Ankyrin	36	37/38	71	17/20	65
WD	38	171/172	66	19/26	30
EF-hands	34	90/90	53	47/71	33
Kringle	54	10/10	96	1/1	97
Sushi	41	41/41	78	9/11	51
Cadherin	33	30/30	44	6/6	45

[a] A dataset of repeats has been created from Pfam and been partitioned into two sets with high and low average percent identity, respectively. The table shows the results for some well-known repeat types.

[b] A repeat was classified as found, if RADAR had aligned at least one residue pair identically to Pfam.

[c] The percentage of identically aligned residue pairs is the fraction of residue pairs in the Pfam alignment that were recovered by RADAR.

removed. With this smaller trace matrix the algorithm recurses to find a new, shorter template repeat unit (step 1). Note that for the alignment of the query sequence to the template repeat the original trace matrix has to be used to collect all repeat instances.

Following each recursion, the current multiple alignment is compared to the previous one. If the number of repeat units has not increased or no multiple alignment has been obtained, the recursion finishes and the previous multiple alignment is returned as a result.

6 Iteration Over Repeat Types. The multiple alignment from the recursion step is output (6.1) and all residues involved are eliminated from the trace matrix (6.2). This purged trace matrix then replaces the original trace matrix from the initialization step (0.1). The algorithm is then restarted to detect further repeat types. When no further repeat types can be found (i.e., no multiple alignment is returned after the recursion), the algorithm ends.

Implementation of Alignment Algorithm

In our protocol, numerous alignments have to be calculated. These alignments are typically constrained. For example, each residue pair has to be part of at least one suboptimal alignment and often the trace is restricted to a region around a diagonal. Therefore, the calculation of the full dynamic programming matrix is usually not necessary. We implement this by keeping a sorted list of residue pairs that are putative candidates for the alignment trace to pass through. Instead of iterating through all columns and rows as in straightforward dynamic programming, we iterate just through this sorted list of residue pairs.

All alignments were calculated using affine gap penalties with a gap opening penalty of -4 and a gap elongation penalty of -0.4.

Implementation and Benchmarking

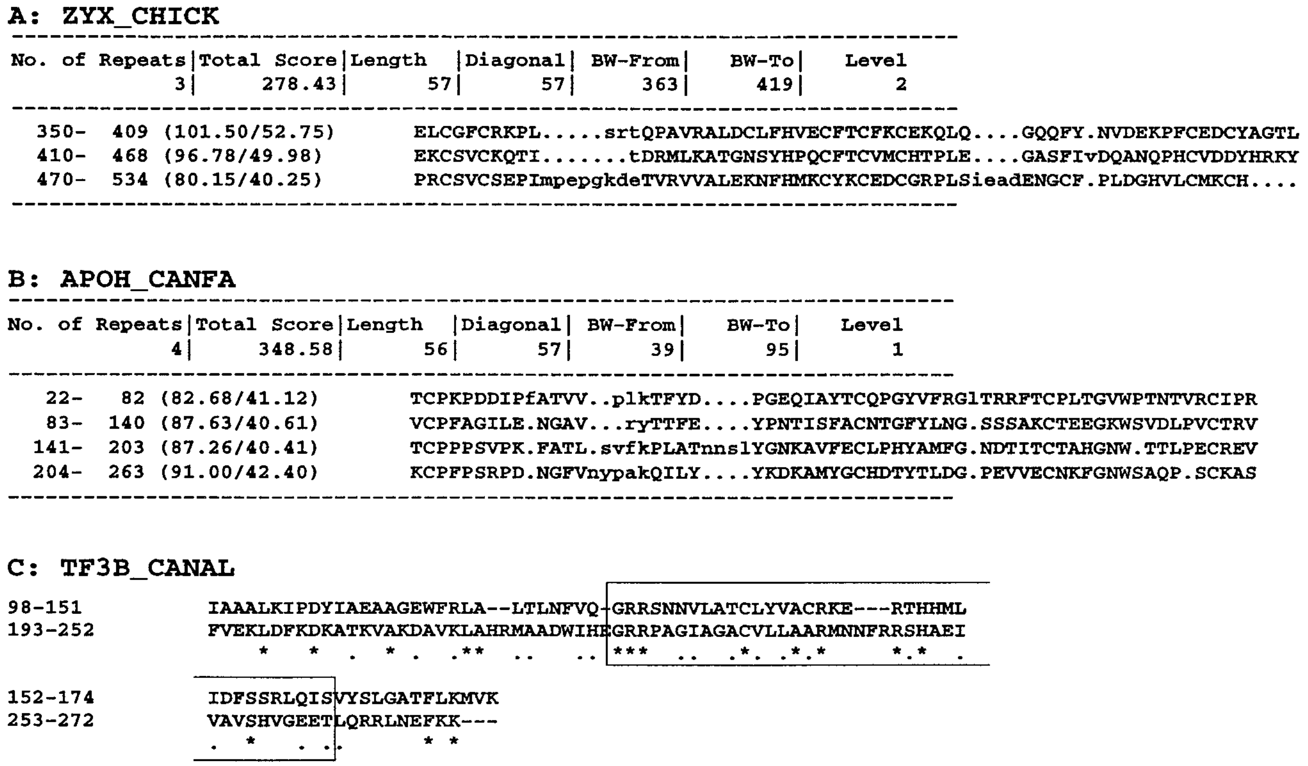
The program was written in Perl and C. The CPU time was recorded on a SGI Power Challenge using up to 8 Mips R10000 processors. The average execution time for all sequences in the Pfam-dataset was 12 sec. Except for one outlier (MUC2_HUMAN, tyrosine-rich region) the execution never took longer than 180s (e.g., fibronectin (FINC_BOVIN): 72s). Sourcecode and executables are available at ftp://ebi.ac.uk/pub/contrib/heger/radar/.

RESULTS

Assessment of Sensitivity and Accuracy by Comparison with Pfam

We assessed the sensitivity and accuracy of the algorithm. To this end, it was necessary to build a large test set of repeat containing protein sequences together with corresponding multiple alignments. We chose Pfam as a source, because it could provide both sequences and alignments. Pfam is a database of protein domain families. Starting from a manually curated multiple alignment of a domain family Pfam calculates a hidden Markov model. Additional members of each family are then added to the original multiple alignment by database search using the hidden Markov model as a query. We selected from Pfam all protein sequences that matched to the same hidden Markov model at least twice. The resulting dataset was filtered so that only sequences with less than 90% sequence identity remained. Alignment artefacts were eliminated by discarding alignments with an overlap of less than ten residues. We obtained 2800 sequences with 3057 repeat types from Pfam (i.e., several sequences contained more than one repeat type).

To better assess the performance of the algorithm, we classified repeats according to their divergence. As a measure, we chose the average percent identity of a repeat



```
A: ZYX_CHICK
---------------------------------------------------------------------------
No. of Repeats|Total Score|Length  |Diagonal| BW-From|   BW-To|   Level
             3|     278.43|      57|      57|     363|     419|       2
---------------------------------------------------------------------------
  350-  409 (101.50/52.75)       ELCGFCRKPL.....srtQPAVRALDCLFHVECFTCFKCEKQLQ....GQQFY.NVDEKPFCEDCYAGTL
  410-  468 (96.78/49.98)        EKCSVCKQTI.......tDRMLKATGNSYHPQCFTCVMCHTPLE....GASFIvDQANQPHCVDDYHRKY
  470-  534 (80.15/40.25)        PRCSVCSEPImpepgkdeTVRVVALEKNFHMKCYKCEDCGRPLSieadENGCF.PLDGHVLCMKCH....
---------------------------------------------------------------------------
```

```
B: APOH_CANFA
---------------------------------------------------------------------------
No. of Repeats|Total Score|Length  |Diagonal| BW-From|   BW-To|   Level
             4|     348.58|      56|      57|      39|      95|       1
---------------------------------------------------------------------------
   22-   82 (82.68/41.12)        TCPKPDDIPfATVV..plkTFYD....PGEQIAYTCQPGYVFRGlTRRFTCPLTGVWPTNTVRCIPR
   83-  140 (87.63/40.61)        VCPFAGILE.NGAV...ryTTFE....YPNTISFACNTGFYLNG.SSSAKCTEEGKWSVDLPVCTRV
  141-  203 (87.26/40.41)        TCPPPSVPK.FATL.svfkPLATnnslYGNKAVFECLPHYAMFG.NDTITCTAHGNW.TTLPECREV
  204-  263 (91.00/42.40)        KCPFPSRPD.NGFVnypakQILY....YKDKAMYGCHDTYTLDG.PEVVECNKFGNWSAQP.SCKAS
---------------------------------------------------------------------------
```

```
C: TF3B_CANAL

98-151            IAAALKIPDYIAEAAGEWFRLA--LTLNFVQ-GRRSNNVLATCLYVACRKE---RTHHML
193-252           FVEKLDFKDKATKVAKDAVKLAHRMAADWIHEGRRPAGIAGACVLLAARMNNFRRSHAEI
                      *   *  .  * .  .**  ..   .. ***  ..  .*. .*.*     *.*  .

152-174           IDFSSRLQISVYSLGATFLKMVK
253-272           VAVSHVGEETLQRRLNEFKK---
                  .  *   . ..      * *
```

Fig. 7. Alignment of repeats generated by RADAR. (**A**) Sample output for zyxin from chicken (ZYX_CHICK). Zyxin contains three LIM domains. Notation: BW, best window; Level, depth of recursion; the alignment row reports first and last residue and, in parentheses, the alignment score and *Z*-score of the repeat. Lower case characters are used for residues that are not aligned to the template repeat unit. (**B**) Output for apolipoprotein H from dog (APOH_CANFA). This protein has four Sushi type repeats. (**C**) The transcription factor TFIIIB (TF3B_CANAL) from *Candida albicans* contains a diverged repeat annotated in SWISS-PROT. The annotated regions are shown here aligned by ClustalX, Version 1.8.[29] Our algorithm detects part of the repeat (boxed region). This repeat is not annotated by DOMO and PRODOM.

and partitioned the dataset into two sets using a threshold of 30%. The first set contained 1,903 repeats with more than 30% average sequence identity. The other set contained the remaining 1,154 more diverged repeats with less than 30% average sequence identity. The test sets and results are available for inspection at ftp://ebi.ac.uk/pub/contrib/heger/radar/.

For repeats with more than 30% average sequence identity our algorithm aligned 73% of the residue pairs correctly (Fig. 6A). The distribution is skewed, the median being 81.6%, so most alignments agreed remarkably well with Pfam. Only 20 repeats were not found at all. In most cases, these repeats were small, consisted of just two repeat units, and were separated by comparatively large gaps. Taking into account all repeats, on average, 60% of the residues were correctly aligned by our algorithm. Even in cases of low sequence similarity, reasonable to good alignments were frequently obtained (Fig. 6B). In this set, on average, 40% of the residue pairs were aligned correctly.

The correctness of the repeat boundaries was checked by calculating the relative average shift between the repeats detected by our algorithm and the reference alignments from Pfam. The shift of two overlapping repeat units is the minimum offset to the right or to the left that separates the two starting positions of the repeats. If one repeat unit was shorter and fully contained inside the other one, the shift was defined to be zero. The offsets were averaged and divided by the length of the reference repeat from Pfam to give the relative average shift per repeat. Using this measure, we found that 2,717 repeats had a relative shift of 25% or less. Three hundred and forty repeats were shifted by more than one-quarter of the repeat length. Inspection showed that those were mostly short repeats. Strict tandem repeats proved especially difficult. The average relative shift of the 93 strict tandem repeats in the Pfam set was 15% whereas that for interspersed repeats was just 7.6%.

Recovery of Well-Known Repeats

The results for some typical repeat types are listed in Table II and sample output from the program can be seen in Figure 7A and B. In cases of moderate to high sequence similarity, the algorithm performed very well, only missing a few repeats. Even in instances of low sequence similarity, the majority of repeats were still detected, albeit the alignment quality dropped as compared to Pfam.

The performance of the algorithm is best illustrated by examples. The following proteins were chosen because they are well known in the literature and contain precise annotation of repeat units. A classical example of repeated motifs are zinc-fingers. Of all zinc-finger

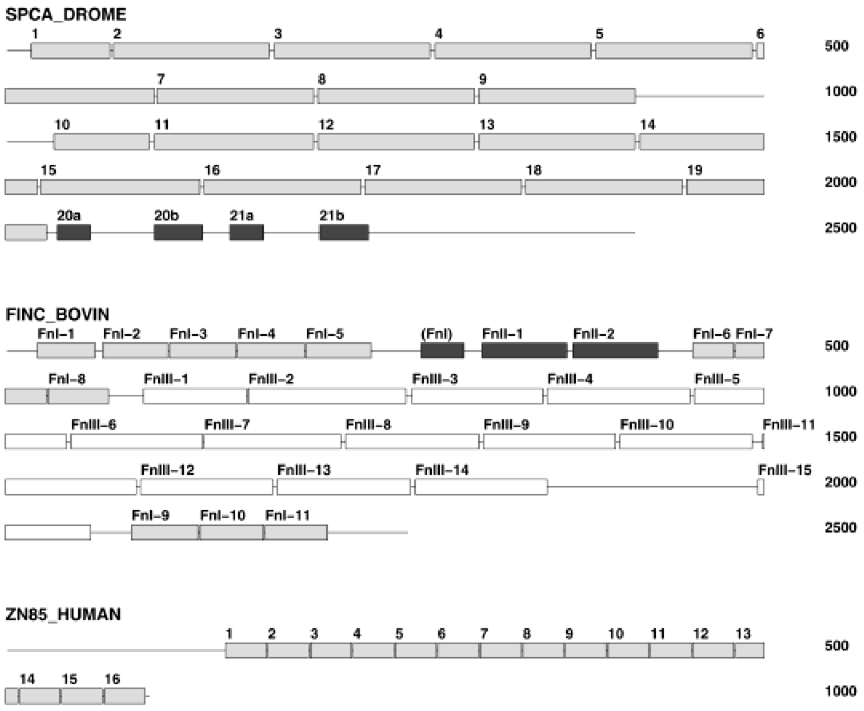


Fig. 8. Examples of the algorithm's performance. (**A**) All 21 repeats in spectrin (SPCA_DROME) were detected, the last two, less similar repeats were split in half. (**B**) Fibronectin contains three different types of fibronectin-like repeats (Fn-1 to Fn-3). Except for one Fn-1 domain, which was erroneously assigned to Fn-2 type repeats, all repeat units were detected and aligned correctly. (**C**) The human zinc-finger protein 85 contains 16 C_2-H_2 zinc-finger repeats. Two of those repeats contain a mutated cysteine and are therefore annotated in SWISS-PROT as degenerate. All of the repeats were found, while Pfam only finds 15.

repeats with more than 30% sequence identity, we missed none. Occasionally, a few repeat units were not detected; however, in other cases, the algorithm could find more repeats than were annotated by Pfam. For example, Pfam detects 15 zinc-finger motifs in human zinc-finger protein 85 (ZN85_HUMAN), whereas we detect 16 (Fig. 8). The additional repeat is annotated in SWISS-PROT as a degenerate repeat. The repeat boundaries as determined by RADAR and annotated by SWISS-PROT coincide (Table III).

The alpha-subunit of spectrin (SPCA_DROME) is composed of a large number of repeat units. Overall there are 21 repeat units which are separated by a unique SH3-domain between repeat units 9 and 10. Repeat units 20 and 21 share less similarity to the other ones. Two EF hands are present at the C-terminus as well. Our algorithm successfully aligned the 19 similar repeat units and split the remaining two into two fragments (Fig. 8). The repeat boundaries are given in Table III. As compared to the SWISS-PROT annotation, we observe a shift of 50 residues, which corresponds to one half of the length of one repeat unit. This is due to some unaligned segments of repeat units next to the SH3 domain and in repeat unit 19. The repetition of the EF hand was not detected, the two sequence fragments having only a sequence identity of 20%.

Fibronectin (FINC_BOVIN) contains three different types of fibronectin-type domains, FnI to FnIII, that are

TABLE III. Boundaries of Repeat Units of Samples in Figure 5

SWISS-PROT identifier	Repeat type	Units
SPCA_DROME	Spectrin	(17–69), (71–174), (177–280), (283–386), (389–492), (495–598), (600–703), (706–809), (812–915), (1032–1095), (1098–1203), (1206–1309), (1312–1415), (1418–1521), (1523–1628), (1631–1734), (1737–1840), (1843–1946), (1949–2027), (2034–2056; 2098–2130)[a], (2148–2170; 2207–2239)[a]
FINC_BOVIN	FibronectinType I	(21–59), (64–107), (108–152), (153–197), (198–241), (453–480), (481–527), (528–568), (2083–2127), (2128–2170), (2171–2212)
	FibronectinType II	(274–302),[b] (314–370), (374–430)
	FibronectinTypeIII	(591–659), (660–764), (768–854), (857–951), (954–1040), (1043–1130), (1131–1221), (1224–1312), (1315–1402), (1405–1492), (1499–1586), (1589–1676), (1679–1767), (1770–1857), (1996–2056)
ZN85_HUMAN	Zinc-finger	(145–172), (173–200), (201–228), (229–256), (257–284), (285–312), (313–340), (341–368), (369–396), (397–424), (425–452), (453–480), (481–508), (509–536), (537–564), (565–592)

[a]These repeat units were split into two fragments.
[b]Actually a Fibronectin type I repeat unit that has been wrongly aligned to Fibronectin type II repeat.

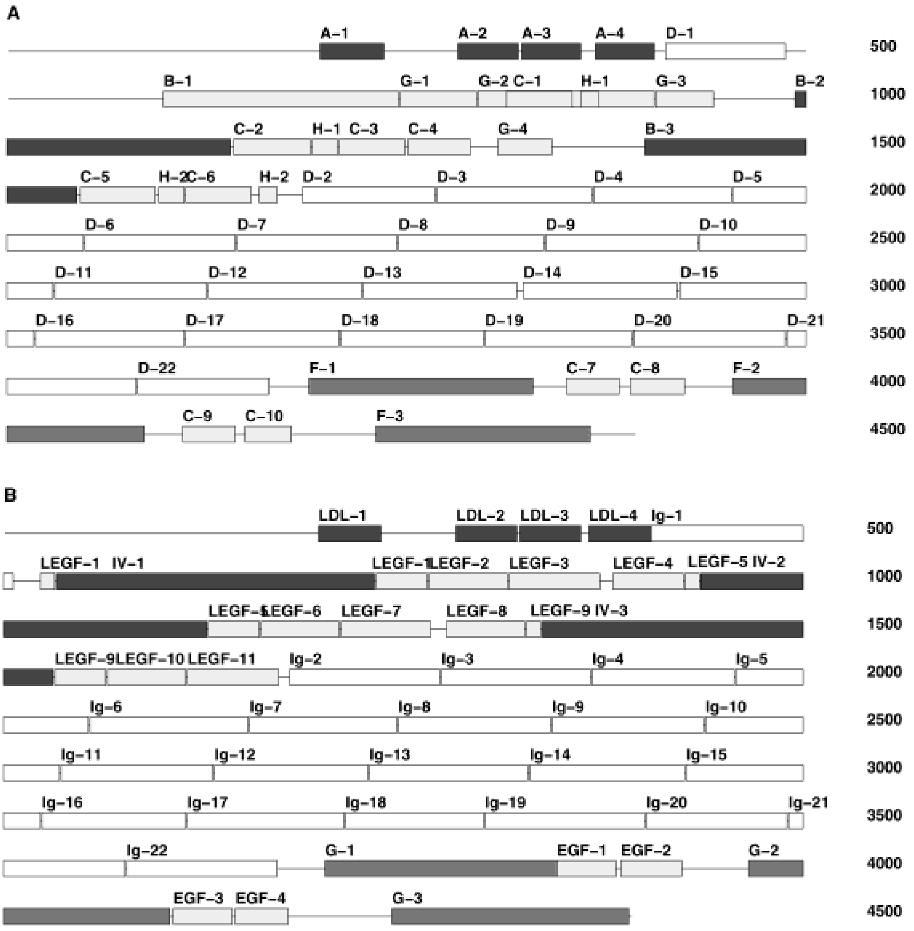


Fig. 9. Comparing our algorithm with SWISS-PROT annotation for the human basement membrane heparan sulfate proteoglycan core protein (PGBM_HUMAN). (**A**) The repeat structure as calculated by RADAR. All 22 Ig-like domains, all three laminin g-like and laminin domain IV domains as well as all four LDL-receptor domains were correctly detected. Ten EGF-like domains were aligned, the EGF-like domains and laminin-EGF-like domains being assigned to the same repeat type (C1–C10). The remaining EGF-like domains were aligned as fragments in smaller alignments. (**B**) Repeats as annotated by SWISS-PROT (Ig: Ig-like C2 type domain; LEGF: laminin-like EGF-domain; EGF: EGF-like domain; LDL: LDL-receptor domain; IV: laminin domain IV; G: laminin G-like domain). Note that EGF-like domains 1, 5, and 9 are split into two segments.

arranged in a complex pattern. Except for one instance of the FnI domain, which was partly aligned to the FnII domains, all repeats were found and aligned correctly (Fig. 8). The repeat boundaries compared well to the ones given in SWISS-PROT: FnI and FNII were shifted by roughly one and five residues, respectively. The shift of FnIII was slightly larger with 11 residues (Table III).

Repeats with remote similarities can be detected as well. The transcription factor TFIIIB from *Candida albicans* (TF3B_CANAL) contains two repeats with a length of about 60 residues and less than 20% sequence identity. Although these repeats went undetected by DOMO[3] and PRODOM,[4] our algorithm successfully aligned 34 residues of the repeat (Fig. 7C).

The human basement membrane heparan sulfate proteoglycan core protein (PGBM_HUMAN) contains six different kinds of repeats in a highly interleaved structure (Fig. 9). We can retrieve most of the repeats correctly and can delineate the domain structure of this protein nicely. However, the conserved cysteines in the EGF-like domains are misaligned in some instances. As this example shows, the algorithm was able to delineate the repeat structure even of complex proteins.

Currently, an ultimate challenge in terms of size and number of domains is the human cardiac muscle protein titin with a length of 26926 residues. Titin has a modular architecture containing a total of 244 copies of 100-residue repeats. One hundred and twelve of these repeats are of the immunoglobulin type and 132 of the fibronectin III type.[6] Our algorithm detects all fibronectin domains correctly but misses a few immunoglobulin domains (Fig. 10).

Mining SWISS-PROT for Novel Repeats

To find unknown repeats and domains, we analysed a representative subset of SWISS-PROT. The set contained only sequences from SWISS-PROT that shared not more than 90% sequence identity among themselves.[20] The 44,217 sequences in the set were analysed using our algorithm. We detected repeats in 12,485 sequences (28%). For 6,637 sequences, SWISS-PROT did not list the key

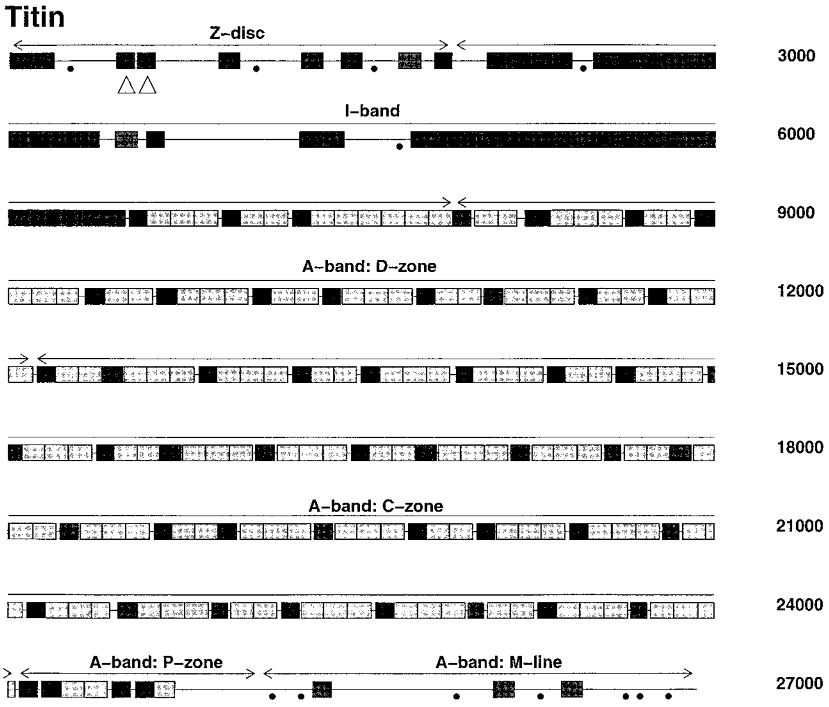


Fig. 10. Automatic analysis of the giant human cardiac muscle protein titin (PIR-accession number: I38344). Titin has a total of 112 immunoglobulin and 132 fibronectin type repeats. All fibronectin domains were correctly detected and aligned (light boxes). The majority of immunoglobulin domains were recognized albeit distributed to three repeat types (dark boxes). The missed instances of immunoglobulin domains are marked by dots. A further repeat type unrelated to fibronectin or immunoglobulin domains in the Z-disc region was detected as well (arrows). The calculation took 3500 s on a SGI Origin2000. A *Z*-score cutoff of 20 was used and the remapping step of dots was skipped (step 5.2, Fig. 9).

word "repeat" in the annotational part. This corresponds well with the 6,548 sequences reported previously.[26] Filtering out sequences which matched to the same Pfam family two or more times reduced the result set further to 4,470 sequences.

This dataset was still much too large to be checked manually. Most of the putative repeats were fairly small with a length of around 20 residues and occurred just twice per sequence (Fig. 11). It is difficult to assess the significance for those short motifs. We therefore filtered the dataset further using an alignment score cutoff of 60. Following this step, 2,920 sequences remained with repeats not covered by SWISS-PROT or Pfam. In the following, we present some interesting examples of repeats found by the algorithm. For every sequence, we verified that it did not contain a match in SMART.[27]

In the exo-β 1,3-glucanase of *Cochliobolus carbonum* (EXG1_COCCA), we "rediscovered" a twofold repeat of 21 residues (Fig. 12A). This repeat was not annotated in SWISS-PROT, DOMO, or PRODOM. However, the repeat has been described in the literature,[28] and been implicated in the interaction with polysaccharides.

Proteins annotated as "putative" are a rich source of novel repeats that are not covered by Pfam, SWISS-PROT, or SMART. For example, the putative protein YQ12_CAEEL from *Caenerorhabditis elegans* contained a duplication of 270 residues length (Fig. 12B) in a sequence with a length of 1,551 amino acids. Analysis with PSI-BLAST did not reveal any sequence neighbours in the repeat region, although in other regions, the protein shared domains with several DNA-binding proteins.

The 997 residues long protein YPX2_CAEEL contains a five-fold repeat with a repeat unit length of 180 residues. Three of the five repeat units contain an insertion of 23 residues in length. This insertion was identified by the

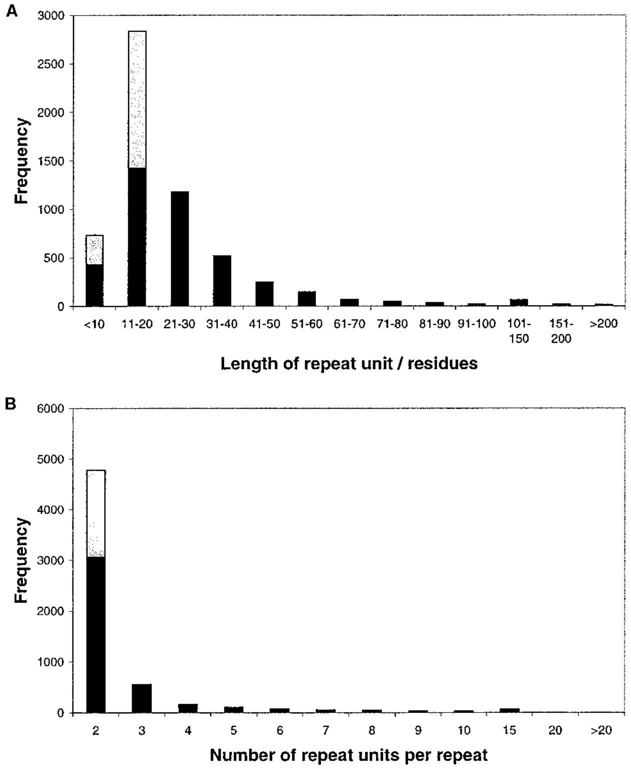


Fig. 11. Results of scanning a non-redundant subset of SWISS-PROT. The result set was filtered to contain only repeats which were not annotated by SWISS-PROT or Pfam. (**A**) Distribution of repeat unit lengths in sequences not annotated as containing repeats in a representative subset of SWISS-PROT. (**B**) Distribution of the number of occurrences of repeat units per repeat. The shaded regions corresponds to the subset of repeats with a total score of less than 60. As can be seen those repeats are mostly short and occur twice per sequence.

```
A: EXG1_COCCA

repeat_1      77 GAKGDGVTDDSDAFNRAISDG   97
repeat_2     430 GATGDGVTDDTRAVQAAVTQA   450
                 ** *******  *   *


B: YQ12_CAEEL
repeat_1     201 DVMREMEKSMNIFTKLFRGSQiRNKNDLPSGRHLVNYRQNHAFDFAAMSHIWKENKMENLPKLPHRANLV   270
repeat_2     496 DLYRQHEMQTNVFTKFFDIGK-SEKRKFGPGKSISNCRHSDAIDFAAMSYIWKNIRQQHLPKLPKLPNMV   564
                 *  *  *   * *** *       *     *    * *   * ****** ***      *****   * *

repeat_1     271 MTLPSSRVRNSKICAFYRVLEKKEDKYILVACHVNIHGFHRGDLRFVEVNNRTVLKGFERSSVIGHLFLG   340
repeat_2     565 LPLPTSMPLDSEICAFYRVLVENKSKYILIACHMNVHGYHRGDLRFLEINESTKLYGFEGRSVIGQLFLG   634
                   ** *    * ********     **** *** * ** *****  * *   * * ***  **** ****

repeat_1     341 DILTVTELTRCNETVSNVASSVEDA--SPDAPCQWMASKVVLFPRHEPVNVQFKFTCNGMASVVSTNEPM   408
repeat_2     635 DIVAVTELARCNGHGSDAFKIPFDVsmSSQNTCTWMASKITLPSRPPPASVLFSFMKNRMAVVKGCDEPM   704
                 **  **** ***   *       *   *     * *****  *  *  *  * * *  * ** *    ***

repeat_1     409 SVILNNLTEAKTNVEYTGIAFKSASYTVH-TEDYTKKWYERIQEEISNIMIYPKALSTIYQYEE         471
repeat_2     705 NVEVKSIQNVEPDLIYKGTAFKPEKPELNfTEGFIKKKRHQIGSITLRIASYPNSFGTIYNFQE         768
                  *             * * ***        **   **     *      *  **    ***   *


C: YPX2_CAEEL
repeat_1     130 GL----PEPSTTTLS-PSQHV-FQTNPIwTILKNSKN--GTQ----GVGAVNLP---------wTPPVSA   178
repeat_2     306 GLSTATPLSSLSTASTPTKTVGFYTRPV-QIVSNPSG--GLQ----GYWIEYFD----------TSSFPR   358
repeat_3     468 --------SDEFETTTVAGRYGFKVDTV-KIVFPNTG--SVEyednGYWVNT----------------PK   510
repeat_4     617 EPTKPSQSTSPSTPSTPPNRIGFYVERV-FLVGPKSGiaGQQ----GYTLKIVS----------NLDPDT   671
repeat_5     816 ----------------------FRLSPV-YIIYPTEY--YLQ----N-------qqegshvatrLGFIQH   849
                                       *

repeat_1     179 SPIEP----TNVCLLNGNSSVLSRCGE-IDRMLQFFD--DTHVYIgfkdSIQDASNTYIYSY--------   233
repeat_2     359 NPLYS----TNVCLFLANSSVITRCGH-ARNLYQYHDTVDNFVFI----SLYTFGRTVESLP--------   411
repeat_3     511 RPLVTfvrdTNVCVFREDLETVEKCGA-TKPLYLYINKNNGGYFM----GIQSFGRNLTSTE--------   567
repeat_4     672 EVRGE----TNVCVITGNFLEIDTCADfTQTFNSYFDANDNYYYV----ARDT-GRPVTKAGiigttfss   732
repeat_5     850 DGAGV----TNACVFLSDPSIVALCGT-TAPLYLYFDRVRVAYFV----GTHSAGRNVSIER--------   902
                          ** *           *

repeat_1     234 -------------------IGyaftssentcgiplepirelykygvgFT--S--------VAGEKYTELI   274
repeat_2     412 -------------------LG------------------------vaFE--TsektyynvVAGDDYQNLL   436
repeat_3     568 -------------------L-------------------------------ydV--------HAGEDYEEIT   581
repeat_4     733 qenlcglnvvpirelykegVG--------------------------YN--A--------VAGDDYQSLL   766
repeat_5     903 -------------------AAtvfesagntcgftllplrefykdgtgYN--L--------HAGDDYELLV   943
                                                                              **  *

repeat_1     275 SNGYSLTRKVLGYTIDCASATN----GELVVTIPD                                      305
repeat_2     437 DQGYNITGNIMGYTIDCKDAND----EFVYGYLPN                                      467
repeat_3     582 AAGYNQTGNIMGYTVDCRDAIGgtvyGHLPNTIPE                                      616
repeat_4     767 DEGYTLNGRIMGYTVDCNDAEN----DFVYGFLP-                                      796
repeat_5     944 NGGFVATGNIMGYAPDCRD--N----G---FHEPD                                      969
                   *        **  **                *

repeat_6     236 YAFTSSENTCGIPLEPIRELYKYG   259
repeat_7     728 TTFSSQENLCGLNVVPIRELYKEG   751
repeat_8     905 TVFESAGNTCGFTLLPLREFYKDG   928
                   * *  * **    * ** ** *
```

Fig. 12. Discovery of unannotated repeats in SWISS-PROT. (**A**) Alignment of a rediscovered repeat in exo-β 1,3-glucanase of *Cochliobolus carbonum* (EXG1_COCCA). (**B**) Alignment of repeats in hypothetical proteins YG12_CAEEL, and (**C**) YPX2_CAEEl of *Caenerorhabditis elegans.* The boxed region corresponds to the insert, that is aligned separately by RADAR. Lowercase characters are used for unaligned residues.

algorithm as such and therefore not aligned. However, in the next iteration step, the three fragments were detected and aligned correctly (Fig. 12c).

In the capsule polysaccharide export protein KPSC from *Escherichia coli* (KSC5_ECOLI), we detected an internal duplication of 300 residues. Analysis with PSI-BLAST revealed that single units of this repeat occur elsewhere (Fig. 13). Weak sequence similarities were found to enzymes involved in teichoic acid biosynthesis (data not shown).

DISCUSSION

In this paper, we introduced an algorithm to detect repeats in protein sequences. The problem is nontrivial, because the repeat structure of a protein can be very complex. Several ideas used in this work have been implemented before. For example, Heringa and Argos[8] use iterative searches with a profile constructed from a few detected repeats to collect more distant members, and Marcotte et al.[26] analyse the trace matrix to estimate the repeat length to scan quickly a large database for the presence of repeats. However, the top-down approach for arriving at the correct length of one repeat unit seems novel. The repeat units are successively broken down into smaller pieces by segmenting the trace matrix until the smallest, nonreducible repeat length is found. RADAR is also sufficiently fast to scan databases

```
CLUSTAL W (1.8) multiple sequence alignment


KSC5_ECOLI_1        WRIPHLEKFLAQPCQKLSLLR-------PVPQ----EVDAIAVWGHRPSAAKPVAIAKAA
KSC5_ECOLI_2        WKSAILKPFLQTATNRLSFSR--------RCT----AASACVVWGVKGEQQWRAEAQRKS
LIPA_NEIME_1        KMIFSTFGLFKQRLLLTKFLD-------KIST----DKTALFWGKKHPKRRKLAHILRLS
LIPA_NEIME_2        WKRAVAKPFFNVPSCRLKFISSTQKLAGVKLS----DDARILAWGN-GKEAIVRFAEQHH
LPSZ_RHIME          WKRDVLARYFS--DFRVAYVRTNTSWTKVQTSFCQFTPQAFVFWGMTEIRAAKNYAIKSS
                             :                                               :

KSC5_ECOLI_1        GKPVIRLEDGFVRSLDLGVNGEPPLSLVVDDYCIYYDASKPSALEKLVQDKAGN-TAL-I
KSC5_ECOLI_2        -LPLWRMEDGFLRSSGLGSDLLPPLSLVLDKRGIYYDATRPSDLEVLLNHSQLTLAQ--Q
LIPA_NEIME_1        ---LLNLEDGFLRSLGLGVAGYPPYS-VYDDIGIYYDTTRPSRLEQLILAADTMPSET-L
LIPA_NEIME_2        -IPLLRMEDGFIRSVGLGSNLVPPLSLVTDDMGIYFNAEAPSRLEHILQNQNFDDQD--F
LPSZ_RHIME          -IPLWRVEDGFLRSVGLGAQHVLPLSLAVDTTGIYFDPSRPSTLETLISEIGVTENATLI
                       : .:****:** .**     * * . *   **::.  ** ** ::

KSC5_ECOLI_1        SQAREAMHTIVTGDLSKYNLAPAFVADESERS----DIVLVVDQTFNDMSVTYGNAGPHE
KSC5_ECOLI_2        MRAEKLRQRLVESKLSKYNLG-ADFSLPAE-AKDK-KIILVPGQVEDDASIKTGTVSIKS
LIPA_NEIME_1        AQARQAMDFILQHHLSKYNHAPELSDDHPLRSPSKPETVLIIDQ-FGDMAIQYGGADAST
LIPA_NEIME_2        QTALKLQKMLTENHISKYNVGSSDFTAP---STDK-TVILVPGQVEDDASIRYGSPQIYR
LPSZ_RHIME          ERARRCMSMISAFGLSKYNVG-QDVPLKRLPPSDR-RRVLVVGQVEDDASIVMGCAARYT
                      * .    :    :**** .          .      :*: .*  .* ::  *

KSC5_ECOLI_1        FAAMLEAAMAENPQAEIWVKVHPDVLEGKKTGYFAALRATQRVRLIAENVSPQSLLRHVS
KSC5_ECOLI_2        NLELLRTVRERNPHAYIIYKPHPDVLVGNRKGNIPTELIAELADYQALDADIIQCIQRAD
LIPA_NEIME_1        FELMFQTALNENPQADIWVKTHPDVLCGKKQGYLTQLAQQHRVNLLAEDINPISLLQNID
LIPA_NEIME_2        NLDLLRTVRERNPNAYIIYKPHPDVVSGNRIGHISPEDAARYADQTAEQADILTCLQYAD
LPSZ_RHIME          NNDIVRITQKENPEAEVIYRPHPDVLGGHRKEFSNPRDVANICTILSGDYDLGSLLDSVD
                       :.. .  .**.* :  : ****: *::          .     : : .    :   .

KSC5_ECOLI_1        RVYVVTSQYGFEALLAGKPVTCFGQPWYAGWGLTDDRHP--QSALLSARRGSATLEELFA
KSC5_ECOLI_2        EVHTMTSLSGFEALLHGKQVHCYGLPFYAGWGLTVDEHH------CPRREQKLTIADLIY
LIPA_NEIME_1        KVYCVTSQMGFEALLCGKPLTTFGLPWYAGWGVSDDRHPKIDSLVQTQRRAPRNLLQLFA
LIPA_NEIME_2        EIHTMTSLTGFEALLRGKKVSCYGLPFYAGWGLTQDLLP------IPRRSRRLELWQLIA
LPSZ_RHIME          HVYTITSLLGFEALIRRKKVTVFGAPFYSGWGLTDDRQP------TPRRTRKPSLDELFA
                    .:: :**  *****:  * :  :* *:*:***:: *          . *     : :*:

KSC5_ECOLI_1        AAYLRYCRYIDP
KSC5_ECOLI_2        QTLIVYPTYIHP
LIPA_NEIME_1        ARYLQYSRYLNP
LIPA_NEIME_2        GTLIHYPDYIHP
LPSZ_RHIME          AAYILYPRYCVG
                       : *  *
```

Fig. 13. Multiple alignment of family members for the capsule polysaccharide export protein KPSC from *Escherichia coli* (KSC5_ECOLI). A duplicated domain was detected in KSC5_ECOLI. Analysis with PSI-BLAST showed that this repeat is present in other sequences as well, but in different copy number. According to SWISS-PROT annotation, all proteins are involved in lipopolysaccharide synthesis or modification. The multiple alignment was created with ClustalX.[29] Note that the translation start for LIPA_NEIME has been corrected.

while producing an explicit multiple alignment of each repeat type found.

In comparison with Pfam, one the most sensitive sequence alignment methods available, we were able to retrieve most of the repeats and to produce good alignments. It should be noted that alignments in Pfam are generated by an automated method as well, but a different gap model is applied. Therefore, complete identity between alignments is not to be expected, especially when the sequence similarity is low.

Scanning through SWISS-PROT, we observed many sequences with hitherto unannotated repeats. Our results compared well to previous surveys.[26] Our algorithm is sensitive and fast enough to be applied to large numbers of sequences and explicitly generates multiple alignments of all repeat units detected in the protein sequence. No prior assumptions about the expected number or length of repeat units are made, and there is no limit imposed on the maximum length of the sequence.

We will use repeat detection as a step toward completing the catalogue of protein domains in the context of global clustering of protein sequences. Describing their modular architecture is important for accurate functional annotation of proteins. Domain borders can be estimated more precisely from internally repeated domains than from a multiple alignment of single occurrences. The explicit multiple alignments generated by RADAR are a useful starting point for profile searches against the complete database to collect all members of the domain family.

Limitations

Despite the good overall agreement to manually checked repeat definitions, our automatic procedure has limitations. We observed various types of errors:

(1) The algorithm is limited by the sensitivity and accuracy achievable by sequence alignments. As with sequence searches in general, there is no guarantee of finding repeats with low sequence similarity among the repeat units. When distant repeats are detected, the placement of gaps in these cases can be ambiguous. Diverse sequences with conserved cysteines (e.g., EGF-like domains) proved especially difficult to align. At low

levels of sequence similarity, it is very difficult to distinguish between significant and nonsignificant matches. By applying a *Z*-score cutoff, we tried to solve this problem using statistics. Although this eliminated obvious mismatches, there remain ambiguous cases.

(2) In some cases, the lengths of repeat units were incorrect. Again, repeats with a conserved cysteine pattern caused problems. In such cases, alignments with twice the number of repeats of half the length can be equally likely as full length repeats. The alignment score tends to be higher for longer repeats due to the longer alignment traces, so it cannot be used as a decision criterion. We decided to discard shorter repeats, when the number of repeats didn't increase substantially compared to the previous recursion.

(3) The task of locating the correct repeat boundaries in those cases where the local alignment length is far from an integer number of repeat unit length is very hard to tackle and essentially unsolved. For example, we have insufficient information in a single sequence to distinguish the cases where there is a large insertion at the middle of a repeart unit or where there is one complete repeat unit.

Most of these limitations can be alleviated by coupling repeat detection to database searches. So far, we use family information only for increasing the sensitivity of the profiles used to collect repeat instances in the query sequence. In essence, the method presented here is still a single sequence method. If information from additional sequences with the same repeat type was incorporated, we would achieve consistency in determining the repeat unit length and boundaries for all members of the family in one go.

The principle of the algorithm might be applicable to nucleic acid sequences as well, if stringent matching criteria were used to limit the density of dots in the trace matrix. We have demonstrated that RADAR can analyse sequences of tens of thousands of residues (titin example), but computer memory limitations could be prohibitive to the analysis of genomic DNA sequences that stretch for megabases.

REFERENCES

1. Wetlaufer DB. Nucleation, rapid folding, and globular intrachain regions in proteins. Proc Natl Acad Sci 1973;70:697–701.
2. Doolittle RF, Bork P. Evolutionary mobile modules in proteins. Sci Am 1993;Oct:34–40.
3. Gracy J, Argos P. Automated protein database classification: I. Integration of compositional similarity search, local similarity search and multiple sequence alignment. II. Delineation of domain boundaries from sequence similarities. Bioinformatics 1998; 14:164–187.
4. Corpet F, Servant F, Gouzy J, Kahn D. ProDom and ProDom-CG: tools for protein domain analysis and whole genome comparisons. Nucl Acid Res 2000;28:267–269.
5. Heringa J. Detection of internal repeats: how common are they? Curr Opin Struct Biol 1998;8:338–345.
6. Labeit S, Kolmer B. Titins: giant proteins in charge of muscle ultrastructure and elasticity. Science 1995;270:293–296.
7. McLachlan AD. Tests for comparing related amino-acid sequences. Cytochromes c and cytochrome c_{551}. J Mol Biol 1971;61: 409–424.
8. Heringa J, Argos P. A method to recognize distant repeats in protein sequences. Proteins 1993;17:391–411.
9. McLachlan AD. Analysis in gene duplication repeats in the myosin rod. J Mol Biol 1983;169:15–30.
10. Lawrence CE, Altschul SF, Boguski MS, Liu JS, Neuwald AF, Wootton JC. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. Science 1993;262:208–214.
11. Benson G. Sequence alignment with tandem duplication. J Comput Biol 1997;4:351–367.
12. Coward E, Drabløs F. Detecting periodic patterns in biological sequences. Bioinformatics 1998;14:498–507.
13. Pellegrini M, Marcotte EM, Yeates TO. A fast algorithm for genome-wide analysis of proteins with repeated sequences. Proteins Struct Funct Genet 1999;35:440–446.
14. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:197.
15. Waterman MS, Eggert M. A new algorithm for best subsequence alignments with applications to tRNA-rRNA comparisons. J Mol Biol 1987;197:723–728.
16. Miller W, Myers E. Approximate matching of regular expressions. Bull Math Biol 1989;51:5–37.
17. Pearson WR, Lipman DJ. Improved tools for biological sequence analysis. Proc Natl Acad Sci 1998;85:2444–2448.
18. Bateman A, Birney E, Durbin R, Eddy SR, Howe KL, Sonnhammer EL. The Pfam Protein Families Database. Nucl Acid Res 2000;28:263–266.
19. Bairoch A, Apweiler R. The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999. Nucl Acid Res 1999;27:49–54.
20. Holm L, Sander C. Removing near-neighbour redundancy from large protein sequence collections. Bioinformatics 1998;14:423–429.
21. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucl Acid Res 1997;25:3389–3402.
22. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci 1992;89:10915–10919.
23. Dayhoff MO, Schwartz RM, Orcutt BC. A model of evolutionary change in proteins. In: Dayhoff MO, editor. Atlas of protein sequence and structure, Vol. 5, Suppl. 3. Washington, DC: National Biomedical Research Foundation; 1978. p 345–352.
24. Gribskov M, McLachlan AD, Eisenberg D. Profile analysis: detection of distantly related proteins. Proc Natl Acad Sci USA 1987;84:4355–4358.
25. Sjölander K, Karplus K, Brown M, Hughey R, Krogh A, Mian IS, Haussler D. Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology. Comput Appl Biosci 1996;12:327–345.
26. Marcotte EM, Pellegrini M, Yeates TO, Eisenberg D. A census of protein repeats. J Mol Biol 1998;293:151–160.
27. Schultz J, Copley RR, Doerks T, Ponting CP, Bork P. SMART: a Web-based tool for the study of genetically mobile domains. Nucl Acid Res 2000;28:231–234.
28. Nikolskaya AN, Pitkin JW, Schaeffer HJ, Ahn JH, Walton JD. EXG1p, a novel exo-beta1,3-glucanase from the fungus Cochliobolus carbonum, contains a repeated motif present in other proteins that interact with polysaccharides. Biochim Biophys Acta 1998; 1425:632–636.
29. Jeanmougin F, Thompson JD, Gouy M, Higgins DG, Gibson TJ. Multiple sequence alignment with Clustal X. Trends Biochem Sci 1998;23:403–405.