4678–4685 *Nucleic Acids Research, 2007, Vol. 35, No. 14* *Published online 27 June 2007*
doi:10.1093/nar/gkm414

The identification of complete domains within protein sequences using accurate *E*-values for semi-global alignment

Maricel G. Kann, Sergey L. Sheetlin, Yonil Park, Stephen H. Bryant and John L. Spouge*

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Department of Health and Human Services, Bethesda, MD 20894, USA

Received December 14, 2006; Revised May 3, 2007; Accepted May 4, 2007

ABSTRACT

The sequencing of complete genomes has created a pressing need for automated annotation of gene function. Because domains are the basic units of protein function and evolution, a gene can be annotated from a domain database by aligning domains to the corresponding protein sequence. Ideally, complete domains are aligned to protein subsequences, in a 'semi-global alignment'. Local alignment, which aligns pieces of domains to subsequences, is common in high-throughput annotation applications, however. It is a mature technique, with the heuristics and accurate *E*-values required for screening large databases and evaluating the screening results. Hidden Markov models (HMMs) provide an alternative theoretical framework for semi-global alignment, but their use is limited because they lack heuristic acceleration and accurate *E*-values. Our new tool, GLOBAL, overcomes some limitations of previous semi-global HMMs: it has accurate *E*-values and the possibility of the heuristic acceleration required for high-throughput applications. Moreover, according to a standard of truth based on protein structure, two semi-global HMM alignment tools (GLOBAL and HMMer) had comparable performance in identifying complete domains, but distinctly outperformed two tools based on local alignment. When searching for complete protein domains, therefore, GLOBAL avoids disadvantages commonly associated with HMMs, yet maintains their superior retrieval performance.

INTRODUCTION

With complete genome sequencing now routine, biology faces the fundamental problem of large-scale automatic annotation of gene function. Local alignment tools (1–7) predominate in automatic annotation, because many of them have the heuristics and accurate *E*-values required for screening large databases rapidly and evaluating search results. In some motif searches, however, including searching for complete domains within a protein sequence, local alignment has two shortcomings. First, it is distracted by strong but incomplete motif matches. Second, it does not align domains over their entire length and does not define their boundaries. Ideally, therefore, complete domains should be aligned to protein subsequences, in a 'semi-global alignment' (8). Accordingly, this article compares the conserved domain (CD) retrieval of a new semi-global alignment tool GLOBAL (GLObal Blocks Aligned Locally) with local and semi-global versions of HMMer (a Hidden Markov model alignment tool) (9,10) and to RPS-BLAST (7) [a local alignment tool, and the current default search tool for NCBI's conserved domain database (CDD) (11)].

To elaborate on the CDD, it offers a comprehensive classification of the CDs composing proteins, modeling each CD as a multiple sequence alignment (MSA). Certain MSAs have been manually curated, to designate a contiguous subset of columns as the 'footprint' of the corresponding CD. For present purposes (Discussion section), we further partitioned the CD footprint into contiguous 'blocks' of conserved columns, separated by 'spacers' of poorly conserved columns. Manual curation constantly refines the MSA to increase the consistency of the blocks with collateral information such as structural alignments or function. Tools related to PSI-BLAST (6) then convert the CD footprint into a position-specific scoring matrix (PSSM).

A benchmarking test set based exclusively on the VAST protein structure alignment program (12) permitted us to compare the CD retrieval performance of GLOBAL, HMMer and RPS-BLAST (Results section). To summarize our main findings, the semi-global alignment tools, GLOBAL and HMMer_semi-global (i.e. HMMer in 'global' mode), essentially had indistinguishable CDD

*To whom correspondence should be addressed. Tel: 301 402 9310; Fax: 301 480 2484; Email: spouge@ncbi.nlm.nih.gov

retrieval performance, and both outperformed the local alignment tools, HMMer_local (i.e. HMMer in 'local' mode) and RPS-BLAST.

Presently, GLOBAL's main advantage over HMMer_semi-global is that GLOBAL has unusually accurate *E*-values. Programs for building protein profiles through iterative search, e.g. PSI-BLAST (6,13), require accurate *E*-values to avoid corrupting their profiles with false positives. GLOBAL's accurate *E*-value therefore opens up the possibility of an iterative program for finding 'complete' domains, either within a single protein or a group of proteins. Moreover, GLOBAL aligns individual blocks to a query protein sequence with gapless local alignment, so it is readily amenable to the word-match heuristics that accelerate RPS-BLAST searches through the CDD (14). We are currently investigating word-match heuristics for GLOBAL.

The layout of the article is as follows. The 'Materials and Methods' section describes the test set for benchmarking CDD retrieval, the $LROC_n$ score used to measure retrieval performance, the GLOBAL algorithm and *E*-value, and the implementation of the retrieval tools.

The 'Results' section assesses first the retrieval performance of the tools, and then the *E*-value accuracy of the tools with the best CDD retrieval (GLOBAL and HMMer_semi-global). Finally, the 'Discussion' section examines the implications of our findings. In particular, the 'independent alignments approximation' in the 'Materials and Methods' section provides an *E*-value for many types of global and semi-global alignments, in response to the statement: 'There is no theory for [the statistical significance of] global alignment.' (15). The independent alignments approximation is not as simple as an extreme-value approximation, but it can be orders of magnitude more accurate (Figures 6 and 7).

The Supplementary Data present the mathematical analysis relevant to the basic concepts in the main article. Interestingly, the Supplementary Data shows that GLOBAL can be viewed either as a classical alignment technique or an unusually simple HMM. Thus, GLOBAL provides a convenient bridge between HMMs and the wealth of statistical and computational techniques available for classical alignment (as aforementioned, e.g. GLOBAL is susceptible to the word-match heuristics used in RPS-BLAST).

MATERIALS AND METHODS

The benchmarking test set

Our standard of truth for comparing different CDD retrieval methods used two databases. First, single-linkage clustering based on BLAST *E*-values of $\leq 10^{-80}$ yielded a non-redundant set of 10 185 proteins from the protein structure database PDB. The non-redundant database, 'DB_10185', is available at: http://www.ncbi.nlm.nih.gov/Structure/VAST/nrpdb.html. Second, the CDD [version 2.02, available at: http://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml (11)] contained a set of 331 MSAs, each not contained within any larger MSA in the familial hierarchy of the CDD, each manually curated and each containing at least one sequence with a known structure. Thus, each of these 331 MSAs could be structurally aligned through that known structure to the members of DB_10185 from PDB. We extracted the set from CDD, to create a database, 'DB_331_CD'.

VAST is a local structural alignment program (12), whose complete list of alignments is available at: http://www.ncbi.nlm.nih.gov/Structure/VAST/vast.shtml. From the list, we found all pairs in DB_331_CD and DB_10185 (from PDB) whose structural alignments had a VAST *E*-value of $\leq 10^{-4}$. Each structural alignment had a first and last column in the corresponding MSA from DB_331_CD, the columns between forming a 'VAST footprint', analogous to the CD footprint described in the Introduction. To discount strong but incomplete local structural similarities, our standard of truth considered an MSA to contain a 'complete' CD, only if the VAST footprint occupied at least 80% of the CD footprint. The two databases (DB_10185 and DB_331_CD), the intersection of the VAST and CD footprints, and the test set containing the structurally related pairs, are available at: ftp://ftp.ncbi.nih.gov/pub/GLOBAL.

Our standard of truth based itself solely on VAST structural alignments, so it avoided subjective judgments derived from human experience with BLAST, which could favor the local alignment tools. Because it held pairwise sequence identity mostly below 25% (see Supplementary Data), it also emphasized subtle protein relationships.

Assessing CDD retrieval performance with the $LROC_n$

The following receiver operating characteristic (ROC) analysis is an established method for measuring the performance of a database search method (16,17). If the protein query contains a CD, the CD is 'relevant'; otherwise, it is 'irrelevant'. Let the total number of irrelevant CDs be *F*. In response to a protein query, a CDD search tool produces a retrieval list, which ranks all CDs in the database. The 'ROC curve' plots the fraction of relevant CDs preceding the *f*-th irrelevant CD against the fraction f/F. The 'ROC score' is the area under the ROC curve. Analogously, the 'ROC_n curve' is the ROC curve truncated on the *X*-axis after the first *n* irrelevant CDs, with the ROC_n score being the area under the ROC_n curve divided by n/F (18,19). The normalization by n/F ensures that an ideal retrieval method (which returns all relevant CDs before any irrelevant CD) receives a ROC_n score of 1.0.

The Supplementary Data describe the 'Localization-Response Operating Characteristic' (LROC) curve (20–22), which accounts for alignments and is therefore slightly preferable to the ROC_n in the context of CD retrieval. $LROC_n$ curves appear in the figures; the ROC_n curves are similar.

To evaluate CDD retrieval over all protein queries, we merged the retrieval lists for each protein query into a single 'pooled list', by sorting the CDs on their *E*-values (6). The ROC_n procedure was carried out on the pooled retrieval list.

The GLOBAL alignment algorithm

A complete mathematical analysis of GLOBAL alignments appears in the Supplementary Data; the intuitive concepts appear here.

GLOBAL exploits the block structure of the CDD directly. Call any (possibly empty or full) subset of contiguous columns in a block, a 'sub-block'. To identify a CD within a query protein sequence, GLOBAL aligns one sub-block from each CD block, in order, against the sequence (Figure 1). GLOBAL only aligns block columns to the sequence, never spacer columns.

On one hand, GLOBAL sometimes aligns all columns in a block or even all blocks to the sequence (e.g. alignment π_1 in Figure 1). On the other hand, it sometimes leaves unaligned arbitrary numbers of columns at the ends of blocks (e.g. alignment π_2) or even entire blocks (e.g. alignment π_3 leaves the purple block $\mathbf{B}_2$ unaligned). Moreover, aligned block columns may overlap with unaligned columns from other blocks (e.g. alignment π_2 makes the purple block $\mathbf{B}_2$ overlap with some unaligned columns from the blue block $\mathbf{B}_3$).

GLOBAL assigns score 0 to unaligned sequence, regardless of its length, between aligned sub-blocks. It also assigns score 0 to unaligned block ends.

In aligning a CD of b blocks $\mathbf{B}_a$ ($a = 1,\ldots,b$) against a sequence $\mathbf{A} = A_1,\ldots,A_n$, GLOBAL aligns the CD blocks in order to the sequence, applying gapless local alignment to each block (Figure 1). GLOBAL alignments have the usual 1-1 correspondence with paths through an alignment graph (e.g. Figure 2). A GLOBAL alignment π has weight W_π equaling the sum of scores from the sub-block columns it aligns to the sequence letters. The GLOBAL score T is the maximum weight W_π over all possible alignments π. A GLOBAL alignment with weight $W_\pi = T$ is 'optimal'. (see Supplementary Data for details.).

Dynamic programming can find the optimal GLOBAL alignment, as follows. Initialize by aligning $\mathbf{B}_1$ against $\mathbf{A}$ with the Smith–Waterman algorithm (1) for gapless local alignment. To induct, note that for $a>1$, an optimal GLOBAL alignment of $\mathbf{B}_1,\ldots,\mathbf{B}_{a+1}$ against $\mathbf{A}$ decomposes into: (i) an optimal GLOBAL alignment of $\mathbf{B}_1,\ldots,\mathbf{B}_a$ against some subsequence $A_1\ldots A_j$; and (ii) an optimal local alignment of $\mathbf{B}_{a+1}$ against $A_{j+1},\ldots,A_n$. Thus, with the optimal alignment scores of $\mathbf{B}_1,\ldots,\mathbf{B}_a$ against all subsequences $A_1,\ldots,A_j$ in hand, optimize the alignment score of $\mathbf{B}_1,\ldots,\mathbf{B}_{a+1}$ against each subsequence $A_1,\ldots,A_j$ by maximizing over $j = 1,\ldots,n$ in the decomposition above. As usual, discover optimal alignments by backtracking through an alignment matrix. The GLOBAL algorithm requires $O(mn)$ time, where CD footprint has length m and the protein sequence has length n. (See Supplementary Data for details.)

GLOBAL alignments have some desirable properties. GLOBAL assigns score 0 to unaligned sequence, regardless of its length, between aligned block columns (23). [In HMM terminology, the blocks are 'free modules' (24)]. GLOBAL therefore respects CDD curation, by freely permitting insertions of arbitrary length between conserved blocks in a protein. GLOBAL assigns score 0 to unaligned columns at the block ends (25). Thus, it recognizes that in evolution, a secondary structure is frequently conserved at its center but not at its end. GLOBAL assigns score 0 to entire unaligned blocks. It therefore recognizes that in evolution, protein domains sometimes experience large deletions.

One evolutionary event does disadvantage GLOBAL, however. In a CD with a long block, unusual insertions into a protein might split the block into sub-blocks. GLOBAL can then align at most one sub-block to the protein sequence correctly. The 'disadvantage' has a trivial *ad hoc* remedy: before retrieving with GLOBAL, just split long blocks in the CDD arbitrarily.

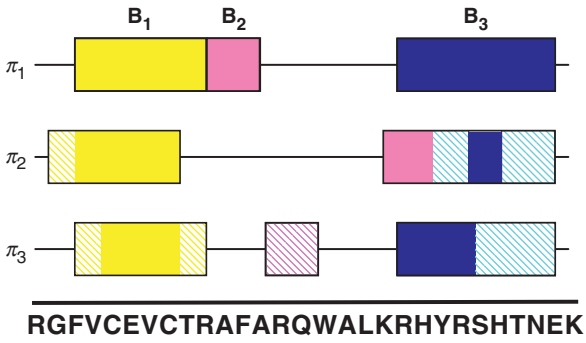

Figure 1. Three possible GLOBAL alignments of a CD to a protein sequence. A protein sequence (bottom); Three alternative GLOBAL alignments of a single CD (π_1, π_2 and π_3) (above). The CD consists of three blocks ($\mathbf{B}_1$, $\mathbf{B}_2$ and $\mathbf{B}_3$, shown as yellow, purple and blue rectangles). Each block corresponds to a PSSM, but for simplicity, PSSM scores are not diagrammed. Usually, GLOBAL aligns only some block columns (solid colors) but not others (diagonally striped colors).

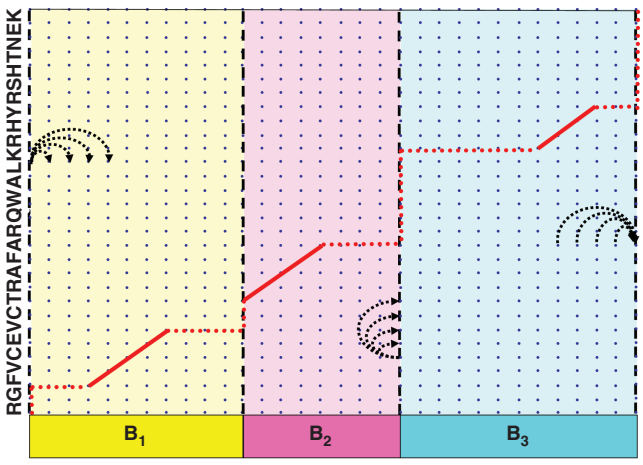

Figure 2. A GLOBAL alignment graph Γ.The alignment graph for a fixed protein sequence (on the y-axis) against $b = 3$ blocks (colored boxes on the x-axis, corresponding to the blocks $\mathbf{B}_1$, $\mathbf{B}_2$ and $\mathbf{B}_3$ in Figure 1). The graph has vertices $\mathbf{V} = \{(i, j):0 \le i \le m,\ 0 \le j \le n\}$ (circles); its directed edges e have integer weights $W(e)$ (not shown). Dotted black arrows correspond to edges of weight 0, indicating unaligned block columns (eastward edges), and unaligned sequence letters (northward edges). The red path corresponds to an optimal alignment, the solid red edges indicating block columns aligned to sequence letters; and dotted red edges (of weight 0) indicating unaligned block columns (eastward edges) and unaligned sequence letters (northward edges).

The GLOBAL *E*-value calculation

DB_331_CD contains $N = 331$ CDs. For a random protein sequence, the *E*-value $E = Np$ is the expected number of CDs with a *P*-value not exceeding p. GLOBAL calculates its *P*-value p under an 'independent letters model', in which a random protein sequence consists of independent, identically distributed amino acids. To optimize empirical retrieval performance, in the random model for each CD, the amino acid frequency was 'composition corrected' (13,26), to match empirical frequencies within the corresponding CD footprint.

Although the Supplementary Data give a mathematical analysis of the GLOBAL *P*-value, the basic concepts appear here.

Before proceeding, note the following concept, 'Markov computation', explained formally in the Supplementary Data and used repeatedly below. A dynamic programming algorithm has a state that changes in response to successive inputs. If the inputs are random and independent of previous states and inputs, the successive states of the dynamic programming computation form a Markov chain. Variants on matrix multiplication can therefore compute the distribution of the successive dynamic programming states. Many articles have been written on special cases of Markov computation (27,28).

The following argument makes many assumptions and is ultimately justified by the success of the resulting *P*-value (Figure 6).

An optimal GLOBAL alignment respects block order. Thus, it usually aligns the a-th block $\mathbf{B}_a$ within some subsequence of **A** of 'effective length' j_a (Figure 3). For convenience, assume that j_a does not depend on a, so $j_a = j$. (In practice, the resulting approximation is both accurate and relatively simple.) Let $\hat{M}_a(j)$ be the optimal gapless local alignment of $\mathbf{B}_a$ against a random sequence of length j. Assume the 'independent alignments approximation', that for some length j, the GLOBAL score T has about the same distribution as the sum $\hat{T} = \sum_{a=1}^{b} \hat{M}_a(j)$ of independent variates $\{\hat{M}_a(j)\}$.

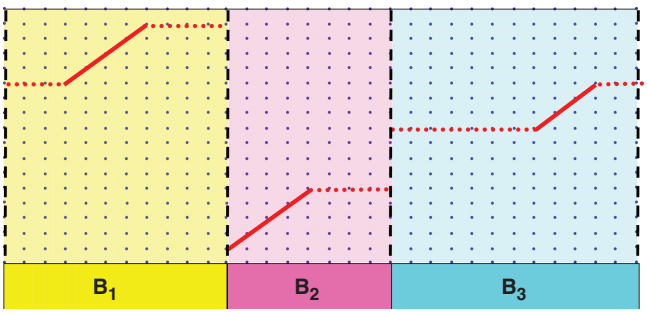

Figure 3. An alignment graph showing the 'independent alignment approximation'. For a random sequence, the alignment graph in Figure 2 corresponds, under the independent alignments approximation, to the $b = 3$ alignment graphs in Figure 3. Each of the three graphs in Figure 3 aligns a random sequence (not shown) against one of the $b = 3$ blocks $\mathbf{B}_1$, $\mathbf{B}_2$ and $\mathbf{B}_3$ from Figure 2. Each of the three alignment matrices shown has the same vertical dimension, the effective length $j = \{(n+b-1)!/[(n-1)!b!]\}^{1/b}$. The length of the sequence in Figure 2 is 29, e.g. so the effective length in Figure 3 is $[(31)!/(28!3!)]^{1/3} \approx 16.5$.The effective length j is determined by Equation (1), which equates the number of combinations of $b = 3$ starting points, for (ordered) optimal matches in Figure 2 and (independent) optimal matches in Figure 3.

The first task is to determine the effective length j. Let the block $\mathbf{B}_a$ have length m_a ($a = 1,\ldots,b$). On one hand, GLOBAL locally aligns the blocks $\mathbf{B}_a$ ($a = 1,\ldots,b$) in order against **A**. The b starting points of the block alignments within the alignment matrix can be chosen in $m_1 \ldots m_b\{n!/[(n-b)!b!]\}$ different ways. (Figure 2; There, the x-coordinates of the starting points can be chosen in $m_1 \ldots m_b$ ways; the y-coordinates, in $n!/[(n-b)!b!]$ ways.)

On the other hand, under the independent alignments approximation, each of the b blocks $\mathbf{B}_a$ ($a = 1,\ldots,b$) is 'independently' aligned locally against a random sequence of effective length j. The b starting points of the block alignments within the b sequences can be chosen in $m_1,\ldots,m_b(j^b)$ different ways. (See Figure 3. There, the x-coordinates of the starting points can be chosen in $m_1,\ldots,m_b$ ways; the y-coordinates, in j^b ways.)

Then, the number of ways of choosing the starting points is the same if

$$m_1,\ldots,m_b\{n!/[(n-b)!b!]\} = m_1,\ldots,m_b(j^b), \qquad \mathbf{1}$$

i.e. if $j = \{n!/[(n-b)!b!]\}^{1/b}$. To make the formula for j apply in the case $n < b$, we replaced n with $n + b - 1$, to produce $j = [(n+b-1)!/\{(n-1)!b!\}]^{1/b}$, the effective length used throughout this article. (The expression $(n+b-1)!/\{(n-1)!b!\}$ is the number of ways of choosing b objects from n objects with replacement, so any object can be chosen several times.)

The next task is to find the distributions of $\{\hat{M}_a(j)\}$. [Approximations based on Gumbel distributions (29) for $\hat{M}_a(j)$ were consistently inferior to the following 'independent diagonals approximation', data not shown]. In the Smith–Waterman alignment matrix (1) for computing $\hat{M}_a(j)$, let the maximum local alignment score on a diagonal d be $\hat{M}_a^{(d)}$. The independent diagonals approximation for gapless local alignment (30) assumes that the diagonals within the alignment matrix are probabilistically independent. Because $\hat{M}_a(j) = \max_d\{\hat{M}_a^{(d)}\}$ (the maximum being over all diagonals d), it follows that $\mathbb{P}\{\hat{M}_a(j) \leqslant y\} = \prod_{(d)} \mathbb{P}\{\hat{M}_a^{(d)} \leqslant y\}$ (the product being over all diagonals d). Thus, the distribution of $\hat{M}_a(j)$ can be computed from the distributions of $\{\hat{M}_a^{(d)}\}$.

To compute the exact distribution of $\hat{M}_a^{(d)}$, note that in 'gapless' local alignment, the Smith–Waterman algorithm applies a dynamic programming recursion along each diagonal d in its alignment matrix. On each step of the recursion, the input for updating the dynamic programming state is an amino acid. In the independent letters model, the amino acid constitutes a random input I_{k+1} independent of previous states and inputs. A Markov computation therefore yields the exact distribution of $\hat{M}_a^{(d)}$.

Finally, the distribution of $\hat{T} = \sum_{a=1}^{b} \hat{M}_a(j)$ can be determined through the usual convolution algorithm (which is itself a Markov computation, with inputs $\{\hat{M}_a(j)\}$). The distribution of $\hat{T}$ serves as an approximation to the distribution of the GLOBAL score T. The GLOBAL *P*-value calculation is complete.

Implementation

Unless otherwise indicated, all implementations used default parameters. The accuracy of an MSA is known to influence retrieval performance (31). All implementations therefore used the same public resource and the same MSAs. If a tool required PSSM input, publicly available tools at NCBI derived the required PSSMs from the MSAs.

The implementation of GLOBAL had no free parameters except the ones inherent in the PSSMs derived from MSAs. (The Supplementary Data support the claim by showing that GLOBAL is a special HMM.) The curated blocks publicly available at: http://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml provided the primary input for GLOBAL's blocks (as described in the 'Discussion' section).

The implementation of HMMer used version hmmer-2.3.2.bin.intel-linux. First, the search database was constructed from the MSAs in DB_331_CD (described above). Second, hmmbuild with the default option hmmls built models for the semi-global alignments; and hmmbuild with the option *–s* built models for the local alignments. We call these variants 'HMMer_semi-global' and 'HMMer_local'. Finally, hmmcalibrate fitted the Gumbel parameters for the HMM *P*-value, and hmmpfam searched the CDD.

The implementation of RPS-BLAST used the NCBI standalone version at: ftp://ftp.ncbi.nih.gov.

RESULTS

Assessing CDD retrieval performance

Our test set emphasized subtle protein relationships with pairwise sequence identity mostly <25% (Supplementary Data). The $LROC_n$ curve and $LROC_n$ score (see 'Materials and Methods' section) measured the CDD retrieval performance of GLOBAL, HMMer_semi-global, HMMer_local and RPS-BLAST on the test set.

As retrieval performance improves, the $LROC_n$ curve for a tool moves higher on the $LROC_n$ plot. The $LROC_n$ plot in Figure 4 was truncated at a 5% false-positive rate, because users rarely examine a CD retrieval list farther. Figure 4 shows that the semi-global methods (GLOBAL and HMMer_semi-global) had comparable retrieval efficacy and dominated the local methods (HMMer_local and RPS-BLAST) throughout CDD retrieval. The domination increased toward the 'twilight zone' at the right of the $LROC_n$ plot, where relationships are most difficult to detect.

Similarly, as retrieval performance improves, the $LROC_n$ score increases. The $LROC_n$ scores displayed in Table 1 (corresponding to ~1, 5, 10 and 20 unrelated CDs per protein query) were chosen because most users examine a CD retrieval list at least up to the first irrelevant CD, but usually not farther than the 20th. Table 1 confirms that the semi-global methods dominated the local methods. As assessed by the bootstrap (13), the small numerical difference in $LROC_n$ scores between GLOBAL and HMMer_semi-global was statistically insignificant in early retrieval, up to about the 10 unrelated CDs per protein query, after which GLOBAL showed a statistically significant improvement over all other tools. Although late retrieval is not as important as early retrieval, it does enter some applications [e.g. SAM T99 uses WU-BLAST retrieval up to *E*-values of 300 (31,32)].

GLOBAL and HMMer_semi-global had similar computation times, seconds to minutes, for: (i) the *E*-value pre-computation for 1 CD; and (ii) the CD retrieval for 10^4 protein queries. To emphasize the computational differences, GLOBAL's pre-computation involves dynamic programming, not simulation, whereas HMMer_semi-global's pre-computation involves simulation.

Assessing *E*-value accuracy

Theoretically, the *E*-value estimates the number of errors (false positives) preceding a CD in a CD retrieval list. To evaluate the accuracy of an *E*-value, the 'EPQ plot' plots the empirical average number of retrieval errors per query against the *E*-value (33). If an *E*-value were to estimate the

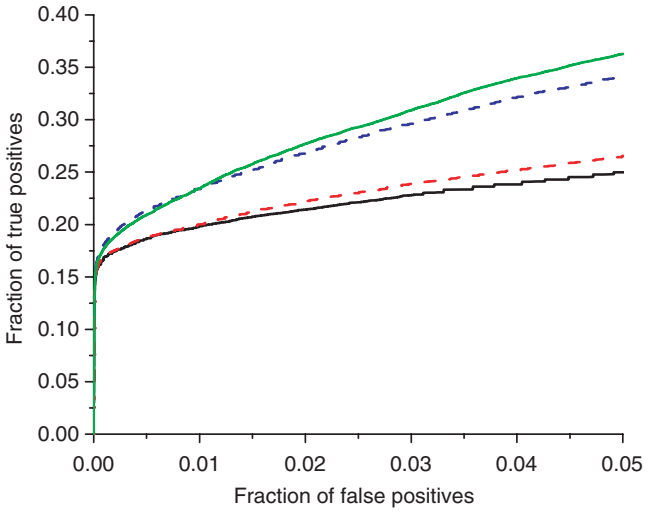

Figure 4. $LROC_n$ curves comparing CD retrieval performances.The $LROC_n$ curves for GLOBAL (green solid line), HMMer_semi-global (blue dashed line), HMMer_local (red dashed line) and RPS-BLAST (black solid line) up to a 5% false-positive rate.

Table 1. The $LROC_n$ score for GLOBAL, HMMer_semi-global, HMMer_local and RPS-BLAST

	$LROC_{10\,000}$	$LROC_{50\,000}$	$LROC_{100\,000}$	$LROC_{200\,000}$
GLOBAL	0.181	0.224	0.260	0.313
HMMer_semi-global	0.185	0.224	0.254	0.299
HMMer_local	0.169	0.194	0.213	0.239
RPS-BLAST	0.168	0.192	0.207	0.229

The $LROC_n$ score is given for n = 10 000; 50 000; 100 000; and 200 000 in the pooled retrieval list (see 'Materials and Methods' section). These values of n correspond to ~1, 5, 10 and 20 unrelated CDs per protein query. All $LROC_n$s indicated have an error of ±0.003, as estimated by bootstrap (13).

errors perfectly for each query, the EPQ plot would place the corresponding point on its diagonal line ($y = x$). The borderline for statistical significance is usually placed somewhere between about 0.01 and 1 error per query (e.g. (31) and ftp://ftp.genetics.wustl.edu/pub/eddy/hmmer/CURRENT/Userguide.pdf). As an initial assessment of *E*-value accuracy, the EPQ plots in Figure 5 for GLOBAL, HMMer and RPS-BLAST were unremarkable, except to indicate that on average, our objective (structural) standard of truth misclassified about 2% of the positives per query as negatives. Although a misclassification rate of 0.02 has little impact on the $LROC_n$ assessment of relative retrieval performance, it could have a large impact on an EPQ assessment of *E*-value accuracy, particularly for *E*-values less than about 0.02.

Rather than make logically circular, *post hoc* subjective judgments to 'correct' the EPQ plot, we used simulations to compare the *E*-value accuracies of GLOBAL and HMMer_semi-global, the two tools with the best CDD retrieval performance.

HMMer derives its *E*-values from fitting the two parameters of a Gumbel distribution. By default, it fits its Gumbel parameters from 5000 random sequences with lengths normally distributed about a mean of 350. To make the conditions of comparison favorable to HMMer_semi-global, the Gumbel parameters in HMMer were fit from 100 000 random sequences instead of the default 5000. Similarly, because HMMer's *E*-value approximation should be most accurate for sequences of length 350 (the mean length used in its Gumbel fit), to favor HMMer further, the *E*-values in the two semi-global tools were tested by aligning CDs against 1 000 000 random sequences of length 350 from a standard (Robinson and Robinson) background amino acid distribution (34).

For a random protein sequence, the *E*-value $E = Np$ is the expected number of CDs with a *P*-value not exceeding p (see 'Materials andMethods' section). For a fixed number N of CDs in the CDD, the *E*- and *P*-values therefore have the same relative error. Accordingly, Figure 6 displays *P*-value accuracies for GLOBAL; Figure 7, for HMMer_semi-global. Although the three

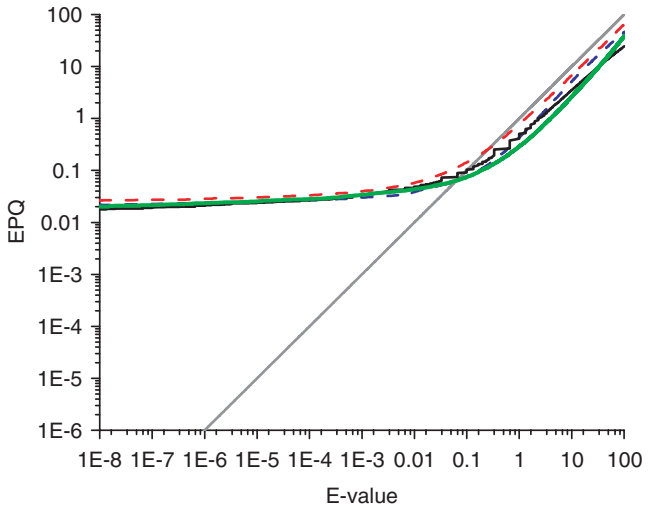

Figure 5. An EPQ plot, graphing errors per query against *E*-value. For a given protein query and a particular *E*-value threshold, a retrieval tool might make 'errors' by assigning *E*-values below the threshold to irrelevant CDs (i.e. CDs not in the query). Figure 5 plots the average number of errors per protein query against the *E*-value threshold for GLOBAL (green solid line), HMMer_semi-global (blue dashed line), HMMer_local (red dashed line) and RPS-BLAST (black solid line). All curves intersect the *y*-axis at about 0.02, probably because our structural standard of truth misclassifies as unrelated about 2% of the related pairs in DB_10185 and DB_331_CD.

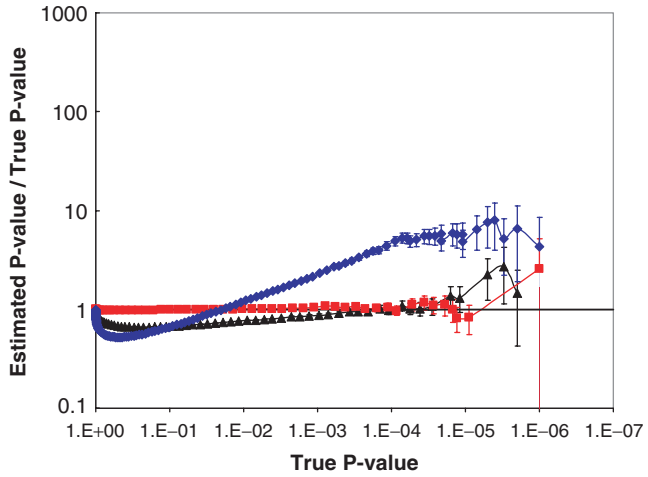

Figure 6. The accuracy of *P*-values for GLOBAL. Figure 6 plots $\hat{p}/p$ on a logarithmic scale against p, where $\hat{p}$ is the calculated GLOBAL *P*-value, and p is the *P*-value from the simulation. Thus, the horizontal solid black line $\hat{p}/p = 1$ (shown) corresponds to perfect *P*-value estimation. The error bars correspond to 1 SEM. (The error bars are asymmetric because of the logarithmic scale. In addition, they were omitted for some points on the right, if they included negative values and could not be plotted on a logarithmic scale.) Figure 6 shows GLOBAL *P*-value results for three CDs: cd00030 (black triangle), having 8 blocks of lengths 16, 16, 12, 6, 11, 15, 17 and 12; cd00083 (red square), having 2 blocks of lengths 34 and 26; and cd00288 (blue diamond), having 44 blocks of lengths 13, 5, 10, 10, 21, 13, 8, 8, 10, 6, 6, 10, 7, 15, 8, 7, 7, 10, 7, 6, 8, 17, 12, 8, 6, 6, 9, 19, 8, 12, 14, 8, 8, 17, 17, 18, 11, 10, 12, 10, 13, 19, 18 and 13.

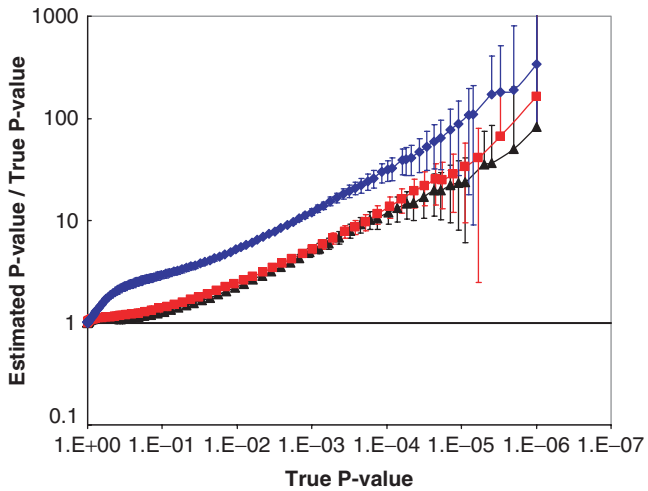

Figure 7. The accuracy of *P*-values for HMMer_semi-global. HMMer_semi-global *P*-value results for the same three CDs and in the same format as in Figure 6: cd00030 (black triangle); cd00083 (red square); and cd00288 (blue diamond).

CDs shown represent the full range of GLOBAL accuracies, from best to worst, they are otherwise arbitrary. Compared with all other CDs in the CDD, the calculated GLOBAL *P*-values for cd00288 were the least accurate, probably because cd00288 has many short blocks. The GLOBAL *P*-values were generally quite accurate, however, whereas the Gumbel *P*-value approximations in HMMer_semi-global were not. The Gumbel *P*-value approximations have increasing errors as the *P*-value decreases, likely reflecting the notorious difficulties in fitting the Gumbel scale parameter λ (40).

The GLOBAL *P*-value is typically an underestimate for sequences longer than about 400, an overestimate for sequences shorter. In our experiments, the *P*-value accuracy usually (but not always) improved with decreasing *P*-values and the number of blocks in a CD. The percentages of the 331 CDs in our test set where the calculated GLOBAL *E*-value $E = 0.1$ differed from simulation estimates by less than a factor of 2.0 were as follows: 60% at sequence length 200; 93% at sequence length 400; and 62% at sequence length 1000. The 'twilight' *E*-value $E = 0.1$ corresponds to the *P*-value $P = 0.1/331 \sim 3 \times 10^{-4}$ in Figures 6 and 7.

DISCUSSION

Intuitively, classification methods should be 'global', in the sense that they should exploit all available information. Correspondingly, in the identification of 'complete' protein domains within protein sequences, Figure 4 shows that the semi-global tools (GLOBAL and HMMer_semi-global) dominate the local alignment tools (HMMer_local and RPS-BLAST). HMMer_local is competitive among local alignment tools based on HMMs (31,32), so potentially, semi-global alignment could dominate local alignment in applications requiring the identification of complete protein domains within protein sequences.

Domain classification methods should be global in the sense above, but they must also maintain flexibility, to handle common evolutionary events like deletions at the ends of a conserved secondary structure. Tools with a strong tendency to align complete blocks (or motifs) might lack such flexibility. In fact, we tested the default mode of several such tools: MAST (35,36), various implementations of META-MEME (37) and the global implementation of SALTO (38). In our hands, according to our benchmarking test set and throughout all of CDD retrieval, none of these tools performed as well as any tool appearing in Figure 4 (data not shown).

In general, the quality of MSAs noticeably influences retrieval performance (31). In the CDD, curators define 'curated blocks' very restrictively, e.g. curated blocks do not contain gap characters. Moreover, within the 'curated spacer' between a consecutive pairs of curated blocks, each MSA sequence is padded in its middle with inserted gap characters, up to the length required. Because CDD curators do not actively align sequence in the curated spacers, the curated spacer alignments are adventitious.

To test whether the curated spacers contain information relevant to domain identification, the 'blocks' used throughout this article were defined by augmenting the curated blocks with all contiguous MSA columns having fewer than 50% gap characters. The adventitious alignments within curated spacers were left unchanged. Thus, curated blocks corresponded to subsets of our blocks, and curated spacers corresponded to supersets of our spacers.

After modifying GLOBAL to use the (shorter) curated blocks, retrieval performance degraded (data not shown), so the curated spacers do indeed contain relevant information (39). Our findings therefore suggest that careful curation of alignments between the curated CDD blocks might noticeably improve the identification and alignment of complete domains within query proteins.

Like HMMer_semi-global, GLOBAL is an HMM (see Supplementary Data). In the GLOBAL HMM, transition probabilities are determined by the block sizes in a CD and are effectively fixed (indeed, they take rather counter-intuitive values). Thus, the GLOBAL HMM fits only emission probabilities, whereas HMMer fits both transition and emission probabilities. The retrieval performances for GLOBAL and HMMer_semi-global were almost indistinguishable, however (Table 1). Within limits, therefore, the retrieval performance of an HMM probably depends more on its emission probabilities than on its transition probabilities.

HMMs sometimes estimate their *E*-values rather poorly (31) (Figure 7). The manual for HMMer, e.g. warns that the HMMer_semi-global Gumbel *E*-value approximation is sometimes very inaccurate (ftp://ftp.genetics.wustl.edu/pub/eddy/hmmer/CURRENT/Userguide.pdf). Some authors even question the theoretical foundations for fitting Gumbel distributions in an HMM (40). In contrast, GLOBAL calculates its *E*-values by dynamic programming, not by simulating or fitting distributional parameters. Consequently, it estimates *E*-values for its null model of independent, identically distributed amino acids quite accurately.

In addition, GLOBAL uses gapless local alignment to align each CD block to a protein sequence (Figure 1). It is therefore amenable to the same heuristics accelerating local alignment computations in RPS-BLAST.

To summarize, GLOBAL is a new semi-global alignment tool for finding complete domains within protein sequences. It has competitive retrieval performance, an accurate *E*-value and the possibility of heuristic acceleration, all of which enhance its potential as a high-throughput tool. The implementation of GLOBAL as the default tool at NCBI for searching the CDD is underway; the current version of GLOBAL is available at: ftp://ftp.ncbi.nih.gov/pub/GLOBAL.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We thank Anna Panchenko for her contribution to building the benchmarking set; Chris Lanczycki and Charlie Liu for their help in implementing PSSM computations; Andy Neuwald, Aron Marchler-Bauer, Yi-Kuo Yu and Stephen Altschul for helpful discussions

and/or comments on the manuscript. Support for this work was provided by the intramural research program of the National Institutes of Health.Funding to pay the Open Access publication charges for this article was provided by the intramural research program of the National Institutes of Health.

Conflict of interest statement. None declared.

REFERENCES

1. Smith,T.F. and Waterman,M.S. (1981) Identification of common molecular subsequences. *J. Mol. Biol.*, **147**, 195–197.
2. Pearson,W.R. (1991) Searching protein sequence libraries: comparison of the sensitivity and selectivity of the Smith-Waterman and FASTA algorithms. *Genomics*, **11**, 635–650.
3. Lipman,D.J. and Pearson,W.R. (1985) Rapid and sensitive protein similarity searches. *Science*, **227**, 1435–1441.
4. Pearson,W.R. and Lipman,D.J. (1988) Improved tools for biological sequence comparison. *Proc. Natl Acad. Sci. USA*, **85**, 2444–2448.
5. Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.
6. Altschul,S.F., Madden,T.L., Schäffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
7. Schäffer,A.A., Wolf,Y.I., Ponting,C.P., Koonin,E.V., Aravind,L. and Altschul,S.F. (1999) IMPALA: matching a protein sequence against a collection of PSI-BLAST-constructed position-specific score matrices. *Bioinformatics*, **15**, 1000–1011.
8. Erickson,B.W. and Sellers,P.H. (1983) In Kruskal,J.B. and Sankoff,D. (ed.), *Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, Reading, MA, pp. 55–91.
9. Karplus,K., Barrett,C. and Hughey,R. (1998) Hidden Markov models for detecting remote protein homologies. *Bioinformatics*, **14**, 846–856.
10. Bateman,A., Birney,E., Cerruti,L., Durbin,R., Etwiller,L., Eddy,S.R., Griffiths-Jones,S., Howe,K.L., Marshall,M. *et al.* (2002) The Pfam protein families database. *Nucleic Acids Res.*, **30**, 276–280.
11. Marchler-Bauer,A., Anderson,J.B., DeWeese-Scott,C., Fedorova,N.D., Geer,L.Y., He,S., Hurwitz,D.I., Jackson,J.D., Jacobs,A.R. *et al.* (2003) CDD: a curated Entrez database of conserved domain alignments. *Nucleic Acids Res.*, **31**, 383–387.
12. Gibrat,J.-F., Madej,T. and Bryant,S.H. (1996) Surprising similarities in structure comparison. *Curr. Opin. Struct. Biol.*, **6**, 377–385.
13. Schäffer,A.A., Aravind,L., Madden,T.L., Shavirin,S., Spouge,J.L., Wolf,Y.I., Koonin,E.V. and Altschul,S.F. (2001) Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res.*, **29**, 2994–3005.
14. Marchler-Bauer,A., Panchenko,A.R., Shoemaker,B.A., Thiessen,P.A., Geer,L.Y. and Bryant,S.H. (2002) CDD: a database of conserved domain alignments with links to domain three-dimensional structure. *Nucleic Acids Res.*, **30**, 281–283.
15. Korf,I., Yandell,M. and Bedell,J. (2003). *BLAST* 1st edn. O'Reilly, Sebastopol, CA, pp. 55.
16. Bamber,D. (1975) The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *J. Math. Psychol.*, **12**, 387–415.
17. Swets,J.A. (1988) Measuring the accuracy of diagnostic systems. *Science*, **240**, 1285–1293.
18. McClish,D.K. (1989) Analyzing a portion of the ROC curve. *Med. Decision Making*, **9**, 190–195.
19. Gribskov,M. and Robinson,N.L. (1996) Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. *Comput.Chem.*, **20**, 25–33.
20. Chakraborty,D.P. (1989) Maximum likelihood analysis of free-response receiver operating characteristic (FROC) data. *Med. Phys.*, **16**, 561–568.
21. Swensson,R.G. (1996) Unified measurement of observer performance in detecting and localizing target objects on images. *Med. Phys.*, **23**, 1709–1725.
22. Edwards,D.C., Kupinski,M.A., Metz,C.E. and Nishikawa,R.M. (2002) Maximum likelihood fitting of FROC curves under an initial-detection-and-candidate-analysis model. *Med. Phys.*, **29**, 2861–2870.
23. Wilbur,W.J. and Lipman,D.J. (1983) Rapid similarity searches of nucleic acid and protein data banks. *Proc. Natl Acad. Sci. USA*, **80**, 726–730.
24. Barrett,C., Hughey,R. and Karplus,K. (1997) Scoring hidden Markov models. *Comput. Appl. Biosci.*, **13**, 191–199.
25. Neuwald,A.F. and Poleksic,A. (2000) PSI-BLAST searches using hidden markov models of structural repeats: prediction of an unusual sliding DNA clamp and of beta-propellers in UV-damaged DNA-binding protein. *Nucleic Acids Res.*, **28**, 3570–3580.
26. Tatusov,R.L., Fedorova,N.D., Jackson,J.D., Jacobs,A.R., Kiryutin,B., Koonin,E.V., Krylov,D.M., Mazumder,R., Mekhedov,S.L. *et al.* (2003) The COG database: an updated version includes eukaryotes. *BMC Bioinformatics*, **4**, 41.
27. Huang,H., Kao,M.C., Zhou,X., Liu,J.S. and Wong,W.H. (2004) Determination of local statistical significance of patterns in Markov sequences with application to promoter element identification. *J. Comput. Biol.*, **11**, 1–14.
28. Staden,R. (1989) Methods for discovering novel motifs in nucleic acid sequences. *Comput. Appl. Biosci.*, **5**, 293–298.
29. Gumbel,E.J. (1958) *Statistics of Extremes* Columbia University Press, New York.
30. Dembo,A., Karlin,S. and Zeitouni,O. (1994) Limit distribution of maximal non-aligned two-sequence segmental score. *Ann Prob.*, **22**, 2022–2039.
31. Madera,M. and Gough,J. (2002) A comparison of profile hidden Markov model procedures for remote homology detection. *Nucleic Acids Res.*, **30**, 4321–4328.
32. Park,J., Karplus,K., Barrett,C., Hughey,R., Haussler,D., Hubbard,T. and Chothia,C. (1998) Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. *J. Mol. Biol.*, **284**, 1201–1210.
33. Brenner,S.E., Chothia,C. and Hubbard,T.J. (1998) Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Natl Acad. Sci. USA*, **95**, 6073–6078.
34. Robinson,A.B. and Robinson,L.R. (1991) Distribution of glutamine and asparagine residues and their near neighbors in peptides and proteins. *Proc. Natl Acad. Sci. USA*, **88**, 8880–8884.
35. Bailey,T.L. and Gribskov,M. (1998) Combining evidence using p-values: application to sequence homology searches. *Bioinformatics*, **14**, 48–54.
36. Bailey,T.L., Williams,N., Misleh,C. and Li,W.W. (2006) MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Res.*, **34**, W369–W373.
37. Grundy,W.N., Bailey,T.L., Elkan,C.P. and Baker,M.E. (1997) Meta-MEME: motif-based hidden Markov models of protein families. *Comput. Appl. Biosci.*, **13**, 397–406.
38. Kann,M.G., Thiessen,P.A., Panchenko,A.R., Schaffer,A.A., Altschul,S.F. and Bryant,S.H. (2005) A structure-based method for protein sequence alignment. *Bioinformatics*, **21**, 1451–1456.
39. Wrabl,J.O. and Grishin,N.V. (2004) Gaps in structurally similar proteins: towards improvement of multiple sequence alignment. *Proteins*, **54**, 71–87.
40. Yu,Y.K., Bundschuh,R. and Hwa,T. (2002) Hybrid alignment: high-performance with universal statistics. *Bioinformatics*, **18**, 864–872.