**Ballast: Blast post-processing based on locally conserved segments**

F. Plewniak¹,* , J. D. Thompson¹ and O. Poch¹

¹Institut de Génétique et de Biologie Moléculaire et Cellulaire, Laboratoire de Biologie Structurale, (CNRS/INSERM/ULP), BP 163, 67404 Illkirch Cedex, France

Received on February 18, 2000; revised and accepted on April 13, 2000

**Abstract**

*Motivation:* Blast programs are very efficient in finding relatively strong similarities but some very distantly related sequences are given a very high Expect value and are ranked very low in Blast results. We have developed Ballast, a program to predict local maximum segments (LMSs—i.e. sequence segments conserved relatively to their flanking regions) from a single Blast database search and to highlight these divergent homologues. The TBLastN database searches can also be processed with the help of information from a joint BlastP search.

**Results:** We have applied the Ballast algorithm to BlastP searches performed with sequences belonging to well described dispersed families (aminoacyl-tRNA synthetases; helicases) against the SwissProt 38 database. We show that Ballast is able to build an appropriate conservation profile and that LMSs are predicted that are consistent with the signatures and motifs described in the literature. Furthermore, by comparing the Blast, PsiBlast and Ballast results obtained on a well defined database of structurally related sequences, we show that the LMSs provide a scoring scheme that can concentrate on top ranking distant homologues better than Blast. Using the graphical user interface available on the Web, specific LMSs may be selected to detect divergent homologues sharing the corresponding properties with the query sequence without requiring any additional database search.

**Availability:** Web service is at http://igbmc.u-strasbg.fr:8080/ballast.html and source code is available at ftp://ftp-igbmc.u-strasbg.fr/Ballast/

**Contact:** plewniak@igbmc.u-strasbg.fr

**Introduction**

In comparative genomics and protein function prediction, the Blast programs (Altschul et al., 1990) are widely used to search protein and nucleotide databases for homologues of query sequences. Though Blast programs are very efficient in finding relatively strong similarities, distantly related sequences scoring poorly are usually difficult to distinguish from the background of non-related ones. This problem is even more significant in TBLastN searches and will become more challenging as genome portions and full cosmids are incorporated into the databases.

Recently, several new implementations of the Blast algorithm have been proposed to improve its sensitivity. PsiBlast (Altschul et al., 1997) iteratively builds a position-specific score matrix from the best high-scoring segment pairs (HSPs) and uses it to search the protein database until it converges and no new similar sequences are found. PhiBlast (Zhang et al., 1998) limits the rate of false positives by highlighting the regions of homology that share *a priori* known functional motifs. A number of post-processing programs have been developed to help the interpretation of Blast results thanks to clear presentation and additional information. BEAUTY (Worley et al., 1995) stacks the HSPs and searches them for Prosite patterns (Hofmann et al., 1999) and known domains to help function assignment. Bla (Tatusov and Koonin, 1994) searches the HSPs for Prosite as well as user-defined patterns. PowerBlast (Zhang and Madden, 1997) combines BLAST searching with filtering of low complexity regions and repeats. Results are displayed graphically and textually as multiple alignments, with annotated features superimposed on the aligned sequences. MSPCrunch (Sonnhammer and Durbin, 1994) studies the compatibility of HSPs found for each database sequence to reduce background noise and its results can be viewed with the Blixem graphical interface (Sonnhammer and Durbin, 1994). Visual Blast (Durand and Mornon, 1997) stacks HSPs as a multiple alignment, builds a conservation profile and provides various HSP analysis tools such as hydrophobic cluster analysis. MView (Brown et al., 1998) formats the results of a Blast search into a coloured multiple alignment of hits stacked against the query. CBLAST (Miller and Fuchs, 1997) sorts the results of a BLAST search according to sequence membership in user-defined ‘clusters’ of sequences.

Here, we present Ballast, a program that concentrates
distant homologues among top ranking hits in a Blast database search without requiring any additional user-provided information nor use of any extra database. One major advantage of Ballast is its ability to predict local maximum segments (LMSs, i.e. segments more conserved than their flanking regions) from the profile of conservation along the query sequence. This local approach is more efficient than a global threshold method since segments conserved in a very small number of database sequences are detected as well as highly represented segments. The LMSs define a new scoring scheme to sort the alignments produced by Blast: alignments overlapping such segments are emphasized, thus reducing the rate of false positives while accentuating the potentially important regions. In addition to the default fully automated scoring scheme, the user can interactively select specific LMSs with the Web graphical interface in order to sort the Blast alignments accordingly.

The Ballast website proposes BlastP searches on the SwissProt + SPT|EMBL non-redundant protein database and/or joint TBlastN searches on the C. elegans genome or the other currently available complete short genomes. This is a unique feature as Ballast is the only program that is able to process a TBlastN search with the help of information drawn from joint a BlastP search.

System and methods
Ballast is written in the ANSI C programming language and has been successfully implemented on a Sparc Server 1000 running Solaris 2.7 and a DEC Alpha Server 8200 running Digital Unix V4.0. Post-processing the Blast search by Ballast takes less than two seconds on the Alpha 8200 running Digital Unix V4.0. Post-processing the Blast alignments accordingly, the user can interactively select specific LMSs with the Web graphical interface in order to sort the Blast alignments accordingly.

The Ballast website proposes BlastP searches on the SwissProt + SPT|EMBL non-redundant protein database and/or joint TBlastN searches on the C. elegans genome or the other currently available complete short genomes. This is a unique feature as Ballast is the only program that is able to process a TBlastN search with the help of information drawn from joint a BlastP search.

Algorithm
Construction of the raw conservation profile
Ballast is able to process both the gapped and ungapped versions of Blast. The gapped alignments though are first divided into ungapped sections that we call ungapped segment pairs (USPs). In the ungapped version of Blast, USPs and HSPs are equivalent while in the gapped version, USPs are portions of the alignments returned by Blast and do not exactly reflect HSPs.

Only USPs coming from alignments with an Expect value $E$ smaller than 0.1 are used to build the profile of conservation in order to provide as much information as possible about sequence variability while keeping a reasonably low proportion of distorted information from false positives. Furthermore, USPs coming from sequences producing strictly identical alignments are counted only once in the profile.

In addition to the filtering of the query sequence by Blast with the seg algorithm (Wootton and Federhen, 1993) Ballast filters out low-complexity (non-informative) regions of the database sequences: for each USP the concentration of residues is computed over a 7-residue-long window sliding along the portion of the database sequence corresponding to the USP. This value, defined as the probability that two residues chosen at random in the window are identical, is given by

$$C = \sum_{a=1}^{A} \frac{n_a(n_a - 1)}{L(L - 1)}$$

where $A$ is the number of different amino acids in the window, $n_a$ is the number of occurrences of amino acid $a$ in the window and $L$ is the window length.

Windows with a concentration value of more than 0.5 are considered non-informative and the positions overlapping all residues $a$ with the maximum $n_a$ are removed from the USP. A USP that is non-informative over more than one-third of its total length is totally removed as we observed that such USPs are generally of low information content (data not shown). If considered informative, the USP is stacked along the query sequence. For each position in the USP the following value is added to the corresponding position in the raw profile of conservation (see Figure 1a):

- $(1 - E)$ for identical residues in both query and database sequences.
- $(1 - E)/2$ for similar residues in query and database.
- $0$ for different amino acids.

where $E$ is the expect value of the HSP or alignment given by BlastP.
The definition for similar and different residues here is based upon the scoring matrix used by Blast (typically BLOSUM 62). Residues scoring positively against each other in the matrix are considered similar, otherwise they are considered different.

**Local maximum segment prediction**

Segments that are more conserved than their local environment are expected to correspond to local maxima in the conservation profile and can be predicted by detecting peaks regardless of their height. We will designate such segments of the query sequence as LMSs. The conservation profile is thus smoothed by averaging over a 9-residue-long sliding window (Figure 1b). Peaks can be easily detected as regions of the smoothed profile between two points of inflection where the second derivative is negative. The sharper the peak, the more negative the minimum of the second derivative; thus, we empirically determined that peaks of interest were those achieving a second derivative less than \(-0.1\) estimated on the smoothed profile. We define an LMS as a segment of the query sequence which verifies the following properties (Figure 1c):

- it is located in a conserved region, i.e. it overlaps a sharp peak (second derivative \(< -0.1\)),
- the values of the smoothed profile at the corresponding positions are at least 30% of the maximum value of the peak,
- it has a length of at least three residues.

**Scoring sequences according to the presence of LMSs**

A LMS pair (LMSP) is defined in both BlastP and TBlastN results as the region of a USP overlapping an LMS in the query sequence. The score for an individual LMSP is computed as

\[
S(LMSP) = \sum_{i=m_0}^{m_l} a \cdot p_i
\]

where \(m_0\) is the first position in the LMSP, \(m_l\) is the last position in the LMSP, \(p_i\) is the value of the raw profile at the position \(i\), and \(a\) is empirically defined as

- 2 for identical residues in both query and database sequences,
- 0.2 for similar residues in query and database sequences,
- \(-1\) for different residues in query and database sequences.

For every database sequence the succession of the LMSPs which fits best to the profile of conservation is determined.
by a dynamic programming algorithm (Figure 1d). The score for the optimal succession of LMSPs ending by the LMSPM is given by

\[ S_{opt}(M) = \max_{L} (S(M); S(M) + S_{opt}(L)) \]

where \( S(M) \) is the score for the individual LMSPM described above,

\[ S_{opt}(L) \] is the score for the optimal succession of LMSPs ending by a LMSPL with start positions in both the query and the database sequences smaller than those of the LMSPM.

This requires the database sequence segments of the LMSPs to be collinear to the LMSs in the query sequence. Patterns describing the observed LMSs are then determined by computing position specific scoring matrices from the observed amino acids at every position of the LMSPs involved in the best succession (see above). Only LMSPs from database sequences used to build the profile of conservation, i.e. with an Expect value less than 0.1, are used here. A residue is considered to be conserved if it contributes to more than 10% of the maximum score at the position while a residue is considered to be highly conserved if it contributes to more than 75%.

Sorting sequences according to the Ballast score

The above score is used to sort sequences from both the BlastP and TBlastN searches. During this process, sequences producing identical USPs are grouped together as they generally correspond to identical or very close sequences, or coding sequences and their protein product.

In order to keep all available information, sequences having achieved an Expect value less than \( 10^{-3} \) in the Blast search are rescued no matter how badly they score with Ballast. Such sequences are frequently fragments of longer sequences which lack some motifs because they are incomplete. They are inserted immediately before the first high Ballast-scoring sequence with a Blast Expect value of more than \( 10^{-3} \).

Implementation

Command line interface

The minimum input to the Ballast command line version is a BlastP result file from either a local or a remote Blast search (version 1.4x or 2.0.x, gapped or ungapped). A TBlastN result file obtained with the same query sequence can also be provided. We empirically determined that the input Blast results should list sequences with an Expect value up to 5000 in order to yield very distant homologies that will be identified by Ballast.

Ballast creates three files containing: (i) the profile of conservation along the query; (ii) an alignment of all the LMSPs found in the USPs used to build the profile; (iii) a list of database sequences sorted by their score according to the LMSPs; this file also contains, for each sequence, a list sorted by position of all the USPs matching LMSSs. A list of proposed patterns is printed to the standard output.

Optionally, a list of the LMSPs found in all the USPs used to build the profile can be listed to a file.

Web graphical interface

A website (see Figure 2) has been implemented which provides concurrent protein and genomic searches as well as an interactive graphical interface. The query protein is submitted through a regular HTML form. Common gateway interface (CGI) scripts run BlastP and TBlastN searches and Ballast on a DEC Alpha Server 8200 with 6 x 525MHz EV6 CPUs. The BlastP and TBlastN searches together with the Ballast post-processing take about 3–5 min with a query sequence of 1000 residues. The actual time used by Ballast itself is less than 2 s. A list of similar sequences is returned with hypertext links to a Sequence Retrieval Software (SRS) server (Etzold et al., 1996) providing fast and easy access to sequence database information. The profile of conservation can be interactively visualized using a GUI written in Tcl/Tk and executed within the Web browser through the Tcl/Tk plugin. Individual LMSs can be selected on the profile to serve as a basis for a user defined scoring scheme. The original BlastP search can also be accessed as a webpage with links to an SRS Web server. For each session, a unique ID number is provided to enable easy and fast access to the results which are stored for one week.

Results and discussion

Construction of profile of conservation and prediction of LMSs

As the Ballast scoring scheme is based on the LMSs predicted from the profile of conservation, the quality of Ballast results depends essentially upon the quality of the profile and the predicted LMSs. We tested this quality using well documented families of sequences with widely accepted signatures. For all the tests, the SwissProt protein database (version 38, July 1999) was used for the quality of its annotations.

When constructing the profile of conservation, Ballast does not penalize the presence of different residues and adds positive values only to the profile. This ensures that the conservation of residues will not be underestimated because of accidental misalignments. Furthermore, the contribution of the alignments to the profile is underestimated according to their Expect value. As the Expect value grows, the contribution to the profile is decreased since alignments with high Expect values are uncertain. The profile of conservation thus gives a good estimation of the observed conservation of the residues along the query sequence and is robust to misalignment. Even when a
very few sequences are used to build the profile, the peaks correspond to the conserved regions and the valleys to the variable regions. However, not all the peaks are of equal height as some regions are more strongly conserved than others. Therefore, instead of using a threshold that would detect only very strong peaks and miss lower ones, peaks are detected by computation of the second derivative of the smoothed profile. Thus, even lower peaks that correspond to less conserved but nevertheless biologically significant regions can be detected. The correspondence between the detected peaks, i.e. the LMSs predicted by Ballast, and the known specificities for three well described query sequences are presented in Figure 3. These examples show that the observed conservation and the LMSs are biologically meaningful. Figure 3a presents the profile for the Haemophilus influenzae tyrosyl-tRNA synthetase (SwissProt P43836). The largest peaks correspond to the class-I tRNA synthetase specific HIGH and KMSKS regions (Eriani et al., 1990). Other LMSs are less conserved (smaller peaks) and are associated with other specific properties.

For multi-domain proteins, the profile typically presents several areas where the LMSs are grouped. These areas are separated by regions where only a few marginal LMSs occur. This situation is exemplified in Figure 3b presenting the profile of conservation of the Saccharomyces cerevisiae GAL4 regulatory protein (SwissProt P04386). The zinc finger domain Zn2-Cys6, which is the family signature, is clearly visible at the N-terminal end of the profile as a large peak. The GAL4 putative inhibitory domain (Poch, 1997) corresponds to the central peaks of the profile and is well separated from the zinc finger domain.
by a region of low conservation with very few LMSs. There is a good correlation between the LMSs and the motifs described by Poch (1997). However, the predictions of motifs III, VII and VIII are very weak. The C-terminal end of the profile does not show any strong conservation and only a very few weak LMSs are predicted.

The profile of conservation for the human multifunctional aminoacyl-tRNA synthetase SYEP_HUMAN (SwissProt P07814) is shown in Figure 3c. This aminoacyl-tRNA synthetase includes two different functions: the N-terminal region contains a glutamyl-tRNA synthetase function (class-I aminoacyl-tRNA synthetase) while the C-terminal region presents a prolyl-tRNA synthetase function (class-II aminoacyl-tRNA synthetase). These two regions can be seen on the profile as two wide areas with high peaks and several LMSs. They are separated by lower conservation regions with fewer LMSs. For each function, strong predicted LMSs correlate with the typical motifs of the corresponding aminoacyl-tRNA synthetase class as described by Eriani et al. (1990).

Concentration of true positives among best ranked similar sequences

Usually, only the highest scoring sequences with very low Expect values in a Blast search are manually examined. However, distant true homologues may diverge greatly from the query sequence, sharing some similarity only in conserved motifs, most of the sequence between these motifs being highly divergent. Such distantly related sequences may thus be overlooked as they have very high Expect values and are ranked very low by Blast. The ability of Ballast to identify these remote homologues within Blast results has been tested and compared with the PsiBlast program using the SCOP PDB40D-J database as a test set. This database was designed to be used in the comparison of a number of database searching programs (Park et al., 1998). The PDB40D-J contains 935 proteins, 219 of which have no homologues within the database at the superfamiliy level. The families represented contain from 2 to 42 structurally related members having pairwise identities less than 40%.

Each sequence from the PDB40D-J was used as a query sequence to perform the BlastP search directly in the database itself with default parameters. On the other hand, PsiBlast and Ballast both require information from multiple intermediate sequences in order to elaborate their scoring scheme. A larger database is therefore necessary to build an accurate position-specific scoring matrix (PSSM) for PsiBlast and to determine LMSs for Ballast. For each sequence from PDB40D-J, we first computed from a single BlastP search in SwissProt 38: (i) the Ballast conservation profile and the corresponding LMSs, (ii) the PsiBlast PSSM. The results from each
BlastP search on PDB40D-J were then processed by Ballast in order to rescore the BlastP hits according to the corresponding LMSs built on SwissProt 38. Similarly, the initial PsiBlast PSSMs built on SwissProt 38 were used to search the PDB40D-J database for new homologues until convergence.

To display and compare the results of these tests we used ‘coverage versus error’ plots based on Brenner et al. (1998). The Blast scores are not absolute as they depend on the length of the query, the number and repartition of the Blast hits used to build the conservation profile as well as the size and number of LMSs. The coverage is therefore defined for every search as the fraction of structurally determined homologues that have scores above a selected threshold. The error is defined as the number of non-homologous pairs above the same threshold. Figure 4 presents the mean of the coverage achieved at a given error rate for BlastP, PsiBlast and Ballast results.

As expected, inspection of the figure shows that PsiBlast at convergence finds on average a higher proportion of true positives at any given error rate than both BlastP and Ballast. This proportion ranges from 34.6% (confidence interval CIα=95%: 31.8%–37.4%) at the level of one non-homologue detected up to 59.2% (CIα=95%: 54.5%–63.9%) for more than 20 non-homologues detected. Comparatively, the default BlastP search only identifies from 30.9% (CIα=95%: 28.3%–36.4%) of true positives for one non-homologue detected to 51.9% (CIα=95%: 46.1%–57.8%) for over 20 non-homologues. These results clearly confirm that the PsiBlast iterative process improves the detection of distantly related sequences by Blast.

At an error rate of less than five non-homologues detected, the coverage for Ballast is roughly equivalent to the Blast one. However, BlastP reaches a plateau of 33.4% (CIα=95%: 32.4%–34.4%) of true homologues between 5 and 15 non-homologues while the proportion of true homologues detected by the Ballast continues to increase up to 40.2% (CIα=95%: 35.5%–45.0%). Over 15 non-homologues detected the coverage for Ballast remains roughly constant while the BlastP coverage increases and even reaches a higher value than Ballast. These data reflect the fact that in BlastP searches some distantly related sequences are given a very high Expect value and thus are rejected after a number of false positives. Presently, on the PDB40D-J dataset, an average of 15 non-homologues is returned before distant structural homologues are detected. On the other hand, these distant homologues are detected at lower error rates by Ballast, thus indicating the ability of the method to concentrate true positives among the top ranking sequences. However, the maximum coverage achieved by Ballast is lower than the maximum value reached by BlastP, indicating that some distant homologues are not detected at all by Ballast. This limitation is due to the fact that, since Ballast examines the hits produced by Blast to rescore the sequences, distantly related sequences that do not produce Blast hits overlapping the LMSs are rejected.

These results demonstrate that concentration of distantly related sequences can be achieved by processing one single BlastP search in less than a few seconds without requiring any additional information. However, since Ballast is a post-processing program and only rescoring sequences that were already proposed by Blast below a given Expect value (5000 is a reasonable threshold here), it cannot identify as many true positives as PsiBlast that iteratively searches the database. When searching a large total database though, PsiBlast is more time consuming since it searches iteratively the whole database until convergence. Furthermore, Ballast is also able to use the LMS scoring scheme from a BlastP search to process a TBlasN search with the same query sequence.

**Use of the GUI: Sorting sequences according to selected LMSs**

In order to demonstrate the use of the GUI to sort sequences according to their similarity to user-selected
Table 1. (a) SYEP_HUMAN (P07814): this sequence is a multifunction t-RNA synthetase (see the text). When the whole protein is used to search the SwissProt database, class-I and class-II aminoacyl t-RNA synthetases are returned. After selection of the LMSs in the class-I part of SYEP_HUMAN, Ballast returns essentially class-I sequences. Alternatively, when the class-II portion of the sequence is selected, Ballast returns class-II sequences. These results are similar to those obtained by a new BlastP search performed with the corresponding portion of the query sequence. (b) GAL4 YEAST (P04386): the GAL4 family is characterized by the Zn2-Cys6 DNA binding domain. Some members also share a putative inhibitory domain consisting of eight conserved motifs (Poch, 1997). When using the full-length GAL4_YEAST sequence, selecting the LMSs predicted in the putative inhibitory domain discriminates between the true positives that contain this domain and those that do not. This discrimination is not significantly different from that obtained with a new BlastP search with the putative inhibitory domain only.

<table>
<thead>
<tr>
<th>(a) SYEP_HUMAN (P07814)</th>
<th>(b) GAL4 YEAST (P04386)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class-I</td>
<td>Class-II</td>
</tr>
<tr>
<td>SYEP_HUMAN</td>
<td>91</td>
</tr>
<tr>
<td>GLU selection(a)</td>
<td>-</td>
</tr>
<tr>
<td>PRO selection(b)</td>
<td>-</td>
</tr>
<tr>
<td>GLU region(c)</td>
<td>85</td>
</tr>
<tr>
<td>PRO region(d)</td>
<td>8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(b) GAL4 YEAST (P04386)</th>
<th>Zn2C6 Inhibitory</th>
<th>Inhibitory domain selection</th>
<th>Inhibitory domain(e)</th>
<th>Zn2C6 Inhibitory</th>
<th>Inhibitory</th>
<th>First false positive</th>
<th>BlastP</th>
<th>Ballast</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAL4 YEAST (230-600)</td>
<td>66</td>
<td>53</td>
<td>37</td>
<td>37</td>
<td>-</td>
<td>22</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>Inhibitory domain(f)</td>
<td>38</td>
<td>38</td>
<td>38</td>
<td>38</td>
<td>-</td>
<td>32</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(a\) Selection in Ballast GUI of the GLU region from SYEP_HUMAN
\(b\) Selection in Ballast GUI of the PRO region from SYEP_HUMAN
\(c\) Search with the GLU region from SYEP_HUMAN
\(d\) Search with the PRO region from SYEP_HUMAN
\(e\) Search with the putative inhibitory domain from GAL4 YEAST (230-600)

LMSs, we performed a search using the human multifunctional aminoacyl-tRNA synthetase SYEP_HUMAN (SwissProt P07814). The class-I and class-II family signatures found in this sequence are mutually exclusive (Eriani et al., 1990). The search with the whole sequence yields mixed results with aminoacyl-tRNA synthetases from both families. Of the 145 aminoacyl-tRNA synthetase sequences with a BlastP Expect value less than 5000, 91 were from the class-I, 52 from the class-II and 2 were multifunctional belonging to both classes. If we select the LMSs predicted in the glutamyl-tRNA synthetase region (class-I) with the GUI and sort the sequences according to their similarity to these LMSs, Ballast returns 81 sequences from class-I, 2 from both classes, and only 2 from class-II. If we select the LMSs predicted within the prolyl-tRNA synthetase region (class-II), Ballast yields 41 class-II sequences, 2 mixed sequences, and 4 class-I sequences. We also compared these results with those obtained from BlastP searches involving only the glutamyl region or the prolyl region. For these searches, in both BlastP and Ballast results, the number of true positives was equivalent to the results obtained after selection of the corresponding region in the full-length search. However, as demonstrated earlier, there were more true positives among the first reported sequences in Ballast results than in BlastP.

Another example of the GUI is given by a search with the Saccharomyces cerevisiae GAL4 regulatory protein (SwissProt P04386). There were 66 true positives with an Expect value less than 5000 in the BlastP full-length search, 53 of them having a putative inhibitory domain described by Poch 1997. In the Ballast results, none was missing and the first 56 reported sequences were all true positives compared to the first 23 sequences in the BlastP search. When the LMSs in the putative inhibitory domain are selected, Ballast returns 37 true positives, all of them containing the requested domain and the first false positive is ranked 29th. These results are not very different from those obtained with a new BlastP search using only the putative inhibitory domain from residues 230 to 600. In this case, 38 true positives were found with an Expect value less than 5000 and none was left out by Ballast. This makes only one more true positive sequence than the Ballast results after domain selection in the full-length search. Interestingly enough, the only additional sequence, SEF1 YEAST (P34228), produced a BlastP alignment only in the Zn2-Cys6 domain when the search was performed with the full-length GAL4 sequence. It was therefore impossible for Ballast to score this sequence positively when only the putative inhibitory domain was selected as there was for this sequence no
alignment overlapping the predicted LMSs in the putative inhibitory domain. On the other hand, with the putative inhibitory domain search, SEF1 YEAST produced BlastP alignments covering motifs I, IV, V and VI and Ballast could thus score this sequence significantly and raise it from rank 144 to rank 28.

These results, summarized in Table 1, show that Ballast is clearly able to discriminate between different families or sub-families according to specific LMSs, thus allowing deep exploration and analysis of the BlastP results without the need for any additional Blast search. Furthermore, after Ballast post-processing, the true positives are more concentrated within the first reported sequences in the original Blast output. However, the Expect values of HSPs depend upon the query sequence. The alignments returned by Blast when the full-length query sequence is used are therefore not always the same as those returned when only a portion of the sequence is used. For instance, some regions not identified with the full-length sequence may be detected with the truncated query sequence. This may affect the ability of Ballast to identify distant similarities though the results are generally equivalent. However, the number of true positives returned by a partial query sequence search is not very different from the number of true positives returned by a partial query sequence after Ballast post-processing, the true positives are more or sub-families according to specific domains within the query sequence. Grouping the predicted motifs according to how well they correlate to certain functions may provide a means to annotate the query sequence.

Conclusion

We present a new Blast post-processing program that not only provides a convenient presentation of Blast results but is also able to bring added value to Blast searches. The prediction of LMSs provides the user with a new scoring scheme leading to powerful exploration of BlastP and TBLASTN results without requiring additional database searches. We have shown that sorting similar sequences according to selected LMSs can discriminate between sub-families and improve identification of specific domains within the query sequence. Grouping the predicted motifs according to how well they correlate to certain functions or protein families can provide a means to annotate the query sequence.

Acknowledgements

We would like to thank J.C.Thierry, R.Ripp and O.Lecompte for helpful discussion and D.Moras for his continuous support. This work was supported by institute funds from INSERM, CNRS and the Fond de Recherche Hoechst Marion Roussel.

References


BLAST application for interactive or automated sequence analy-
Zhang,Z., Schäffer,A.A., Miller,W., Madden,T.L., Lipman,D.J.,
searches using patterns as seeds. Nucleic Acids Res., 26, 3986–
3990.