Identification of Protein Coding Regions by Database Similarity Search

Warren Gish and David J. States[1]

Running title: Database search by simultaneous translation and alignment

Keywords: database search algorithms, sequence errors

National Center for Biotechnology Information
National Library of Medicine
Building 38A, Room 8N806
8600 Rockville Pike
Bethesda, MD 20894-0001

[1]Present address:
Institute for Biomedical Computing
Washington University
700 S. Euclid Avenue
St. Louis, MO 63110

Correspondence should be addressed to W.G.

Summary

Sequence similarity between a translated nucleotide sequence and a known biological protein can provide strong evidence for the presence of a homologous coding region, and such similarities can often be identified even between distantly related genes. The computer program BLASTX performed conceptual translation of a nucleotide query sequence followed by a protein database search in one programmatic step. The BLAST search algorithm combined with Karlin-Altschul statistics yields a predictable selectivity that has been parameterized. We characterized the sensitivity of BLASTX recognition to the presence of substitution, insertion and deletion errors in the query sequence and to sequence divergence. Reading frames were reliably identified in the presence of 1% query errors, a rate that is typical for primary nucleotide sequence data. BLASTX is appropriate for use in moderate and large scale sequencing projects at the earliest opportunity, when the data are most prone to containing errors.

Introduction

Most primary sequence data is obtained as nucleic acid, while much of the biological interest lies in the encoded protein. Inference of likely protein coding regions is often based on statistical features, such as codon usage[1,2,3,4] and the locations of putative splice site signals[5], but significant false positive rates are common. In contrast, similarity between a conceptually translated nucleotide sequence and a known protein sequence may be highly significant statistically, which suggested a more discriminating approach to inferring coding potential. We present a software tool, BLASTX, that can be used to probe a nucleotide sequence directly for the presence of protein coding regions by identifying segments that encode significant similarity to members of a protein sequence database, a technique that may also be used to assign putative function.

Molecular sequence determination is a complex process, the course of which may be significantly altered when homologs or related sequences are identified by database search. Search tools should therefore be amenable to use in early stages of a sequencing project, even though the data may be more prone to containing errors. Protein-protein comparison methods are important in nucleotide sequencing projects because distant evolutionary relationships which appear to be merely coincidental at the nucleotide sequence level can be meaningfully discerned at the protein sequence level[6]. The amino acid alphabet is expected to be superior to the nucleic acid when sequences are diverged by as few as 50 amino acid PAMs, due to degeneracy in the genetic code and functional constraints on protein structure[6]. For example, the coding sequences for cystic fibrosis transmembrane regulatory protein[7] and the human multiple drug resistance protein[8], two members of the family of ATP-binding proteins, share at best a region of 70% identity in 60 nucleotides (only marginally significant statistically), while the two sequences are highly similar at the protein level ($P<10^{-13}$).

The sequence data available early in a project may not only be more prone to containing errors, but may also be more abundant. These factors cause increased dependence to be placed on the speed, sensitivity, and convenience of the software tools employed. The selectivity of a search method, or its ability to rank matches by precise statistical criteria, takes on greater importance as well. When commingled with biologically relevant matches, purely chance matches add to the tedium of data analysis and may obscure the presence of the former. These issues are addressed by the software tool presented here.

The BLASTX program has been successfully employed to identify likely protein coding sequences in thousands of partial cDNA sequences from human brain tissue[9] and in genomic cosmid clones from *Caenorhabditis elegans*[10]. BLASTX allowed protein-protein comparisons to be considered when only uncharacterized nucleotide query sequence was available. The program conceptually translated query sequences in all six reading frames (three on each strand) and compared each of these full-length translation products with a comprehensive protein sequence database in a single pass. Non-coding sequence tends to yield alignments of marginal significance at best that were selectively excluded from BLASTX output on the basis of a statistically determined cutoff score[11,13]. In Monte Carlo simulations, BLASTX was effective at recognizing statistically significant sequence similarities in the presence of 1% data errors, an error rate that is typical for raw molecular sequence data. Homologs were identifiable by the program in approximately 17% of the raw, human EST sequences examined[9].

BLASTX offers distinct advantages over related tools such as BLASTP[13], by combining the conceptual translation and database search procedures into a single programmatic step and by collating the results from searches with all six reading frames into a single report; furthermore, with its knowledge of the matches appearing in all reading frames, BLASTX applied Poisson statistics to combine marginally significant matches in different frames from the same strand to yield higher estimates of statistical significance than would be obtained in separate BLASTP searches.

The converse problem, that of searching with protein query sequence against a translated nucleotide sequence database, is addressed by other programs such as TFASTA[14] and PATMAT[15,16]. Coercing TFASTA to perform the converse of its designed function would be impractical for even modest sized protein databases, as each protein sequence would require a separate invocation of the program. PATMAT does however provide the flexibility of using nucleic acid sequence as the query in a search, translating the nucleic acid in all six reading frames, and comparing the full-length products against either a standard protein sequence database or a concise database of protein sequence “blocks”[16]. Searches against a blocks database can be practically performed with lower thresholds of significance than against a comprehensive protein database, thus permitting increased search sensitivity; the smaller size of a blocks database may yield shorter search times, as well. In contrast, the use of Karlin-Altschul statistics[11] by BLASTX permits precise significance estimates and thresholds to be established; and for comprehensive database searches, BLASTX is several fold faster.

Acceptable computational efficiency is required for practical use in modest to large scale sequencing applications. Without the aid of an automated sequencer, an individual experimentalist may generate daily ten to twenty runs of raw sequence, each 300 or more nucleotides in length and totalling several kilobases; automated sequencing laboratories may exceed this throughput by several fold. Practical database search tools must be able to maintain pace with the laboratory, using available computational resources and without overburdening the experimentalist with random, uninformative matches that must be individually evaluated in subsequent steps. If the sense strand or reading frame has not been clearly established, it may be necessary to compare all six reading frames in these data against a comprehensive database of published sequence to find significant similarities. BLASTX was found effective in this capacity due to its speed, sensitivity, selectivity and convenience; additional speed was yet achieved by employing parallel processing on a common multi-purpose, multi-processor computer platform.

Results and Discussion

To search a protein sequence database with a nucleotide query sequence, BLASTX translated the query in all six reading frames, built one neighborhood table[13] containing pointers into each of these reading frames, and searched the database in a single pass. An example of the effectiveness of this strategy is shown in Figure 1, the partial BLASTX output generated using the gene sequence for *Saccharomyces cerevisiae* tetrahydrofolate synthase[17] as a query to search the PIR® protein sequence database[18]. The enzyme itself and several closely related homologs were seen at the top of the list of matches, along with the previously noted TEC1 protein encoded in flanking sequence[19], which were translated in positive- or plus-strand reading frames. These matches were followed by a set of highly significant ($P<10^{-56}$) alignments with several L19 ribosomal protein sequences[20], but the reading frame involved in these alignments resided on the opposite strand in an unannotated segment of the query sequence. It appeared very likely that the yeast homolog of the L19 ribosomal protein was cloned and sequenced incidentally. Two fragments of the L19 protein have been identified by direct peptide sequencing[21], but the segment identified here was more extensive and provided the encoding nucleotide sequence.

The reliability of simultaneous translation and alignment is described in Table 1, for which comparative searches were performed with BLASTP and BLASTX using sequences for human cdc2 kinase (CDC2)[22] and the human cystic fibrosis transmembrane regulatory protein (CFTR)[7] as queries. All of the alignments scoring over 80 with direct protein searches of the protein database (BLASTP) were identified by BLASTX using mRNA queries, and no erroneous alignments were introduced (Table 1, column 3). All 258 of the distantly related CDC2 homologs found by BLASTP (maximum HSP scores between 70-79, see Methodology) were correctly identified by BLASTX without introducing false positive matches. All but one of the 77 distantly related CFTR homologs found by BLASTP were correctly identified by BLASTX, again without introducing false positive matches. For alignments with maximum HSP scores of less than 70, many discrepancies were seen between the BLASTP and BLASTX searches. These alignments were not statistically significant and the discrepancies can therefore be attributed to random events.

A. The cost of missing data -- reading frame and orientation

Karlin and Altschul derived an analytic expression for the probability of finding an alignment above any set score starting at given positions in two sequences[11]. The effective number of such starting points in a database search is proportional to the product of the query sequence length and the size of the database. In BLASTX searches, the reading frame and orientation of the query sequence were unknown prior to the search. In querying with all six reading frames, BLASTX was searching a space roughly six times the size of a single-frame BLASTP search, thus decreasing the significance of any alignments found by a factor of about 6, or by a score of $\log_2(6) \approx 2.6$ bits. In practice, the impact of this ambiguity was minimal (Table 1). To ensure a manageable size for the neighborhood word list (see Methodology) with nucleotide query sequences of potentially great length, BLASTX used a marginally higher neighborhood word score threshold, T, as compared to the value used by BLASTP. This resulted in a slight relative reduction in sensitivity of BLASTX in its detection of lower-scoring alignments and accounts for the BLASTP-BLASTX difference seen with CFTR (Table 1).

B. Sensitivity to sequence data errors

Nucleotide sequences, like any experimental data, can be expected to contain errors, and historical surveys suggest that the GenBank® sequence database[23] contains errors at an overall rate of approximately 1 in 300 nucleotides, of which about half are insertions or deletions (indels)[24]. For MSP-based search algorithms such as BLAST (see Methodology), sequencing errors may alter a statistically significant MSP and prevent its recognition in two ways: indels in one or both sequences may break the MSP into two or more shorter segments that individually score below the threshold of significance; and substitution errors may decrease the score of the MSP to a value below the cutoff and perhaps alter its end-points.

The degradation of an alignment score by individual substitution errors is often small relative to the cutoff score for reporting matches, so little or no significant loss of search sensitivity is expected even at error rates encountered in raw molecular sequence data (<1 substitution per 100 amino acids)[25]. In contrast, frameshift sequencing errors have a significant impact on search sensitivity, particularly when the query and target sequences are highly diverged[25]. Even in the absence of sequencing errors, naturally occurring indel mutations can prevent homolog detection. With increasing divergence, the expected contribution to an alignment score by each pair of aligned residues falls while the expected length of a statistically significant HSP rises[12]. The increased length provides greater opportunity for indel mutations and frameshift errors to impart their deleterious effects on subsequent search sensitivity, a situation which is exacerbated by the tendency of indel mutations to appear more frequently with increasing PAM distance[26].

Frameshift sequencing errors are shown in Figure 2 to have reduced the probability that an MSP was long enough to accumulate a score detectable above random. Under optimal conditions of error-free data at 120 PAMs divergence, an alignment significant to at least 35 bits was detected with a probability of 94%; in the presence of 1% indel errors, the probability was 86%. Overall, sequencing errors at rates as high as 3 indels per 1,000 nucleotides made little difference in the probability of detection; but even with perfect sequencing data, homologs could not be reliably identified at 250 PAMs divergence, due in part to the use of the sub-optimal PAM120 scoring scheme[6,12]. Neglecting indel mutations and all sequencing errors, if the optimal, PAM250 matrix for scoring homologs diverged by 250 PAMs is used instead, the expected score between true homologs is about 0.36 bits per aligned residue pair,[12] or 36 bits over the entire modeled length of 100 codons; when the same homologs are scored with the PAM120 matrix, the expected score drops to 0.17 bits per aligned residue pair[12], or 17 bits over the entire length. This is 18 bits (or a factor of $2^{18} \approx 10^{5.4}$) below the threshold of 35 bits deemed necessary to achieve marginal significance in a typical database search (Fig. 2 legend), but due to a large variance in the distribution of segment scores (data not shown), the observed probability of detection at 250 PAMs divergence was >4% (Fig. 2A).

The empirical behavior of BLASTX in the presence of sequencing errors was assessed by comparing the results of BLASTX searches performed with 275 error prone cDNA sequence fragments (ESTs)[9,10,27,28] to the results obtained by BLASTP with the corresponding finished protein sequences used as queries against the SWISS-PROT database[29]. EST identification was originally based on searches of the NCBI non-redundant database[30]; the extended coverage of this database permitted more ESTs to be positively characterized. Figure 3 exhibits the effects had upon search performance by sequence errors and EST fragmentation. For approximately one quarter of the ESTs, all of the matches found by BLASTP were correctly identified by BLASTX. At the other end of the spectrum, in twelve of the 275 cases all of the matches found by BLASTP were missed by BLASTX. Manual inspection revealed that in most of these cases, failure to identify a homolog was a result of frameshift errors or truncation of the cDNA sequence in the EST, leaving little or no intact protein coding sequence. For the remaining three quarters of the ESTs, BLASTX correctly identified at least some of the matches found by BLASTP, thus permitting a homolog to be identified without resorting to the use of the latter program. In addition to the effects imposed by sequence errors and truncation, some BLASTP matches may have gone unreported by BLASTX because they fell just below the cutoff threshold for BLASTX yet just above the threshold for BLASTP, where the two score thresholds differ by the approximately 2.6 bits explained earlier; with the expectation cutoff of 0.01 that was used, this threshold effect should have been relatively small. As a caveat to this study, the 275 ESTs were chosen on the basis of previous, positive characterization by BLASTX, which constitutes some degree of selection for ESTs of higher quality--particularly with fewer frameshift errors and more complete coding regions. While BLASTX was responsible for the initial characterization, the results in Fig. 3 may tend to present a best-case distribution.

The results in Figures 2 and 3 show that inaccurate sequence data can degrade the sensitivity of database searches. As more error prone data enters the public databases, it becomes increasingly difficult and costly for other investigators to check their results. When BLASTX reports adjacent HSPs appearing in different reading frames on the same strand of the query, the possibility of frameshift errors--the errors which are of greatest concern--should be investigated and corrected as appropriate. While minimizing the error rate is one concern in a sequencing project, practical considerations often demand that error prone data be released to the public. This provides some of the impetus for maintaining a specialized, public repository of EST sequence data[27,28].

C. Caveats to the statistical model: some causes of misleading results

Several phenomena complicate the statistical analysis of similarity searches. These are independent of the algorithm used to perform the search, but must be considered in the interpretation of BLASTX output. Genomes contain local regions of strongly biased residue composition and reduced information content. Such regions of low complexity sequence are present in the public databases and may be present in a query sequence or its translations, such that a large number of high scoring but biologically uninformative alignments are observed. Inappropriately high statistical significance may be assigned to these alignments because local biases in amino acid composition are not encompassed by the random sequence model assumed in Karlin-Altschul statistics. The biological significance of matches against low complexity regions must, therefore, remain suspect.

At the next level of order above low-complexity sequences, complex but repetitive sequence elements, such as the human *Alu* family [31], are present in genomes and are particularly frequent in higher eukaryotic genomes. It is common for a query sequence to contain a segment derived from such a repetitive element, even if the query is a cDNA. Searches performed with such a query may produce alignments that appear statistically significant wherever members of the same repetitive sequence family are present in the database. Proteins may be partially encoded by repetitive sequence elements, as well. For example, the PIR® entries for human complement factor 5 (C5HU), decay accelerating factors 1 and 2 (A26359, B26359), the HLA-DR beta 1 precursor (S01441), and platelet glycoprotein IIb (A28411) all appear to contain translated *Alu* sequences[32]. BLASTX searches with a query sequence containing an *Alu* element may score highly against these database sequences, even if the *Alu* sequence in the query is not translated *in vivo* and no further homology exists outside of the repetitive element.

Repetitive sequence elements may be filtered from a query sequence by searching against a concise database of exceptional sequences that includes the six frame translations of *Alu*[32]. Segments of the query that match against the exceptions database can then be excluded from the full database search by masking with the IUB nucleic acid code N (for “any”); such masked segments were translated by BLASTX into the IUB amino acid code X (for “unknown”) and would not appear in alignments when the default PAM120 matrix was used. Low complexity sequences can be filtered by analyzing local residue composition (Wootton and Federhen, submitted) or by identifying stretches of short period, internal repeats in the amino acid translations (Claverie and States, submitted) and then masking these segments.

The results presented show that BLASTX was a computationally efficient tool for finding gene homologs without prior knowledge of the coding regions or reading frames in a nucleotide query sequence. Pseudogenes are true homologs and may achieve significant alignment scores, as well, but they can be distinguished from functional coding regions by other criteria, such as the presence of stop codons and the absence of promotor elements and splice sites. The BLAST algorithm was able to identify many related sequences even if the query sequence was error prone, but at lower sensitivity. The greatest impact of query errors on search performance was expected in comparisons between distantly related proteins, and this effect was confirmed empirically.

The BLAST algorithm identifies local regions of similarity which are ungapped. Multiple local regions of similarity may contribute to the overall score, but algorithms such as the dynamic programming approach of Smith and Waterman [33] may also provide increased sensitivity when insertion and deletion errors are present[25], albeit at increased computational cost. Even with that algorithm, however, gap initialization is often heavily penalized, such that a small number of insertion or deletion errors rapidly degrades the significance of an alignment.

Acknowledgments

We wish to thank Dr. Stephen Altschul for many hours of informative discussion and Drs. David Landsman, Mark Boguski, Jean-Michel Claverie, and David Lipman for their critical reading of the manuscript.

Methodology

A.Algorithms and Implementations

The BLAST algorithm approximates a well defined measure of local sequence similarity based on a matrix of similarity or substitution scores for all possible pairs of residues[13]. The algorithm identifies ungapped, aligned pairs of sequence segments with locally maximum scores which meet or exceed a parameterized cutoff score, *S*. These segments are referred to as “high-scoring segment pairs” (HSPs), and the highest scoring segment pair derivable from any two sequences is their maximal-scoring segment pair, or MSP. A program, BLASTX, based on this rapid, probabilistic algorithm, was used to find statistically significant HSPs between a translated nucleotide query sequence and a target protein sequence database. When an HSP was found, the analysis of Karlin and Altschul[11] was used to estimate the significance of its score (Equation 1 below).

No prior knowledge of the reading frame or direction was assumed by BLASTX; all possible reading frames in both orientations of the query sequence were translated into protein sequence using the standard genetic code, with eight other genetic codes available to choose from using a simple command line parameter. The PAM (point accepted mutation) amino acid substitution model was typically used for scoring similarity between peptide sequences[34], wherein identities and conservative replacements receive positive scores, and non-conservative replacements, *e.g.*, leucine for aspartic acid, receive negative scores. By default, BLASTX used a PAM120 matrix scaled to 0.5 bit per unit score[12].

Stop codons were not explicitly included in development of the PAM theoretical framework[34]. We chose to score alignments between amino acids and stop codons as equivalent to the least-favorable (most negative) substitution score observed between any two amino acids in the PAM matrix. All substitution scores were readily accessible to the user as rows and columns in the score matrix file read by the program and, as such, were user-modifiable. Alignments incorporating a stop codon could be effectively forbidden by applying a large negative penalty to any substitution for a stop codon, or scores might represent the log-odds that a stop codon resulted from experimental error.

The expected number of alignments scoring S or greater in a comparison between two random sequences of lengths m and n is

$$E = mnKe^{-\lambda S} \qquad \text{(EQ 1)}$$

where K and λ are parameters dependent on the amino acid compositions of the sequences[11]. For values less than about 0.1, E is often an acceptable approximation to P, the probability of occurrence of one or more matches scoring S or greater. In a true coding region, one reading frame may have a predicted amino acid composition typical for biologically occurring proteins, while the other reading frames exhibit anomalous compositions[1,3,2]. For this reason, BLASTX calculated separate K and λ values for each reading frame. Alignment scores are often cited here in units of bits (binary digits), such that their significance can be evaluated independently of the scale of the scoring system employed[12]. Each unitary increase in the bits of information corresponds to a factor of two increase in the statistical significance.

The BLAST algorithm operates in two successive stages, “neighborhood” word generation followed by the actual search, with an implicit trade-off in speed versus sensitivity imparted in the first stage[13]. A list of neighborhood words of length W is generated from consecutive, overlapping words of length W in the query sequence, using a specified scoring matrix. The neighborhood list contains all words which satisfy a threshold scoring parameter, T, when aligned with words in the query sequence. Raising T decreases the size of the neighborhood and, consequently, increases the search speed in the algorithm’s second stage, but at the expense of decreased sensitivity[13]. In BLASTX, the neighborhood word list was built from the conceptual translations of all six reading frames on both strands of the query sequence, and this word list was stored in a class of data structures known as a deterministic finite-state automaton (DFA)[35]. Depending on the scoring matrix and value chosen for T, a given word of length W in the query sequence may yield no neighborhood words. The fundamental BLAST algorithm was enhanced by including any query word in the DFA, as long as the score of the word when aligned with itself was positive.

During the second stage of the BLAST algorithm, the neighborhood words from the first stage are searched for in the database or “target” sequence; the presence of a neighborhood word match indicates the possible location of an HSP. Individual neighborhood word matches (or word “hits”) are extended in both directions along the matrix diagonal until the ends are reached or the cumulative alignment score falls from its maximum achieved value by a parameterized quantity X. In BLASTX and BLASTP, the initial word hits were found by streaming database sequences through the automaton; after hit extension, only those segment pairs whose scores met or exceeded a cutoff score, S, were reported to the user. Rather than choose a value for S explicitly, users often found it more natural to specify a maximum expected frequency of occurrence, E, for HSPs to be reported by the program. From the specified value of E and the relationship shown in Equation 1, BLASTX calculated the value for S necessary to achieve the desired level of significance, as a function of the length and amino acid composition of the query sequence in each reading frame, the length of the database, and the particular scoring matrix employed; a fixed set of amino acid frequencies characteristic of general protein databases was also used in these calculations[13].

The BLAST algorithm is heuristic but has the property that any desired sensitivity in the detection of MSPs may be obtained at the cost of increased computation time. For the particular parameter values used to obtain the BLASTX results presented here (word length W=3 with T determined by an *ad hoc* formula and a PAM120 matrix), the predicted sensitivity of BLAST in finding an MSP with a score of 32 bits or greater is very nearly 100%[13]. (In contrast, the program BLASTN for comparing nucleotide sequences uses a longer word length of 12 and builds its DFA from a very restricted set of neighborhood words--just the query words themselves--thus achieving high speed at the expense of sensitivity)[13].

Simultaneous translation and alignment strategies multiplied the effective size of the query sequence. Instead of searching a single reading frame translation of the query against the database, BLASTX searched a total of six reading frames, so the DFA used to implement the neighborhood search in BLASTX was roughly six times the size it would have been for one frame only. The search speed of the DFA itself was unaffected by the query length. Even for modest queries (>50 nucleotides or 100 amino acids) the rate limiting step was extension of the word hits found by the DFA. The number of such hits was expected to be proportional to the product of the query and database lengths. In one example, BLASTX searched a 20 million residue protein sequence database in 120 seconds for a 2,115 nucleotide (4,226 amino acids translated) query sequence (GenBank® 71 locus HUMBARR), or only about 6-fold longer than the 20 seconds required for BLASTP to search the same database with the encoded 413 amino acid protein sequence, a sequence roughly one-tenth as long. This modest discrepancy in relative execution times is largely attributable to the use by BLASTP of a lower (more sensitive) neighborhood word score threshold, T, resulting in an increased number of word hits and subsequent word hit extensions.

B. Parallel processing of BLAST calculations

Multiprocessing capabilities were exploited when either BLASTX or BLASTP executed on Silicon Graphics, Inc. (SGI) PowerIRIS platforms, including the models 4D/240 and 4D/480. These systems provided respectively 4 and 8 processors, for which real-time processing improvements of about 3- and 6-fold were observed. For example, in probing a 703 nucleotide mRNA sequence encoding bovine lactalbumin against release 32.0 of the PIR® database with BLASTX, 0.2 seconds of CPU time in serial mode was required to build the DFA, followed by 21 seconds real time in 8-processor parallel search mode, and finishing with 3 seconds of single-processor time to report the results, yielding an overall efficiency of 86%.

Construction of the DFA to recognize neighborhood words was performed serially on one processor. The search was then run in parallel with individual processors accessing the same DFA (as read only memory) to search dynamically assigned segments of the database. While searching, each processor compiled a local list of HSPs in common heap storage, with only intermittent interprocessor communication required; the efficiency of multiple processor utilization during this phase approached 100%. After all of the processors had searched their database segments, the program reverted to single processor mode and the individual HSP lists were merged into a single list that was then sorted by statistical significance. No interprocessor communication overhead was incurred for merging the lists because of their placement in common heap storage.

To facilitate repetitive or concurrent searches, BLASTX and BLASTP optionally searched database files loaded semi-permanently in shared memory, thus significantly reducing the overhead of disk I/O and contention for disk-based resources. Shared memory was managed using AT&T UNIX® System V interprocess communication (IPC) facilities, including semaphores and message queues to arbitrate access and signal updates to the memory-resident database files.

C. Analysis of sensitivity to sequence errors

The probability that an MSP is recognizable by the BLAST algorithm using an error-prone query sequence was estimated by sampling using randomly generated peptide sequence data constrained to an amino acid composition typical for that observed in protein sequence databases[34]. For each query sequence, a target sequence was generated using the transition probabilities underlying the PAM model[34,12]. Insertion and deletion error sites were then generated randomly in the query sequence and alignments of the remaining continuous segments were scored with a PAM120 matrix. 2,000 trials were performed at each integral value of PAM from one through 250, and a least-squares curve was fit through the resulting data. Calculations were implemented in the S statistical analysis language (AT&T).

D. Software compatibility and availability

The programs described here were implemented in the C programming language and are compatible with many UNIX®-based computing platforms. Complete, public domain source code is available via anonymous ftp on ncbi.nlm.nih.gov (130.14.20.1) or by contacting the corresponding author.

References

1. Staden, R. & McLachlan, A. D. Codon preference and its use in identifying protein coding regions in long DNA sequence *Nuc. Acids Res.* **10**, 141-156 (1982).

2. Fickett, J. W. Recognition of protein coding regions in DNA sequences *Nuc. Acids Res.* **10**, 5303-5318 (1982).

3. Staden, R. Finding protein coding regions in genomic sequences *Methods Enzymol.* **183**, 163-180 (1990).

4. Uberbacher, E. C.& Mural, R. J. Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach *Proc. natn. Acad. Sci. U.S.A.* **88**, 11261-11265 (1991).

5. Fields, C. A. & Soderlund, C. A. gm: a practical tool for automating DNA sequence analysis *Comput. Appl. Biosci.* **6**, 263-270 (1990).

6. States, D. J., Gish, W. & Altschul, S. F. Improved sensitivity of nucleic acid database searches using application-specific scoring matrices *Methods: A Companion to Methods Enzymol.* **3**, 66-70 (1991).

7. Riordan, J. R. *et al.* Identification of the cystic fibrosis gene: cloning and characterization of complementary DNA *Science* **245**, 1066-1073 (1989).

8. Chen, C.-J. *et al.* Genomic organization of the human multidrug resistance (MDR1) gene and origin of P-glycoproteins *J. Biol. Chem.* **265**, 506-514 (1990).

9. Adams, M. D. *et al.* Sequence identification of 2,375 human brain genes *Nature* **355**, 632-634 (1992).

10. Sulston, J. *et al*. The C. elegans genome sequencing project: a beginning *Nature* **356**, 37-41 (1992).

11. Karlin, S. & Altschul, S. F. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes *Proc. natn. Acad. Sci. U.S.A.* **87**, 2264-2268 (1990).

12. Altschul, S. F. Amino acid substitution matrices from an information theoretic perspective *J. Mol. Biol.* **219**, 555-565 (1991).

13. Altschul, S. F., Gish, W., Miller, W. Myers, E. W. & Lipman, D. J. Basic local alignment search tool *J. Mol. Biol.* **215**, 403-410 (1990).

14. Pearson, W. R. & Lipman, D. J. Improved tools for biological sequence comparison *Proc. natn. Acad Sci U.S.A.* **85**, 2444-2448 (1988).

15. Henikoff, S., Wallace, J. C., & Brown, J. P. Finding protein similarities with nucleotide sequence databases *Methods Enzymol.* **183**, 111-132 (1990).

16. Wallace, J. C. & Henikoff, S. PATMAT: a searching and extraction program for sequence, pattern and block queries and databases *Comp. Appl. Bio. Sci.* **8**, 249-254 (1992).

17. Shannon, K. W. & Rabinowitz, J. C. Isolation and characterization of the Saccharomyces cerevisiae MIS1 gene encoding mitochondrial C1-tetrahydrofolate synthase *J. Biol. Chem.* **263**, 7717-7725 (1988).

18. Barker, W. C., George, D. G. & Hunt, L. T. Protein sequence database *Methods Enzymol.* **183**, 31-49 (1990).

19. Laloux, I., Dubois, E., Dewerchin, M. & Jacobs, E. TEC1, a gene involved in the activation of Ty1 and Ty1-mediated gene expression in Saccharomyces cerevisiae: cloning and molecular analysis *Mol. Cell. Biol.* **10**, 3541-3550 (1991).

20. Chan, Y. L. *et al.* The primary structure of rat ribosomal protein L7. The presence near the amino terminus of L7 of five tandem repeats of a sequence of 12 amino acids *J. Biol. Chem.* **262**, 1111-1115 (1987).

21. Otaka, E., Higo, K. I. & Itoh, T. Isolation of seventeen proteins and amino-terminal amino acid sequences of eight proteins from cytoplasmic ribosomes of yeast *Mol. Gen. Genet.* **191**, 519-524 (1983).

22. Lee, M. G. & Nurse, P. Complementation used to clone a human homologue of the fission yeast cell cycle control gene cdc2 *Nature* **327**, 31-35 (1987).

23. Burks, C. *et al.* GenBank: current status and future directions *Methods Enzymol.* **183**, 3-22 (1990).

24. Krawetz, S. A. Sequence errors described in GenBank: a means to determine the accuracy of DNA sequence interpretation *Nuc. Acids Res.* **17**, 3951-3957 (1989).

25. States, D. J. & Botstein, D. Molecular sequence accuracy and the analysis of protein coding regions *Proc. natn. Acad. Sci. U.S.A.* **88**, 5518-5522 (1991).

26. Gonnet, G. H., Cohen, M. A. & Benner, S. A. Exhaustive matching of the entire protein sequence database *Science* **256**, 1443-1445 (1992).

27. Boguski, M. S. dbEST, a database of expressed sequence tagged sites (National Center for Biotechnology Information, National Library of Medicine, Bethesda, MD 20894-0001, Internet electronic mail: boguski@ncbi.nlm.nih.gov, 1992).

28. Update on expressed sequence tag database. *NCBI News* **1**(3) 6 (1992).

29. Bairoch, A. & Boeckmann, B. The SWISS-PROT protein sequence data bank. *Nuc. Acids Res.* **20**, 2019-2022 (1992).

30. *Entrez* CD ROM Pre-release 6 (National Center for Biotechnology Information, National Library of Medicine, Bethesda, MD 20894-0001, 1992).

31. Rubin, C. M., Houck, C. M., Deininger, P. L., Friedmann, T. & Schmid, C. W. Partial nucleotide sequence of the 300-nucleotide interspersed repeated human DNA sequences *Nature* **284**, 372-374 (1980).

32. Claverie, J.-M. Identifying coding exons by similarity search: Alu-derived and other potentially misleading protein sequences *Genomics* **12**, 838-841 (1992).

33. Smith, T. F. & Waterman, M. S. Identification of common molecular subsequences *J. Mol. Biol.* **147**, 195-197 (1981).

34. Dayhoff, M. O., Schwartz, R. M. & Orcutt, B. C. in *Atlas of Protein Sequence and Structure* (ed. Dayhoff, M. O.) **5**(3), 345-352 (Natl. Biomed. Res. Found., Washington, D.C., 1978).

35. Hopcroft, J. E. & Ullman, J. D. *Introduction to automata theory, languages, and computation,* pp. 42-43 (Addison-Wesley Publishing, Reading, MA, 1979).

Table 1. The effect of search strategy on search sensitivity

The sensitivity of searches performed using BLASTP and BLASTX were compared in searching release 22.0 of the SWISS-PROT database[29]. Two sample query sequences were used, human cdc2 kinase (CDC2)[22], and the cystic fibrosis transmembrane regulatory protein (CFTR)[7], a large protein with many distant homologs but few closely related ones. For both genes, BLASTP searches were performed using the protein sequence from SWISS-PROT and BLASTX searches were performed using the corresponding cDNA (mRNA) sequence from the GenBank® database. Low complexity and repetitive sequences were eliminated from the query sequences by pre-filtering with XNU (Claverie and States, submitted). Alignments were scored with the PAM120 matrix[34,12]. The number of matches for each score interval are reported.

Figure 1. Sample BLASTX output

A yeast MIS1 tetrahydrofolate synthase gene sequence[17] (GenBank® accession number J03724) was used as a BLASTX query sequence against release 32 of the PIR® database[18]. The complete list of one-line descriptions of matching database sequences is shown for a probability cutoff of 0.1. The reading frame of the single highest-scoring segment is indicated alongside its score, followed by the Poisson probability for the most significant cluster of segments (which can conceivably involve segments in other reading frames, but in this example never did). In addition to the expected synthase protein alignments, highly significant alignments between several L19 ribosomal proteins[20] (data not shown) and an unannotated region of the query sequence were identified. Only a short fragment of the yeast L19 sequence has been published[21]. The presence of TEC1 coding sequence located 5' to the MIS1 gene has been noted[19]. Residue coordinate numbers for the query sequence are given with respect to the original nucleotide query sequence. Cpu times were reported in seconds for user (u), system (s), and the total (t) of user + system for one of the 8 processors.

Figure 2. Probability of finding an MSP in the presence of indel errors and mutations

The probability that a query sequence 300 nucleotides (100 codons) long would retain at least one ungapped alignment of sufficient length to permit its identification in a database search was plotted versus PAM distance between the encoded amino acid sequence and a homolog for alignments of (A) at least 35 bits desired significance and (B) at least 45 bits desired significance. Frameshift errors were introduced at rates of 0, 0.001, 0.003, 0.01, 0.03 and 0.1 per nucleotide, as indicated. Naturally occurring frameshift mutations were modeled at a rate of $5x10^{-5}$ per nucleotide per (amino acid) PAM; this rate was chosen by extrapolating from results of multiple protein sequence alignments (data not shown; ref. [26]). Scoring was performed in all cases with a PAM120 substitution matrix. The value of K (Methodology, Equation 1) in the Karlin-Altschul analysis[11] is typically about 0.2, so a score of 45 bits corresponds to a less than 1 in 10,000 chance of finding an alignment of equal or greater significance, when comparing a query of length 750 nucleotides in all six reading frames against a protein sequence database of 10 million residues; and 35 bits corresponds to about a 1 in 10 chance under the same conditions, or a marginal threshold for statistical significance[12].

Figure 3. Comparison of EST and homologous protein database searches

For each of the 275 non-*Alu* containing cDNA sequences in pre-release 1.0 of the dbEST database[27,28] for which a homolog was identified by a BLASTX search of the NCBI non-redundant database[30], the results of a BLASTX search against release 22.0 of SWISS-PROT[29] were compared to a BLASTP search using the highest scoring protein homolog as the query. Both the protein queries and the protein database sequences were pre-filtered with the XNU program (Claverie and States, submitted) to eliminate improbable residue clusters and short period repetitive sequences. Based on the sizes and compositions of both the database and the queries, only matches with less than a 1% chance of random occurrence[11] were counted. For each EST, the figure shows the fraction of the BLASTP sequence similarities which were identified by BLASTX, after sorting by this fraction. The abscissa is labeled with both the EST index number and the fraction of the total number of ESTs.

.

Table 1: Comparison of BLASTX and BLASTP database searches to identify ESTs

Query	Maximum	Number of Database Sequence Matches		
Sequence	HSP Score	BLASTX and BLASTP	BLASTX Only	BLASTP Only
CDC2	40-49	835	1850	37
	50-59	307	75	0
	60-69	282	1	0
	70-79	258	0	0
	80-89	210	0	0
	90-99	124	0	0
	100-109	65	0	0
	110-119	53	0	0
	≥120	17	0	0
CFTR	40-49	295	132	2705
	50-59	231	196	6
	60-69	92	4	0
	70-79	76	0	1
	80-89	56	0	0
	90-99	34	0	0
	100-109	23	0	0
	110-119	19	0	0
	≥120	17	0	0

FIGURE 1. Sample BLASTX output

```
BLASTX 1.2.7MP [31-Mar-92]

Query= YSCMIS1A S.cerevisiae mitochondrial C-1-Tetrahydrofolate synthase gene
        (4359 residues)
 Translating both strands of query sequence in all 6 reading frames

Database:  PIR 32.0 (complete), March 31, 1992
           40,298 sequences; 11,831,134 total residues.
                                                                     Smallest
                                                                     Poisson
                                                     Reading  High  Probability
Sequences producing High-scoring Segment Pairs:        Frame Score  P(N)      N

A28174 C1-tetrahydrofolate synthase precursor, mitocho... +1  5008  0.0       1
A29550 C1-tetrahydrofolate synthase - Yeast (Saccharom... +1  1775  0.0       2
A31903 Methylenetetrahydrofolate dehydrogenase (NADP+)... +1   800  1.4e-218  2
A35367 *C1-tetrahydrofolate synthase - Rat                +1   797  1.0e-214  2
A35942 *Formate--tetrahydrofolate ligase - Clostridium... +1   382  9.2e-108  4
A28185 *Formate--tetrahydrofolate ligase - Clostridium... +1   344  2.2e-105  4
A35667 *TEC1 protein - Yeast (Saccharomyces cerevisiae)   +3   581  1.2e-86   1
R5RT19 Ribosomal protein L19 - Rat                        -1   429  9.0e-59   1
A36554 Ribosomal protein L19 - Mouse                      -1   428  1.3e-58   1
R5DO9E Ribosomal protein L19e - Slime mold (Dictyostel... -1   419  3.4e-57   1
JS0662 *Methylenetetrahydrofolate dehydrogenase (NADP+... +1   187  1.2e-42   3
DEHUMT Methylenetetrahydrofolate dehydrogenase (NAD+) ... +1   203  4.5e-39   2
A33267 *Methylenetetrahydrofolate dehydrogenase/ methy... +1   199  4.6e-39   2
R5MXE  Ribosomal protein E - Methanococcus vannielii |... -1   157  6.4e-16   1
S16540 Ribosomal protein L19eR - Haloarcula marismortui   -1   112  7.9e-09   1
R5HSH4 Ribosomal protein HL24 - Haloarcula marismortui... -1   104  1.4e-07   1

 { Several alignments deleted }

>R5RT19 Ribosomal protein L19 - Rat  Length = 196
 Score = 429 (224.4 bits), Expect = 9.0e-59, P = 9.0e-59
 Identities = 77/144 (53%), Positives = 109/144 (75%), Frame = -1

Query:  4341 KLVKNGTIVKKSVTVHSKSRTRAHAQSKREGRHSGYGKRKGTREARLPSQVVWIRRLRVL 4162
             KL+K+G I++K+VTVHS++R R ++ ++R GRH G GKRKGT +AR+P  V W+RR+R+L
Sbjct:    43 KLIKDGLIIRKPVTVHSRARCRKNTLARRKGRHMGIGKRKGTANARMPEKVTWMRRMRIL 102

Query:  4161 RRLLAKYRDAGKIDKHLYHVLYKESKGNAFKHKRALVEHIIQAKADAQREKALNEEAEAR 3982
             RRLL +YR++ KID+H+YH LY   KGN FK+KR L+EHI   KAD  R K L ++AEAR
Sbjct:   103 RRLLRRYRESKKIDRHMYHSLYLKVKGNVFKNKRILMEHIHKLKADKARKKLLADQAEAR 162

Query:  3981 RLKNRAARDRRAQRVAEKRDALLK 3910
             R K + AR RR +R+  K++ ++K
Sbjct:   163 RSKTKEARKRREERLQAKKEEIIK 186

Parameters:
  E = 0.100, S = see table
  W = 3, T = see table, X = see table
  M = PAM120
  C = 0 (Standard genetic code)
  H = n/a, V = 500, B = 250

  Frame  Length      E    S   T   X
   +3     1452    0.076  65  13  19
   +2     1452    0.094  67  13  20
   +1     1453    0.082  71  13  21
   -1     1453    0.097  67  13  20
   -2     1452    0.094  68  13  20
   -3     1452    0.081  65  13  19

Statistics:                                Expected        Observed
  Frame Lambda   Lambda/ln2   K       H    High Score      High Score
   +3   0.378    0.546      0.212    1.45  58 (31.6 bits)   581 (317.0 bits)
   +2   0.363    0.524      0.205    1.21  60 (31.4 bits)    59 (30.9 bits)
  NOTE:  the cutoff score is greater than the expected high score
   +1   0.343    0.495      0.187   0.879  63 (31.2 bits)  5008 (2480.2 bits)
   -1   0.363    0.523      0.203    1.24  60 (31.4 bits)   429 (224.4 bits)
   -2   0.357    0.515      0.195    1.07  61 (31.4 bits)    63 (32.5 bits)
  NOTE:  the cutoff score is greater than the expected high score
   -3   0.378    0.545      0.218    1.41  58 (31.6 bits)    55 (30.0 bits)
  NOTE:  the cutoff score is greater than the expected high score

  # of neighborhood words in query = 114,650
  # of exact words scoring below T = 1461
  # of word hits against database = 82,603,593
  # of failed hit extensions = 70,774,341
  # of successful extensions = 42
  # of overlapping HSPs discarded = 6
  # of HSPs reportable = 36
  # of database sequences with at least one HSP = 16
No. of states in DFA:  598 (59 KB)    Total size of DFA:  1010 KB (1024 KB)
Time to generate neighborhoods:  1.15u 0.09s 1.24t
No. of processors used:  8
Time to search database:  141.78u 0.24s 142.02t
Total cpu time:  143.61u 2.63s 146.24t
```

FIGURE 2. Probability of finding an MSP in the presence of indel errors and mutations.

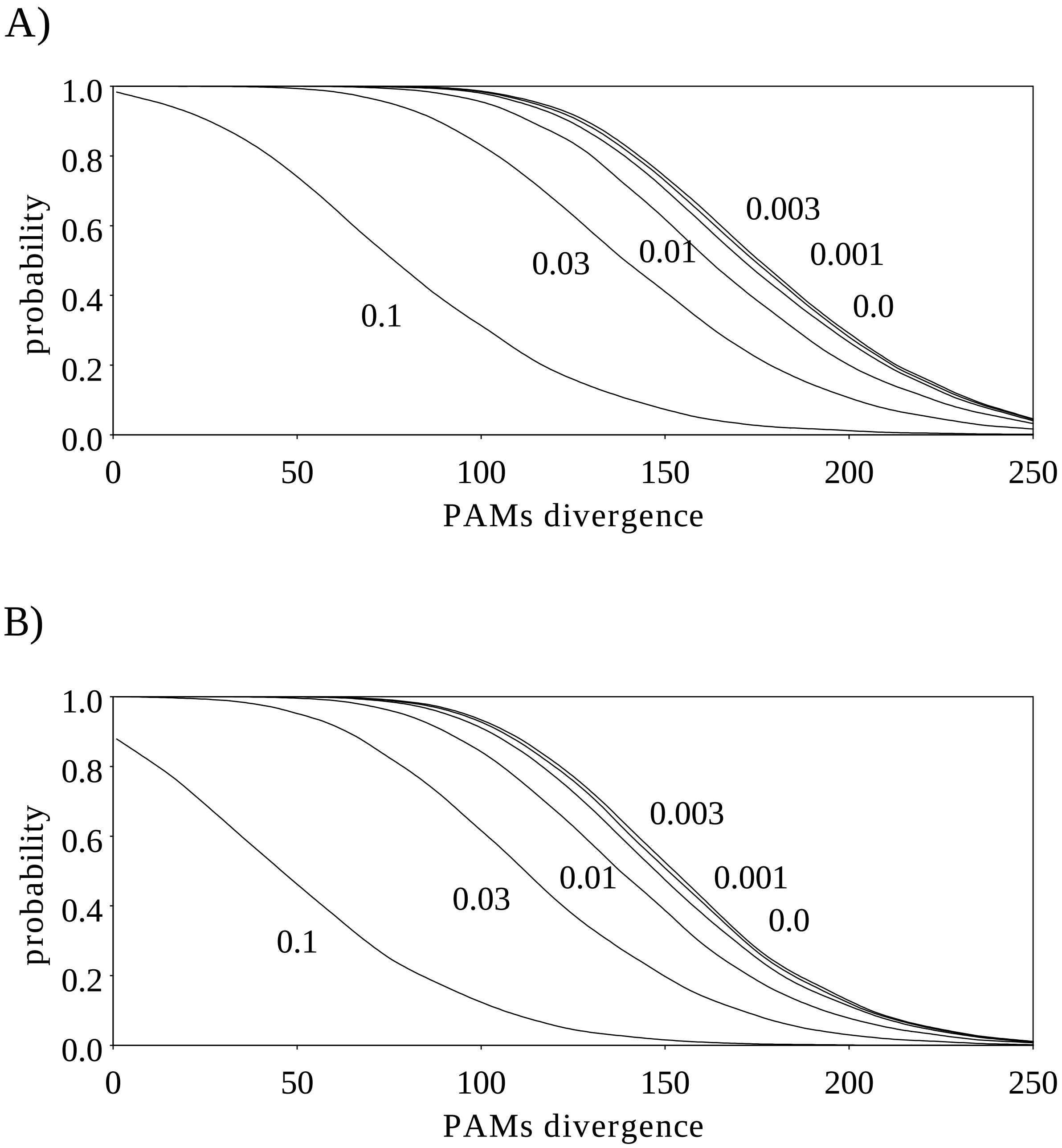
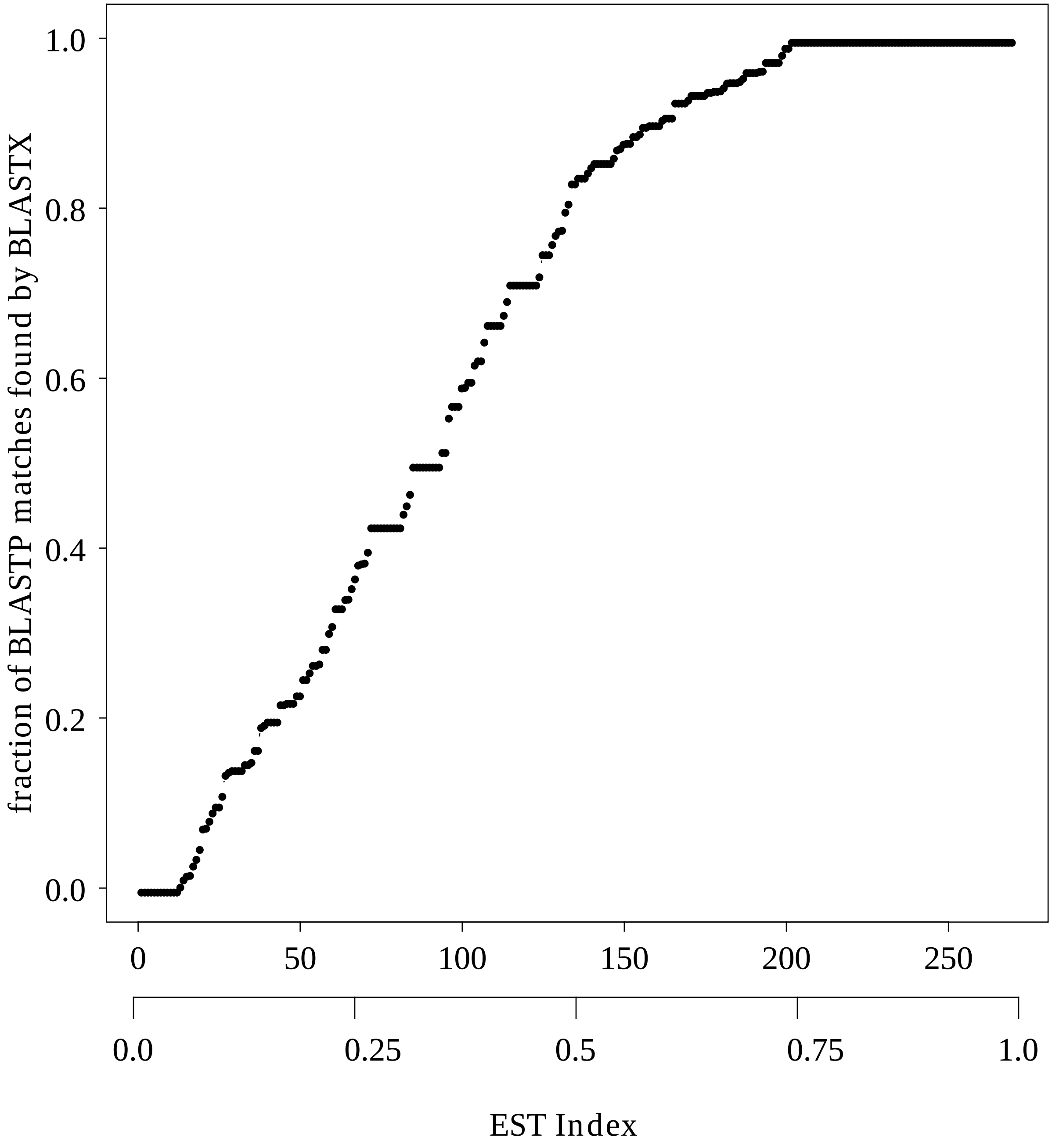


FIGURE 3. Comparison of EST and homologous protein database searches


fraction of BLASTP matches found by BLASTX
EST Index