Searching DNA databases for similarities to DNA sequences: when is a match significant?

Isobel Anderson and Andy Brass*

School of Biological Sciences, University of Manchester, 2.205 Stopford Building, Oxford Road, Manchester M13 9PT, UK

Received on October 31, 1997; revised on December 22, 1997; accepted on January 5, 1998

Abstract

Motivation: Searching DNA sequences against a DNA database is an essential element of sequence analysis. However, few systematic studies have been carried out to determine when a match between two DNA sequences has biological significance and this is limiting the use that can be made of DNA searching algorithms.

Results: A test set of DNA sequences has been constructed consisting of artificially evolved and real sequences. This set has been used to test various database searching algorithms (BLAST, BLAST2, FASTA and Smith–Waterman) on a subset of the EMBL database. The results of this analysis have been used to determine the sensitivity and coverage of all of the algorithms. Guidelines have been produced which can be used to assess the significance of DNA database search results. The Smith–Waterman algorithm was shown to have the best coverage, but the worst sensitivity, whereas the default BLASTN algorithm (word length set to 11) was shown to have good sensitivity, but poor coverage. A sensible compromise between speed, sensitivity and coverage can be obtained using either the FASTA or BLAST (word length set to 6) algorithms. However, analysis of the results also showed that no algorithm works well when the length of the probe sequence is <200 bases. In general, matches can accurately be identified between coding regions of DNA sequences when there is >35% sequence identity between the corresponding proteins. Searching a DNA sequence against a DNA sequence database can, therefore, be a useful tool in sequence analysis.

Availability: The test sets used are available via anonymous ftp from mbisg2.sbc.man.ac.uk in the directory /pub/cabios/testdata/

Contact: I.Anderson@stud.man.ac.uk; abrass@man.ac.uk

Introduction

Sequence comparison is a key tool in bioinformatics analysis. One of the best ways of assigning a putative function to a given sequence is to compare it to sequences of known function. There are, therefore, many occasions when a newly determined DNA sequence must be compared against sequence databases.

If a DNA sequence contains a protein coding region, then sequence comparison at the protein level is normally preferred. There are a number of reasons for this, such as:

1. Searches at the protein level should be more sensitive. There is degeneracy at the DNA level. Different codons can encode the same amino acid. This means that two identical protein sequences can differ greatly at the DNA level.

2. We have a better understanding of what constitutes a significant protein–protein hit. A number of papers have considered the problem of comparing protein sequences against protein sequence databases. Sander and Schneider (1991) showed that there must be 25% sequence identity over >80 residues to be confident of topological similarity.

3. Protein sequences are shorter than the corresponding DNA sequences. Protein sequence databases are smaller than DNA databases. The latest release of Swissprot contains 59 021 entries (October 1996). The main DNA sequence databases, EMBL, GenBank and DDBJ, contain > 1 432 941 sequences (June 1997). Protein database are therefore much quicker to search.

Therefore, in the majority of cases, sequence comparisons of DNA sequences containing a protein coding region are run using a program such as BLASTX which translates the DNA sequence into all possible reading frames and then compares the resultant protein sequences against protein sequence databases.

There are a number of occasions when this approach might not give the best results, or when additional information can be obtained by comparing the raw DNA sequence against a DNA sequence database. For example:
1. The primary repository of sequence data is DNA databases. Therefore, DNA databases contain the most up-to-date sequences. Annotation bottlenecks can mean that there is a delay in the coding regions of DNA sequences being deposited in protein sequence databases.

2. The protein databases may contain false translations of raw DNA sequences (Bork and Bairoch, 1996).

3. The query sequence may be non-coding, or, especially in the case of error-prone expressed sequence tags, the correct open reading frame may be difficult to determine (Mann, 1996).

In such circumstances, running a DNA database search might be a sensible option. However, although the techniques for assigning significance for running protein sequence searches against the databases have been well studied, very little work has been done on assessing searches of DNA sequences against DNA sequence databases. Three questions in particular need to be addressed. (i) What are the best algorithms for comparing DNA sequences against a DNA sequence database? (ii) Is it possible to tell from the alignment scores when a hit is significant? (iii) What error rate is expected from the search, i.e. what sort of false-positive/false-negative rates should be expected for the searches?

In this paper, we have therefore generated a model system in which to explore the performance of algorithms which compare DNA sequences against DNA sequence databases. The aims of the work are 2-fold: firstly, to assess the strengths and weaknesses of existing search algorithms, and secondly to provide researchers with a set of guidelines by which to assess the significance of DNA search results.

**Systems and methods**

**Creation of test sets**

Test sets for analysing the performance of protein sequence database searches have been created (Shpaer et al., 1996) or can be derived from protein classification systems such as SCOP (Murzin et al., 1995). No such test sets are available for testing DNA database searches. For this work, we have therefore constructed an extensive, carefully controlled test data set in which the sequences to be compared against the database have either been artificially generated or carefully chosen from real sequence families.

Consider choosing some DNA sequence, \( S(0) \), from a DNA sequence database, \( D \), where \( S(0) \) is reasonably long and does not consist entirely of low-complexity sequences. A sequence comparison algorithm run using \( S(0) \) as a probe against the database should find that the best match to \( S(0) \) is \( S(0) \) itself. Now consider modifying the sequence \( S(0) \), using some form of evolutionary algorithm, to produce a new sequence at a distance \( d \) away from the original sequence, \( S(d) \), where \( d \) is measured using some form of sequence space metric. Consider now using the sequence \( S(d) \) as a probe to search the database \( D \). If \( d \) is small, we would expect that \( S(d) \) would find the original sequence \( S(0) \) as one of its top matches. However, for large values of \( d \), we would have to move so far away from \( S(0) \) that it is no longer recognized as a homologous sequence. In principle, the more sensitive the sequence comparison algorithm, the larger the value of \( d \) that can be used, such that the sequence \( S(d) \) finds a significant match against \( S(0) \). We can therefore explore the behaviour of different comparison algorithms by exploring how effectively they can locate homologous matches to a range of different seed sequences, \( S(0) \), as a function of the distance that the probe sequence, \( S(d) \), has been evolved.

For this technique to work, we therefore need an algorithm to evolve DNA sequences and a metric to measure distance in sequence space. Sequence evolution has been studied extensively for regions of DNA that code for protein. Models of DNA evolution are much less well understood in other regions of DNA. In this study, we have therefore concentrated on testing DNA sequence searching for protein coding regions in DNA. Many metrics have been proposed for such applications, for this study we are using PAM (point accepted mutations) distance (Dayhoff et al., 1978). PAM distance is the number of substitutions that occur per 100 amino acids as one protein sequence evolves to another, allowing for the fact that multiple substitutions will have occurred at some positions. The PAM matrix gives the log-odds probability of one amino acid being substituted for another based upon mutational data. We are using the PAM distance between the corresponding aligned protein sequences as a measure of distance between two DNA sequences.

To create the test sets of artificially evolved sequences, ‘seed’ sequences were mutated using the Evolve algorithm (Slater, 1995). This algorithm simulates evolution in molecular sequences, with mutation taking place at the DNA level, and selection at the protein level, until a specified PAM distance between the original and evolved protein sequences has been reached. It is worth noting that insertions and deletions (indels) are added to the sequence. The gap length is determined by means of a Zipfian distribution (Benner et al., 1993), and is usually zero. There is an equal chance of the indel being an insertion or a deletion. When a deletion is selected, the corresponding region of the DNA sequence is deleted. When an insertion is selected, the composition of the DNA sequence to be added is determined using a codon usage table to weight the decisions as to which codons to use.

In all cases, the searches were performed against a database containing real sequences, the primate subset of EMBL which contains around 60 000 sequences. DNA databases do not contain sequences randomly distributed through sequence space. For example, the current data contained within EMBL database are biased towards genes that are or have
be of scientific interest. It is possible to show that the chance of finding a match to an unrelated sequence in a database (i.e. a false-positive result) is proportional to $\sqrt{\ln(N)}$, where $N$ is the number of sequences in the databases (Anderson, unpublished results), i.e. only very weakly dependent on the size of the database. The primate subset of EMBL is, therefore, a convenient subset to use to provide representative results at a relatively modest computational cost.

The first test set was constructed from a number of ‘seed’ sequences, shown in Table 1. The seed sequences were chosen to reflect different protein structures and different lengths of sequence so as to remove as far as possible bias in the test set. Three sequences were chosen according to secondary structure classification of the corresponding proteins (either all $\alpha$ helices, all $\beta$ sheets, or both $\alpha$ helices and $\beta$ sheets) using SCOP as a reference (Murzin et al., 1995; http://www.bio.cam.ac.uk/scop). Another seven sequences were chosen to have lengths varying from 100 to 1000 bp. RepeatMasker (http://ftp.genome.washington.edu/) was used to ensure that the seed sequences were free from interspersed repeats and low-complexity regions. The Evolve program was used to generate randomly a set of 10 new sequences at a specific PAM distance from each of the seed sequences. Sets of sequences were generated at distances of 25, 50, 75, 100, 125, 150, 175, 200, 225 and 250 PAMs from each of the seed sequences. In total, there were therefore 1000 sequences in the artificially created test set, 100 sequences being generated from each of the 10 seed sequences.

### Table 1. Genes included in the test set

<table>
<thead>
<tr>
<th>Gene name</th>
<th>Accession no.</th>
<th>ORF length</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ciliary neurotrophic factor</td>
<td>x60542</td>
<td>597</td>
<td>All $\alpha$</td>
</tr>
<tr>
<td>Tyrosine phosphatase</td>
<td>u02681</td>
<td>633</td>
<td>$\alpha/\beta$</td>
</tr>
<tr>
<td>Retinol binding protein</td>
<td>x00129</td>
<td>594</td>
<td>All $\beta$</td>
</tr>
<tr>
<td>Human paired-box-protein Pax8</td>
<td>S77904</td>
<td>99</td>
<td>Length 100</td>
</tr>
<tr>
<td>Human adenosine deaminase (ADA) gene</td>
<td>X02195</td>
<td>146</td>
<td>Length 150</td>
</tr>
<tr>
<td>Epsilon globin gene</td>
<td>U11712</td>
<td>210</td>
<td>Length 200</td>
</tr>
<tr>
<td>Pro-alpha-2 chain of type I procollagen</td>
<td>V00503</td>
<td>402</td>
<td>Length 400</td>
</tr>
<tr>
<td>AQP3 gene</td>
<td>D25280</td>
<td>606</td>
<td>Length 600</td>
</tr>
<tr>
<td>cAMP responsive element binding protein $\gamma$ subunit</td>
<td>L05913</td>
<td>811</td>
<td>Length 800</td>
</tr>
<tr>
<td>Progesterone receptor-associated p48 protein</td>
<td>U28918</td>
<td>1005</td>
<td>Length 1000</td>
</tr>
</tbody>
</table>

### Table 2. Summary of database search methods and parameters used

<table>
<thead>
<tr>
<th>Database search method</th>
<th>Database searched</th>
<th>Parameters</th>
<th>Scoring matrix</th>
<th>Type of query sequence test set used</th>
<th>Average search time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smith–Waterman</td>
<td>PRI</td>
<td>gap = –4.5, −0.05</td>
<td>Match = +1</td>
<td>Gene type</td>
<td>720</td>
</tr>
<tr>
<td>BLASTN (default)</td>
<td>PRI</td>
<td>Word size = 11</td>
<td>Match = +5</td>
<td>Gene length</td>
<td>34</td>
</tr>
<tr>
<td>BLASTN ($w = 6$)</td>
<td>PRI</td>
<td>Word size = 6</td>
<td>Match = +5</td>
<td>Gene length</td>
<td>347</td>
</tr>
<tr>
<td>FASTA (default)</td>
<td>PRI</td>
<td>K-tuple = 6</td>
<td>Match = +20,</td>
<td>Gene length</td>
<td>358</td>
</tr>
<tr>
<td>BLASTN2 (default)</td>
<td>PRI</td>
<td>K-tuple = 6</td>
<td>Mismatch = −6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLASTX (default)</td>
<td>PRI</td>
<td>Word size = 11</td>
<td>Match = +5</td>
<td>Gene length</td>
<td>42</td>
</tr>
<tr>
<td>FASTX (default)</td>
<td>TREMBL</td>
<td>Word size = 3</td>
<td>BLOSUM62</td>
<td>Tyrosine phosphatase</td>
<td>575 (SWISS)</td>
</tr>
<tr>
<td>FASTX (default)</td>
<td>SWISS</td>
<td>K-tuple = 2</td>
<td>BLOSUM50</td>
<td>Tyrosine phosphatase</td>
<td>423 (SWISS)</td>
</tr>
</tbody>
</table>

PRI, Primate division of EMBL (Version 46); SWISS, SWISSPROT protein database (Release 33); TREMBL, translated EMBL database (Version 1); gap, gap insertion penalty, gap extension penalty.

Algorithms used: Smith–Waterman (Smith and Waterman, 1981), implemented on a Bioccelerator at the SEQNET facility, Daresbury; BLAST (Altschul et al., 1990); BLAST2 (BLAST Version 2; Altschul et al., 1997); FASTA (Pearson and Lipman, 1988).
Testing the database searching algorithms

Each of the 1000 artificially evolved sequences was used as a query for database searches. Table 2 gives a summary of the database searching algorithms, parameters and test sets used for each search. Smith–Waterman aligns the query sequence against each sequence in the database, whereas BLAST and FASTA look for exact matches of a specific size (specified by the word size or K-tuple parameters) between the query and database sequences. For each database search, the top hit and corresponding search statistics were recorded. The hit was labelled ‘correct’ if the evolved sequence matched against the ‘seed’ sequence or homologue, otherwise it was labelled ‘incorrect’. All but one of the search sets were carried out using the default parameters.

Data analysis

Coverage of the database searches

We define the coverage of a database search to be the ability of the algorithm to pick out the appropriate homologous sequence from the database to the query sequence, irrespective of the score given. The coverage of the database searches was defined as the percentage of correct top hits out of all the searches carried out. The effectiveness of database search methods can also be illustrated by the distance the query sequences can be evolved before the search method no longer finds the homologous sequence in the database. We can therefore define a PAM sensitivity for a search algorithm as being the distance, $d$, to which a seed sequence, $S(0)$, can be evolved before $S(d)$ is no longer the top sequence found using $S(d)$ as a probe in 50% of cases. The larger the PAM sensitivity of an algorithm, the more effective the search method.

Discriminatory power of the database search statistics

Coverage does not take account of whether the database search statistics would be able to recognize the correct match as significant or not. It is perfectly possible for a correct top hit to be given a score which would lead to the hit being ignored. The relative operating characteristic (ROC) curves allowed us to test the usefulness of the search statistics [see Swets and Pickett (1982) and Shah and Hunter (1997)]. The curves were obtained by plotting the sensitivity ($P^+$) of the searches against the specificity ($P^-$).

\[
\text{Sensitivity} : P^+ = \frac{t^+}{t^+ + f^-} \quad \text{Selectivity} : P^- = \frac{t^-}{t^- + f^+}
\]

where $t^+$ is a true positive: correct hit, with a score above threshold; $f^-$ is a false negative: correct hit, with a score below threshold; $t^-$ is a true negative: incorrect hit, with a score below threshold; $f^+$ is a false positive: incorrect hit, with a score above threshold.

For each search carried out using both the artificially evolved and the real sequence, we knew which of these results were correct or incorrect, and the search score given for that hit. If the score for a database hit was above the threshold, the hit was assigned as positive, below the threshold they were assigned as negative. If a specific threshold value was selected, it was therefore possible to assign all hits as either true positives, true negatives, false positives or false negatives. For BLAST probabilities and FASTA expectation values, the converse is true. $P^+$ and $P^-$ were calculated for a range of threshold values. The range was selected to ensure that $P^-$ ranged from 0 to 1.

The area under the curve was found by numerical integration. This gave us a performance measure $P$. $P$ is essentially a measure of how well the statistic is able to discriminate between correct and incorrect hits. If $P = 1$, there is a threshold that allows us to discriminate between correct and incorrect hits perfectly. When $0 < P < 1$, there are always going to be false results, whatever threshold is set.

We can define the optimal cut-off to be the one that minimizes the total number of errors. The number of true positives which have been missed is given by $(1 - P^+)$. Similarly, the number of true negatives that have been missed is given by $(1 - P^-)$. Therefore, the optimal cut-off is the one that minimizes $(1 - P^+) + (1 - P^-)$. The optimal cut-off was calculated for each of the database search methods.

Results and discussion

Coverage of the database searches: BLASTN and Smith–Waterman

The results of the database searches using the evolved sequences were plotted. Figure 1 shows the number of correct top-ranked hits out of 10 obtained for each set of sequences of different gene type using the default Smith–Waterman and BLASTN searches. A reverse sigmoidal shape is observed. The same shape curve was obtained for each gene type used, indicating that the search results are independent of gene type. The PAM distance at which only 50% of the correct top hits are found gives the PAM sensitivity, a useful measure of the coverage of a database search method. For BLASTN, the coverage falls to 50% at around PAM 75; for Smith–Waterman, the coverage falls to 50% at around PAM 125. This indicates that the Smith–Waterman default searches perform better than the default BLASTN searches in identifying the more distantly related sequences.

If the per cent identity between the original and evolved DNA sequences was >70%, the hits were significant. However, this was only the case for sequences that had been evolved to PAM 25. For sequences had been evolved to PAM
50 or greater, the average per cent identity fell to ~60%, at which point it remained constant as a function of increasing PAM distance (data not shown). This suggests that the percentage identity between the probe sequence and a match in the database is not a good statistic for determining whether a match is significant. This is different from the case of proteins where it is well established that sequence matches showing >25% sequence identity are significant (provided that the probe sequence is long enough). It does not seem to be possible to detect significant matches between DNA sequences when the distance between them is greater than PAM 125. This again is in contrast to protein sequence database searches where the ‘twilight zone’ is generally considered to occur at about PAM 250, equivalent to 25% sequence identity (see below for a more detailed analysis).

Figure 2 shows the PAM distance at which the coverage falls to 50% as a function of probe sequence length using Smith–Waterman. From this figure, it can be seen that there is a strong effect of probe sequence length of the coverage, with a sharp fall off in performance for probe sequences shorter than 200 bp long. Therefore, we would not expect database searches using a query sequence of <200 nucleotides to be very effective. These results are reassuring in the context of expressed sequence tag (EST) analysis, which plays such an important role in current genome research. ESTs are generally 300–400 nucleotides in length (Boguski et al., 1993), longer than the minimum length suggested here.

Table 3 shows the percentage coverage of the database searches, using the sets of sequences of different length longer than 200 nucleotides. These values provide a measure of the sensitivity of the algorithms. Smith–Waterman searches have the best coverage, and are therefore the most sensitive. BLASTN with a word size of six has the second highest percentage coverage. FASTA had a very similar coverage to that seen for BLASTN with a word size of 6. The worst performing algorithm was the default BLASTN.

A number of searches using real tyrosine phosphatase nucleotide sequences were carried out against databases containing just one member of the tyrosine phosphatase gene family. The performance measure, \( P \), for the Smith–Waterman and BLAST algorithms (word length = 6) was 0.858001 and 0.876623, respectively. For Smith–Waterman, the discriminatory power of the \( Z \)-score statistic was the same as observed for the model data. For BLAST searches, the statistics did not perform quite as well for the real data as they do for the model data.

Shah and Hunter (1997) calculated ROC curves for FASTA \( Z \)-score and BLAST expectation to evaluate the performance of these statistics in classifying enzymes according to their International Enzyme Commission (EC) classifica-
tion. They obtained $P$ values of 1 for $\sim 40\%$ of EC classes, and $0.8 < P < 0.99$ for $\sim 48\%$ of classes. These $P$ values were the result of protein database searches and so we would not expect the DNA database search statistics to perform as well because of the degeneracy and smaller alphabet of DNA sequences. However, the $P$ values of 0.85–0.87 for the real DNA data searches are within the range observed for protein database searches. This again suggests that comparing a DNA sequence against a DNA database can produce useful results.

**Coverage of the database searches: BLASTX and FASTX**

The BLASTX and FASTX searches of the evolved sequences against SWISSPROT gave 50\% coverage at PAM 50 (data not shown). This might appear to be an anomolously low value. However, SWISSPROT did not contain a translation of the original tyrosine phosphatase sequence, only related sequences. For the BLASTX and FASTX searches against TREMBL, 50\% coverage was yielded at PAM 210 (data not shown). TREMBL did contain a translation of the original ‘seed’ sequence. The searches against TREMBL were more sensitive than those against EMBL. This is expected as proteins are non-degenerate and comprise a larger alphabet than DNA. These results demonstrate the importance of searching SWISSPROT, EMBL and TREMBL. TREMBL contains the translations of all coding sequences (CDS) specified in EMBL, but not already in SWISSPROT (Bairoch and Apweiler, 1996). As translations only appear in TREMBL if they have been specified in the nucleotide sequence annotation, it is important to search a nucleotide database such as EMBL, which in any case is the most up to date. These results again demonstrate the fact that searching a DNA sequence against a DNA sequence database can sometimes give better results than using the more traditional FASTX or BLASTX.

**Discriminatory power of the database search statistics**

Figure 3 shows ROC curves for the BLAST, FASTA and Smith–Waterman searches using the sets of sequences of different lengths (200, 400, 600, 800 and 1000). The values of $P^+$ and $P^-$, which were determined over a range of thresholds, are not shown.

The performance measures $P$ for each search method are shown in Figure 3. The BLAST and FASTA searches all give a similar $P$ value ($P = 0.96-0.97$). This indicates that the search statistics are performing well. The Smith–Waterman searches produce a $P$ value of 0.85. Therefore, although Smith–Waterman is the algorithm which gives the best coverage (see above), it has the least discriminating score for determining whether a top hit is significant. The Z-score does not work as well as the $P$ and $E$ values used by BLAST and FASTA, respectively.

The optimal cut-offs for the database search statistics were calculated by finding a cut-off value for the database search statistic which minimized $(1 - P^+) + (1 - P^-)$. A good default cut-off value to use for the $P$ or $E$ score is either 0.01 or 0.005. At these values of the cut-offs, the rate of false positives is $<2\%$ and the rate of false negatives is $<5\%$. For Smith–Waterman searches, the cut-off value for the Z-score should be set at 5. At this value, the rate of false positives is 1.5\% and the rate of false negatives is 20\%. From these numbers, it is clear that the Smith–Waterman Z-score seriously underestimates positive hits. Altschul et al. (1994) provide a useful introduction to the strengths and weaknesses of the different scoring statistics used, and particularly into the problems associated with using the Z-score as a statistic.
Conclusions

Our aims in this paper were to assess the strengths and weaknesses of different DNA database searching algorithms, and to provide guidelines to help assess the significance of the results obtained. Because of the problems of creating suitable test sets, we have had to concentrate on DNA sequences that contain protein coding regions. However, the greatest increase in DNA sequences is currently coming from ESTs and the results from this work should be applicable for this important class of sequences.

The Smith–Waterman algorithm is the best algorithm for finding remote homologues of DNA sequences; however, it suffers from having the least discriminating statistics (Z-score). BLASTN run at a word length of 11 (default) has the worst coverage of the algorithms tested; however, it does have the most discriminating statistics. The results show that a sensible compromise for coverage, sensitivity and speed is to use either FASTA or BLASTN with a word length of 6. This gives almost as good a coverage as can be obtained with Smith–Waterman, but with the advantage of better sensitivity. The cut-off for the BLASTN P value should be set at 0.01, the cut-off for the FASTA E value should be 0.005. Any matches found with scores below these E or P values should be significant 98% of the time.

Analysis of the results of the test set has suggested optimal cut-off values of the search statistics. These are given below:

<table>
<thead>
<tr>
<th>Algorithm (default)</th>
<th>P value: ≤ 0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLASTN (w = 6)</td>
<td>P value: ≤ 0.01</td>
</tr>
<tr>
<td>BLASTN2 (default)</td>
<td>P value: ≤ 0.005</td>
</tr>
<tr>
<td>FASTA</td>
<td>E value: ≤ 0.005</td>
</tr>
<tr>
<td>SW</td>
<td>Z-score: ≥ 5 (Implementation: Biocellerator bic_sw)</td>
</tr>
</tbody>
</table>

These thresholds should only be applied when a query sequence of at least 200 nucleotides has been used as searches using query sequences shorter than this have been shown to be ineffective. It is important to realize that these scores have been calculated for searches of a database of 70 000 sequences. If the simulations had been run on a larger database, the cut-off values obtained would have been larger. Therefore, these cut-offs are conservative.

A PAM distance of 130 between protein sequences seems to be the limit of sequence homology detection between the corresponding DNA sequences. A distance of PAM 130 corresponds to ~35% amino acid identity between two protein sequences (Dayhoff et al., 1978)—good enough to identify members of the same superfamily. Therefore, comparison of a DNA sequence for a coding region against a DNA database can provide useful structural and functional information on the associated protein product. However, translating the DNA coding region to protein and comparing at the protein level is still the most sensitive way to search for matches—picking up matches at a distance of PAM 210, equivalent to a 20–25% sequence identity at the protein level. However, this only works if either the protein translation of the matching DNA sequence can either be created sensibly on the fly (for algorithms such as BLASTX) or has already been correctly entered into the protein sequence database. This need not necessarily be the case, particularly for EST sequences, in which case a comparison of a DNA sequence against a DNA database can provide useful information.

Acknowledgements

This work benefited from the use of the SEQNET facility. The authors are grateful to Crispin Miller, Paul Higgs and Nick Lawson for useful comments, and to Guy Slater for making his Evolve program available. One of the authors (I.A.) is the grateful recipient of a BBSRC studentship.

References


