An Expert System for Processing Sequence Homology Data

Erik L.L. Sonnhammer and Richard Durbin
The Sanger Centre
Hinxton Hall, Hinxton
Cambridge CB10 1RQ, UK
esr@sanger.ac.uk rd@sanger.ac.uk

Abstract

When confronted with the task of finding homology to large numbers of sequences, database searching tools such as Blast and Fasta generate prohibitively large amounts of information. An automatic way of making most of the decisions a trained sequence analyst would make was developed by means of a rule-based expert system combined with an algorithm to avoid non-informative biased residue composition matches. The results found relevant by the system are presented in a very concise and clear way, so that the homology can be assessed with minimum effort. The expert system, HSPcrunch, was implemented to process the output of the programs in the BLAST suite. HSPcrunch embodies rules on detecting distant similarities when pairs of weak matches are consistent with a larger gapped alignment, i.e. when Blast has broken a longer gapped alignment up into smaller ungapped ones. This way, more distant similarities can be detected with no or little side-effects of more spurious matches. The rules for how small the gaps must be to be considered significant have been derived empirically. Currently a set of rules are used that operate on two different scoring levels, one for very weak matches that have very small gaps and one for medium weak matches that have slightly larger gaps. This set of rules proved to be robust for most cases and gives high fidelity separation between real homologies and spurious matches. One of the most important rules for reducing the amount of output is to limit the number of overlapping matches to the same region of the query sequence. This way, a region with many high-scoring matches will not dominate the output and hide weaker but relevant matches to other regions. This is particularly valuable for multi-domain queries.

Introduction

Large scale genome sequencing projects (Wilson et al., 1994; Oliver et al., 1992), generate new sequences at such a high rate that analyzing their homology is becoming a bottleneck. Database searching programs such as Blast (Altschul et al., 1991), Blaze (Brutlag et al., 1993), and Flash (Rigoutsos and Califano, 1993), are designed to optimize the search speed with little trade-off in sensitivity, but their output is usually only designed for single-gene queries, for which detailed manual reading of large amounts of data is acceptable. If the task is to analyze hundreds of proteins or several megabases of DNA however, a severe bottleneck lies in the manual evaluation of the matches reported by the search programs, which often form a list of many thousands of potential homologies.

One of the most distracting and time-consuming problems are spurious matches to regions of biased amino acid composition. Often significant similarities are missed because they are overshadowed by enormous amounts of such "junk" matches to other domains that are rich in a few amino acids. A similar problem is encountered if one domain is member of a large protein domain family and the strong and numerous matches to this region "drowns out" a few weak but relevant matches in other domains.

Restricting the amount of results by using a high score cutoff only makes the problem of missing distant similarities worse. This is very undesirable, since the distant matches generate just as much scientific interest as the close ones. However, manual reading of exceedingly long search result lists becomes an inhuman task for sequencing projects of several megabases.

These problems have been addressed by various approaches, such as deleting regions of biased composition in the query sequence, detected by internal repeats (Claverie and States, 1993) or entropy measures (Wootton and Federhen, 1993). By deleting these regions from the query, the unwanted matches they would generate are avoided.

A different approach to circumvent these problems is to view the database hits region by region instead of inspecting them in hi-scoring order. An example is the interactive multiple alignment browser for database matches *Blixem*. (See Sonnhammer and Durbin (1994) where also an early version of HSPcrunch is applied to DNA sequences). This facilitates domainwise analysis of the database matches and virtually eliminates the risk of "twilight zone" matches being drowned out by

other regions with numerous matches.

This paper describes an expert system which "weeds out" as many unwanted matches as possible. It is currently implemented to work on output from the programs Blastp, Blastx, Blastn and Tblastn which produce a list of ungapped alignments, or HSPs (High-scoring Segment Pairs). A set of rules are applied to filter out as many unwanted matches as possible, by compensating the score for compositional bias and by limiting the number of matches in congested regions. Weak matches that are potentially distant similarities are kept, however, if they support each other as being conserved regions of a gapped alignment. After filtering, the accepted matches from Blast can be viewed either as a graphical "Big Picture" display with one database sequence per line, showing matches symbolically as lines or a detailed pairwise alignment à la Blast, but sorted N- to C-terminal.

HSPcrunch rules

Biased composition

Biased composition HSPs are detected by a rule that compares the score of the HSP with the score of an HSP with no composition bias, in relation to the amino acid composition of the HSP in question.

The expected score of an HSP S_{exp} is the average score two random sequences of that length and amino acid composition would have. For a typical HSP the expected score is negative, but if the composition is very biased the expected score may be positive. The expected score is calculated the following way: Two vectors Q and S with the observed frequencies of the amino acids in the two sequences making up the HSP are constructed. The vectors are then scored against each other so that

$$S_{exp} = L \sum_{i=1}^{20} \sum_{j=1}^{20} Q_i S_j M_{ij}$$

where L is the length of the HSP and M is the scoring matrix. This method yields the same result as random shuffling methods would asymptotically, but is faster. To estimate if the score S of the HSP is the result of biased composition, we calculate the bias-ratio β:

$$\beta = \frac{S - S_{exp}}{S - LM_{exp}}$$

where M_{exp} is the frequency-weighted expected score of random (unbiased) sequences according to the scoring matrix used. For BLOSUM62, $M_{exp} = -0.945$. β can be used as an index of how biased the composition of the HSP is. As a rule, $\beta < 0.75$ is a sign that the HSP has a biased composition and should be rejected. Table 1 shows to what degree the unwanted biased composition HSPs are removed for different values of β. For values of β above .75, loss of good matches with slight bias becomes a problem.

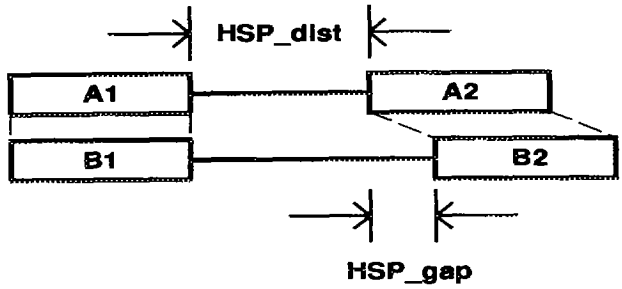

Figure 1: Adjacency of neighbor HPSs is checked by applying rules to the inter-HSP parameters shown here. HSP_dist is the smallest distance between the two segments of protein A or B. HSP_gap is the shift of protein A relative to protein B.

HSP adjacency

Low-scoring HSPs may be due to fragmentation caused by gaps in a larger alignment. Since these gapped alignments are potentially real, a lower score threshold should be used for adjacent HSPs that can be concatenated within some limits of allowed overlaps and gaps in the query and subject sequences. Figure 1 depicts two HSPs of the proteins A and B.

Adjacency is verified if A1 is N-terminal to A2 and B1 is N-terminal to B2. If not, there is no evidence that the HSPs are part of a larger gapped alignment and the HSPs are treated separately. The definition of adjacency completely depends on the chosen parameters for how big the gaps and overlaps between HSP may be. After testing various rules of adjacency, the most efficient combination of rules proved to be to test the adjacencies on two levels. A pair of neighboring HSPs are said to be either non-adjacent, loosely adjacent (on the correct side, but distant) or stringently adjacent. The parameter that gave the best results in terms of least missed homologies and most rejection of false ones are: HSP_gap < 25 for both loose and stringent adjacency. For loose adjacency, $-20 <$ HSP_dist < 300 and for stringent adjacency $-20 <$ HSP_dist < 50.

With no adjacency at all the score of an HSP is required to be at least 75. With loose adjacency (defined below), the score cutoff is 60, while with stringent adjacency we accept HSPs scoring 40. Note that these values are strictly meant for protein-protein HSPs. For DNA-DNA and DNA-protein other thresholds are used.

Coverage cutoff

If the segment in the query of an HSP already is covered by many other HSPs that score higher, the HSP is rejected to avoid redundant data, which can happen if the query is part of a family with many members. We limit the coverage by default to 10 fold. If every residue in the query has this coverage already, the HSP

	YMH5_CAEEL	YMH5_CAEEL	B0284.1	CA14_CAEEL	GRP_ARATH
Blastp cutoff	64†	40	40	65†	59†
Blastp HSPs	3465	30125	10377	5903	4427
β	biased | good	biased | good	biased | good	biased | good	biased | self
0.1	331 - 132	487 - 179	292 - 3	2503 - 27	1625 - 1
0.2	298 - 132	454 - 179	288 - 3	2485 - 27	847 - 1
0.3	221 - 132	377 - 179	277 - 3	2443 - 27	326 - 1
0.4	133 - 132	285 - 179	251 - 3	2418 - 27	66 - 1
0.5	23 - 132	117 - 179	191 - 3	2378 27	15 - 1
0.6	7 - 132	27 - 179	67 - 3	2255 - 27	3 - 0
0.7	0 - 132	8 - 179	15 - 3	1867 - 27	0 - 0
0.8	0 - 132	4 - 179	3 - 3	623 - 25	0 - 0
0.9	0 - 129	0 - 170	0 - 0	22 - 22	0 - 0

Table 1: Separation of biased composition matches from good ones by HSPcrunch as a function of the bias-ratio β. The numbers refer to counts of HSPs that passed the HSPcrunch adjacency criteria. No coverage limit was used. YMH5_CAEEL (Swissprot P34472) has a stretch of biased composition (acid-rich) in the N-terminus as well as a reverse transcriptase domain and 3 C-type lectin domains (see Figure 2). Most of the lectin domains are missed when using the default Blastp score cutoff but appear when it is is lowered to 40. The extra HSPs produced this way (26660) are efficiently filtered out by HSPcrunch. B0284.1 (Wormpep CE00650) has a charged-residue biased region. CA14_CAEEL (Swissprot P17139) is a collagen, containing mainly $[Gxy]_n$ repeats. Although these matches have biased composition, they are to other collagens, and it is therefore useful that HSPcrunch does not reject all of them. GRP_ARATH is the most biased composition protein in Swissprot 28 (72% Glycine, relative entropy 2.0 bits). In this extreme case even the match to itself does not pass the biased composition test when $\beta > .6$. †: The used Blastp cutoff was calculated by Blastp.

is rejected. If a set of HSPs are adjacent, all HSPs must be covered to be rejected.

Displaying results

HSPcrunch currently supports the following output formats:

- As a graphical "Big Picture" of the relevant matches, with one database sequence per line as shown in Figure 2. This way one rapidly gets a good picture of which proteins match a certain region of the query. In this display, matches that are consistent with the adjacency rules are combined onto one line, and the sum of their scores is given as the score. The number of adjacent segments is also shown. Non-adjacent HSPs are displayed on separate lines. If an HSP with a positive expected score passes the biased composition filter, its score will be marked by an asterisk.
- As a listing of accepted HSP alignments in N to C-terminal order. This verbose ASCII output of HSPcrunch is shown Figure 3a. The layout has been designed to be easy to read as well as easy to parse by other programs. Instead of sorting the HSPs in score order, like Blast does, HSPcrunch sorts them by position from N to C-terminus in the database sequence. This way a much better appreciation of the global alignment with gaps is gained, if it exists.
- View the HSPs as a multiple sequence alignment in the X-windows viewer Blixem (Sonnhammer and Durbin, 1994), either directly or called from ACEDB (Mieg and Durbin, unpublished).
- One line per HSP output, for parsing by other programs. A variety of one line formats are supported, of which one is shown in Figure 3b.

Methods and Materials

For generating the HSPs, Blastp version 1.3.11 was used. The B parameter was set to a high value so that it does not limit the number of HSPs reported, and the S (score cutoff) was usually set to 40. The protein sequence database searched, Swir, is a low-redundancy collection of sequences from SwissProt, PIR and Wormpep. Redundancy was removed by a program DBcomm, which rejected any sequence from PIR that was identical to or included in any Wormpep or SwissProt entry.

Release 5 of swir consisted of 57115 sequences, consisting of 781 sequences from Wormpep release 4, 35488 sequences from SwissProt release 28 (Bairoch and Boeckman, 1991) that were not derived from Wormpep, and 20846 sequences from PIR release 38 (Barker et al., 1992). Wormpep is a Sanger Centre in-house database containing all predicted proteins so far from the C. elegans genome sequencing project.

All programs were written in ANSI C and run on UNIX workstations from Silicon Graphics running Irix 4.0.5, and SUN running SunOS 4.1.3.

Availability

HSPcrunch and auxiliary programs such as Blixem, Fetch, Seqsplit, Blastunsplit are available by anonymous FTP from ftp.sanger.ac.uk in /pub/HSPcrunch

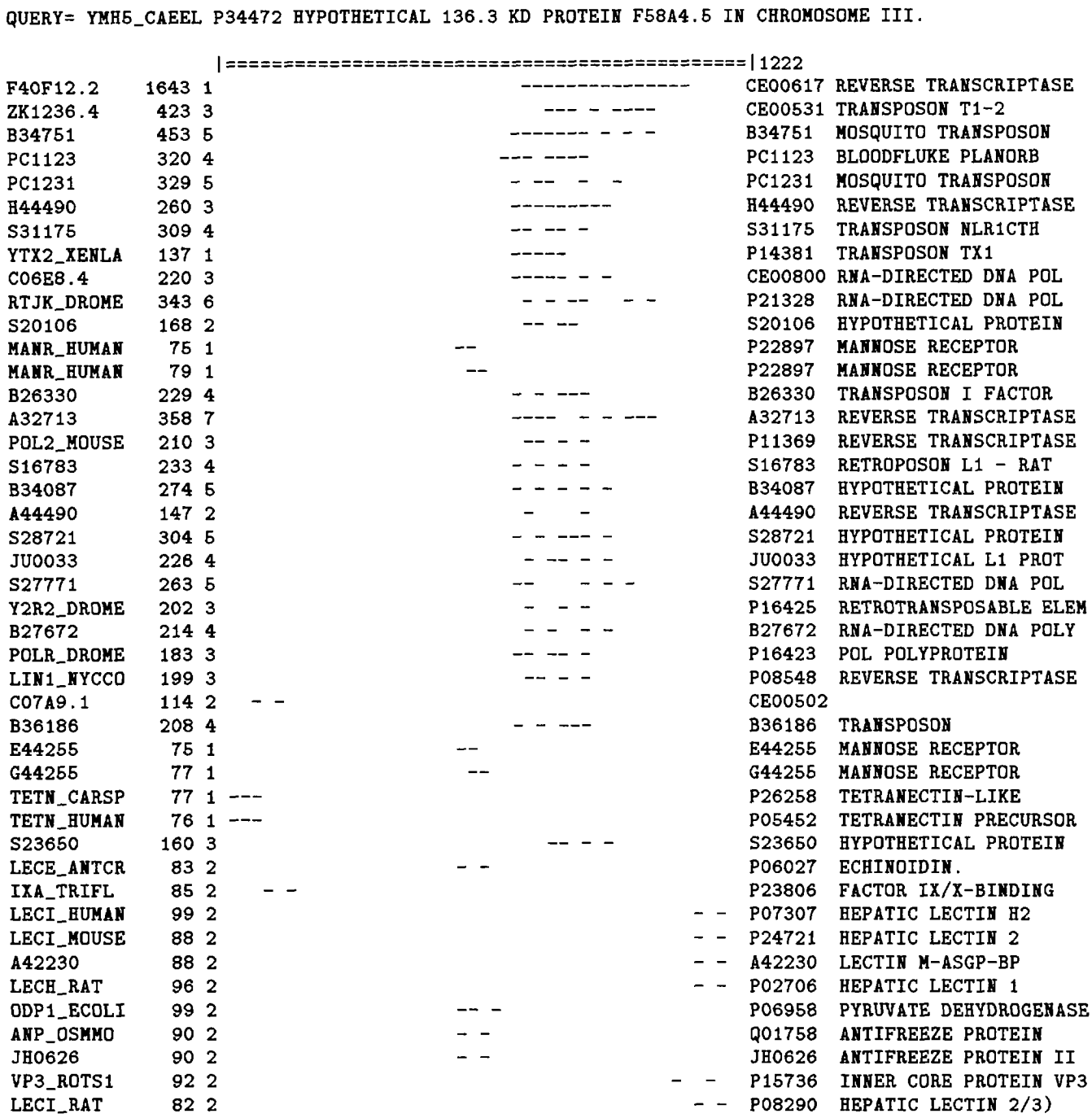

Figure 2: Big Picture display of HSPcrunched Blastp results. The sequence YMH5_CAEEL (Swissprot P34472) was searched against swir5 with a Blastp score cutoff of 40. The domain organization of this protein is C-type lectin (30-160), an acid-rich stretch (160-540), C-type lectin (540-620), Reverse Transcriptase (650-980) and C-type lectin (1080-1150). All matches to the acid-rich stretch were removed by the biased composition rule ($\beta = .75$). In the original output from Blastp, the 62 highest-scoring HSPs were all biased composition matches, apart from the close relatives F40F12.2 and ZK1236.4 from the same chromosome. Nearly all lectin matches are missed if the Score cutoff is not lowered from the default 64 to 40. The columns are: Entry name, combined score, nr. of HSPs, alignment, accession nr. and abbreviated description. Sequences from Wormpep include a dot and the ones from Swissprot an underscore. Other sequences are from PIR.

A

```
QUERY = YOW3_CAEEL  Length = 482

> VIPR_HUMAN  P32241 VASOACTIVE INTESTINAL POLYPEPTIDE RECEPTOR 1 PRECURSOR (VIP-R-1).
--------------------------------------------------------------------------------

Score= 45, Expected score= -30, Matrix_Expected= -30.2, bias-ratio= 1.00, Adjacency= 1
Query: ZK643.3 309 - 340    VPGVITVVYIFVRSLNDDVGMCWIENSTVAWI
                            VP   T+V+   R   +D G     NS++ WI
Sbjct: VIPR_HUMAN 265 - 296 VPSTFTMVWTIARIHFEDYGCWDTINSSLWWI

Score= 57, Expected score= -10, Matrix_Expected= -28.4, bias-ratio= 0.78, Adjacency= 1
Query: ZK643.3 342 - 371    WMIITPSLLAMGVNLLLLGLIVYILVKKLR
                            W+I  P L ++ VN +L   I+ IL++KLR
Sbjct: VIPR_HUMAN 295 - 324 WIIKGPILTSILVNFILFICIIRILLQKLR

Score= 53, Expected score= -13, Matrix_Expected= -19.8, bias-ratio= 0.91, Adjacency= 1
Query: ZK643.3 381 - 401    YRKAVRGALMLIPVFGVQQLL
                            Y +  R  L+LIP+FGV  ++
Sbjct: VIPR_HUMAN 336 - 356 YSRLARSTLLLIPLFGVHYIM

Score= 40, Expected score= -38, Matrix_Expected= -35.9, bias-ratio= 1.03, Adjacency= 1
Query: ZK643.3 418 - 455    LNGLQGMFVSFIVCYTNRSVVECVLKFWSSHQEKRALG
                            +   QG  V+ + C+ N  V   + + W     +  LG
Sbjct: VIPR_HUMAN 376 - 413 VGSFQGFVVAILYCFLNGEVQAELRRKWRRWHLQGVLG
```

B

```
45  34.38  309  340  YMH5_CAEEL  265  296  VIPR_HUMAN  ... POLYPEPTIDE RECEPTOR ...
57  43.33  342  371  YMH5_CAEEL  295  324  VIPR_HUMAN  ... POLYPEPTIDE RECEPTOR ...
53  42.86  381  401  YMH5_CAEEL  336  356  VIPR_HUMAN  ... POLYPEPTIDE RECEPTOR ...
40  23.68  418  455  YMH5_CAEEL  376  413  VIPR_HUMAN  ... POLYPEPTIDE RECEPTOR ...
```

Figure 3: HSP output. A. The alignment format, showing one of the G-protein coupled receptors that match YOW3_CAEEL (Swissprot P30650). Note that all of these HSPs would have been missed if the default Blastp score cutoff of 59 was used. Expected score: S_{exp}, Matrix_Expected: M_{exp}, bias-ratio: β, adjacency: 1 if the adjacency rules are satisfied. B. Same as A but in a one-line format. The columns are: score, % identity, query start, query end, query name, subject start, subject end, subject name and title.

or by sending Email to esr@sanger.ac.uk. Wormpep is in /pub/databases/wormpep.

Discussion

An often heard criticism of using ungapped alignments is that distantly related proteins generally can only be aligned by inserting gaps. However, the regions which require gaps usually correspond to loops between secondary structure elements in the 3-dimensional structure, where the length of the loop may vary. The loop residues can often not be aligned structurally, which makes sequence alignments of these regions rather meaningless. Also, the results of algorithms that produce gapped alignment depend strongly on a somewhat arbitrary gap penalty. We have therefore not attempted to concatenate adjacent HSPs into gapped alignments, but feel that most information is already present in the ungapped HSPs. A further advantage of ungapped alignments is that repeated and shuffled domains in one sequence can be detected, something which is often compromised by programs that produce a gapped alignment.

One drawback of ungapped alignments is the difficulty of calculating an appropriate composite score for all HSPs with the same protein. Here we put the emphasis on making sure that a series of HSPs are truly consistent with a single gapped alignment. We then simply add the individual scores. A different approach is taken by the Blast programs. For n HSPs from the same database sequence, they calculate a combinatorial Poisson probability P(n) that they were found by chance. This probability can be derived analytically

but their "consistently ordered" criterion (Karlin and Altschul, 1993) is much weaker than our adjacency criteria and falsely high Poisson rankings may arise from spurious hits that are not adjacent, especially those involving biased composition matches.

An additional practical problem with the Poisson probabilities calculated in BLAST is that they increase with the size of the database, because the expected number of spurious matches increases slowly as the database grows. However, the true match scores do not change, and because many of the new sequences are homologous to existing ones, the Poisson correction often overestimates the drop in significance. In any case it is more convenient to work with a measure of similarity that remains stable for a particular match. For these reasons we designed HSPcrunch to work only with the raw scores (which are log odds ratios).

The reduction of redundant results due to large protein families was achieved here by rejecting excess matches to a given region. A more subtle way of accomplishing this is to search a pre-clustered database. Instead of finding similarities to every member of the family, a single match would be found to the entire family, thus giving the relations to all other members of the family, not only the closest relatives. Presently available collections of protein families such as BLOCKS (Henikoff and Henikoff, 1992) and ProDom (Sonnhammer and Kahn, 1994) could be used for this. Since the number of protein sequences grows faster than the number of families (Green et al., 1993), a pre-clustering approach seems very appealing. Given a family of proteins, one can build statistical models such as Profiles or Hidden Markov Models (Krogh et al., 1994) to improve sensitivity. Whether searching a collection of aligned families is better than searching against all sequences is however not entirely clear. Sensitivity may also decrease if the family is not well defined, or if the query is much closer to one of the members than to the average of the family. Therefore, we have here pursued a higher quality of traditional single-sequence database searching, which most likely will remain an important tool complementary to family-based searching techniques.

Acknowledgments

The Sanger Centre is supported by the Wellcome Trust and the MRC.

References

Altschul, S. F.; Gish, W.; Miller, W.; Myers, E. W.; and Lipman, D. J. 1990. Basic local alignment search tool. *J. Mol. Biol.* 215: 403-410.

Bairoch, A. and Boeckmann B. 1991. The SWISS-PROT protein sequence data bank. *Nucleic Acids Res.* 19: 2247-2249.

Barker, W. C.; George, D. G.; Mewes, H. W.; and Tsugita A. 1992. The PIR-International protein sequence database. *Nucleic Acids Res.* 20: 2023-2026.

Brutlag, D. L.; Dautricourt, J. P.; Diaz, R.; Fier, J.; and Stamm R. 1993. BLAZE (tm): an implementation of the Smith-Waterman sequence comparison algorithm on a massively parallel computer. *Comput. Chem.* 17: 203-207.

Claverie, J. M. and States. D. J. 1993. Information enhancement methods for large scale sequence analysis. *Comput. Chem.* 17: 191-201.

Green, P.; Lipman, D. J.; Hillier, L.: Waterson, R.; States, D.; and Claverie, J. M. 1993. Ancient conserved regions in new gene sequences and the protein databases. *Science* 259: 1711-1716.

Henikoff, S. and Henikoff, J. G. 1991. Automatic assembly of protein blocks for database searching. *Nucleic Acids Res.* 19: 6565-6572.

Karlin, S. and Altschul, S. F. 1993. Applications and statistics for multiple high-scoring segments in molecular sequences. *Proc. Natl. Acad. Sci. USA* 90: 5873-5877.

Krogh A.; Brown M.; Mian I. S.; Sjoelander K.; and Haussler D. 1994. Hidden Markov model in computational biology. Applications to protein modeling. *J. Mol. Biol.* 235: 1501-1531.

Oliver, S. G.; Van der Aart, Q. J. M.; Agostini-Carbone, M. L.; Aigle, M.; Alberghina, L.; Alexandraki, D.; Antoine, G.; Anwar, R.; Ballesta, J. P. G.; Benit, P.; Berben, G.; Bergantino, E.; Biteau, N.; Bolle, P. A.; Bolotin-Fukuhara, M.; Brown, A.; Brown, A. J. P.; et al. 1992. The complete DNA sequence of yeast chromosome III. *Nature* 357: 38-46.

Rigoutsos, I. and Califano, A. 1993. dFLASH: A Distributed Fast Look-Up Algorithm for String Homology. IEEE *Computational Science and Engineering* In Press.

Sonnhammer, E. L. L and Durbin, R. 1994. A workbench for Large Scale Sequence Homology Analysis. *Comput. Appl. Biosci.* In Press.

Sonnhammer, E. L. L. and Kahn, D. 1994. Modular structure of proteins as inferred from analysis of homology. *Protein Science* 3: 482-492.

Wilson et al. 1994. 2.2 Mb of contiguous nucleotide sequence from chromosome III for C. elegans. *Nature* 368: 32-38.

Wootton J.C. and Federhen S. 1993. Statistics of local complexity in amino acid sequences and sequence databases. *Comput. Chem.* 17: 149-163.