Performance Evaluation of A New Algorithm for the Detection of Remote Homologs With Sequence Comparison

Maricel G. Kann[1] **and Richard A. Goldstein**[2*]
[1]*Department of Chemistry, University of Michigan, Ann Arbor, Michigan*
[2]*Biophysics Research Division, University of Michigan, Ann Arbor, Michigan*

ABSTRACT **A detailed analysis of the performance of *hybrid*, a new sequence alignment algorithm developed by Yu and coworkers that combines Smith Waterman local dynamic programming with a local version of the maximum-likelihood approach, was made to access the applicability of this algorithm to the detection of distant homologs by sequence comparison. We analyzed the statistics of hybrid with a set of nonhomologous protein sequences from the SCOP database and found that the statistics of the scores from hybrid algorithm follows an Extreme Value Distribution with lambda ~1, as previously shown by Yu et al. for the case of artificially generated sequences. Local dynamic programming was compared to the hybrid algorithm by using two different test data sets of distant homologs from the PFAM and COGs protein sequence databases. The studies were made with several score functions in current use including OPTIMA, a new score function originally developed to detect remote homologs with the Smith Waterman algorithm. We found OPTIMA to be the best score function for both both dynamic programming and the hybrid algorithms. The ability of dynamic programming to discriminate between homologs and nonhomologs in the two sets of distantly related sequences is slightly better than that of hybrid algorithm. The advantage of producing accurate score statistics with only a few simulations may overcome the small differences in performance and make this new algorithm suitable for detection of homologs in conjunction with a wide range of score functions and gap penalties. Proteins 2002;48:367–376.**

Key words: sequence alignment; protein evolution; score matrices; extreme value distribution; dynamic programming

INTRODUCTION

Over the past few years scientist have gathered an enormous number of protein sequences from the various genome projects. However, we are generally interested in knowing more than the sequence; we want to know the structure, function, and metabolic role of those newly sequenced proteins. Experimental, and more recently, computational techniques can provide us with information that can help us to solve this one aspect of the puzzle of life.

One of the most popular first steps in such a process is called sequence comparison. Its popularity is due to the fact that it can provide a fair amount of information in a few seconds at low cost to the user. In this technique, the sequence of the target protein is aligned and compared with those of all the other sequences in various protein databases. An alignment score S is generally computed by using an expression such as

$$S = \sum_{i,j} n_{i,j}\,\mathrm{sim}(i,j) + n_{\mathrm{gap-I}}\,d + n_{\mathrm{gap-E}}\,e \qquad (1)$$

where $n_{i,j}$ refers to the number of times that amino acid type i is aligned with amino acid type j, $n_{\mathrm{gap-I}}$ is the total number of gaps in the alignment, $n_{\mathrm{gap-E}}$ is the total number of residues in each gap beyond one, and sim(i, j), d, and e represent the contribution to the score for any amino acid match or mismatch, initialization of a gap, and extension of a gap, respectively; sim(i, j) is known as the score function, substitution matrix, or exchange residue matrix, whereas d and e represent the gap penalties. This linear representation of the gap penalties is referred to as an affine gap penalty. sim(i, j) can be interpreted as proportional to the log of the probability for such a pair of amino acids to be found in a set of homologs compared with what would be expected at random.[1,2] Considering the various aligned positions as statistically independent, the resulting total score then can be seen as representing the log of the probability of the entire set of aligned amino acids for a pair of homologous proteins, again relative to what would be expected at random.

Proteins in the database with strong sequence similarities to the target protein likely represent homologous

Grant sponsor: National Institutes of Health; Grant number: LM0577, Grant Sponsor: National Science Foundation; Grant number: BIR9512955.

M.G. Kann's present address is National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD 20894.

*Correspondence to: Richard A. Goldstein, Department of Chemistry, University of Michigan, Ann Arbor, MI 48109-1055. E-mail: richardg@umich.edu

Received 20 September 2001; Accepted 17 January 2002

proteins, that is, related by common ancestry to the target protein. The identification of homologous proteins can provide a wealth of information about the target protein. To identify homologies, it is necessary to identify matches yielding high scores or better yet, scores with a low probability to arise by chance alone. To make this discrimination, we require an accurate statistical model of the distribution of scores from alignments of nonhomologous protein pairs.

One of the standard methods for sequence comparison is the local dynamic programming (LDP) algorithm developed by Smith and Waterman.[3] This algorithm finds the highest scoring alignment for the pair of sequences given the score function and returns this maximum score. For this method, the statistics of random scores for the case of alignments without gaps has been studied[4–6] and the probability $p(S_r > x)$ of a score S_r generated by a nonhomologous pair of proteins being larger than any particular score x can be well represented by a Gumbel or Extreme Value Distribution (EVD)

$$p(S_r > x) = 1 - \int_{-\infty}^{S_r} \rho(x)dx = 1 - \exp(-KMNe^{-\lambda x}) \quad (2)$$

where M and N are the lengths of the two sequences, λ represents the scale characterization parameter, and K is related to the EVD's localization parameter.[7] For those alignments where gaps are allowed, of particular relevance when homology is distant, there is no such theory. It has been shown, however, that the distribution of random scores for alignments with gaps can still be approximated as an EVD.[8] To describe the EVD distribution, it is necessary to estimate the parameters λ and K for each choice of score function and gap penalty. Estimation of λ is particularly difficult and requires the simulation of thousands of alignments, a computationally intensive task.[9] Generally, these parameters are precalculated for certain choices of the scoring system, allowing commonly used search engines such as BLAST[6,10] to estimate the statistical significance of alignment scores rapidly. However, precalculation limits these methods, to a predetermined set of score functions and gap penalties. This is especially problematic for programs such as PSI-BLAST[11,12] where the score matrix adjusts during the procedure, as well as for any attempt to use a wider range of gap penalties.

In the LDP algorithm, information about suboptimal alignments is lost regardless of how close to the optimal they are. To overcome this problem and potentially improve our ability to discriminate between homologs and nonhomologs, it is possible to calculate the score for each alignment and sum, rather than optimize, over all possible alignments. This procedure corresponds to calculating the probability of the set of aligned amino acids occurring in a pair of homologous proteins integrated over all possible alignments. This probabilistic approach has been used for comparison of protein sequences[13–16] and structures.[17] The major disadvantage of the probabilistic approach is that the null distribution of random scores has not yet been characterized, making the significance of the scores difficult to estimate.

Yu and coworkers[18,19] recently developed a hybrid between the probabilistic approach and dynamic programming. The new algorithm, which we refer to as *hybrid*, first sums over all alignments ending in any given pair of amino acids and then assigns the final score to that of the maximum sum. In this way, the score represents the integral over the set of alignments ending in one particular pair of aligned amino acids, with this pair chosen to maximize the resulting score. Their theory predicts that the scores obtained with the hybrid method will satisfy the EVD distribution with parameter λ asymptotically approaching one for longer sequences. This prediction was verified numerically for certain score functions for an artificially created set of random sequences with fixed length and composition. Real sequences have different compositions as well as correlations between the amino acids found in various locations. To provide a useful algorithm, the hybrid algorithm's well-characterized score statistics must extend to pairs of nonhomologous protein sequences from the databases. In this article, we reproduce the results for an artificial data set and extend the study to searches on real biological sequences. We show that the statistics for nonhomologous biological sequences still obeys EVD statistics with asymptote $\lambda \simeq 1$ as predicted.

The hybrid alignment's well-characterized statistics and its ability to reproduce the null score statistics for different score systems with only a few simulations make the hybrid method attractive for searching large protein sequence databases. These findings prompt the question of whether the performance for the detection of distant homologs of hybrid algorithm is comparable to that of local dynamic programming with affine gap penalties. To answer this question, we compared the two algorithms by using various score schemes, finding only slightly lower performance in the task of detecting distant homologs for the hybrid algorithm.

The choice of score function and gap penalties is crucial in protein sequence comparison regardless of the algorithm in use. Because both the alignments and the final scores depend on the score system used, the performance of both algorithms was tested for two different databases with a wide range of score matrices and gap penalties. We found OPTIMA,[20] a score function originally developed to detect distant homologies with local dynamic programming, to be the best score function for both algorithms in both data sets.

MATERIALS AND METHODS

Database Preparation

The hybrid algorithm's null statistic has been studied analytically and numerically by Yu and coworkers[18,19] by using a set of artificial randomly generated sequences. The question of whether the null models will still be applicable when using real sequences is still untested. To reproduce the results of Yu et al., we prepared 100,000 random synthetic sequences of length 300 each, with average compositions based on the observed distribution of amino acids tabulated by Robinson and Robinson.[21] To describe the statistics of alignments scores generated with random nonhomologous pairs of biological proteins, we chose a test set of 122 sequences pairs from SCOP database release 1.53[22] with lengths between 200 and 600 residues in which each member of the pairs belong to a different fold. Therefore, each pair of protein sequences can be assumed to be nonhomologous.

When comparing the performance of the hybrid algorithm with LDP, we focused on those sequence pairs that are difficult to detect with current methods. For that purpose, we developed test sets from two databases in which each pair of homologous sequences share <25% sequence identity. A set of 321 pairs of proteins sequences from the Cluster of Orthologs Groups (COGs) database developed by Tatusov and coworkers[23] was chosen so each pair of homologs belongs to the same cluster and shares <25% sequence identity. Proteins from the COGs set had lengths between 200 and 1000 residues. A second test was prepared by using the PFAM database release 5.2 (PFAM 5.2)[24]; 103 pairs of protein sequence between 200 and 600 residues long were selected where each pair belonged to the same PFAM family. For both test sets, only one target pair of sequences was taken from each cluster or family; thus, the other sequences in the test set (belonging to a different family or cluster) could be used to represent decoy sequences nonhomologous to the target pairs. To provide a fair comparison between OPTIMA and the other score functions, we eliminated all proteins from the COGs and PFAM test sets that had significant similarities (E-values < 10) or were members of the same COG set as any of the proteins used to adjust the parameters of the OPTIMA model.[20] Listings of the proteins in the various training and test sets are available from the authors.

Comparisons

The implementation of the local dynamic programing and hybrid algorithms are described in the Appendix.

The key element in any practical alignment procedure is a characterization of the null statistics, that is, the statistics of scores for nonhomologous pairs of sequences. As the first part of this work, we evaluated how closely the null distribution matched an EVD distribution, and estimated the parameters λ and K that allow us to calculate $p(S_r > x)$ in conjunction with Eq. 2. To test the statistics of the hybrid algorithm scores with a set of artificially created sequences, we aligned each of the queries from the COGs and PFAM test sets to 100,000 randomly created sequences by using the BLOSUM62 score function with $(d, e) = (-11, -1)$[25], which we will notate by BLOSUM62$_{-11,-1}$. To evaluate the null statistics with nonhomologous biological proteins, we used an assortment of score functions, including BLOSUM62$_{-11,-1}$ and BLOSUM50$_{-13,-2}$,[25] PAM250$_{-14,-2}$,[1] and OPTIMA$_{-20,-4}$,[20] to align each of the queries of SCOP with the proteins sequences belonging to other fold categories. For OPTIMA, a score function corresponding to one fifth of the score matrix published by Kann et al[20] was used, with correspondingly adjusted gap penalties.

The next part of the work involved comparing the performance of the LDP and hybrid algorithms for a choice of score functions and gap penalties, evaluating how well the different approaches could discriminate between homologous and nonhomologous proteins. We characterized the performance of the different algorithms and score functions for each of the pairs of homologous proteins in the two test sets. In all cases, other target proteins in the test sets were used as decoys. We used the default gap parameters as included in BLAST. Because sensitivity is significantly affected by the choice of gap penalty, we also performed an exhaustive search for the optimal gap penalty for the hybrid algorithm for each of the score matrices (results not shown). For both algorithms, we computed the E-value, that is, the expected number of scores between the target protein and random pairs that is greater than the score for the correct homolog. For a search of a database of size D, the E-value is estimated as $E = D\,p(S_r > x)$, where x is the score of the homolog. In this study, we used a value of $D = 100{,}000$. We also calculated the probability P of observing at least one alignment of random protein sequences with score $\geq x$; assuming Poisson statistics this value is given by

$$P = 1 - \exp(-E) \qquad (3)$$

We also calculated another statistical parameter, C, that represents the confidence in identification of a true homolog and is calculated as the total number of correct matches divided by number of matches, both correct and incorrect, with the same score or higher.[20] One advantage of the C score is that it concentrates on protein pairs at the limit of detectability. Assuming there is one true homolog in the data set, C can be calculated from the scores of the alignments as

$$C = \frac{1}{1+E} = (1 + D(1 - \exp(-KMNe^{-\lambda s}))^{-1} \qquad (4)$$

The values of $\langle p(S_r > x)\rangle$, $\langle P\rangle$, and $\langle C\rangle$ for hybrid and local dynamic programming algorithms were obtained by using Eq. 2, 3, and 4, respectively, by averaging over all the sequence pairs in the test set.

The detection of homologous proteins using any of the algorithms in consideration can be seen as the discrimination between two alternatives. A certain threshold S_0 can be defined so that pairs of protein sequences with pairwise alignment scores greater than the threshold will be classified as homologs and below as nonhomologs. (Any other monotonic function of the scores can be used in a similar manner, i.e., lower p or E values than a certain threshold will indicate homology.). The data can be then classified as either positive or negative corresponding to homolog or nonhomolog, respectively. This decision, based on the threshold, can be true or false. Therefore, there are two possible outcomes, true-positives and true-negatives, in which the decision was correct. And there are two where the method fails: false-positives and false-negatives. To evaluate the performance of dynamic programming, we plotted the fraction of true positives, FTP = (number of

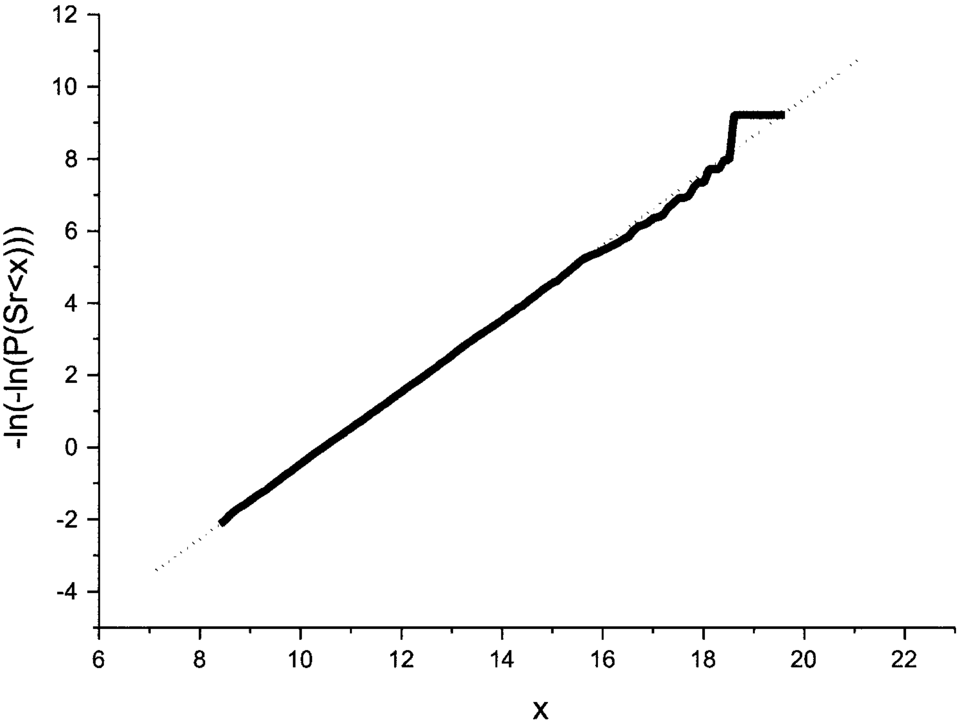

Fig. 1. Linear fit of the double logarithmic plot of the probability density function (pdf) of the scores obtained from the alignment of 100,000 synthetic random sequences to test set proteins. BLOSUM62$_{-11,-1}$^{25} was used as a scoring system.

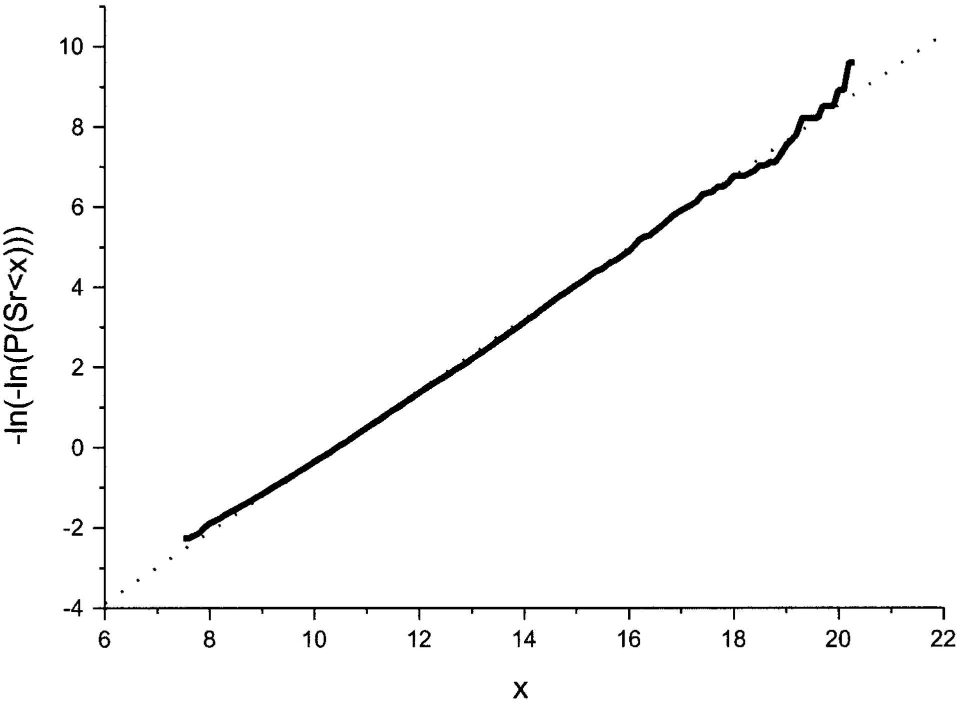

Fig. 2. Linear fit of the double logarithmic plot of the probability density function (pdf) of the scores obtained from the alignment of 14,762 pairs of nonhomologous sequences belonging to different SCOP fold classifications. BLOSUM62$_{-11,-1}$^{25} was used as a scoring system.

pairs of true positives with score > S_0)/(total number of true-positives) also known as sensitivity, versus the fraction of false positives, FFP = (number of true-negatives with score > S_0)/(total number of true-negatives), known as specificity. This kind of plot, called a “Relative Operating Characterization” [ROC] plot, has been used since the early 1950s, especially for diagnostic medical systems.[26] The ROC plot is related to Brenner and coworkers[27] coverage versus error per queries curves.

TABLE I. Distribution of λ Values for Hybrid Algorithm

Score matrix	Gap penalties	λ
OPTIMA[20]	−20.0/−4.0	0.87 ± 0.15
BLOSUM62[25]	−11/−1.0	0.89 ± 0.14
PAM250[1]	−14/−2.0	1.00 ± 0.19
BLOSUM50[25]	−13/−2.0	0.88 ± 0.15

Hybrid algorithm was used in conjunction with four different scoring systems and the COGs and PFAM test databases for the estimation of λ. The parameter λ was calculated as the slope of the linear fit to the double logarithmic plot of the score probability distribution function (pdf).

RESULTS

Statistics of the Scores of Random Alignments With Hybrid and Local Dynamic Programing

Figure 1 shows the double logarithmic plot of the probability density function (pdf) of the scores obtained from the alignment of 100,000 synthetic random sequences to the queries from the COGs and PFAM test sets using the hybrid algorithm. An EVD distribution on such a plot yields a straight line. The closeness of the fit to an EVD distribution is clear. The parameter λ can be estimated as the slope of the line and is ~1, confirming Yu and Hwa's[18] results.

To simulate the "real" conditions of a sequence search where the comparisons are made against biological sequences, we calculated λ for structurally dissimilar pairs

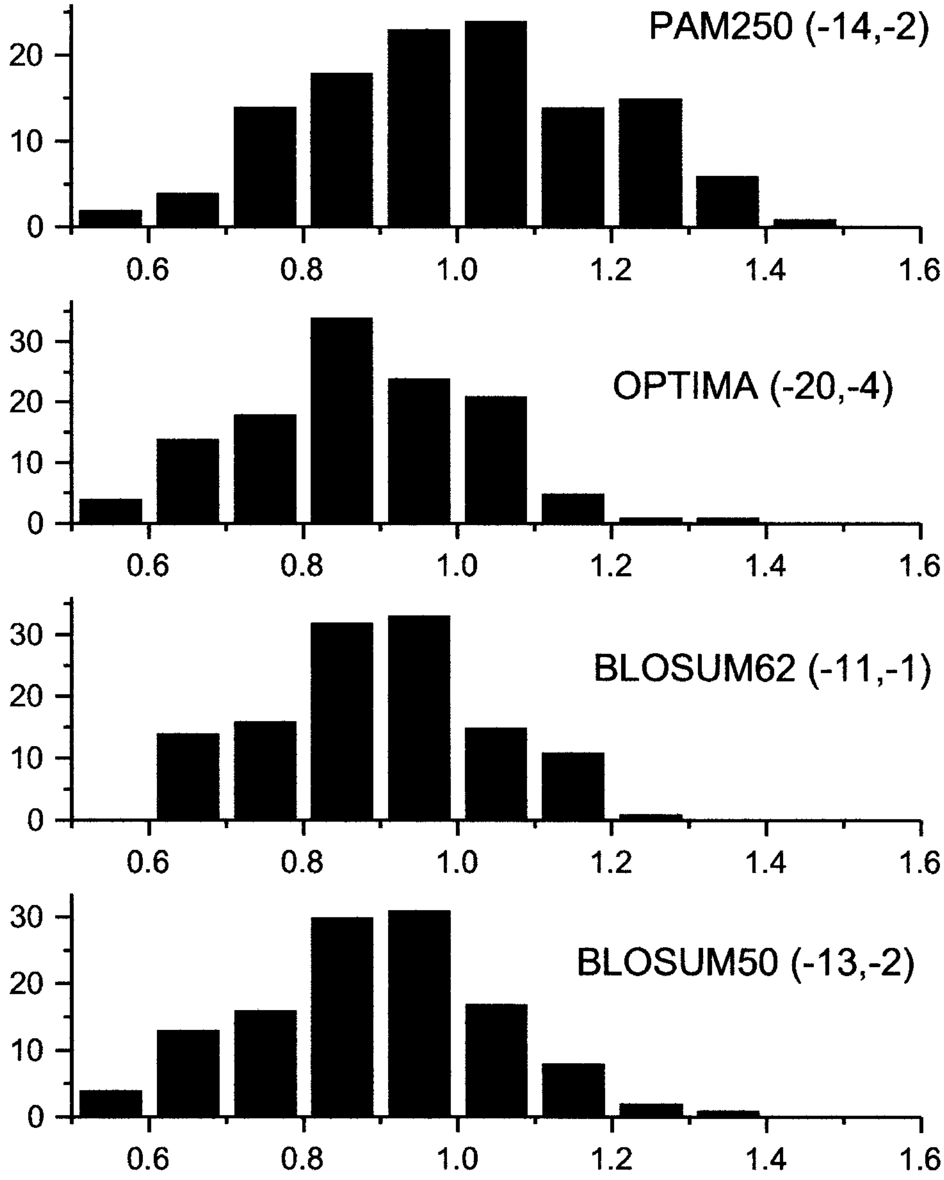

Fig. 3. Distribution of the values of λ for various score functions and gap penalties for scores obtained from the alignment of nonhomologous sequences from the SCOP test dataset.

TABLE II. Comparison of the Performance of Various Scoring Systems With the Hybrid and Smith-Waterman (LDP) Algorithms

		Hybrid			LDP		
Score matrix	Gap penalties	$\langle C\rangle$	$\langle p(S_r > x)\rangle$	$\langle P\rangle$	$\langle C\rangle$	$\langle p(S_r > x)\rangle$	$\langle P\rangle$
OPTIMA[20]	−20.0/−4.0	0.884	0.004	0.125	0.915	0.003	0.091
OPTIMA[20]	−20.0/−3.0‡	0.901	0.002	0.107			
BLOSUM62[25]	−11.0/−1.0	0.840	0.005	0.172	0.857	0.005	0.157
BLOSUM62[25]	−9.0/−2.0‡	0.854	0.005	0.159			
PAM250[1]	−14.0/−2.0	0.613	0.024	0.414	0.835	0.004	0.181
PAM250[1]	−20.0/−2.0‡	0.750	0.010	0.268			
BLOSUM50[25]	−13.0/−2.0	0.830	0.009	0.185	0.888	0.004	0.120
BLOSUM50[25]	−15.0/−2.0‡	0.866	0.006	0.147			

Results are based on a set of distant homologous sequences from COGs test database. Gap penalties correspond to BLAST defaults as well as those penalties found to maximize $\langle C\rangle$ for the hybrid algorithm (‡).

TABLE III. Comparison of the Performance of Various Scoring Systems With the Hybrid and Smith-Waterman (LDP) Algorithms

		Hybrid			LDP		
Score Matrix	Gap penalties	$\langle C\rangle$	$\langle p(S_r > x)\rangle$	$\langle P\rangle$	$\langle C\rangle$	$\langle p(S_r > x)\rangle$	$\langle P\rangle$
OPTIMA[20]	−20.0/−4.0	0.865	0.0004	0.151	0.886	0.003	0.125
BLOSUM62[25]	−11.0/−1.0	0.793	0.0016	0.230	0.816	0.007	0.199
PAM250[1]	−14.0/−2.0	0.687	0.0164	0.329	0.851	0.003	0.164
BLOSUM50[25]	−13.0/−2.0	0.820	0.0020	0.194	0.862	0.007	0.148

Results are based on a set of distant homologous sequences from PFAM test database.

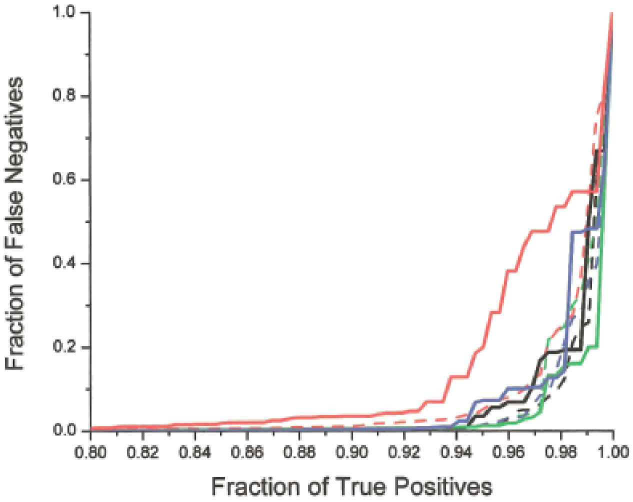

Fig. 4. Sensitivity-sensibility curves (equivalent to an inverse ROC plots). Hybrid method (solid lines) is compared with the Smith Waterman algorithm (dashed lines) by using the following scoring systems: BLOSUM62$_{-11,-1}$[25] (green), BLOSUM50$_{-13,-2}$[25] (blue), PAM250$_{-14,-2}$[1] (red), and OPTIMA$_{-20,-4}$[20] (black). The tests were performed in a database of distant homologs from the COGs database.

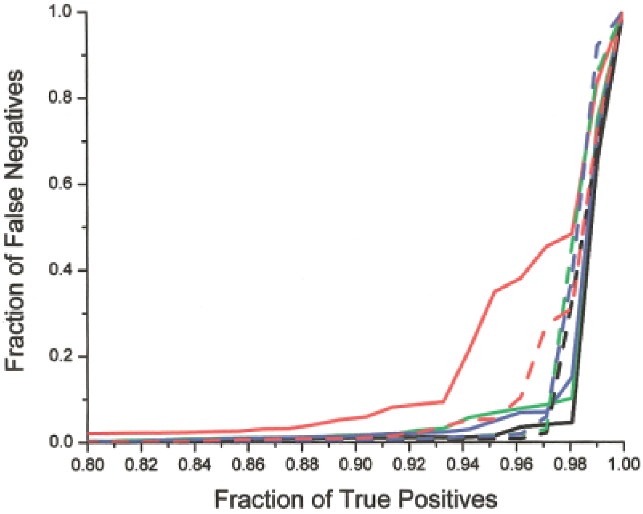

Fig. 5. Sensitivity-sensibility curves (equivalent to an inverse ROC plots). Hybrid method is compared with the Smith Waterman algorithm, for a variety of scoring systems (symbols as in Fig. 4). The tests were performed in a database of distant homologs from PFAM database release 5.2.

of sequences from the SCOP database. Similar to the synthetic sequences, the double log of the pdf of the scores obtained with the hybrid algorithms is well fitted to an EVD as shown in Figure 2. We calculated the appropriate value of λ for each of the target proteins individually. The distribution of λ values for each of the various score functions is shown in Table I and Figure 3. In all the cases, the theory prediction of $\lambda = 1$ is followed within the error.

Performance Evaluation of Hybrid Versus LDP

We then evaluated the performance of the hybrid algorithm on the correct detection of distantly related protein sequences and compared it with local dynamic programming. The values of $\langle C\rangle$, $\langle p(S_r > x)\rangle$, and $\langle P\rangle$ for hybrid and local dynamic programming algorithms for the two different test sets are shown in Tables II and III. To explore the sensitivity of these results to changes in the gap penalty, we took advantage of our ability to determine the null statistics quickly for the hybrid algorithm and performed an exhaustive search over a wide range of gap penalties. The gap penalties that maximized performance for the COGs dataset are also included in Table II.

The sensitivity-sensibility curves or inverse ROC plots for hybrid and LDP algorithms for four different score systems are shown in Figures 4 and 5 for the PFAM and COGs data sets, respectively.

The LDP algorithm seems to perform slightly better than the hybrid algorithm. We also found the ranking of the score systems to be similar for both algorithms, that is, a scoring matrix that performs well with the LDP will perform comparably well with the hybrid algorithm. In particular, we found OPTIMA to outperform other scores systems for both the LDP and hybrid algorithms.

Conclusion

The hybrid method developed by Yu, Bundschuh, and Hwa[18,19] is an alignment algorithm with a well-characterized score statistics, well approximated as an EVD. The fact that λ, one of the most crucial parameters to describe the EVD, can be assumed to be equal to 1 makes it possible to obtain an accurate representation of the null statistics with a small number of simulations. This ability of the hybrid alignment is of special interest for those applications such as PSI-BLAST program[11,12] in which the score function is adjusted after each iteration, and for searches with multiple gap penalties. We have proved that for a real test database, the hybrid alignment statistics are within the error for different score functions and different compositions of the database, even when the statistics are derived from real protein sequences, mimicking the situation of a real database search.

Four scoring systems were chosen for the comparison of hybrid and LDP algorithms. The Smith-Waterman LDP algorithm performs slightly better in the task of distinguish distant pairs of homologs from nonrelated sequences. It is important to note that OPTIMA, the best score function for both the LDP and hybrid algorithms, was optimized for use with LDP. This finding suggests that a similar optimization process could be used to improve performance with the hybrid algorithm.

ACKNOWLEDGMENTS

We thank Bin Qian, Stephen Altschul, Ralf Bundschuh, Yi-Kuo Yu, and Terry Hwa for helpful discussions. We also thank Stephen Altschul, Ralf Bundschuh, Yi-Kuo Yu, and Terry Hwa for providing us with preprints of their manuscripts and copies of their software. Todd Raeker and Michael Kitson furnished computational assistance.

REFERENCES

1. Dayhoff MO, Schwartz RM, Orcutt BC. A model of evolutionary change in proteins. In: Dayhoff MO, editor. Atlas of protein sequence and structure. Washington, DC: National Biomedical Research Foundation; 1978;5(suppl 3):45.
2. Altschul SF. Amino acid substitution matrices from an information theoretic perspective. J Mol Biol, 1991;219:555–565.
3. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:195–197.
4. Karlin S, Altschul SF. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. Proc Natl Acad Sci USA 1990;87:2264–2268.
5. Dembo A, Karlin S, Zeitouni O. Limit distribution of maximal non-aligned two-sequence segmental score. Ann Prob 1994;22: 2022–2039.
6. Altschul SF, Gish W. Local alignment statistics. Methods Enzymol 1996;215:460–480.
7. Gumbel EJ. Statistics theory of extreme values and some practical applications. National Bureau of Standards Applied Mathematics Series 33, Washington, DC: U.S. Government Printing Office, 1954.
8. Pearson WR. Empirical statistical estimates for sequence similarity searches. J Mol Biol 1998;276:71–84.
9. Altschul SF, Bundschuh R, Olsen R, Hwa T. The estimation of statistical parameters for local alignment score distributions. J Mol Biol 2001;29(2):351–361.
10. Altschul SF, Gish W, Miller W, Myers E, Lipman D. Basic local alignment tool. Nucleic Acids Res 1990;215:403–410.
11. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman D. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25: 3389–3402.
12. Schaffer AA, Aravind L, Madden TL, Shavirin S, Spouge JL, Wolf YI, Koonin EV, Altschul SF. Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. Nucleic Acids Res 2001;29:2994–3005.
13. Baldi P, Chauvin Y, Hunkapiller T, McClure MA. Hidden Markov Models of biological primary sequence information. Proc Natl Acad Sci USA 1994;91:1059–1063.
14. Krogh A, Brown M, Mian IS, Sjölander K, Haussler D. Hidden Markov Models in computational biology. J Mol Biol 1994;235: 1501–1531.
15. Eddy SR, Mitchison G, Durbin R. Hidden Markov Models. Curr Opin Struct Biol 1995;6:361–365.
16. Eddy SR. Maximum discrimination Hidden Markov Models of sequence consensus. J Comput Biol 1996;2:9–23.
17. Bienkowska JR, Yu L, Zarakhovich S, Rogers RG Jr, Smith TF. Protein fold recognition by total alignment probability. Proteins 2000;40:451–462.
18. Yu Y-K, Hwa T. Statistical significance of probabilistic sequence alignment and related local Hidden Markov Models. J Comput Biol 2001;8:249–282.
19. Yu Y-K, Bundschuh R, Hwa T. Hybrid alignment: high-performance with universal statistics 2001. Submitted to Bioinformatics for publication.
20. Kann M, Qian B, Goldstein RA. Optimization of a new score function for the detection of remote homologs. Proteins 2000;41: 498–503.
21. Robinson AB, Robinson LR. Distribution of glutamine and asparagine residues and their near neighbors in peptides and proteins. Proc Natl Acad Sci USA 1991;88:8880–8884.
22. Murzin AG, Brenner SE, Hubbard TJP, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–540.
23. Tatusov RL, Galperin MY, Koonin EV. The COG database: a tool for genome-scale analysis of proteins functions and evolution. Nucleic Acids Res 2000;28:33–36.
24. Bateman A, Birney E, Durbin R, Eddy SR, Finn RD, Sonnhammer ELL. PFAM 3.1: 1313 multiple alignments match the majority of proteins. Nucleic Acids Res 1999;27:260–262.
25. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci USA 1992;89:10915–10919.
26. Swets A. Measuring the accuracy of diagnostic systems. Science 1988;240:1285–1293.
27. Brenner SE, Chothia C, Hubbard TJP. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. Proc Natl Acad Sci USA 1998;95:6073–6078.
28. Arratia R, Waterman MS. A phase transition for the score in matching random sequences allowing deletions. Ann Appl Prob 1994;4:200–225.
29. Olsen R, Bundschuh R, Hwa T. Rapid assessment of extremal statistics for gapped local alignment. Proc Int Conf Intel Sys Mol Biol (ISMB) 1999;1999;211–222.

APPENDIX

Local Dynamic Programming

Here, we briefly review the Smith Waterman local dynamic programming algorithm, followed by a description of the modifications made by Yu and coworkers.[18,19] We use the latter's notation for the description of both algorithms.

The Smith and Waterman algorithm[3] with affine gap penalties is the standard when looking for the best alignment between two sequences $\mathbf{a} = a_1 a_2 \dots a_\mathrm{m}$ and $\mathbf{b} = b_1 b_2 \dots b_\mathrm{n}$ of length M and N. Because of its ability to find the best local alignment between two substrings, this local version of dynamic programming algorithm is of special interest in biology, especially when aligning sequences with long evolutionary distances that are more likely to conserve only local regions with high sequence similarity.

To obtain the final score and the alignment, the LDP algorithm uses a score function ($\mathrm{sim}(a_m, b_n)$) and a set of penalties d and e for the initialization and extension of the gaps, respectively. When aligning two sequences, there are only three possible choices at each step, either amino acid a_m is aligned to b_n, a gap is created in sequence $\mathbf{b}$ or a gap is placed in sequence $\mathbf{a}$; we refer to those choices as substitution (S), deletion (D), or insertion (I). At each step of dynamic programing a matrix $H\,(m, n)$ is constructed for each of these cases, and the best choice is stored through the following recursion

$$\begin{aligned} H^s_{m,n} = \max\{0,\, & H^s_{m-1,n-1} + \mathrm{sim}(a_m, b_n), \\ & H^D_{m-1,n} + \mathrm{sim}(a_m, b_n), \quad H^I_{m,n-1} + \mathrm{sim}(a_m, b_n)\}, \\ H^D_{m,n} &= \max\{H^S_{m-1,n} - d,\, H^D_{m-1,n} - e\}, \\ H^I_{m,n} &= \max\{H^S_{m,n-1} - d,\, H^I_{m,n-1} - e,\, H^D_{m,n-1} - d\}, \end{aligned} \tag{5}$$

with the following boundary conditions

$$H^S_{0,n} = H^D_{0,n} = H^I_{0,n} = H^S_{m,0} = H^D_{m,0} = H^I_{m,0} = 0 \tag{6}$$

and the final score $S(\mathbf{a}, \mathbf{b})$ is given by

$$S[\mathbf{a}, \mathbf{b}] = \max_{\substack{1 \le i < m \\ 1 \le j < n}} \{H(m, n)\} \tag{7}$$

Hybrid Algorithm

The semiprobabilistic or hybrid algorithm developed by Yu and coworkers[18,19] is an alternative approach that combines the advantages of integrating over a larger subset of possible alignments with a well-understood null statistics. Let us first consider the case without insertions and deletions. One approach to understanding this hybrid algorithm is to interpret the substitution score for each pair of amino acid $\mathrm{sim}(a, b)$ as a log-odds ratio[1,2]

$$\mathrm{sim}(a, b) = \frac{1}{\lambda_{\mathrm{ug}}} \log \frac{q(a, b)}{p(a)p(b)} \tag{8}$$

where $q(a,\ b)$ is the probability that that pair will be observed in a pair of homologous proteins, $p(a)$ is the probability of amino acid a occurring in the target protein, $p(b)$ the probability of b occurring in the database, and $1/\lambda_{\mathrm{ug}}$ is an unknown scaling parameter; the subscript reminds us that we are analyzing the statistics for ungapped alignments. In this way, H is proportional to the log of the ratio of the probability that the observed aligned amino acids would be observed in a given pair of homologs compared to the probability of observing those pairs at random. In an analogous way, in the hybrid alignment we calculate the alignment's weight Z as the likelihood of the observed amino acids given the homologous relationship. We first define all the input parameters and then the recursion to calculate Z.

We convert the log likelihood ratios to simple likelihood ratios through exponentiation.

$$w_{\mathrm{ug}}(a, b) = \exp[\lambda_{\mathrm{ug}}\mathrm{sim}(a, b)] = \left(\frac{q(a, b)}{p(a)p(b)}\right) \tag{9}$$

Because we know that for an ungapped alignment $\Sigma_{a,b}\, q(a, b) = 1$, we can adjust λ_{ug} so it is the unique positive root of the following equation

$$\sum_{a,b} e^{\lambda_{\mathrm{ug}}\mathrm{sim}(a,b)} p(a)p(b) = \sum_{a,b} w_{ug}(a, b)p(a)p(b) = \langle w_{\mathrm{ug}}(a, b)\rangle_0 = 1 \tag{10}$$

where $\langle \dots \rangle_0$ represents the average over the distribution of amino acids in the target and database proteins. In fact, $\{w_{\mathrm{ug}}(a, b)\}$ must fulfill stricter conditions. If we consider $q(a, b|a) = q(a, b)/p(a)$, the conditional probability of the alignment in a pair of homologs of an a and a b given that the target sequence has an a, it is clear that $\Sigma_b\, q(a, b|a) = 1$, or using Eq. 9

$$\sum_b w_{\mathrm{ug}}(a, b)p(b) = 1 \tag{11}$$

Note that Eq. 10 follows directly from this condition using $\Sigma_a\, p(a) = 1$.

We need to represent gap penalties in a similar framework, including their effect in the normalization of the probabilities. Similar to Eq. 9, one can express μ, the weight of gap initialization, and νv, the weight of gap extension, as

$$\mu = \exp[-\lambda_{\mathrm{ug}}(d + e)] \tag{12}$$

$$\nu = \exp[-\lambda_{\mathrm{ug}}e] \tag{13}$$

where d and e are the affine gap penalties used in LDP algorithm.

One also needs to define μ^{I1} and μ^{D1} associated with the penalties to terminate an insertion or deletion, respectively, as well as μ^{I2} and μ^{D2}, the costs for creating a deletion or insertion. In the LDP algorithm we assigned no penalties in those situations. In the hybrid approach, those penalties must fulfill the following condition

$$\mu^{D1} \cdot \mu^{D2} = \mu = \mu^{I1} \cdot \mu^{I2} \tag{14}$$

where μ represents the cost for starting a gap.

In addition, for the probability weights Z to be interpreted as a likelihood, the sum of all the possibilities at each point in the alignment should be equal on average to 1. $w_{ug}(a, b)$ must be replaced by $w(a, b)$ which must satisfy

$$\langle w(a, b)\rangle_0 + \mu^{I2} + \mu^{D2} = 1, \tag{15}$$

$$\mu^{I1}\langle w(a, b)\rangle_0 + \nu v = 1, \tag{16}$$

$$\mu^{D1}\langle w(a, b)\rangle_0 + \nu + \mu^{I2}\mu'\mu^{D1} = 1. \tag{17}$$

The sum over all the possibilities after a substitution, which are either to align the next pair of amino acids $\langle w(a, b)\rangle$ or create a gap in one of the sequences (μ^{I2} or μ^{D2}), is represented by Eq. 15. Similarly, insertions and deletions can only be either extended or terminated and followed by a substitution; the total sum is represented by Eq. 16 and 17, respectively. A third possibility after a set of gaps in either sequence is to start a set of gaps in the other sequence. This is represented by the double gap cost parameter $\mu' \, \varepsilon \, \{0, 1\}$, which is set to 1 when double gaps are allowed and to 0 when they are not.

These constraints, combined with Eq. 14, determine all the input parameters as following

$$\mu^{I1} = [(1 + \mu - \nu)^2 + (\mu' - 1)\mu^2]/(1 - \nu), \tag{18}$$

$$\mu^{D1} = [(1 + \mu - \nu)^2 + (\mu' - 1)\mu^2]/(1 + \mu'\mu - \nu), \tag{19}$$

$$\mu^{I2} = \mu(1 - \nu)/[(1 + \mu - \nu)^2 + (\mu' - 1)\mu^2], \tag{20}$$

$$\mu^{D2} = \mu(1 + \mu'\mu - \nu)/[(1 + \mu - \nu)^2 + (\mu' - 1)\mu^2], \tag{21}$$

and

$$\langle w(a, b)\rangle_0 = \frac{(1 - \nu)^2}{(1 + \mu - \nu)^2 + (\mu' - 1)\mu^2}. \tag{22}$$

Therefore, to fulfill the conservation condition one needs to choose $w(a, b)$ so that

$$w(a, b) = \frac{(1 - \nu)^2}{(1 + \mu - \nu)^2 + (\mu' - 1)\mu^2}\frac{q(a, b)}{p(a)p(b)}. \tag{23}$$

Finally, substitution weights Z can be interpreted as a likelihood and calculated as the product of the gap weights (defined by Eq. 18–21) times the substitutions weight defined by each pair $(w(a, b))$ using the following recursion.

$$\begin{aligned} Z^S_{m,n} &= 1 + \eta W(a_m, b_n) \\ &\qquad \times [Z^S_{m-1,n-1} + \mu^{D1}Z^D_{m-1,n-1} + \mu^{I1}Z^I_{m-1,n-1}], \\ Z^D_{m,n} &= \mu^{D2}Z^S_{m-1,n} + \nu Z^D_{m-1,n}, \\ Z^I_{m,n} &= \mu^{I2}Z^S_{m,n-1} + \nu Z^I_{m,n-1} + \mu\mu'\mu^{I2}\mu^{DI}Z^D_{i,j-1} \end{aligned} \tag{24}$$

with the following boundary conditions for local alignments

$$Z^D_{m \le 0, n \ge 0} = Z^D_{m \ge 0, n \le 0} = Z^I_{m \le 0, n \ge 0} = Z^I_{m \ge 0, n \le 0} = 0 \tag{25}$$

$$\begin{gathered} Z^S_{m \prec 0, n \ge 0} = Z^D_{m \ge 0, n < 0} = 0, \\ Z^D_{m = 0, n \ge 0} = 1, Z^D_{m \ge 0, n = 0} = 1. \end{gathered} \tag{26}$$

The alignment maximum likelihood score S is computed as

$$S[\mathbf{a}, \mathbf{b}] = \max_{\substack{1 \le i < m \\ 1 \le j < n}} \{\ln Z_{m,n}\} \tag{27}$$

where

$$Z_{m,n} = Z^S_{m,n} + Z^D_{m,n} + Z^I_{m,n}. \tag{28}$$

In the probabilistic approach, one simply adds all the probabilities for each alignment as in Eq. 28 and adds, instead of optimizing, over all possible alignments. Hence, replacing Eq. 27 by

$$e^S = \sum_{i,j} Z_{i,j} \tag{29}$$

will turn the hybrid algorithm into a similar version of the probabilistic local alignment.

The various score matrices used in this study did not fulfill the condition represented by Eq. 23. To properly normalize the score functions, the scores were rescaled by using the factor λ_{ug} calculated by using Eq. 10. We then adjusted the main diagonal terms so that the modified w_{ug} (a, b) fulfill the condition described by Eq. 11. $w(a, b)$ and the other various score parameters were then calculated by using Eq. 12–23. To calculate the scores with hybrid algorithm, we first calculated the probability of all the alignments ending with a certain pair of amino acids ($Z_{m,n}$) by using Eq. 24 and summed over all the alignment ending in that pair (Eq. 28). The maximum likelihood score was then obtained by using Eq. 27 similarly to the last step in LDP algorithm.

Correction for Sequence Length

When a target sequence is aligned to unrelated sequences, the average score will increase with increasing length of the proteins. We need to adjust our null distribution to take this correlation into account. The scores can increase with sequence length either linearly or logarithmically, with the former applying when the best alignment behaves like a global alignment.[28] For our purposes, we assume a logarithmic dependence. In general, one wants to find the parameters that define the length dependence of the scores and adjust either the values of M and N or the EVD distribution parameters, or alternatively adjust the raw scores.

Recent work by Olsen et al.[29] introduced the island method for the estimation of the parameters λ and K. A simple modification in the algorithm in question (either to LDP or to the hybrid algorithms) can be made to keep track of all the alignments starting at a certain amino acid pair with score > 0 (the islands). In particular, one can keep track of the number of sequences $q(S_i > x)$ with a score $> x$, which has a Poisson distribution with a mean

$$\langle q(S_i > x)\rangle \approx KMNe^{-\lambda x} \tag{30}$$

from which the values of λ and K can be estimated. Because the average random scores of high scoring islands has a linear relation with length that can be described as $\langle\sigma\rangle(l) \approx \alpha l + \beta$, one can estimate α and β, the length dependence parameters, from the slope and intersection of the linear fit of $\langle\sigma\rangle(l)$ versus the length (This procedure is described in more detail by Altschul et al.[9]).

To calculate the statistical parameter $p(S_r > x)$ when using the LDP algorithm, we first calculated λ, K, α, and β as described by Altschul et al.[9] (software provided by the authors). We then applied the length correction parameters α and β to the lengths of the sequences, M and N, by using the following equation to obtain the "effective" lengths M' and N'

$$M' = M - \alpha x - \beta \tag{30a}$$

$$N' = N - \alpha x - \beta. \tag{30b}$$

We then used the effective lengths in conjunction with Eq. 2 to obtain the probability $p(S_r > x)$ that the score of a random alignment is greater than the score x in question.

In an alternative approach, Yu and coworkers describe how the parameters characterizing the EVD, K, and λ can be adjusted for the values of M and N.[18] As in LDP, α and β were first estimated from the slope and intersection of the linear fit of the average score of the high scoring islands $\langle\sigma\rangle(l)$ versus the length. $\lambda(M, N)$, the length-adjusted values of K and λ, can then be calculated as

$$\lambda(M, N) = 1 + \frac{1}{(\alpha M + \beta)} + \frac{1}{(\alpha N + \beta)}. \tag{31}$$

Given an estimate of (), $K(M,N)$ can then be determined rapidly via the expression

$$\langle\sigma\rangle = \frac{[\ln K(M, N)MN + \gamma]}{\lambda(M, N)} \tag{32}$$

where $\gamma \cong 0.5772$ is the Euler's constant. K_∞ can be then be obtained by using

$$K(M, N) = K_\infty\left(1 + \frac{\beta}{\alpha M}\right)\left(1 + \frac{\beta}{\alpha N}\right). \tag{33}$$