Published in final edited form as:
FEBS J. 2005 October ; 272(20): 5101–5109.

Protein Database Searches Using Compositionally Adjusted Substitution Matrices

Stephen F. Altschul, **John C. Wootton**, **E. Michael Gertz**, **Richa Agarwala**, **Aleksandr Morgulis**, **Alejandro A. Schäffer**, and **Yi-Kuo Yu**
National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD 20894

Abstract

Almost all protein database search methods use amino acid substitution matrices for scoring, optimizing, and assessing the statistical significance of sequence alignments. Much care and effort has therefore gone into constructing substitution matrices, and the quality of search results can depend strongly upon the choice of the proper matrix. A long-standing problem has been the comparison of sequences with biased amino acid compositions, for which standard substitution matrices are not optimal. To address this problem, we have recently developed a general procedure for transforming a standard matrix into one appropriate for the comparison of two sequences with arbitrary, and possibly differing compositions. Such adjusted matrices yield, on average, improved alignments and alignment scores when applied to the comparison of proteins with markedly biased compositions.

Here we review the application of compositionally adjusted matrices and consider whether they may also be applied fruitfully to general purpose protein sequence database searches, in which related sequence pairs do not necessarily have strong compositional biases. Although it is not advisable to apply compositional adjustment indiscriminately, we describe several simple criteria under which invoking such adjustment is on average beneficial. In a typical database search, at least one of these criteria is satisfied by over half the related sequence pairs. Compositional substitution matrix adjustment is now available in NCBI's protein-protein version of BLAST.

Keywords

substitution matrices; compositional adjustment; protein database searches; BLAST; BLOSUM

Introduction

With the introduction in 1970 of protein alignment algorithms [1], a need was created for matrices of amino acid substitution scores. Over time, many different rationales were advanced for constructing such matrices [2–8], based on a variety of considerations, such as the genetic code and amino acid physico-chemical properties. However, for many years the *log-odds* matrices [4] derived from the PAM model of protein evolution [3] gained the widest use. These matrices were generally employed as well, unaltered, with the local alignment methods introduced in the 1980s [9], which largely supplanted the earlier global alignment algorithms.

The statistical theory of ungapped local alignment scores described in the early 1990s [10, 11] demonstrated that all local alignment matrices are implicitly of the log-odds form, and are optimized for the recognition of alignments characterized by certain amino-acid-pair *target frequencies* [12]. It could then be recognized that what had given the PAM matrices an edge

Corresponding author: Stephen F. Altschul; altschul@ncbi.nlm.nih.gov; Tel: (301) 435-7803; Fax: (301) 480-2288

was their explicit and purposeful, rather than implicit, specification of target frequencies. Accordingly, the subsequently described BLOSUM matrices [13] retained the log-odds formalism for constructing substitution scores, and replaced only the PAM model for estimating target frequencies. This has been true as well of other approaches to constructing substitution matrices [14–17].

The sensitivity of a protein database search can depend strongly on the choice of a substitution matrix [18, 19]. The BLOSUM and other commonly used matrices, constructed from particular sets of related proteins, are tailored to target frequencies in the context of implied standard *background* amino acid compositions. When used to compare proteins with markedly non-standard compositions, these matrices have new target frequencies which are incompatible with the new compositional context, implying non-optimal performance [20].

Proteins with non-standard compositions are far from rare. They may arise in specialized (e.g. hydrophobic or cysteine-rich) protein families, or wholesale in organisms with AT- or GC-rich genomes [21, 22]. For the analysis of such proteins, we have previously described a rationale and an efficient algorithm, improved here, for transforming a standard matrix into one appropriate for any specified non-standard compositional context [20, 23]. This procedure is fully applicable to the comparison of proteins with differing compositions, in that case yielding asymmetric substitution matrices. On average, when used to compare proteins with markedly biased compositions, the adjusted matrices yield alignments that are in better agreement with structural evidence and that have higher scores [20].

An important factor in the effectiveness of protein database programs is the evolutionary distance for which the substitution matrix employed is tailored. This is conveniently measured by the matrix's relative entropy [12, 24]. When adjusting a standard matrix for compositional bias, one may simultaneously control its relative entropy [20, 23], and we here discuss various rationales for doing so. Among the relative entropy strategies we consider, the best on average is to fix the relative entropy of adjusted matrices at a standard value.

Finally, we study the effectiveness of compositional adjustment in the context of general purpose protein database searches, in which there is no expectation of pervasive strong compositional biases. Although it is not advisable to employ compositional adjustment universally, we describe several simple criteria for invoking such adjustment, which predict its utility for a majority of pairwise comparisons of related proteins. Compositional score matrix adjustment has been added as an option to NCBI's protein-query protein-database BLAST program [25, 26].

Statistical Underpinnings

For ungapped local alignments, a statistical theory of substitution matrices has been developed, which assumes a random protein model in which the twenty amino acids appear independently with background probabilities $\vec{p}$ [10, 11]. A substitution matrix should have negative expected score, and can then always be written in the form

$$s_{ij} = \frac{1}{\lambda} \ln \frac{q_{ij}}{p_i p_j} \tag{1}$$

where the implicit q_{ij} are positive target frequencies that sum to 1, and the positive parameter λ provides a natural scale for the matrix. This matrix is optimal for distinguishing from chance those local alignments whose aligned amino acid pairs appear with frequencies characterized by **q**. In practice, equation (1) is widely used to construct log-odds matrices after estimating target and background frequencies directly from carefully curated sets of "true" biological alignments. The target frequencies generally are estimated as symmetric, with $q_{ij} = q_{ji}$, and the background frequencies are then generally chosen to be consistent with the target frequencies, with $p_i = \Sigma_j q_{ij}$.

Because different evolutionary distances imply different target frequencies, sets of substitution matrices, such as the PAM [3, 4] and BLOSUM [13] series, have been optimized for differing degrees of evolutionary divergence. The relative entropy of a matrix [12], defined as $H = \Sigma_{ij} q_{ij} \ln(q_{ij} / p_i p_j)$, with the unit of *nats*, is a convenient parameter for characterizing the evolutionary distance to which the matrix corresponds; the higher H, the lesser the degree of evolutionary divergence.

Compositionally Adjusted Matrices

Generalizing to the comparison of sequences with possibly unequal background compositions $\vec{P}$ and $\vec{P}'$, it is reasonable to assume that the target frequencies **Q** best characterizing true alignments will be consistent with these background frequencies, so that

$$\sum_j Q_{ij} = P_i; \quad \sum_i Q_{ij} = P'_j. \tag{2}$$

We call a substitution matrix *valid* in the context of the background frequencies $\vec{P}$ and $\vec{P}'$ if its implicit target frequencies satisfy equations (2). Except for certain degenerate cases unimportant in practice, a substitution matrix can be valid in only a unique context [20, 23]. This implies that it is not ideal to use a substitution matrix derived from standard target and background frequencies in a non-standard context, but leaves open the question of how to construct an appropriate matrix.

For the comparison of proteins with biased compositions, it is possible to replicate the PAM or BLOSUM procedure by constructing special sets of true alignments for such proteins, as has been described for hydrophobic and transmembrane proteins [27, 28]. From such alignment sets, target and background frequencies may be extracted. Problems with this approach are that it is laborious, that each new context requires a new curatorial effort, and that it is difficult to apply consistently to the comparison of proteins with differing amino acid biases. Accordingly, we have proposed a rationale for automatically transforming any standard matrix, constructed using equation (1) with a unique valid **q**, into a matrix valid in a non-standard context, specified by new background frequencies $\vec{P}$ and $\vec{P}'$ [20]. In short, we propose finding new target frequencies **Q** that minimize the Kullback-Liebler distance from the standard **q**, i.e. $\Sigma_{ij} Q_{ij} \ln(Q_{ij} / q_{ij})$, but subject to the consistency constraints of equations (2). In addition, one may wish to constrain the relative entropy of the new substitution matrix to equal some constant H:

$$\sum_{ij} Q_{ij} \ln \frac{Q_{ij}}{P_i P'_j} = H. \tag{3}$$

Previously we have described a Newtonian procedure for this purpose [23]. Here, we have implemented a modified procedure, with improved speed and stability, which we detail below.

Controlling Relative Entropy

If one adjusts a substitution matrix for compositional bias, why might one wish to constrain its relative entropy, and how should one do so? We will study this question by analyzing the performance of four modes of substitution matrix construction (Table 1). For these evaluations, we use the 143 homologous sequence pairs with validated alignments described in [20], which we call the “biaspair143” data set; these pairs were chosen specially for evaluating substitution matrix compositional adjustment and include various compositional biases.

Mode A is simply the standard BLOSUM-62 substitution matrix while modes B–D are versions of BLOSUM-62 compositionally adjusted for each sequence pair (Table 1). In mode B, the relative entropy of the matrix is left unconstrained. In mode C, the relative entropy is constrained to equal a constant, here chosen as 0.44 nats. Finally, in mode D, the relative entropy is constrained to equal that of the standard BLOSUM-62 matrix in the context of the two sequences being compared. The rationale for constraining relative entropy, as in modes C and D, is elaborated below. Note that for mode A, *composition-based statistics* are used to rescale the matrix, as described in [29], so that it has the same ungapped scale parameter λ as the matrices calculated by modes B–D. Therefore, the bit scores and *E*-values for alignments computed by all four modes are accurate and comparable. Note also that modes B–D use pseudocounts for defining $\vec{P}$ and $\vec{P}'$, as described in [20].

For the comparison of any particular pair of related sequences, it is best to use a matrix whose relative entropy reflects the sequences' degree of evolutionary divergence [12, 24]. However, a database search generally entails comparing a query sequence to related sequences diverged to varying extents. If a single matrix is to be employed, it is best to use one focused on alignments near the limits of detectability. The BLOSUM-62 matrix [13], whose standard rounded version has a relative entropy of 0.44 nats, has been found to be among the most effective [18, 19]. Matrices with much larger relative entropies are tuned to alignments so strong that, using most reasonable scoring systems, they will likely be found in any case; those with much smaller relative entropies are tuned to alignments so weak they will likely be missed in any case.

When BLOSUM-62 is compositionally adjusted for a given pair of sequences, there is no guarantee that its relative entropy will remain near 0.44 nats. If the relative entropy decreases, then this is fortunate if the sequences compared are very distantly related, but unfortunate if they are closely related. However, there is no theoretical reason or empirical evidence that, when unconstrained, the relative entropy of a matrix compositionally adjusted for two related sequences will tend to reflect their evolutionary divergence. Therefore, it would seem best on average for the adjusted matrix to retain a relative entropy near 0.44 nats. This is the rationale for employing mode C of compositional adjustment.

Because relative entropy is a key element in the effectiveness of substitution matrices, it can be a confounding factor when trying to establish that compositional adjustment is of value per se. Specifically, when in [20] we compared the performance of the standard BLOSUM-62 matrix to that of compositionally adjusted versions of BLOSUM-62, we faced the possible objection that any observed improvement was due not to the compositional adjustment itself, but rather to incidental changes in relative entropy. This criticism could be leveled at either mode B or C, because when the standard BLOSUM-62 is used in a non-standard compositional context, its implicit relative entropy changes as well. Mode D was designed to deal with this issue. For any particular pair of sequences, with attendent amino acid compositions, BLOSUM-62 will have a particular and calculable implicit set of target frequencies, and therefore a particular and calculable implicit relative entropy *H*. By constraining the relative entropy of the compositionally adjusted matrix to this *H*, one removes relative entropy as a confounding factor when comparing the standard to a compositionally adjusted BLOSUM-62.

In [20] we used mode D for all compositional adjustments, and were therefore able to show that such adjustment is fruitful per se. However, once this has been established, there is little argument in favor of mode D, relative to modes B or C, as a general approach to sequence comparison. To study this issue more fully, we use modes A–D to analyze the biaspair143 data set of related sequence pairs; a summary of the results is presented in Table 2. Composition-based statistics [29] and compositional matrix adjustment yield accurate *E*-values, as shown by the essentially identical score distributions of unrelated sequence pairs for modes A–D [20]. Therefore, it is valid to compare score adjustment strategies using normalized bit scores [24].

For the biaspair143 data set, the mean bit score of modes B and C exceeds that of mode A by approximately 3 bits, whereas mode D yields an average improvement of only about 2 bits. When considered on a case-by-case basis, and ignoring the magnitude of score changes, it is true that mode D improves on mode A the most consistently. This can be understood by recognizing that the relative entropy change implicit in mode A may on occasion be fortuitous. When this is so, it may be a deciding factor in favor of mode A vis-a-vis either modes B or C, but it will not help vis-a-vis mode D. Nevertheless, when one confines attention to only substantial *E*-value changes, of greater than a factor of 10, i.e. score changes greater than 3.3 bits, the case-by-case advantage of mode D is vitiated. We therefore prefer modes B and C to mode D.

Mode B is simpler than mode C both conceptually and algorithmically, and may be preferred in some contexts. However, Table 2 suggests that mode C (with $H = 0.44$ nats) has a slight advantage to mode B by the criteria of mean bit score, and case-by-case improvement vis-a-vis mode A. For this reason, as well as for the theoretical considerations presented above, we will base our further study of compositional adjustment in this paper on mode C.

Search Program Evaluation Protocol

Most of the biaspair143 comparisons include at least one sequence known to have considerable compositional bias [20]. However, the comparisons that arise in general purpose protein database similarity searches are likely on average to have much less bias. Accordingly, to evaluate the utility of compositional adjustment for such searches, we employ two distinct data sets constructed previously. The first is the expert-curated “aravind103” data set [29], consisting of 103 query sequences, and associated true positive lists from a non-redundant version of the yeast (*S. cerevisiae*) proteome. The second is the “astral40” data set [30, 31], based upon the SCOP [32, 33] structure-based protein classification. Only those 3586 astral40 sequences related to at least one other sequence in the set were included as queries; all 4013 astral40 sequences served as the associated test database.

For assessing the accuracy of database search methods, the truncated receiver-operator characteristic ROC_n for *n* false positives [34] has become a popular measure. Here, we compare all queries to their associated test databases, and then calculate ROC_n curves and scores for the pooled results, ordered by *E*-value [29]. Our application of composition-based statistics to database searching requires some parameter tuning, so we use the smaller aravind103 set for development, and the astral40 set for evaluation.

Although the compositional adjustment of a substitution matrix can be accomplished in a small fraction of a second, comprehensive protein sequence databases now have hundreds of thousands of sequences. It would slow down a search program unduly if such an adjustment needed to be performed for each one. Accordingly, and in keeping with the heuristic nature of

BLAST and related programs, we adjust substitution matrices only as a final step. Specifically, BLAST is executed using a standard matrix, and only alignments with a preliminary *E*-value lower than a certain threshold, here set to 100, are passed on to a second step. In this step, the score matrix is adjusted, the query and database sequences are re-aligned, and a final *E*-value is calculated. This heuristic approach rarely alters which matching sequences appear in the output, but it saves execution time. The same approach and much of the same code is used in BLAST when it calculates composition-based statistics [29]. Note that composition-based statistics are applied only if the *E*-value of the initial alignment would not improve, but compositional score matrix adjustment may decrease, as well as increase, the *E*-value. Therefore score matrix adjustment must be invoked for alignments that initially appear far from significant.

Criteria for Invoking Compositional Adjustment

When comparing standard BLOSUM-62 (mode A) to compositionally-adjusted BLOSUM-62 (mode C) on the aravind103 data set, our initial results were unpromising. However, we find that several simple sequence properties, suggested by theoretical considerations, tend to characterize those sequence pairs that profit from score adjustment. Experiment yields three specific criteria for invoking compositional adjustment.

LENGTH RATIO

For related proteins of very different lengths, the longer may tend to contain domains, missing from the shorter, sufficient to render compositional adjustment unreliable. We find that compositional adjustment is on average preferred if the length ratio of the longer to the shorter sequence is less than 3.0.

COMPOSITIONAL DISTANCE

If the amino acid compositions of two sequences are very similar, this may reflect a common organismal or protein family bias. An appropriate, recently developed distance metric [35] for two probability distributions $\vec{r}$ and $\vec{s}$ is given by

$$D^2(\vec{r}, \vec{s}) = \frac{1}{2}\sum_i \left(r_i \ln\frac{2r_i}{r_i + s_i} + s_i \ln\frac{2s_i}{r_i + s_i}\right). \tag{4}$$

Using this measure, we find that compositional adjustment is on average preferred for two sequences if their compositions $\vec{r}$ and $\vec{s}$ have a distance D less than 0.16.

COMPOSITIONAL ANGLE

A common compositional bias in two sequences may be reflected in similar compositional drift vis-a-vis a standard protein composition $\vec{p}$. Given the metric of equation (4), we can use the law of cosines to calculate the angle θ formed by the vectors from $\vec{p}$ to $\vec{r}$ and from $\vec{p}$ to $\vec{s}$:

$$\theta = \cos^{-1}\frac{D^2(\vec{p}, \vec{r}) + D^2(\vec{p}, \vec{s}) - D^2(\vec{r}, \vec{s})}{2D(\vec{p}, \vec{r})D(\vec{p}, \vec{s})}. \tag{5}$$

We find that compositional adjustment is on average preferred for two sequences whose compositions make an angle with the standard composition of less than 70°. Note that in the 19-dimensional amino acid composition space, random departures from the standard composition are likely to be nearly perpendicular, so that 70° in fact represents a strong correlation. Angles substantially larger than 90° may be due to unrelated domains, and so do not on average favor compositional adjustment.

The criteria we have described favoring compositional adjustment are by no means independent. However, there is both a theoretical and an empirical basis for employing each criterion individually, and we therefore invoke compositional adjustment for sequence pairs that pass any of the three. We call this procedure *conditional adjustment*. In practice, for the data sets we studied, the single criterion most likely to trigger compositional adjustment is that of length ratio. For related sequence pairs from the aravind103 data set, approximately 69% pass the conditional adjustment test, and for related but non-identical pairs from the astral40 data set, approximately 98% do. To a large extent, the much greater percentage for astral40 is due to the "processed" nature of SCOP [32, 33]: because this database contains single domains rather than complete proteins, related sequence pairs tend to be similar in length. Note that in generating Table 2, we applied compositional adjustment universally rather than conditionally, because the biaspair143 data set was constructed from organisms with known substantial compositional biases.

In Figures 1a and 1b, we show ROC$_n$ curves for BLAST applied to the aravind103 and the astral40 data sets. For each data set, curves are shown for BLOSUM-62 (BL62) and for conditionally compositionally adjusted BLOSUM-62 (CA-BL62). For aravind103, the ROC$_{100}$ score is 0.521 ± 0.005 for BL62 and 0.530 ± 0.003 for CA-BL62, where standard errors are calculated as described in [29]. For astral40, the ROC$_{10,000}$ score is 0.1148 ± 0.0001 for BL62 and 0.1214 ± 0.0001 for CA-BL62. The different numbers of false positives allowed for pooled search results reflect the relative sizes of the test sets. For the astral40 test set, the difference in ROC$_n$ scores between CA-BL62 and BL62 is statistically significant. The greater effectiveness of compositional adjustment in the astral40 context probably is partly due to the processed nature of SCOP, discussed above.

Examination of Figure 1 suggests that for a given number of true positives, the conditional use of compositional score matrix adjustment reduces the number of false positives by approximately 50%; this corresponds to an average increase of about 1 bit in the score of true but marginally significant alignments. The performance of compositional adjustment in this test, while positive, is weaker than that described in Table 2. This is due to the intentional selection, for the biaspair143 test set, of sequence pairs for which compositional adjustment is particularly suited.

Implementation

We have added compositional substitution matrix adjustment as an option to NCBI's protein-query, protein-database BLAST program, named blastpgp, available at http://www.ncbi.nlm.nih.gov/BLAST/. By default, the program performs no compositional adjustment, but the user may choose to invoke adjustment either universally or conditionally, i.e. for just for those sequence pairs that pass one of the three criteria described above. (When conditional adjustment is chosen and the three criteria fail for a specific match, composition-based statistics [29] are applied to scale the matrix for that match.) In either case, substitution matrices are actually adjusted only for those sequence pairs whose initial (non-adjusted) *E*-values are no more than 10 times the *E*-value specified for reporting a result. Also, the relative entropy of the adjusted matrix is always constrained to equal the relative entropy of the standard matrix specified, in its implicit compositional context. For the standard BLOSUM-62 matrix, this is 0.44 nats (mode C of Table 1).

Previously, we had described a multi-dimensional Newtonian method for calculating compositionally adjusted matrices [23]. However, we have implemented a modified procedure, to achieve greater stability and speed, especially in the worst case. Rather than expressing the target frequencies sought in terms of Lagrange multipliers, and then solving for the multipliers

[23], we instead use the Newtonian method to solve for the target frequencies and Lagrange multipliers simultaneously. A test of the new procedure on 1,000,000 pairs of compositions derived from real proteins showed that it takes an average of 7 iterations to converge, with 15 iterations the maximum number observed. The new procedure is summarized in the Appendix.

Using a single 3.2 GHz Xeon processor (within a four processor pentium 4 PC, with 4GB of RAM), we found that a single compositional adjustment of a standard substitution matrix required on average slightly over one millisecond. In the context of a single BLAST search, hundreds of adjustments may need to be performed, depending upon the number of alignments found with sufficiently low initial *E*-value. Also, some adjustments may add additional overhead in the form of an extra pairwise local alignment. Using the aravind103 data set as representative queries, we executed BLAST on the machine described above to search a frozen non-redundant protein sequence database, with 1,242,768 sequences and 395,571,179 total amino acids. From three runs, the median aggregate execution time was: 1107 seconds for BLAST using mode A, 1164 seconds for conditionally invoked compositional score adjustment, and 1179 seconds for universally invoked compositional score adjustment. In other words, even invoking compositional adjustment universally, the new method on average adds well under 10% to BLAST's running time.

Conclusion

Compositional score matrix adjustment was originally developed for the comparison of sequences with strongly biased compositions, and in this context it may be useful to apply it universally. Here, we have shown that compositional adjustment is useful also in the context of general purpose protein database similarity searches. We have described several simple criteria under which invoking adjustment is recommended, and shown that adding compositional adjustment to the BLAST database search program yields improved retrieval results at a nominal cost in execution time. Future work includes the extension of compositional adjustment to position-specific database search programs such as PSI-BLAST [26], and the investigation of whether compositional adjustment permits lighter use of low-complexity filtering procedures such as SEG [36].

References

1. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 1970;48:443–53. [PubMed: 5420325]
2. McLachlan AD. Tests for comparing related amino-acid sequences. Cytochrome c and cytochrome c 551. J Mol Biol 1971;61:409–24. [PubMed: 5167087]
3. Dayhoff, M. O., Schwartz, R. M. & Orcutt, B. C. (1978) A model of evolutionary change in proteins in Atlas of Protein Sequence and Structure (Dayhoff, M. O., ed) pp. 345–52, Natl Biomed Res Found, Washington, DC.
4. Schwartz, R. M. & Dayhoff, M. O. (1978) Matrices for detecting distant relationships in Atlas of Protein Sequence and Structure (Dayhoff, M. O., ed) pp. 353–58, Natl Biomed Res Found, Washington, DC.
5. Feng DF, Johnson MS, Doolittle RF. Aligning amino acid sequences: comparison of commonly used methods. J Mol Evol 1984;21:112–25. [PubMed: 6100188]
6. Taylor WR. The classification of amino acid conservation. J Theor Biol 1986;119:205–18. [PubMed: 3461222]
7. Rao JKM. New scoring matrix for amino acid residue exchanges based on residue characteristic physical parameters. Int J Peptide Protein Res 1987;29:276–81. [PubMed: 3570667]
8. Risler JL, Delorme MO, Delacroix H, Henaut A. Amino acid substitutions in structurally related proteins. A pattern recognition approach. Determination of a new and efficient scoring matrix. J Mol Biol 1988;204:1019–29. [PubMed: 3221397]

9. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1981;147:195–7. [PubMed: 7265238]
10. Karlin S, Altschul SF. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. Proc Natl Acad Sci U S A 1990;87:2264–8. [PubMed: 2315319]
11. Dembo A, Karlin S, Zeitouni O. Limit distribution of maximal non-aligned two-sequence segmental score. Ann Prob 1994;22:2022–39.
12. Altschul SF. Amino acid substitution matrices from an information theoretic perspective. J Mol Biol 1991;219:555–65. [PubMed: 2051488]
13. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proc Natl Acad Sci U S A 1992;89:10915–9. [PubMed: 1438297]
14. Gonnet GH, Cohen MA, Benner SA. Exhaustive matching of the entire protein sequence database. Science 1992;256:1443–5. [PubMed: 1604319]
15. Jones DT, Taylor WR, Thornton JM. The rapid generation of mutation data matrices from protein sequences. Comput Appl Biosci 1992;8:275–82. [PubMed: 1633570]
16. Muller T, Vingron M. Modeling amino acid replacement. J Comput Biol 2000;7:761–76. [PubMed: 11382360]
17. Crooks GE, Brenner SE. An alternative model of amino acid replacement. Bioinformatics 2005;21:975–80. [PubMed: 15531614]
18. Henikoff S, Henikoff JG. Performance evaluation of amino acid substitution matrices. Proteins 1993;17:49–61. [PubMed: 8234244]
19. Pearson WR. Comparison of methods for searching protein sequence databases. Protein Sci 1995;4:1145–60. [PubMed: 7549879]
20. Yu YK, Wootton JC, Altschul SF. The compositional adjustment of amino acid substitution matrices. Proc Natl Acad Sci U S A 2003;100:15688–93. [PubMed: 14663142]
21. Sueoka N. Directional mutation pressure and neutral molecular evolution. Proc Natl Acad Sci U S A 1988;85:2653–7. [PubMed: 3357886]
22. Wan H, Wootton JC. A global compositional complexity measure for biological sequences: AT-rich and GC-rich genomes encode less complex proteins. Comput Chem 2000;24:71–94. [PubMed: 10642881]
23. Yu YK, Altschul SF. The construction of amino acid substitution matrices for the comparison of proteins with non-standard compositions. Bioinformatics 2005;21:902–11. [PubMed: 15509610]
24. Altschul SF. A protein alignment scoring system sensitive at all evolutionary distances. J Mol Evol 1993;36:290–300. [PubMed: 8483166]
25. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol 1990;215:403–10. [PubMed: 2231712]
26. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25:3389–402. [PubMed: 9254694]
27. Ng PC, Henikoff JG, Henikoff S. PHAT: a transmembrane-specific substitution matrix. Predicted hydrophobic and transmembrane. Bioinformatics 2000;16:760–6. [PubMed: 11108698]
28. Muller T, Rahmann S, Rehmsmeier M. Non-symmetric score matrices and the detection of homologous transmembrane proteins. Bioinformatics 2001;17(Suppl 1):S182–9. [PubMed: 11473008]
29. Schäffer 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. [PubMed: 11452024]
30. Chandonia JM, Walker NS, Lo Conte L, Koehl P, Levitt M, Brenner SE. ASTRAL compendium enhancements. Nucleic Acids Res 2002;30:260–3. [PubMed: 11752310]
31. Green RE, Brenner SE. Bootstrapping and normalization for enhanced evaluations of pairwise sequence comparison. Proc IEEE 2002;90:1834–47.

32. Murzin AG, Brenner SE, Hubbard T, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–40. [PubMed: 7723011]
33. Brenner SE, Chothia C, Hubbard TJ. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. Proc Natl Acad Sci U S A 1998;95:6073–8. [PubMed: 9600919]
34. Gribskov M, Robinson NL. Use of receiver operating characteristic ROC analysis to evaluate sequence matching. Comput Chem 1996;20:25–33.
35. Endres DM, Schindelin JE. A new metric for probability distributions. IEEE Trans Info Theo 2003;49:1858–60.
36. Wootton JC, Federhen S. Statistics of local complexity in amino acid sequences and sequence databases. Comput Chem 1993;17:149–63.
37. Fourer, R., Gay, D. M. & Kernighan, B. W. (2002) AMPL: A Modeling Language for Mathematical Programming, 2nd edn, Duxbury Press, Pacific Grove, CA.
38. Golub, G. H. & Van Loan, C. F. (1996) Matrix Computations, Johns Hopkins University Press, Baltimore, MD.
39. Nocedal, J. & Wright, S. (1999) Numerical Optimization, Springer, New York, NY.
40. Gotoh O. An improved algorithm for matching biological sequences. J Mol Biol 1982;162:705–8. [PubMed: 7166760]
41. Altschul SF, Erickson BW. Optimal sequence alignment using affine gap costs. Bull Math Biol 1986;48:603–16. [PubMed: 3580642]
42. Altschul SF, Bundschuh R, Olsen R, Hwa T. The estimation of statistical parameters for local alignment score distributions. Nucleic Acids Res 2001;29:351–61. [PubMed: 11139604]

Appendix

Our problem is to find a set of target frequencies **Q** that minimizes the Kullback-Leibler distance from a standard **q**, while remaining consistent with a specified pair of background compositions $\vec{P}$ and $\vec{P}'$. In addition, we seek to constrain the relative entropy H of the resulting substitution matrix. We use Newton's method to solve a nonlinear system of equations. This system is composed of 39 linearly independent consistency constraints (2), the constraint (3) that fixes the relative entropy, and a set of 400 equations specifying that the gradient of the Lagrangian function is zero [23]. This yields a set of 440 equations in 440 variables.

Newton's method involves solving a linear system at each iteration to generate a new iterate. It is desirable to reduce the size of the linear system, but this goal should be balanced by the goal of reducing the total number of iterates calculated [37]. In general, Newton's method behaves well on functions that are well-approximated by their derivatives. The relative entropy constraint (3) and the Kullback-Leibler distance both involve terms of the form $x \ln x$ which are well-approximated by their derivatives for most positive x, but are singular at $x = 0$. Reducing the size of the system [23] in the presence of constraint (3) results in the introduction of exponential terms that have singularities and are poorly approximated by their derivatives. Therefore, to reduce the number of iterates required, we propose to solve the 440-equation system directly.

Fortunately, the matrix of the system of linear equations contains few nonzero elements, and these elements occur in a regular pattern. The matrix has the form

$$\begin{pmatrix} D & A^T \\ A & 0 \end{pmatrix},$$

where D is positive definite and diagonal, A is rectangular, and A^T is the transpose of A. One may use block-elimination [38] to transform the matrix of the problem to the form

$$\begin{pmatrix} D & A^T \\ 0 & -AD^{-1}A^T \end{pmatrix},$$

Systems with this matrix may be solved by factoring $AD^{-1}A^T$, a 40 × 40 symmetric positive-definite matrix. It takes roughly half as many operations to factor $AD^{-1}A^T$ as it does to factor the matrix described in [23]. The cost of applying the block-reductions and solving using the block reduced system is less than the cost of evaluating the functions and derivatives in [23], so the optimization method requires less time per iteration.

The only modification to Newton's method required for this problem is explicitly enforcing the positivity of the variables q_{ij}. To obtain a positive iterate, we decrease the magnitude of the displacement suggested by Newton's method whenever necessary [39]. With this modification, the optimization algorithm is robust and efficient in practice.

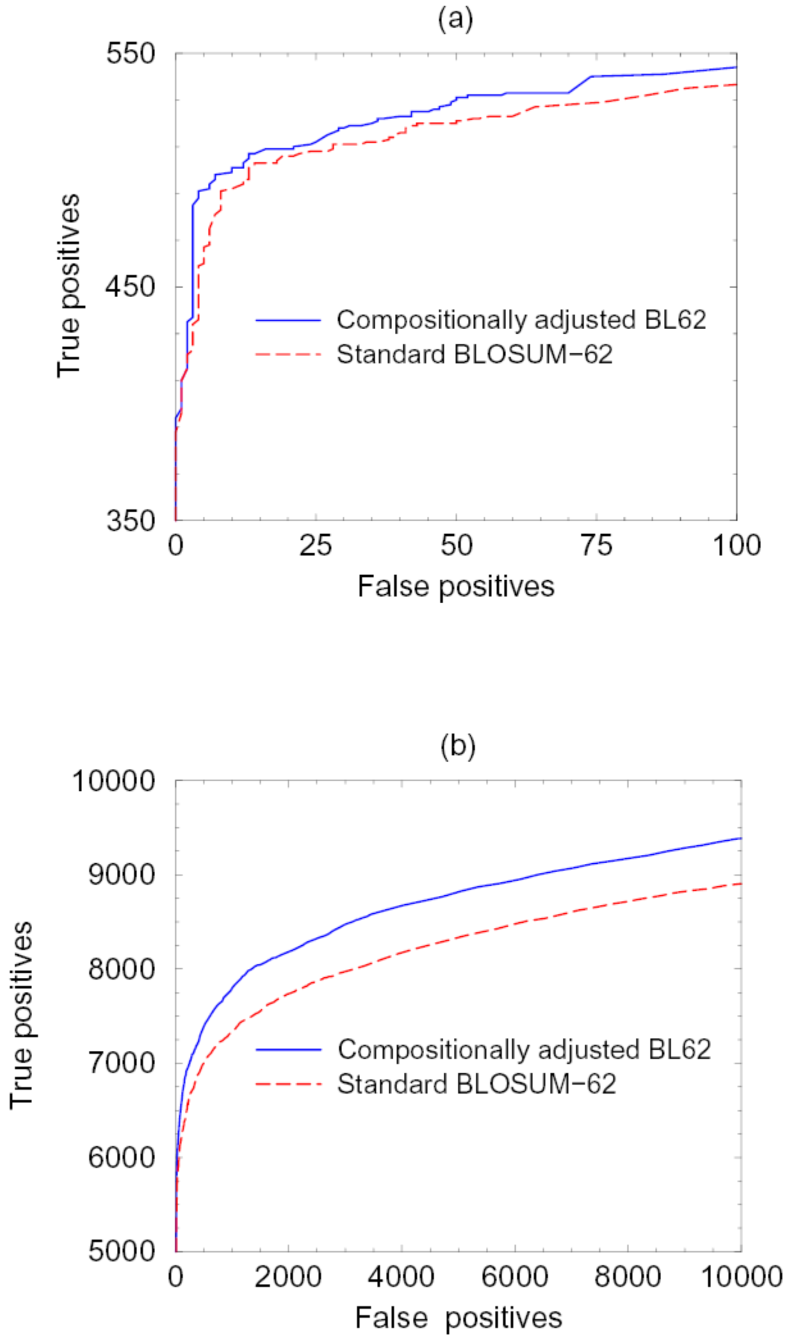

Figure 1. ROC*n* curves for the aravind103 and astral40 data sets using standard BLOSUM-62 and conditionally compositionally adjusted BLOSUM-62

The BLAST program [25, 26, 29] was used to compare the test query sets to the test databases, with database sequences filtered of low-complexity segments using the SEG program [36] with parameters (10, 1.8, 2.1). Search results were pooled and ranked by *E*-value, and ROC_n curves [29, 34] were obtained by plotting true positives versus false positives for increasing *E*-values. For each test set, local alignment scores [9] were calculated using BLOSUM-62 substitution scores [13] and affine gap costs [40, 41]. Composition-based statistics [29] were employed in order to obtain accurate *E*-values. Specifically, for sufficiently high-scoring alignments, the

BLOSUM-62 substitution scores were scaled to have an ungapped λ [10] of 0.006352 in the context of the two sequences being compared, and were used in conjunction with scores of -550-50*k* for a gap of length *k*. Gapped statistical parameters have been estimated for this scoring system using random simulation [42], and scaling arguments [26, 29]. Also, for each test set, a second run was performed with conditionally compositionally adjusted BLOSUM-62 substitution scores, constrained to have a relative entropy of 0.44 nats in the context of the two sequences being compared (mode C). (a) The aravind103 test set was compared to a yeast protein sequence database that had been edited to remove extra copies of highly similar sequences [29]. (b) A subset of 3586 sequences from the astral40 data set [30, 31] was used as queries against astral40; all self-comparisons were excluded.

Table 1

Modes of compositional substitution matrix adjustment.

Mode	Description
A	The standard matrix with no compositional adjustment
B	Relative entropy left unconstrained
C	Relative entropy constrained to equal a constant value
D	Relative entropy constrained to equal that of the standard matrix in the compositional context of the two sequences being compared

Table 2

Performance of substitution matrices on the related sequence pairs of the biaspair143 data set.

Mode	Mean bit Score	Percent of cases improved vis-a-vis mode A	Percent of cases with *E*-value improved/worsened by a factor > 10
A	59.8		
B	62.7	81	40 / 2.1
C	62.9	86	41 / 2.1
D	61.9	88	26 / 1.4