IMPROVING SEQUENCE ALIGNMENTS FOR INTRINSICALLY DISORDERED PROTEINS

PREDRAG RADIVOJAC, ZORAN OBRADOVIC
Center for Information Science and Technology, Temple University, U. S. A.

CELESTE J. BROWN, A. KEITH DUNKER
School of Molecular Biosciences, Washington State University, U. S. A.

Here we analyze sequence alignments for intrinsically disordered proteins. For 55 disordered protein families we measure the performance of different scoring matrices and propose one adjusted to disordered regions. An iterative algorithm of realigning sequences and recalculating matrices is designed and tested. For each matrix we also test a wide range of gap penalties. Results show an improvement in the ability to detect and discriminate related disordered proteins whose average sequence identity with the other family members is below 50%.

1 Introduction

Amino acid sequence alignment is the cornerstone of bioinformatics. Alignment algorithms include optimal pairwise comparisons, either global[1] or local[2], as well as heuristic algorithms such as FastA[3] and BLAST[4]. Optimal multiple sequence alignments[5] suffer from exponential complexity with increasing numbers of sequences. Indeed, the multiple alignment problem is NP-complete; furthermore, a scoring system is difficult to define[6]. These facts gave rise to different suboptimal algorithms based on progressive alignments[7, 8]. Finally, there are sequence profiles[9] and hidden Markov models[10], which exploit position-specific dependencies within protein families. All alignment methods require a scoring system, which is typically adjusted to optimize sensitivity and specificity.

In a given twenty-by-twenty scoring matrix, each entry, s_{ij}, is the score when amino acids i and j are aligned opposite one another. Typically, s_{ij} is a function of

$$\log \frac{p(i, j)}{p(i)\, p(j)}, \tag{1}$$

where $p(i, j)$ is the joint probability[f-1] of the aligned pair of residues i and j, and $p(i)$ the probability of occurrence of residue i. This expression is called the log-odds ratio (mutual information) and when the logarithm is to base two it is measured in bits. The total score of two aligned sequences is finally calculated as the sum of scores of each aligned amino acid pair, along with empirically determined gap-opening and gap-extension penalties that provide the means to accommodate length variability.

[f-1] The scoring function is defined in terms of probabilities that are approximated by observed relative frequencies. We also use terms relative frequency and probability interchangeably.

Two important sets of scoring matrices are the PAM (accepted point mutation) series[11, 12] and the BLOSUM (block substitution) series[13]. The initial PAM matrix was based on just 1,572 substitutions. Evolutionary modeling was then used to boost the data and develop a series of matrices, but this modeling was imprecise[14]. The BLOSUM series was based on 2,106 aligned multiple-sequence segments with more than 15 million amino acid pairs and used only segments in highly conserved regions between gaps (e.g. blocks) to calculate substitution probabilities. Grouping the aligned sequences by sequence identity gave the BLOSUM series. After extensive testing, the BLOSUM62 matrix was identified as the best general scoring matrix[13].

Many additional scoring matrices have been developed[15, 16]. These are based on various criteria such as amino acid properties, structural superposition, minimum number of base changes per codon, evolutionary properties, etc. These matrices have idiotypic advantages, but PAM and BLOSUM remain the most widely used.

The development of scoring matrices has focused on ordered proteins that fold into 3-D structures. In contrast, many proteins have functional regions that exist as a structural ensemble at either the secondary or the tertiary level, that is, these regions are intrinsically disordered[17]. The realization that such disorder is not uncommon and is important for the function of essential proteins has led to a call for the reassessment of the view that function always follows from a protein's 3-D structure[18].

Amino acid compositions for ordered and intrinsically disordered protein are clearly different[19]. Also, insertions and deletions are more common in disordered as compared to ordered regions[20]. Thus, the scoring matrices and gap penalties developed from ordered proteins are likely to be inappropriate for disordered protein. Here we report the first attempts to develop disorder-specific scoring matrices with appropriately weighted gap-opening and gap-extension penalties.

2 Materials and Methods

2.1 Databases, hardware and software

A set of proteins with structurally characterized regions of disorder of length ≥ 40 consecutive residues was identified by database and literature searches. Homologous proteins were compiled using the BLAST algorithm[4]. For proteins with both ordered and disordered regions, it was assumed that segments aligning to the disorder were also disordered. The result was the following database[f-2]: 1d1r (10, 23, 32), 4E binding protein (7, 115, 58), ssDNA binding protein (15, 50, 49), α_tubulin (54, 48, 84), DNA-lyase (7, 40, 80), Bcl-x_L (7, 50, 81), calcineurin (22, 164, 34), cyclin-

[f-2] The database is presented in the format: family name (number of sequences, average sequence length, average sequence identity with all family proteins). The latter two numbers were rounded to the nearest integer. For proteins containing both ordered and disordered regions, only disordered regions were used.

dependent kinase inhibitor (4, 162, 80), chloroperoxidase (2, 41, 44), ubiquinol cytochrome C reductase (5, 45, 73), eukaryotic translation initiation factor 4γ (4, 98, 68), carrot embryonic protein 1 (37, 96, 69), epidermal growth factor (8, 38, 78), Phe-tRNA synthetase (14, 88, 34), flagellin (34, 102, 48), negative regulator of flagellin synthesis (8, 98, 45), fibronectin binding protein C (2, 129, 96), oncogene fos (21, 145, 44), Gly-tRNA synthetase (23, 52, 27), glycine methyltransferase (8, 40, 86), gonadotropin (7, 34, 51), transcription factor VP16 (3, 93, 83), histone 5 (9, 114, 67), HMG14 (6, 101, 64), HMG17 (14, 87, 75), HMGI(Y) (10, 153, 39), HMGT (43, 209, 71), I-tRNA synthetase (23, 39, 33), inosine monophosphate dehydrogenase (52, 174, 40), lactose operon repressor (37, 61, 45), metaminopeptidase (3, 123, 86), HIV1 negative factor (3, 119, 25), osteocalcin (20, 47, 61), transcription factor p65 (4, 127, 41), prion (55, 98, 75), prothymosin α (4, 111, 96), *Pvu* II methyltransferase (12, 24, 22), anti-termination protein N (3, 120, 47), regulator of G-protein signalling 4 (17, 80, 38), acidic ribosomal protein P2β (56, 117, 41), replication protein A (7, 61, 21), southern bean mosaic virus capsid (6, 64, 71), translocase sec61 (9, 44, 47), sindbis virus capsid (6, 101, 45), small heat shock protein (6, 40, 42), sulfotransferase (12, 69, 32), α synuclein (22, 134, 62), tomato bushy stunt virus capsid (7, 58, 64), T-cell receptor α (10, 112, 38), telomere bindin gprotein (5, 39, 61), transcription initiation factor IID (3, 59, 53), thyroid transcription factor (10, 187, 47), Topoisomerase II (26, 99, 29), T-tRNA synthetase (24, 95, 28), yeast heat shock protein (2, 195, 23). Overall, this database contains 55 families with 828 segments of disorder containing 81,491 residues in total. Minimum and maximum observed sequence identities between any two aligned sequences were 10% and 99.53%, respectively.

A set of unrelated proteins was taken from reference 21. This set contains 131 proteins and 26,692 residues.

The various experiments were performed on a Windows based 800 MHz Pentium computer using C++ and MATLAB software packages.

2.2 Scoring Matrices

To build scoring matrices we applied a simple iterative algorithm consisting essentially of two steps:

1) for a given scoring matrix, align every protein in every family to all the other proteins belonging to the same family
2) for a given set of alignments calculate a new scoring matrix

Using BLOSUM62 as the initial matrix, these two steps were repeated until the scoring matrices in two successive iterations remained essentially unchanged.

2.3 Aligning sequences

We use both multiple alignment[7] and a series of global pairwise alignments[1] in Step 1. Both methods result in aligning every residue of every sequence in a family opposite only one residue or gap of every other sequence in the same family. Pairs of aligned sequences are used for calculating the final entries in the scoring matrices.

2.4 Assigning weights to sequences

In a scheme that differs from previous approaches, we assign a weight to every sequence as an inverse of its average sequence identity with all proteins of the same family (including itself). We take a soft approach (without thresholds) as compared to reference 13. Note that this method reduces the influence of large families of highly similar sequences, and all families contribute according to their size.

2.5 Counting mutations

No matter which strategy of alignment is used, substitutions are counted as shown in Fig. 1. Note that no counting is done when a residue is aligned to a gap.

This algorithm is applied to all families of disordered proteins and the overall substitution count matrix M is calculated as the sum of all family count matrices.

Input: family f of n aligned proteins $s_1, s_2, \ldots s_n$
with corresponding weights $w_1, w_2, \ldots w_n$
Output: family count matrix M_f
$M_f \leftarrow 20 \times 20$ zero matrix
for every pair of sequences s_i, s_j
for every two superimposed amino acids x, y
$M_f(x, y) \leftarrow M_f(x, y) + w_i \cdot w_j$

Figure 1. *Calculating family count matrix*

2.6 Constructing the scoring matrix

The elements of the scoring matrix S are calculated for each count matrix M, in the following way. The joint probability matrix Q that residue i will be aligned opposite residue j (for every i and j) is computed as

$$Q = \frac{M + M^T}{2\sum_i \sum_j m_{ij}}, \tag{2}$$

where m_{ij} is an element of matrix M. Adding a transpose matrix M^T and dividing by two annuls any effects from the counting order. The double sum in the denominator normalizes entries of Q to sum to one. From the elements of Q the conditional probabilities of substitutions $p(i \mid j)$ are calculated, yielding the elements of a substitution probability matrix P. In order to adjust the ensuing scoring matrix for longer evolutionary times we can transform matrix P before evaluating expression (1). Modeling the evolution by a discrete time-invariant Markov process with the unknown transition matrix, P can be modified as

$$P = P^{\alpha}, \tag{3}$$

where $\alpha \in (1, \infty)$. This generalizes the idea of Dayhoff *et al.*[11, 12] that models longer evolutionary times by extrapolating from proteins with shorter distances. However, the matrix P is already developed using the available spectrum of divergence from the available data, thus reflecting moderate evolutionary distances. Naturally, since the assumptions about underlying Markov processes do not hold strictly, the model becomes less accurate as α increases.

Even with all amino acid exchanges observed in the database, there is no guarantee that P is positive definite. As a result, in order for expression (3) to be well-defined, a test for all positive eigenvalues is performed before raising P to a non-integer power. Failing the test, although such never occurred, would cancel the powering step. Finally, all entries s_{ij} of a scoring matrix are calculated as

$$s_{ij} = \text{round}\left(C \cdot \log_2 \frac{p(i, j)}{p(i)p(j)}\right), \tag{4}$$

where we multiply by $C = 2$ before rounding to the nearest integer in order to have s_{ij} entries in half bit units. Multiplying by two increases the resolution of the matrix elements before rounding, which is performed for the convenience of using integer arithmetic during the course of alignment. We use the same gap penalty system as for BLOSUM62. Once chosen at the start, the gap penalties are not changed during the refinement of matrix S, making the construction process far less expensive.

2.7 Evaluating the matrices

Scoring matrices were evaluated by building family-specific hidden Markov models (HMMs) from a set of aligned training proteins as described in Fig. 2. Test family proteins as well as a number of unrelated proteins were aligned to the HMM and the resulting discriminatory capabilities were measured as indicated.

Briefly, this testing procedure consists of two steps. In the first step all proteins from the test families and non-family set are assigned a score for each scoring matrix. Reliable performance assessment is achieved by applying a cross-validation procedure that also emulates real situations where only a small number of known homologues are available. Random division (line two in Fig. 2) was the same for

each matrix. In order to build a model we used ClustalW [7] for multiple sequence alignment and the HMMER[10] software for profile HMMs. Aligning a protein to a HMM results in a score and an E-value. While the score reflects the log-likelihood that the query sequence is generated by a HMM the E-value is an estimate of statistical significance of the match. Overall, the best model is the one that provides the smallest overlap between two distributions of scores (family and non-family proteins). However, in a situation when rigid statistical tests are not conclusive, we compared the Z-scores generated by different models. The greater the value of a Z-score the lesser the probability that a query protein is one of the unrelated sequences. The model that best discriminates family from non-family sequences is the one with highest Z-scores. Consequently, in the second step, all length-normalized scores from step one are converted into Z-scores, and the maximum score for each protein is found over all matrices. Then, a cumulative score is calculated for each matrix as indicated in Fig. 2. Note that this score depends on the set of matrices being compared; however it preserves numeric differences and hence the relative order between any two models. Note also that our testing procedure is not optimal for discrimination purposes. As probabilistic models and local optimizers, hidden Markov models can only approximate dependencies in protein sequences. Still, successful application of HMMs to protein representation justifies their use.

for each scoring matrix $S \in \boldsymbol{S}$, corresponding gap penalties, and family $f \in \boldsymbol{f}$
  randomly divide f into 4 equal sized test groups
  for each test group
    multiply align the other 3 groups using S and gap penalties
    construct a HMM based on the multiple alignment
    align proteins in test group to the HMM and record scores*
    align non-family proteins to the HMM and record scores*
  end
end
for each scoring matrix S and family f
  calculate mean (m) and standard deviation (σ) for non-family protein scores
  for each family protein p (whose score is s), calculate $Z_{S,p} = (s - m) / \sigma$
end
for each protein sequence $p \in \boldsymbol{f}$
  $max_p = \max_{S \in \boldsymbol{S}} \{Z_{S,p}\}$
end
for each scoring matrix S
  cumulative score $= \sum_{p \in \boldsymbol{f}} (max_p - Z_{S,p})$
end

Figure 2. *Testing procedure.* * *All scores are length normalized.*

3 Results

In the development of our testing procedures, we compared the performance of several previously published matrices and corresponding gap penalties optimized in reference 15 as applied to the 21 largest families of disorder (Table 1). The matrices were ranked by their cumulative scores (see *2.7*) over all of the test proteins up to a threshold sequence identity that was calculated as the average of pairwise sequence identities between a query protein and the set of training proteins used in model construction. A sequence identity threshold of 50% was set to include a reasonable number of proteins while emphasizing the more divergent ones. Furthermore, although both multiple sequence alignments[7] and a series of optimal pairwise alignments[1] were tried in step one of our iterative procedure (section *2.2*), the latter gave slightly better results so that only these results are presented.

Table 1. *Comparative performance of different matrices with given gap penalties for all test proteins whose average sequence identity to the training sequences is less than 50%*

Matrix	Gap-opening penalty	Gap-extension penalty	Cumulative score
GONNET	6	0.8	55.54
BLOSUM30	9	1	65.34
PAM250	12.5	0.1	68.76
BLOSUM62	7.5	0.9	74.55
BLOSUM62	10	0.6	76.03
BENNER74	7	0.8	77.83
BLOSUM30	10	1.5	79.13
BENNER74	9.5	0.8	83.19
IDENT	12	0.5	83.82
BLOSUM80	7	1.5	87.48
PAM300	12.5	0.4	90.78
IDENT	7	1.4	95.40
PAM250	11	0.5	96.45
PAM120	6	1.4	100.29
GONNET	14	0.2	103.47
PAM300	9	2	119.52
BLOSUM80	14.5	0.04	126.44
PAM120	12.5	1	159.28
OPTIMA *	120	20	276.57

*The scale difference in gap penalties for OPTIMA arises from the ten times greater values used to increase alignment sensitivity

In addition to representatives from the BLOSUM and PAM series, we also evaluated two updates of PAM250, Gonnet *et al.*[22] and Benner *et al.*[23]. The matrix IDENT assigns +6 for a match and −1 for a mismatch, and the OPTIMA matrix was taken from reference 24. For each matrix we used gap penalties from reference 15

and both original and modified all-positive scoring matrices. We also tried gap penalties 12/2 exploited in reference 24 but they exhibited poor performance and are excluded from Table 1, except for the OPTIMA matrix for which they are optimal.

Overall, the results of Table 1 suggest that, of the previously published scoring matrices, the matrix of Gonnet *et al*.[22] performed the best on our disordered protein families with less than 50% sequence identity.

The matrix DISORDER was obtained as described in Section 2 with $\alpha = 1.75$ (α values were tested in steps of 0.25 in the interval from 1 to 2). This new, disorder-specific scoring matrix (Fig. 3) differs significantly from all of the other scoring matrices in Table 1. For example, DISORDER differs in 100 out of 210 positions (47.6%) from BLOSUM62, which was used as the initial matrix in the development cycle (section *2.2*). Differences in values were between –3 and 3, and exhibited an almost normal distribution (not shown).

	C	**S**	**T**	**P**	**A**	**G**	**N**	**D**	**E**	**Q**	**H**	**R**	**K**	**M**	**I**	**L**	**V**	**F**	**Y**	**W**
C	10																			
S	0	3																		
T	1	1	4																	
P	–2	0	–1	6																
A	–1	1	0	–1	3															
G	–3	0	–2	–1	0	5														
N	–1	1	0	–1	–1	0	4													
D	–3	0	–1	–2	–1	–1	1	4												
E	–4	–1	–1	–1	–1	–2	0	2	4											
Q	–3	0	0	–1	–1	–2	1	0	0	5										
H	–1	–1	0	–2	–2	–1	2	–1	–1	1	8									
R	–1	–1	–1	–2	–2	–2	0	–2	–1	1	0	5								
K	–3	–1	0	–1	–1	–2	0	–1	0	0	–1	2	4							
M	0	–2	–1	–2	–1	–4	–2	–4	–3	–1	–2	–1	–2	7						
I	0	–2	–1	–2	–1	–5	–3	–4	–3	–2	–2	–2	–2	1	4					
L	–1	–2	–2	–1	–1	–4	–3	–4	–3	–2	–2	–2	–2	2	2	4				
V	1	–2	0	–1	0	–4	–3	–4	–2	–2	–2	–2	–2	1	3	1	4			
F	–1	–2	–2	–3	–2	–4	–2	–4	–4	–2	0	–3	–3	1	1	1	0	7		
Y	0	–2	–1	–3	–2	–3	–1	–4	–3	0	2	–2	–2	–1	0	0	–1	4	8	
W	–5	–3	–5	–1	–5	–4	–3	–4	–4	–1	–2	0	–3	–1	–2	–2	–4	–1	3	13

Figure 3. *DISORDER matrix*

To compare DISORDER with the other scoring matrices, we measured the performance of each with varying gap penalties and saved the best performing example in each case. The gap-opening penalty was varied over the range from 1 through 14 (in steps of 0.5), and the gap-extension penalty was varied from 0.5 through 2 (in steps of 0.5). The resulting matrices are ranked in Table 2. From the definition given in section *2.7*, it is evident that the cumulative score depends on the set of matrices being evaluated. Thus, the cumulative score values in Tables 1 and 2 cannot be

compared directly: only the rankings are important. The DISORDER matrix outperforms the others, but changes in the gap penalties alter the ranking of the other matrices so that BLOSUM62 now becomes better than the others. DISORDER only marginally outperforms BLOSUM62 by the cumulative score measure.

Table 2. *Comparative performance of matrices with optimized gap penalties for all test proteins with average sequence identity with the training sequences less than 50%*

Matrix	Gap-opening penalty	Gap-extension penalty	Cumulative score
DISORDER	3	0.5	56.54
BLOSUM62	3.5	0.5	57.01
BLOSUM30	2	0.5	57.15
PAM250	1.5	0.5	71.32
GONNET	3.5	0.5	71.59
BENNER74	3	0.5	76.21
GONNET	6	0.8	90.61

The distribution of scores for different HMMs were compared (Fig. 4). Shaded bars represent the number of test proteins for which the DISORDER matrix obtained higher scores when aligned to the appropriate HMM, while white bars represent the same number for the BLOSUM62 matrix. These comparisons are plotted as a function of average sequence identity (quantized into 10 bins) as defined above, in the description of Table 1, but without any threshold.

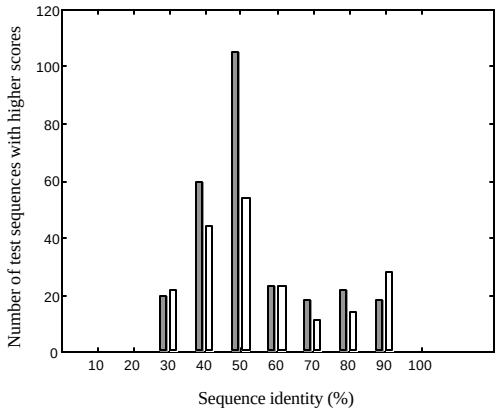

Figure 4. *Comparing DISORDER (shaded) and BLOSUM62 (open)*

Since the DISORDER matrix exhibited the best performance, we further refined its gap penalties. Values of 3.2/0.1 provided the best alignments.

5 Discussion

Although development of a scoring matrix from a set of sequence alignments is straightforward, evaluation of the resulting matrix is not. In reference 13, matrices were tested on an independent dataset of 504 blocks and the matrix that correctly classified a query block to its group the most times for a given level of statistical significance was declared the winner. In reference 24, a scoring matrix was created by maximizing the ability of the system to discriminate between homologous and non-homologous proteins. Performance tests were evaluated on 1,542 pairs of distantly related proteins with less than 40% pairwise sequence identity according to the average confidence value and the probabilities that random scores would be higher than the score for a query homolog. In reference 15 many different matrices were compared using both global and local optimal pairwise alignment algorithms on a database of aligned sequences resulting from superposition of three-dimensional protein structures representing correct alignments. The tests were carried out on 204 structurally aligned proteins from 37 families.

Since our database is rather small as compared to those from references 13 and 24 and since disordered segments are conformational ensembles and so cannot be structurally aligned as was done in reference 15, we developed an alternative method to evaluate the resulting matrices as described herein. The idea behind our evaluation protocol was to mimic how the matrix would likely be used, namely in connection with position-specific modeling.

Reports on new matrices usually contain calculations of the average mutual information (relative entropy, transinformation) and the expected score. The higher values of the average mutual information indicate that the matrix is better adjusted to shorter evolutionary distances. Longer distances, on the other hand, are characterized by smaller differences between diagonal and non-diagonal elements in the transition matrix P, resulting in a smaller relative entropy. The expected score represents an estimate of a per amino acid score of any two aligned proteins with the same distribution of amino acids.

The relative entropy of 0.54 in our matrix is different from that of BLOSUM62 (0.69) and similar to BLOSUM55 (0.56) and PAM180 (0.59). However, these matrices have a different scale so that immediate comparisons are not possible. The expected score obtained for the DISORDER matrix is –0.43.

During the course of designing matrices we have noticed that there is only a small dependence on any individual family in the training set (leaving out any individual family did not change things much), which basically enabled us to test the matrices on the training set. Also, differences in cross-validation steps were small. We have repeated the matrix design procedure with IDENT as the initial matrix and the final results were different at several positions and at most for ±1. The maximum number of iterations was set to 10, but usually a matrix will converge fast to its local

optimum in 47 iterations. In the current paper, the matrix was optimized followed by a separate optimization of gap penalties. Future research will explore optimization of a matrix and gap penalties at the same time, a procedure that should lead to improved alignments. Also, we will continue to enlarge the database of intrinsically disordered segments, which at the very least should improve the statistics.

The quality of multiple alignments is improved by using the DISORDER matrix. Even though the new gap penalties are smaller than are typically used for ordered protein sequences, the average number of gaps in aligned disordered sequences actually decreases when the DISORDER matrix is used. When PAM 001 is used to calculate pairwise genetic distances between sequences aligned by either the GONNET or the DISORDER matrices, the average distances of disordered sequences aligned by the DISORDER matrix are smaller than for GONNET (data not shown).

Over the last several years, we have published several predictors of natural disordered regions (PONDRs)[25-28]. We envision an approach in which order/disorder predictions are first carried out using the most appropriate PONDR. During the subsequent HMM (or profile) construction process, BLOSUM62 (or another suitable matrix) would be used as the initial scoring matrix for those regions predicted to be ordered and DISORDER would be used for those regions predicted to be disordered. As the PONDRs and DISORDER are improved over time, this approach should yield improved alignments for proteins containing regions of intrinsic disorder.

Acknowledgement

NIH Grant 1R01 LM06916 awarded to AKD and ZO and NSF Grant CSE-IIS-9711532 awarded to ZO and AKD are gratefully acknowledged. Chris Oldfield, Sachiko Takayama and others did yeoman's work in developing the database proteins with physically characterized regions of intrinsic disorder. Finally, Slobodan Vucetic is thanked for numerous helpful discussions.

References

1. S. B. Needleman, C. D. Wunsh, *J. Mol. Biol*., **48**, 443 (1970).
2. T. Smith, M. Waterman, *J. Mol. Biol*., **147**, 195 (1981).
3. W. Pearson, D. Lipman, *Proc. Natl. Acad. Sci. USA*, **85**, 2444 (1988).
4. S. F. Altchul, W. Gish, W. Miller, E. W. Myers, D. J. Lipman, *J. Mol. Biol*., **215**, 403 (1990).
5. M. Murata, J. Richardson, J. Sussman, *Proc. Natl. Acad. Sci. USA*, **82**, 3073 (1985).
6. D. Gusfield, Algorithms on Strings, Trees, and Sequences, Cambridge University Press, 1997.

7. J. D. Thompson, D. G. Higgins, T. J. Gibson, *Nucleic Acids Research*, **22**, 4673 (1994).
8. R. Durbin, S. R. Eddy, A. Krogh, G. Mitchison, *Biological sequence analysis: probabilistic models of proteins and nucleic acids*, Cambridge University Press, 1998.
9. M. Gribskov, R. Luthy, D. Eisenberg, *Methods in Enzymology*, **183**, 146 (1990).
10. A. Krogh, M. Brown, I. S. Mian, K. Sjolander, D. Haussler, *J. Mol. Biol.*, **235**, 1501 (1994).
11. M. O. Dayhoff, R. M. Schwartz, B. C. Orcutt, *Atlas of protein sequence and structure*, **5**, suppl. 3, 345 (1978).
12. R. M. Schwartz, M. O. Dayhoff, *Atlas of protein sequence and structure*, **5**, suppl. 3, 353 (1978).
13. S. Henikoff, J. Henikoff, *Proc. Natl. Acad. Sci. USA*, **89**, 10915 (1992).
14. W. J. Wilbur, *Molecular Biology and Evolution*, **2**, 434 (1985).
15. G. Vogt, T. Etzold, P. Argos, *J. Mol. Biol.*, **249**, 816 (1995).
16. R. Luthy, A. D. McLachlan, D. Eisenberg, *PROTEINS: Structure, Function, and Genetics*, **10**, 229 (1991).
17. P. E. Wright, H. J. Dyson, *J. Mol. Biol.* **293**, 321 (1999).
18. R. M. Williams, Z. Obradovic, V. Mathura, W. Braun, E. C. Garner, J. Young, S. Takayama, C. J. Brown, and A. K. Dunker. *Pacific Symp. Biocomputing*, **6**, 89 (2001).
19. A. K. Dunker, J. D. Lawson, C. J. Brown, R. M. Williams, P. Romero, J. S. Oh, C. J. Oldfield, A. M. Campen, C. M. Ratliff, K. W. Hipps, J. Ausio, M. S. Nissen, R. Reeves, C. H. Kang, C. R. Kissinger, R. W. Bailey, M. D. Griswold, W. Chiu, E. C. Garner, Z. Obradovic, *J. Mol. Graphics Model.*, **19**, 26 (2001).
20. W. L. Shaiu, T. Hu, T. S. Hsieh, *Pacific Symp. Biocomputing*, **4**, 578 (1999).
21. V. Geetha, V. Di Francesco, J. Garnier, P. J. Munson, *Protein Engineering*, **12**, 527 (1999).
22. G. H. Gonnet, M. A. Cohen, S. A. Benner, *Science*, **256**, 1433 (1992).
23. S. A. Benner, M. A. Cohen, G. H. Gonnet, *Protein Engineering*, **1**, 88 (1994).
24. M. Kann, B. Quiann, R. A. Goldstein, *PROTEINS: Structure, Function, and Genetics*, **41**, 498 (2000).
25. P. Romero, Z. Obradovic, C. R. Kissinger, J. E. Villafranca, A. K. Dunker, *Proc. IEEE. Int. Conf. on Neural Networks*, **1**, 90 (1997).
26. X. Li, P. Romero, M. Rani, A. K. Dunker, Z. Obradovic, *Genome Informatics*, **10**, 30 (1999).
27. P. Romero, Z. Obradovic, X. Li, E.C. Garner, C. J. Brown, A. K. Dunker, *PROTEINS: Structure, Function, and Genetics*, **42**, 38 (2001).
28. S. Vucetic, P. Radivojac, C. J. Brown, A. K. Dunker, Z. Obradovic, *Proc. Int. Joint INNS-IEEE Conf. on Neural Networks*, Washington D.C., **4**, 2718 (2001).