Journal of Bioinformatics and Computational Biology
Vol. 4, No. 3 (2006) 769–782
© Imperial College Press
ICP Imperial College Press
www.icpress.co.uk

AN AMINO ACID SUBSTITUTION MATRIX FOR PROTEIN CONFORMATION IDENTIFICATION

XIN LIU
Institute of Mechanics, Chinese Academy of Sciences
Beijing 100080, China

WEI-MOU ZHENG
Institute of Theoretical Physics, China, Beijing 100080, China

Received 23 September 2005
Revised 9 January 2006
Accepted 9 January 2006

Amino acid substitution matrices play an essential role in protein sequence alignment, a fundamental task in bioinformatics. Most widely used matrices, such as PAM matrices derived from homologous sequences and BLOSUM matrices derived from aligned segments of PROSITE, did not integrate conformation information in their construction. There are a few structure-based matrices, which are derived from limited data of structure alignment. Using databases PDB_SELECT and DSSP, we create a database of sequence-conformation blocks which explicitly represent sequence-structure relationship. Members in a block are identical in conformation and are highly similar in sequence. From this block database, we derive a conformation-specific amino acid substitution matrix CBSM60. The matrix shows an improved performance in conformational segment search and homolog detection.

Keywords: Amino acid substitution matrix; protein secondary structure; sequence alignment.

PACS number(s): 87.10.+e,02.50.-r

1. Introduction

The similarity of amino acids is the basis for protein sequence alignment, protein design, and protein structure/function prediction. The point accepted mutation matrices of Dayhoff[1] and the substitution matrices of Henikoff[2] from protein blocks are not only the standard choices for amino acid similarity evaluation, but also a basis for amino acid classification.[3,4] Efforts have been made to develop different score matrices, for example, for best fitting distances between aligned sequences,[5] or using also the information of associated evolutionary trees.[6] Most of the matrices widely used in protein design and protein structure prediction are obtained

from sequence samples with certain homologous relationship, while conformation similarity are taken as a secondary consideration.

Protein conformation is more conservative in evolution and is more directly related to function than sequences. It is shown that residue substitution behavior is influenced by protein conformation.[7,8] Several structure-base matrices have been published.[9–11] However, they were generally constructed from limited examples of structure alignment of homologous sets with low sequence identity.

Many sequence-structure motifs have been identified from the nonredundant PDB_SELECT-25 database.[12] Extracting the propensity of residue substitution to conformations from such sequence-structure motifs will be useful for protein structure identification. We collect ungapped similar segments of amino acids from PDB_SELECT[13] database. The protein secondary structures of these residue segments are given in the database of secondary structure in proteins (DSSP).[14] From PDB_SELECT and DSSP, we derive a database of sequence-conformation segments or blocks which explicitly represent sequence-structure relationship. It is our purpose to construct a conformation-based amino acid substitution matrix from such blocks, and examine its ability in protein conformation identification.

2. Materials and Methods

The popular amino acid substitution matrices BLOSUM were constructed from the BLOCKS database,[2] which is derived from the homologous proteins in PROSITE[15] catalog by PROTOMAT[16] algorithm. Counts of residue substitution pairs obtained from ungapped multiple alignments of amino acid segments in each block can be converted to entries of matrices. This approach has also been used in constructing substitution matrices from blocks of structure alignment. However, the alignment of three-dimensional structures, especially the multiple alignment, requires heavy computation. The purpose of the structure alignment is to find residue pairs in similar conformation segments. A feasible approach to find such residue pairs without doing structure alignment is based on the idea of sequence-structure motifs.[12]

We collected a nonredundant set of 1612 nonmembrane proteins from PDB_SELECT with amino acid identity less than 25% issued on September, 25, 2001. The secondary structure for these sequences were taken from DSSP database. In DSSP algorithm, Kabsch and Sander defined eight states of secondary structure according to the hydrogen-bond pattern. As in most methods, we considered three states $\{h, e, c\}$ generated from the eight by the coarse-graining $H, G, I \to h$, $E \to e$ and $X, T, S, B \to c$.

2.1. *Conformational block database*

Two requirements are used in constructing our blocks database: (1) each amino acid segment in a block has the same protein secondary structure; and (2) each

amino acid segment in a block is similar enough with at least one other member in the same block, i.e. the two segments have a high sum of some ungapped pair similarity score (so-called single linkage clustering). A sliding window of width l is used to scan every sequence in the original dataset with the DSSP conformation annotation. Two amino acid segments $A = a_0a_1 \ldots a_{l-1}$ and $B = b_0b_1 \ldots b_{l-1}$ with the same secondary structure $s_0s_1 \cdots s_{l-1}$ is compared by calculating the similarity score

$$T(A, B) = \sum_{i=0}^{i=l-1} \text{Score}(a_i, b_i), \tag{1}$$

where $\text{Score}(a_i, b_i)$ may be the entry of the BLOSUM62 matrix for the residues pair a_i and b_i.[17] If score $T(A, B)$ is above a preset threshold $T*$, A and B are in a same block. Suppose that there are already n_α members in block Σ_α. They must have the same conformation σ. A block has an index being an existing secondary structure segment σ. However, a single conformation index σ may correspond to several blocks each of which containing a class of sequence segments. A new segment C of conformation σ will be compared with all the n_α members in block Σ_α of conformation σ. If one of the n_α members, say A', satisfies $T(A', C) > T*$, the C also belongs to block Σ_α. If no such member exists in block Σ_α, then C will not belong to the block. The new segment C has to be compared with all the existing blocks with the same conformation σ. If C belongs two blocks, say Σ_α and Σ_β, then the two blocks have to be merged as a single block. This can be done by moving all members of Σ_β to block Σ_α, and removing the empty block Σ_β. In this way, each segment will belong to one and only one block. Once a segment is assigned to a block, we skip the whole window by l sites to avoid any overlapping of segments. The obtained blocks represent frequently occurring sequence-structure elements of proteins.

Generally, samples in a block are biased, i.e. many members are very similar to each other. To reduce the bias, those closely related members will be clustered, and counted as a single segment. By specifying the identity rate 60% (which is close to 62% of the BLOSUM62) as another similarity score threshold, another single linkage clustering is conducted within each block. After this second clustering, we obtain our database of blocks.

Entries in the nonredundant database PDB_SELECT are close to the so-called twilight zone. The homologous relationship between any pair of sequences in the database is rather weak. No alignment in a sequence level is performed in constructing the block database. This is a fundamental difference between our method and others.

2.2. *Derivation of the amino acid substitution matrix*

Following the way to derive substitution matrices from blocks, we count all pairs of amino acid substitutions in each column of blocks. All these counts are summed.

The result of this counting forms a frequency table whose entries are the numbers of times for each of the 210 possible different amino acid pairs to occur in all blocks. The table is then used to calculate a matrix representing the logarithmic odds ratio between these observed frequencies and those expected by chance.

Denote by f_{ij} the total number of pairs of amino acids i and j, $1 \leq j \leq i \leq 20$. The observed probability of occurrence for the ij pair is then

$$q_{ij} = f_{ij} \Big/ \sum_{i=1}^{20} \sum_{j=1}^{i} f_{ij}, \tag{2}$$

and the probability for amino acid i is

$$p_i = q_{ii} + \sum_{j \neq i} q_{ij}/2. \tag{3}$$

The expected probability for pair ij by chance is e_{ij}

$$e_{ii} = p_i^2; \quad \text{and} \quad e_{ij} = 2p_i p_j, \quad \text{for } i \neq j. \tag{4}$$

In consistence with the BLOSUM matrices, a log ratio is calculated in unit of half bit as

$$s_{ij} = 2\log_2(q_{ij}/e_{ij}), \tag{5}$$

which is rounded to the nearest integer value to finally produce our conformational blocks substitution matrix (CBSM60). The mutual information per amino acid pair H as a relative entropy, and the expected score E are calculated as

$$H = \sum_{i=1}^{20} \sum_{j=1}^{i} q_{ij} \times s_{ij}, \quad E = \sum_{i=1}^{20} \sum_{j=1}^{20} p_i \times p_j \times s_{ij}. \tag{6}$$

For more details on matrix driving, we refer the reader to Ref. 2.

3. Results

Sample size of conformational blocks is controlled by parameters $T*$ and l. A high $T*$ keeps only highly similar segments in a block, so the block size reduces. Similarly, a larger l results in a smaller block size. In order to obtain enough blocks above a reasonable size, the setting of $T* = 27$ and $l = 10$ is practicable. The block database consists of 3133 blocks and 7235 segments. After clustering within blocks, the final effective number of segments are 4899. The resulting amino acid substitution matrix CBSM60 is shown in Table 1.

3.1. *Comparison of CBSM60 with BLOSUM*

It is interesting to make a comparison between the matrix CBSM60 and the commonly used BLOSUM62. There are many remarkable differences. A lot of amino

Table 1. CBSM60 substitution matrix (Lower) and the difference matrix (Upper) obtained by subtracting the BLOSUM62 from CBSM60 entry by entry.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	
	1	0	−1	−1	−1	0	−1	−1	0	−1	0	0	1	0	−1	1	0	−3	0	0	A
		1	0	−1	−2	0	0	−1	0	−1	−1	0	0	0	−1	0	0	−3	−1	−1	R
A	5		0	0	−4	0	−1	0	0	−2	−1	0	−3	−1	−2	1	0	−2	−2	−2	N
R	−1	6		0	−5	0	0	−1	−1	−4	−3	−1	−2	−2	−2	0	−1	−6	−3	−3	D
N	−3	0	6		0	−5	−3	−1	−2	−1	−1	−4	−1	−4	−2	−2	−1	−3	−2	−2	C
D	−3	−3	1	6		0	0	−1	0	−1	0	0	−1	−3	−3	0	0	−3	−1	−2	Q
C	−1	−5	−7	−8	9		−1	−2	0	−2	−2	0	−1	−3	−2	0	−1	−2	−2	−2	E
Q	−1	1	0	0	−8	5		1	−2	−3	−1	−1	−1	−2	0	0	−1	−2	−2	−1	G
E	−2	0	−1	2	−7	2	4		0	−2	0	0	−1	−1	−2	0	−1	−3	−1	−1	H
G	−1	−3	0	−2	−4	−3	−4	7		0	1	−1	1	0	−3	−1	0	−2	−1	1	I
H	−2	0	1	−2	−5	0	0	−4	8		0	−1	1	0	−3	−1	0	0	−1	1	L
I	−2	−4	−5	−7	−2	−4	−5	−7	−5	4		0	−1	−1	−2	0	0	−4	−1	−1	K
L	−1	−3	−4	−7	−2	−2	−5	−5	−3	3	4		0	1	−2	0	−1	−1	0	0	M
K	−1	2	0	−2	−7	1	1	−3	−1	−4	−3	5		1	0	−1	−1	−1	1	0	F
M	0	−1	−5	−5	−2	−1	−3	−4	−3	2	3	−2	5		0	−1	−1	−7	−3	−2	P
F	−2	−3	−4	−5	−6	−6	−6	−5	−2	0	0	−4	1	7		0	1	0	−1	−2	S
P	−2	−3	−4	−3	−5	−4	−3	−2	−4	−6	−6	−3	−4	−4	7		1	−2	−1	0	T
S	2	−1	2	0	−3	0	0	0	−1	−3	−3	0	−1	−3	−2	4		−1	−1	−3	W
T	0	−1	0	−2	−2	−1	−2	−3	−3	−1	−1	−1	−2	−3	−2	2	6		0	0	Y
W	−6	−6	−6	−10	−5	−5	−5	−4	−5	−5	−2	−7	−2	0	−11	−3	−4	10		0	V
Y	−2	−3	−4	−6	−4	−2	−4	−5	1	−2	−2	−3	−1	4	−6	−3	−3	1	7		
V	0	−4	−5	−6	−3	−4	−4	−4	−4	4	2	−3	1	−1	−4	−4	0	−6	−1	4	
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	

acid pairs have scores more negative in CBSM60 than in BLOSUM62. For example, the scores for pairs CN, CD, ID, WD, QC, KC, FC, WK, and WP are more than two bits lesser in CBSM60. This means that the substitution of dissimilar residues is strongly forbidden by the conservation of conformation. On the other hand, some similar residue pairs, such as SA, SN, LI, VI, MI, ML, VL, FM, YF, and ST have their scores slightly more positive.

An uncorrelated residue pair distribution has a vanishing relative entropy H; H measures the residue pair correlation. As far as H is concerned, CBSM60 is comparable with BLOSUM90, and both have $H \approx 1.2$ (see Table 2). Compared with CBSM60, residues are more conservative in BLOSUM90, which has its diagonal elements more positive. For some amino acid substitutions, especially for PW, WD, and QC, matrix CBSM60 is less tolerant to mismatches than BLOSUM90. On the contrary, for some other residue pairs, such as MA, FM, SN, VL, and VY, substitutions are more tolerable in CBSM60.

3.2. *Detection of homologous pairs*

Another evaluation of substitution matrices is the ability in detecting homologous pairs. All-against-all sequence alignment is carried out on test sets with

Table 2. The difference matrix obtained by subtracting BLOSUM90 from CBSM60 entry by entry.

A	0																		
R	1	0																	
N	−1	1	−1																
D	0	0	0	−1															
C	0	0	−3	−3	0														
Q	0	0	0	1	−4	−2													
E	−1	1	0	1	−1	0	−2												
G	−1	0	1	0	0	0	−1	1											
H	0	0	1	0	0	−1	1	−1	0										
I	0	0	−1	−2	0	0	−1	−2	−1	−1									
L	1	0	0	−2	0	1	−1	0	1	2	−1								
K	0	0	0	−1	−3	0	1	−1	0	0	0	−1							
M	2	1	−2	−1	0	−1	0	0	0	1	1	0	−2						
F	1	1	0	0	−3	−2	−1	0	0	1	0	0	2	0					
P	−1	0	−1	0	−1	−2	−1	1	−1	−2	−2	−1	−1	0	−1				
S	1	0	2	1	−1	1	1	1	1	0	0	1	1	0	0	−1			
T	0	1	0	0	0	0	−1	0	−1	0	1	0	−1	0	0	1	0		
W	−2	−2	−1	−4	−1	−2	0	0	−2	−1	1	−2	0	0	−6	1	0	−1	
Y	1	0	−1	−2	0	1	0	0	0	0	0	0	1	1	−2	0	−1	−1	−1
V	1	−1	−1	−1	−1	−1	−1	1	0	1	2	0	1	1	−1	−2	1	−3	2	−1
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

Blast2.2.6.[18,19] The gap insertion and elongation parameters used for alignment are set to 11/1. The detective ability is illustrated by the number of successfully identified homologous pairs as a function of errors per query.[20] The error per query is defined as the ratio of the total number of nonhomologous protein sequences detected with expectation value not greater than the threshold to the total number of aligned sequence pairs. By varying the expectation value cutoff of Blast, we obtain the curve to measure the detection ability.

For remote homologous sequences in the twilight zone, we examine whether CBSM60 performs better than that BLOSUM62 for sequence alignments. The 176 sequences extracted by Elber[21] are selected as the test set. Each homologous pair in this test set have a sequence identity less than 25%, but bear a very similar structure. The detection results are shown in Fig. 1. CBSM60 is able to find nearly one-third more homologs than BLOSUM62.

The SCOP40[20,22] clustered database developed by Brenner for assessing sequence comparison methods is selected as the test set for further assessment. It contains 1323 proteins assigned to 639 folding families; no two homologous sequences share more than 40% sequence identity. For comparison, CBSM60, BLOSUM62, and other two score matrices of Gonnet and Overington are used in the detection of homologous pairs. The matrix of Overington is also scaled in the unit of half bit as BLOSUM.[17] The results are shown in Fig. 2. As shown in the figure, for common homolog detection where sequence similarity is not so weak, CBSM60 still performs slightly better than other matrices.

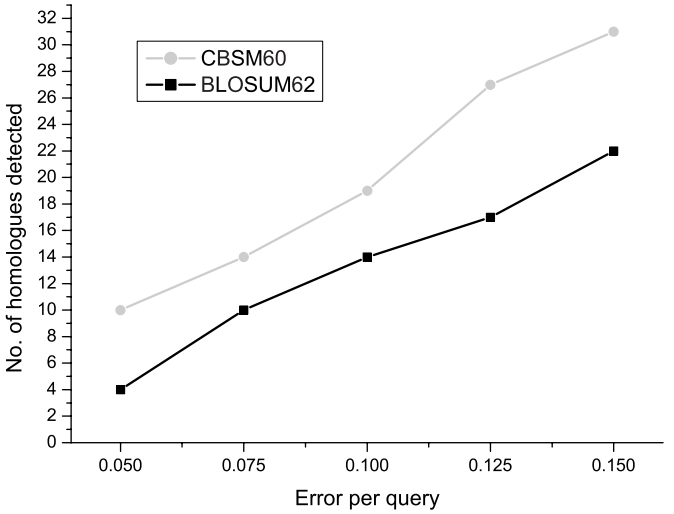

Fig. 1. The number of successfully identified remote homologous pairs in the test set of 176 sequences as a function of errors per query.

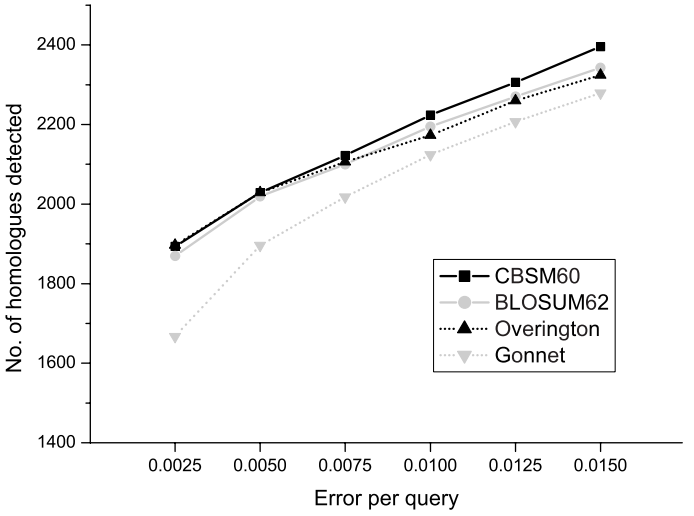

Fig. 2. The number of successfully identified homologous pairs in SCOP40 clustered database as a function of errors per query.

As a specific example to illustrate the performance of CBSM60, the sequence alignment of homologous pair 1EAG_A and 3APR_E by Blast2.2.6 is shown in Table 3. Segment ASEF of 1EAG_A (271–274) has conformation GGGG. This ASEF/GGGG is aligned to DSLV/GGGE of 3APR_E (274–277) by CBSM60.

However, BLOSUM62 aligns the segment to GASF/TEEE of 3APR_E (266–269), a segment of distinct conformation. As a whole, CBSM60 supports more conformation pairs *ee* than BLOSUM.

3.3. *Secondary structure segment search*

If a score matrix is more sensitive to conformation, we would expect that in a set of similar segments extracted according to some threshold score a higher proportion will share the same conformation. As an evaluation of the performance of CBSM60 matrix, we calculate all-against-all pair similarity scores for pairs of width 10, using the original dataset of PDB_SELECT. If two segments A and B have their similarity score $T(A, B) \geq \tau$ and they share the same protein secondary structure, they give one count of 'true positive' (TP); otherwise, a 'false positive' (FP). We then obtain the total counts of TP and FP.

We do the same thing with BLOSUM62 matrix at some threshold τ' to obtain the counts of TP and FP. By varying the threshold τ, we make the total counts of FP to be equal for both the matrices. Consequently, we can evaluate the improvement by the raise in counts of TP. For example, at $\tau' = 23$ BLOSUM62 finds 86,601 TP events and 1,389,714 FP. In the next step, by varying threshold τ, CBSM60 detects 1,368,517 FP samples. At the nearly equal FP, CBSM60 detects 101,745 TPs, gaining an increase of nearly 17.5%. The results of secondary structure segment search are shown in Table 4. The improvement of CBSM over BLOSUM is remarkable. The TP counts increase by nearly 15% at all τ'. Furthermore, the proportion of samples with only tiny structural discrepancy in all the FP cases increases nearly by one percent. For example, the proportions of FP with only one single mismatch in conformation are 6.9 and 7.8% in total pairs for BLOSUM62 and CBSM60, respectively.

4. Conclusion

We have derived from databases PDB_SELECT and DSSP a database of sequence-conformation blocks which explicitly represent sequence-structure relationship. We have constructed an amino acid substitution matrix called CBSM60 from this block database. Compared with other substitution matrices, CBSM60 is more sensitive to conformation, and performs better in protein secondary structure identification and (either remote or common) homolog detection. CBSM60 can be used as an alternative score matrix in protein design and protein structure prediction.

When creating our block database, we have to start with some existing matrix to measure sequence similarity. Here, BLOSUM62 is used at the beginning. We may regard the construction of CBSM60 as the first step of iteration. We may replace BLOSUM62 with the newly found CBSM60 to construct an updated CBSM60, and repeat the procedure until a final convergence is met.

Table 3. Alignments between 1EAG_A and 3APR_E given by Blast2.2.6 with CBSM60 and BLOSUM62. The gap insertion and elongation parameters are set to 11/1. Query = 1EAG_A, Sbjct = 3APR_E. The secondary structures of disgreed alignment are indicated in upper cases. Some identical alignments are omitted.

```
Aligned with CBLM60

Query: 62  TYDPSGSSASQDLNTPFKIGYGDGSSSQGTLYKDTVGFGGVSIKNQV--LADVDSTSI-- 117
SS          xbxgggxttxeeeeeeeeeeexttsxeeeeeeeeeeeeeettеeeeeee--EEEEEEESS--
             YDP+ SS  Q       I YGDGSS+ G L KD V  GG+ IK Q   LA  ++ S
SS          -bxgggxttxeeeeeeeeeeexttsxeeeeeeeeeeeeeettеeeeeeeEEEEEEEXHHHHT
Sbjct: 58  -YDPNQSSTYQADGRTWSISYGDGSSASGILAKDNVNLGGLLIKGQTIELAKREAASFAS 116

Query: 118 ---DGILGVGYKT-NEAGG---SYDNVPVTLKKQGVIAKNAYSLYL-NSPDAATGQIIFG 169
SS          ---sxeeexsxgg GXSSX---SXXXHHHHhhhttsssssseeeeex-xxttxseeeeeet
               DG+LG+G+ T     G     DN    L  QG+I++  + +YL  + +   G+ IFG
SS          SSXseeeexsxggGXSSTTXXXHHHH----hhhttsxsssseeeeeexxgggtxxeeeeet
Sbjct: 117 GPNDGLLGLGFDTITTVRGVKTPMDN----LISQGLISRPIFGVYLGKAKNGGGGEYIFG 172

Query: 225 DLADQIIKAFNGLTQDSNGNSFYEVDCNLSG--DVVFNFSKNAKISV-PASEFAASLDGQ 281
SS          hhhhhhhhhttXEEExttsxeeeeeeesxxxs--EEEEEXSTTXEEEE-EGGGGEEEXXXS
            ++A  V +A+       NG+  Y I C+ S    +VF+ +  A   V P S       GQ
SS          hhhhhhhhhht---Xeexsssxeeexsxggg XXXEEEEET-TEEEEEXGGGGEEEEETTE
Sbjct: 229 NIAASVARAYG---ASDNGDGTYTISCDTSAFKPLVFSIN-GASFQVSPDSLVFEEFQGQ 284

Aligned with BLOSUM62

Query: 62  TYDPSGSSASQDLNTPFKIGYGDGSSSQGTLYKDTVGFGGVSIKNQVL-------ADVDS 114
SS          xbxgggxttxeeeeeeeeeeexttsxeeeeeeeeeeeeeettеeeeeeE-------EEEEE
             YDP+ SS  Q     + I YGDGSS+ G L KD V  GG+ IK Q +       A   S
SS          -bxgggxttxeeeeeeeeeeexttsxeeeeeeeeeeeeeettеeeeeeEEEEEEEXHHHHT
Sbjct: 58  -YDPNQSSTYQADGRTWSISYGDGSSASGILAKDNVNLGGLLIKGQTIELAKREAASFAS 116

Query: 115 TSIDGILGVGYKTNEAGGSYDNVPVTLKKQGVIAKNAYSLYL-NSPDAATGQIIFGGVDN 173
SS          ESSsxeeexsxggGXSSXSXXXHHHHhhhttsssssseeeeex-xxttxseeeeeeteeet
               DG+LG+G+ T             L  QG+I++  + +YL  + +   G+ IFGG D+
SS          SSXseeeexsxggGXSSTTXXXHHHHhhhttsxsssseeeeeexxgggtxxeeeeetxxxg
Sbjct: 117 GPNDGLLGLGFDTITTVRGVKTPMDNLISQGLISRPIFGVYLGKAKNGGGGEYIFGGYDS 176

Query: 229 QIIKAFNGLTQDSNGNSFYEVDCNLSGDVVFNFSKNAKISVPASEFAASLDGQPYD---- 284
SS          hhhhhttXEEExttsxeeeeeeesxxxsEEEEEXSTTXEEEEEGGGGEEEXXXSXTT----
             + +A+       NG+  Y + C+ S        K    S+  + F  S D   ++
SS          hhhhhhtX---eexsssxeeexsxggg------XXXEEEEETTEEEEEXGGGGEEEEETT
Sbjct: 233 SVARAYGA---SDNGDGTYTISCDTSA------FKPLVFSINGASFQVSPDSLVFEEFQG 283
```

Table 4. Counts of segment pairs with complete conformation match (TP) and pairs with mismatch (FP) at different thresholds τ' for BLOSUM62. The thresholds τ for CBSM60 (not shown) are adjusted to make its FP close to the corresponding BLOSUM's FP. Columns 3–13 are proportions counted according to the number of mismatched conformation sites.

		Samples/proportion counted according to the number of mismatched conformation sites											
τ'		0(TP)	1	2	3	4	5	6	7	8	9	10	FP
23	CBSM60	101,745	**7.8**	9.2	10.5	11.0	11.3	11.3	10.7	9.7	7.7	10.7	1,36,8517
	BLOSUM62	86,601	6.9	8.7	10.5	11.4	12.0	12.0	11.3	9.9	7.6	9.7	1,389,714
25	CBSM60	46,999	**8.1**	9.5	10.8	11.1	11.4	11.3	10.6	9.5	7.5	10.2	597,618
	BLOSUM62	39,522	7.2	9.0	10.8	11.6	12.1	12.0	11.1	9.7	7.3	9.2	588,929
27	CBSM60	20,612	**8.5**	9.9	11.0	11.3	11.5	11.3	10.5	9.3	7.2	9.6	242,292
	BLOSUM62	17,841	7.6	9.3	11.0	11.8	12.2	12.0	10.9	9.5	7.0	8.6	243,169
29	CBSM60	9272	**8.9**	10.3	11.2	11.4	11.5	11.3	10.3	9.0	6.9	9.1	98,926
	BLOSUM62	8175	8.1	9.7	11.3	11.8	12.4	12.0	10.6	9.1	6.8	8.1	98,219
31	CBSM60	4356	**9.7**	10.7	11.4	11.6	11.6	11.2	9.9	8.7	6.7	8.5	39,101
	BLOSUM62	3942	8.9	10.2	11.5	11.9	12.5	11.9	10.1	9.0	6.7	7.4	38,688

Unlike PAM or BLOSUM, matrix CBSM60 includes the information of conformation explicitly. Compared with other structure-base substitution matrices, our constructing procedure does not involve in any heavy computation of structure alignment in a whole protein level. In this way, we can calculate matrix entries from a much larger size of samples.

Since the conformation of each column in any block is known, we may group residue pairs according to their conformation. We can then derive substitution matrices for each group separately. The obtained matrices are shown in Tables 5–7.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China.

Appendix

Notable differences are seen among the three matrices. For example, the similarities of CA, SR, MQ, PH, and TP change drastically from helix to sheet. The score of CA is positive in helix, while it is negative in sheet. It is expected that such a set of matrices would be very useful when aligning a query sequence to the sequences whose structures are known. This will be discussed elsewhere.

Table 5. Amino acid substitution matrix CBSM60c for coil state (Lower) and difference matrix (Upper) obtained by subtracting the CBSM60h matrix from CBSM60c entry by entry. Distinct entries are in boldface to guide view.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	
	−1	1	−1	−1	**−3**	0	2	0	1	1	2	2	1	0	1	0	0	2	2	1	A
		1	0	0	1	1	0	−1	−2	2	2	1	1	4	−3	1	0	1	4	3	R
A	4		−2	0	1	0	−1	**−2**	0	3	1	−1	2	2	−2	0	−1	7	0	3	N
R	0	6		−1	2	−1	0	−2	0	1	1	0	1	5	1	−1	−1	4	0	−1	D
N	−3	0	5		−3	−4	2	2	−4	1	0	−1	−1	−5	−6	−4	−2	7	−3	−3	C
D	−3	−3	1	5		1	1	−2	0	2	3	1	**3**	5	1	0	1	−1	3	3	Q
C	−2	−4	−7	−8	8		1	−1	0	4	2	1	3	4	2	0	1	1	1	3	E
Q	−1	2	0	−1	−11	5		−4	−3	−1	−1	0	2	1	1	**−2**	−3	1	−1	1	G
E	0	0	−1	2	−6	3	5		1	−1	1	1	4	0	2	−1	0	4	1	4	H
G	0	−3	−1	−3	−5	−3	−3	5		2	1	0	1	0	2	2	2	4	1	1	I
H	−1	−2	1	−2	−7	0	0	−5	9		1	1	1	1	1	1	0	−2	0	1	L
I	−1	−2	−3	−6	0	−3	−2	−6	−6	5		0	2	1	1	0	1	1	3	1	K
L	0	−1	−3	−6	−1	0	−4	−5	−3	4	5		1	1	−3	1	3	−3	0	1	M
K	0	3	−1	−2	−8	2	1	−3	−1	−4	−2	5		1	3	2	0	**3**	1	1	F
M	0	0	−3	−4	−2	1	−1	−2	−1	3	4	0	6		−6	0	−1	−3	4	3	P
F	−3	−1	−3	−3	−8	−3	−3	−4	−2	0	1	−3	1	8		0	0	3	2	2	S
P	−1	−4	−5	−3	−7	−3	−2	−4	−4	−5	−4	−2	−4	−3	5		−1	2	1	1	T
S	2	0	1	−1	−5	0	0	−1	−2	−1	−2	0	0	−2	−3	4		1	0	4	W
T	0	−1	0	−2	−3	0	−1	−4	−3	0	−1	−1	0	−4	−3	2	5		1	−1	Y
W	−5	−5	−3	−7	−3	−7	−4	−3	−4	−2	−4	−6	−5	2	−11	−2	−2	11		1	V
Y	−1	−1	−4	−5	−5	0	−3	−5	2	−1	−2	−1	−1	5	−5	−2	−2	1	8		
V	1	−2	−3	−6	−4	−2	−2	−3	−2	5	3	−2	2	0	−2	−2	0	−2	−2	5	
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	

Table 6. Amino acid substitution matrix CBSM60e for sheet state (Lower) and difference matrix (Upper) obtained by subtracting the CBSM60c matrix from CBSM60e entry by entry.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	
	1	−1	2	0	−1	1	−1	1	0	−1	−1	−1	0	1	−4	0	0	−1	−3	−1	A
		0	1	1	0	1	2	0	2	−3	−3	0	−1	−2	3	1	0	−2	−1	−1	R
A	5		3	−1	2	2	**2**	**2**	−1	−6	−1	**2**	−1	−4	2	2	0	−3	1	−3	N
R	−1	6		4	2	**2**	0	1	1	−1	−1	1	−1	−5	−1	**2**	−3	1	−3	0	D
N	−1	1	8		2	8	−1	−2	2	−4	−2	3	−2	0	4	−1	0	−1	−3	1	C
D	−3	−2	0	9		1	0	2	1	0	−4	0	1	−3	0	0	−1	5	−3	−2	Q
C	−3	−4	−5	−6	10		1	2	0	−2	−3	1	−5	−3	2	0	1	−1	−2	−2	E
Q	0	3	2	1	−3	6		3	−3	0	0	0	−3	−1	0	**2**	0	−4	0	0	G
E	−1	2	1	2	−7	3	6		−1	3	1	0	−3	−1	**5**	2	2	1	−1	−2	H
G	1	−3	1	−2	−7	−1	−1	8		−3	−2	0	−2	−2	2	−2	−2	−6	−3	−3	I
H	−1	0	0	−1	−5	1	0	−8	8		−1	−2	−1	−1	−3	−1	−1	2	0	−2	L
I	−2	−5	−9	−7	−4	−3	−4	−6	−3	2		1	−2	−1	−3	1	0	−2	−4	−2	K
L	−1	−4	−4	−7	−3	−4	−7	−5	−2	2	4		−1	−1	2	−1	−2	0	−2	−1	M
K	−1	3	1	−1	−5	2	2	−3	−1	−4	−4	6		−2	−2	0	2	**−3**	−2	−2	F
M	0	−1	−4	−5	−4	2	−6	−5	−4	1	3	−2	5		6	1	**4**	7	−1	−6	P
F	−2	−3	−7	−8	−8	−6	−6	−5	−3	−2	0	−4	0	6		1	0	−5	0	−4	S
P	−5	−1	−3	−4	−3	−3	0	−4	1	−3	−7	−5	−2	−5	11		0	−3	−2	−1	T
S	2	1	3	1	−6	0	0	1	0	−3	−3	1	−1	−2	−2	5		−2	−1	−9	W
T	0	−1	0	−5	−3	−1	0	−4	−1	−2	−2	−1	−2	−2	1	2	5		−2	−1	Y
W	−6	−7	−6	−6	−4	−2	−5	−7	−3	−8	−2	−8	−5	−1	−4	−7	−5	9		−2	V
Y	−4	−2	−3	−8	−8	−3	−5	−5	1	−4	−2	−5	−3	3	−6	−2	−4	0	6		
V	0	−3	−6	−6	−3	−4	−4	−3	−4	2	1	−4	1	−2	−8	−6	−1	−11	−3	3	
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	

Table 7. Amino acid substitution matrix CBSM60h for helix state (Lower) and difference matrix (Upper) obtained by subtracting the CBSM60e matrix from CBSM60h entry by entry.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	
	0	0	−1	1	**4**	−1	−1	−1	−1	0	−1	−1	−1	−1	3	0	0	−1	1	0	A
		−1	−1	−1	−1	−2	−2	1	0	1	1	−1	0	−2	0	**−2**	0	1	−3	−2	R
A	5		−1	1	−3	−2	−1	0	1	3	0	−1	−1	2	0	−2	1	−4	−1	0	N
R	−1	5		−3	−4	−1	0	1	−1	0	0	−1	0	0	0	−1	4	−5	3	1	D
N	−2	0	7		1	−4	−1	0	2	3	2	−2	3	5	2	5	2	−6	6	2	C
D	−2	−3	1	6		−2	−1	0	−1	−2	1	−1	**−4**	−2	−1	0	0	−4	0	−1	Q
C	1	−5	−8	−10	11		−2	−1	0	−2	1	−2	2	−1	−4	0	−2	0	1	−1	E
Q	−1	1	0	0	−7	4		1	6	1	1	0	1	0	−1	0	3	3	1	−1	G
E	−2	0	0	2	−8	2	4		0	−2	−2	−1	−1	1	**−7**	−1	−2	−5	0	−2	H
G	0	−2	1	−1	−7	−1	−2	9		1	1	0	1	2	−4	0	0	2	2	2	I
H	−2	0	1	−2	−3	0	0	−2	8		0	1	0	0	2	0	1	0	0	1	L
I	−2	−4	−6	−7	−1	−5	−6	−5	−5	3		−1	0	0	2	−1	−1	1	1	1	K
L	−2	−3	−4	−7	−1	−3	−6	−4	−4	3	4		0	0	1	0	−1	3	2	0	M
K	−2	2	0	−2	−7	1	0	−3	−2	−4	−3	5		1	−1	−2	−2	0	1	1	F
M	−1	−1	−5	−5	−1	−2	−4	−4	−5	2	3	−2	5		0	−1	**−3**	−4	−3	3	P
F	−3	−5	−5	−8	−3	−8	−7	−5	−2	0	0	−4	0	7		−1	0	2	−2	2	S
P	−2	−1	−3	−4	−1	−4	−4	−5	−6	−7	−5	−3	−1	−6	11		1	1	1	0	T

Table 7. (*Continued*)

S	2	−1	1	0	−1	0	0	1	−1	−3	−3	0	−1	−4	−3	4		1	1	5	W
T	0	−1	1	−1	−1	−1	−2	−1	−3	−2	−1	−2	−3	−4	−2	2	6		1	2	Y
W	−7	−6	−10	−11	−10	−6	−5	−4	−8	−6	−2	−7	−2	−1	−8	−5	−4	10		1	V
Y	−3	−5	−4	−5	−2	−3	−4	−4	1	−2	−2	−4	−1	4	−9	−4	−3	1	7		
V	0	−5	−6	−5	−1	−5	−5	−4	−6	4	2	−3	1	−1	−5	−4	−1	−6	−1	4	
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	

References

1. Dayhoff MO, Eck RV, Atlas of protein sequence and structure, *Natl Biolmed Res Found Silver Springs, MD* **3**:33–45, 1968.
2. Henikoff S, Henikoff JG, Amino acid substitution matrices from protein blocks, *Proc Natl Acad Sci* **89**:10915–10919, 1992.
3. Murphy LR, Wallqvist A, Levy RM, Simplied amino acid alphabets for protein fold recognition and implications for folding, *Protein Eng* **13**:149–152, 2000.
4. Liu X, *et al.*, Simplified amino acid alphabets based on deviation of conditional probability from random background, *Phys Rev E* **66**:021906, 2002.
5. Gonnet GH, Cohen MA, Benner SA, Exhaustive matching of the entire protein sequence database, *Science* **256**:1433–1445, 1992.
6. Koshi JM, Goldstein RA, Context-dependent optimal substitution matrices, *Protein Eng* **8**:641–645, 1995.
7. Bowie JU, Lüthy R, Eisenberg D, A method to identify protein sequences that fold into a known three-dimensional structure, *Science* **253**:164–170, 1991.
8. Liu X, *et al.*, Distances and classification of amino acids for different protein secondary structures, *Phys. Rev. E* **67**:051927, 2003.
9. Risler JL, *et al.*, Amino acid substitutions in structurally related proteins. A pattern recognition approach. Determination of a new and efficient scoring matrix, *J Mol Biol* **204**:1019–1029, 1988.
10. Johnson MS, Overington JP, A structural basis for sequence comparisons. An evaluation of scoring methodologies, *J Mol Biol* **233**:716–738, 1993.
11. Prlić A, Domingues FS, Sippl MJ, Structure-derived substitution matrices for alignment of distantly related sequences, *Protein Eng* **13**:545–550, 2000.
12. Bystroff C, Baker D, Prediction of local structure in proteins using a library of sequence-structure motifs, *J Mol Biol* **281**:565–577, 1998.
13. Hobohm U, Sander C, Enlarged representative set of protein structures, *Protein Sci* **3**:522–524, 1994.
14. Kabsch W, Sander C, Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features, *Biopolymers* **22**:2577–2637, 1983.
15. Bairoch A, PROSITE: a dictionary of sites and patterns in proteins, *Nucleic Acids Res* **19**:2241–2245, 1991.
16. Henikoff S, Henikoff JG, Automated assembly of protein blocks for database searching, *Nucleic Acids Res* **19**:6565–6572, 1991.
17. Henikoff S, Henikoff JG, Performance evaluation of amino acid substitution matrices, *Proteins* **17**:49–61, 1993.
18. Altschul SF, Amino acid substitution matrices from an information theoretic perspective, *J Mol Biol* **219**:555–565, 1991.

19. Altschul SF *et al.*, Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, *Nucleic Acids Res* **25**:3389–3402, 1997.
20. Brenner SE, Chothia C, Hubbard JP, Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships, *Proc Natl Acad Sci* **95**:6073–6078, 1998.
21. Teodorescu O *et al.*, Enriching the sequence substitution matrix by structural information, *Proteins: Struct Funct Bioinform* **54**:41–48, 2004.
22. Murzin AG *et al.*, SCOP: a structural classification of proteins database for the investigation of sequences and structures, *J Mol Biol* **247**:536–540, 1995.

Xin Liu received his B.S. and M.S. degree in Physics from Beijing Normal University. His PhD degree is received from Institute of Theoretical Physics, Chinese Academy of Sciences. His research interests cover Statistical Biophysics, Bioinformatics, and Biomechanics.

Wei-Mou Zheng received his B.S. degree in Physics from the Peking University, and his Ph.D. degree in Physics from the Free University of Brussels. He is Professor of Theoretical Physics at the Institute of Theoretical Physics, Academia Sinica. His research interests cover quantum mechanics, statistical physics, nonlinear dynamics and bioinformatics.