

## Sequence analysis

# ARCS: an aggregated related column scoring scheme for aligned sequences

Bin Song<sup>1,†</sup>, Jeong-Hyeon Choi<sup>2,†</sup>, Guangyu Chen<sup>1</sup>, Jacek Szymanski<sup>1</sup>, Guo-Qiang Zhang<sup>1</sup>, Anthony K. H. Tung<sup>3</sup>, Jaewoo Kang<sup>4</sup>, Sun Kim<sup>2,\*</sup> and Jiong Yang<sup>1,\*</sup>

<sup>1</sup>Electrical Engineering and Computer Science Department, Case Western Reserve University, Cleveland, OH, USA, <sup>2</sup>School of Informatics, Indiana University, Bloomington, IN, USA, <sup>3</sup>Department of Computer Science, National University of Singapore, Singapore and <sup>4</sup>Department of Computer Science and Engineering, Korea University, Seoul, Korea

Received on November 14, 2005; revised on June 6, 2006; accepted on July 18, 2006

Advance Access publication July 26, 2006

Associate Editor: Christos Ouzounis

## ABSTRACT

**Motivation:** Biologists frequently align multiple biological sequences to determine consensus sequences and/or search for predominant residues and conserved regions. Particularly, determining conserved regions in an alignment is one of the most important activities. Since protein sequences are often several-hundred residues or longer, it is difficult to distinguish biologically important conserved regions (motifs or domains) from others. The widely used tools, Logos, Al2co, Confind, and the entropy-based method, often fail to highlight such regions. Thus a computational tool that can highlight biologically important regions accurately will be highly desired.

**Results:** This paper presents a new scoring scheme ARCS (Aggregated Related Column Score) for aligned biological sequences. ARCS method considers not only the traditional character similarity measure but also column correlation. In an extensive experimental evaluation using 533 PROSITE patterns, ARCS is able to highlight the motif regions with up to 77.7% accuracy corresponding to the top three peaks.

**Availability:** The source code is available on <http://bio.informatics.indiana.edu/projects/arcs> and <http://goldengate.case.edu/projects/arcs>

**Contacts:** [jiong.yang@case.edu](mailto:jiong.yang@case.edu), [sunkim2@indiana.edu](mailto:sunkim2@indiana.edu)

**Supplementary Material:** <http://bio.informatics.indiana.edu/projects/arcs> and <http://goldengate.case.edu/projects/arcs>

## 1 INTRODUCTION

One of the most important and challenging problems in biological sequence analysis is to find the predominant residues or conserved regions in a set of biological sequences. Analysis of positional conservation in an amino acid sequence alignment can aid in detection of motifs and functionally and/or structurally important residues, e.g. at the binding sites (Pei and Grishin, 2001; Villar and Kauvar, 1994; Ouzounis *et al.*, 1998). Mapping the conservation information on to a protein 3D structure helps to visualize spatial conservation patterns and to deduce potential functional surfaces of

a protein molecule (Sander and Schneider, 1991; Lichtarge *et al.*, 1996; Landgraf *et al.*, 1999; Makarova and Grishin, 1999; Zhang *et al.*, 2000). Several methods of conservation analysis are used, such as the vectorial method (Casari *et al.*, 1995), evolutionary tracing (Lichtarge *et al.*, 1996) and Entropy-based conservation analysis (Sander and Schneider, 1991; Shenkin *et al.*, 1991). A typical approach for conservation analysis is to align the sequences using a multiple sequence alignment tool and then determine conserved regions of these aligned sequences.

There is a significant body of literature on the multiple sequence alignment problem (MSA), e.g. MSA algorithms, such as, the dynamic programming method, central-star approach (Gusfield, 1993, 1997), *l*-star algorithm (Bafna *et al.*, 1997) and Partial Order Alignment algorithms (POA) (Lee *et al.*, 2002); existing multiple sequence alignment tools, such as Clustal W (Higgins *et al.*, 1994), T-coffee (Notredame *et al.*, 2000), MuSiC (Tsai *et al.*, 2004), etc. However, determining conserved regions in the aligned sequences remains a challenging problem. Computational tools that highlight potential conserved regions effectively can help biologists to determine conserved regions fast and accurately. To the best of our knowledge, there only exist a few tools, e.g. Logos (Schneider and Stephens, 1990), AL2CO (Pei and Grishin, 2001), COMPASS (Sadreyev and Grishin, 2003) and ConFind (Smagala *et al.*, 2005). In this paper, we present a novel algorithm that can highlight potential conserved regions effectively.

### 1.1 Motivation

Several methods are known for discovery of conserved regions from aligned sequences. The main idea of Logos was to compute the frequency of each letter at the position in the aligned sequences. Logos could present the consensus sequences and display the patterns in the aligned sequences. (In Section 3.2 we will show the disadvantages of Logos empirically.) AL2CO calculated a conservation index at each position in a multiple sequence alignment using several methods. Amino acid frequencies at each position are estimated and the conservation index is calculated from these frequencies. Two different strategies (unweighted frequencies and weighted frequencies) and three conceptually different approaches (entropy-based, variance-based and matrix score-based) were

\*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

utilized in the AL2CO algorithm. COMPASS was a method for the comparison of multiple protein alignments. The method derived numerical profiles from alignments, constructs optimal local profile-profile alignments and analytically estimates  $E$ -values for the detected similarities. The scoring system and  $E$ -value calculation were based on a generalization of the PSI-BLAST approach to profile-sequence comparison, which was adapted for the profile-profile case. However, COMPASS focused on the comparison of different alignments, instead of highlighting the conserved regions from aligned sequences. ConFind was designed to work with a large number of closely related, highly variable sequences. Conserved regions were defined in terms of minimum region length, maximum informational entropy (variability) per position, number of exceptions allowed to the maximum entropy criterion and the minimum number of sequences that must contain a non-ambiguous character at a position to be considered for inclusion in a conserved region. Though ConFind provided robust handling of alignments containing partial sequences and ambiguous characters, the method could not deal with general alignments well. Thus more effective methods for highlighting true conserved regions of the alignment are still needed. Moreover, the above methods did not consider the correlation information among columns in the aligned sequences although they took into account the similarity within each (aligned) column.

In biological sequences, the columns or positions in biologically important domains are usually highly correlated and the sequences in rows are similar (Cline *et al.*, 2002; Martin *et al.*, 2005). Until now, to the best of our knowledge, the approaches of conserved regions discovery on alignments consider the similarity within each aligned column only. No one takes into account the correlation among columns which is significant in biological domains. If the column correlation information is incorporated into the discovery function of conserved regions, the results could be improved greatly. Therefore, we introduce a new aggregated related column scoring (ARCS) scheme for aligned sequences. In detail, ARCS consists of two factors. The first factor is the similarity of residues in an aligned column, which the LOGOS value (Schneider *et al.*, 1990) can measure. If the alignments are of similar sequences, then the score of ARCS will be high. The second factor reflects the correlation among positions. If the domains are more correlated, then it will also receive a higher score. The functional dependency (Giannella *et al.*, 2004) could be used for this purpose. We apply the ARCS scheme to highlight the conserve regions on alignments. PROSITE (Nicolas *et al.*, 2004) is a database of motif signatures in proteins and it is compiled by human experts. In an extensive experiment with randomly chosen 533 PROSITE patterns in correctly aligned sequences, ARCS is able to successfully highlight true motif regions up to 77.7%, corresponding to the three highest peaks. Both Logos and AL2CO are not as effective as ARCS.

The multiple sequence alignment is a difficult problem and, in reality, alignment of sequences may be incorrect. Thus, we compute the ARCS score for 47 randomly chosen PROSITE families that can not be aligned correctly. ARCS can still detect part of conserve regions up to 40.4%.

The remainder of this paper is organized as follows. In the next section, we will introduce some formal definitions of ARCS and present a method to compute. An extensive empirical study is done in Section 3. The final conclusion is drawn in Section 4.

## 2 ARCS METHOD

In this section, we present the ARCS model in detail. The main idea is that we make use of the biological knowledge that the elements in different columns of a domain are usually highly correlated and rows have great similarity. As a result, the functional dependency is used to represent the correlation between columns, which is  $FD_{i \rightarrow j}$  in Definition 2. LOGOS reflects the similarity of residues within a column, which is function LOGOS() in Definition 1.

The notations in this section are similar to those in Giannella and Robertson (2004) and Schneider and Stephens (1990).

**DEFINITION 1.** Given a set of  $n, n \geq 2$ , aligned sequences  $\{S_1, S_2, \dots, S_n\}$  with the same length  $m$ , the LOGOS score is defined as

$$LOGOS(i) = H_{max} - H(i) \quad (1)$$

where  $LOGOS(i)$  denotes the  $i$ -th column's LOGOS value in the aligned sequences. It tries to quantify the useful, ordered information that is available in the  $i$ -th column. The  $i$ -th column's  $H$  value,  $H(i)$  represents the disorder degree of the  $i$ -th column. It is defined as

$$H(i) = - \sum_e F_{ie} \log_2(F_{ie}) \quad (2)$$

where  $F_{ie}$  is the frequency of letter  $e$  in column  $i$ , that is

$$F_{ie} = c_{ie}/n \quad (3)$$

Moreover,  $c_{ie}$  is the observed count for letter  $e$  in column  $i$ ;  $c_{ie} = \sum_j \delta(S_j(i) = e)$ , where  $\delta(S_j(i) = e)$  is 1 if  $S_j(i) = e$  and 0 otherwise.  $S_j(i)$  denotes the  $i$ -th letter in the aligned sequence  $S_j$ .  $H_{max}$  is defined as

$$H_{max} = \log_2(\text{Min}(NL, n)) \quad (4)$$

$NL$  denotes the number of letters appear in the aligned sequences set  $\{S_1, S_2, \dots, S_n\}$ .

**EXAMPLE 1.** Consider an aligned sequence set in Fig. 1. There are 4 sequences (i.e.  $n = 4$ ) with 5 distinct letters (MLQW\_) (i.e.  $NL = 5$ ).  $H_{max} = \log_2(\text{Min}(NL, n)) = \log_2(\text{Min}(5, 4)) = 2$ ;  $F_{1M} = 2/4 = 1/2$ ;  $F_{1W} = 2/4 = 1/2$ ;  $F_{2L} = 2/4 = 1/2$ ;  $F_{2Q} = 1/4$ ;  $F_{2_} = 1/4$ ;  $F_{3Q} = 3/4$ ;  $F_{3L} = 1/4$ ;  $F_{4L} = 3/4$ ;  $F_{4W} = 1/4$ .  $H(1) = -(F_{1M} \log_2(F_{1M}) + F_{1W} \log_2(F_{1W})) = 1$ ;  $H(2) = -(F_{2L} \log_2(F_{2L}) + F_{2Q} \log_2(F_{2Q}) + F_{2_} \log_2(F_{2_})) = 1.5$ ;  $H(3) = -(F_{3Q} \log_2(F_{3Q}) + F_{3L} \log_2(F_{3L})) = 0.8113$ ;  $H(4) = -(F_{4W} \log_2(F_{4W}) + F_{4L} \log_2(F_{4L})) = 0.8113$ .  $LOGOS(1) = H_{max} - H(1) = 1$ ;  $LOGOS(2) = 0.5$ ;  $LOGOS(3) = 1.1887$ ;  $LOGOS(4) = 1.1887$ .

**DEFINITION 2.** A functional dependency from  $A$  to  $B$  is defined as the existence of a map from  $A$  to  $B$ . Giannella *et al.* presented an approximation measure for functional dependency, which will be applied in the method of ARCS.

Given a set of  $n, n \geq 2$ , aligned sequences  $\{S_1, S_2, \dots, S_n\}$  with the same length  $m$ , for the  $i$ -th column and the  $j$ -th column in the aligned sequences,  $c_{ip,jq}$  is the observed count for letter  $p$  in column  $i$  and letter  $q$  in column  $j$ , i.e.  $c_{ip,jq} = \sum_k \delta(S_k(i) = p, S_k(j) = q)$ . The functional dependency from column  $i$  to column  $j$  is defined as

$$FD_{i \rightarrow j} = 1 - H_{i \rightarrow j} / \log_2(n) \quad (5)$$

$H_{i \rightarrow j}$  is the information dependency measure of column  $j$  given column  $i$ ,

$$H_{i \rightarrow j} = p(c_{ip}/n) (q(c_{ip,jq}/c_{ip}) \log_2(c_{ip}/c_{ip,jp})) \quad (6)$$

**EXAMPLE 2.** Consider the aligned sequence set in Fig.1, the functional dependency from column 4 to column 1 is,

$$H_{4 \rightarrow 1} = 3/4 (1/3 \log_2(3) + 2/3 \log_2(3/2)) + 1/4(1 \log_2(1)) = 0.689$$

$$FD_{4 \rightarrow 1} = 1 - 0.689 / \log_2(4) = 0.656$$

$S'_1$  M L Q W  
 $S'_2$  M \_ Q L  
 $S'_3$  W L L L  
 $S'_4$  W Q Q L

Fig. 1. Aligned Sequences Set.

The functional dependency from column 1 to column 4 is,

$$H_{1 \rightarrow 4} = 2/4 (1 \log_2(1)) + 2/4(1/2 \log_2(2)) + 1/2 \log_2(2) = 0.5$$

$$FD_{1 \rightarrow 4} = 1 - 0.5/\log_2(4) = 0.75$$

We can see that the definition of functional dependency is not symmetrical, i.e.  $FD_{i \rightarrow j}$  is not necessarily equal to  $FD_{j \rightarrow i}$ .

DEFINITION 3: Given a set of  $n, n \geq 2$ , aligned sequences  $\{S_1, S_2, \dots, S_n\}$  with the same length  $m$ , the Aggregated Related Column Score (ARCS) is defined as

$$ARCS(i) = \sum_{j \in N(i)} FD_{j \rightarrow i} LOGOS(j) \quad (7)$$

where  $N(i)$  is the set of neighboring columns of column  $i$ . In the paper, we define a neighborhood size  $N = |N(i)|$ . Column  $j$  belongs to set  $N(i)$  if  $|j-i| \leq (N-1)/2$ .

EXAMPLE 3. Consider the aligned sequence set in Fig. 1. Let  $N3$ .

$$ARCS(1) = FD_{1 \rightarrow 1} LOGOS(1) + FD_{2 \rightarrow 1} LOGOS(2) = 1.375$$

$$ARCS(2) = FD_{1 \rightarrow 2} LOGOS(1) + FD_{2 \rightarrow 2} LOGOS(2) + FD_{3 \rightarrow 2} LOGOS(3) = 1.482$$

$$ARCS(3) = FD_{2 \rightarrow 3} LOGOS(2) + FD_{3 \rightarrow 3} LOGOS(3) + FD_{4 \rightarrow 3} LOGOS(4) = 2.343$$

$$ARCS(4) = FD_{3 \rightarrow 4} LOGOS(3) + FD_{4 \rightarrow 4} LOGOS(4) = 1.968$$

ARCS can be used to obtain some information about reserved regions among aligned sequences. For example, we use the aligned protein sequence set PS00702. Figure 2 shows the ARCS score of each column with neighborhood size 9, that is  $N(i) = [i - 4, i + 4]$ .

From Figure 2, we can see that ARCS shows the conserved information for each sequence position. In order to let the curve highlight the conserved regions clearly, we smooth the ARCS result. That is,

$$\text{Smoothed ARCS}(i) = \frac{\sum_{0 \leq j \leq \lfloor (w-1)/2 \rfloor} ARCS(i \pm j)}{w} \quad (8)$$

We let  $w$  denote the smoothing window size. Figure 3 shows the ARCS score curve by the smoothing window size 3.

### 3 EXPERIMENTS

The performance of ARCS was extensively evaluated using the PROSITE database (Release 17.01 of January 2002). For each PROSITE pattern, we extracted a set of sequences with the pattern and aligned the sequence set with ClustalW. Column scores were then calculated using the ARCS method, which was implemented with Matlab and Octave. Among 1320 patterns, we randomly chose 709 patterns where the number of sequences was not  $>50$ . Of 176 patterns whose corresponding multiple sequence alignment failed to align the motif regions correctly, 47 patterns were randomly chosen. Thus we used 533 multiple sequence alignments to evaluate our method for the case that the alignment is correct (details in Section 3.2). A total of 47 alignments were tested for the case

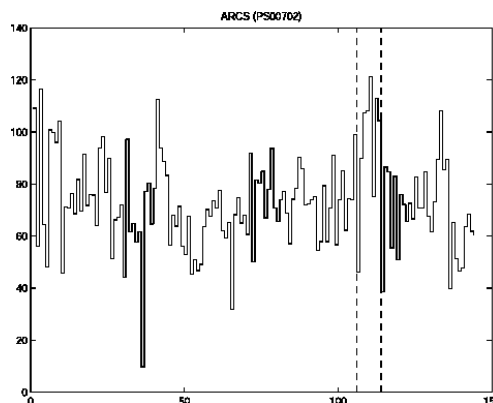


Fig. 2. ARCS score of PS00702 with neighborhood size 9. The motif region is between two lines.

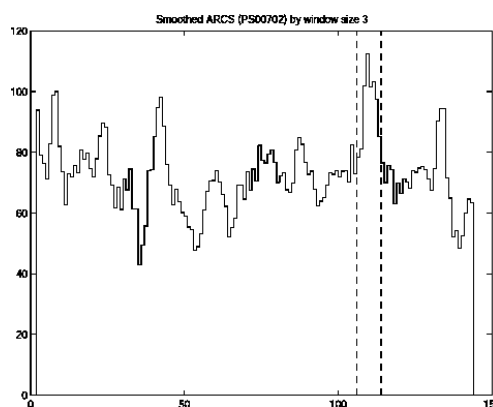


Fig. 3. The ARCS score smoothed by window size 3 with neighborhood size 9.

that Clustal W aligned part or none of the motifs (details in Section 3.3).

ARCS method transforms the multiple sequence alignment to a series of real numbers, one for each column, and we can define peaks in the number series. For each correct alignment and the corresponding PROSITE pattern, performance was measured in terms of the rank of the peak that the true motif region corresponds to. The highest peak will be assigned rank 1 while the second highest peak will be assigned peak 2, and so on. Since there were 533 patterns randomly selected to test for correct alignments, we were not able to manually verify; manual verification would also be subjective. Thus we implemented a peak-finding program that is described below. For the 47 alignments that Clustal W aligns ‘incorrectly’, we manually find whether the highest peaks indicate part of motif. We also measured the performance of our method in terms of the complexity of the PROSITE patterns (Section 3.4).

*Automatic peak detection method.* It is not trivial to define peaks in a series of numbers. One challenge is to handle adjacent peaks. For example, if two high peaks are nearby, should we define two separate peaks or a single peak merging these two peaks? A widely used technique is to smooth the values within a window of a fixed size. As a result of smoothing, we have a new series of numbers where peaks will be defined. To define peaks we need to

define local minima and local maxima. We define a peak as a data position of a local maximum where the difference between the local maximum and any of two nearby local minima is greater than a parameter. The parameters for the peak finding program are  $T_{\text{mh}}$ , the minimum height of the local maximum from the local minimum (Li and Fenimore, 1996), and  $T_{\text{ew}}$ , the half window size for evaluation.

In the following figures from Fig. 4 to Fig. 8, the  $x$ -axes represent the column position of the aligned sequences and the  $y$ -axes indicate the score.

### 3.1 Effects of the neighborhood size for ARCS

We explore the peaks of ARCS with various neighborhood sizes for PS00568, which are illustrated in Figure 4. In Figure 4a, the positions of the highest peaks are not the known domains. However, at window length of 5, 7 (Fig. 4b and c), the highest peak corresponds to the true motif region. Table 1 shows experiments with a varying window size from 3 to 11 with 533 different PROSITE patterns. From neighborhood size 7 to neighborhood size 11, ARCS performance does not change much in highlighting the conserved regions.

We evaluate ARCS performance with different neighborhood sizes on 533 datasets. Table 1 shows the results. From Table 1, when the neighborhood size is 3, 40.2% of motifs corresponded to the first peak, 60.0% of motifs corresponded to the top two peaks and 71.3% of motifs corresponded to the top three peaks. When the neighborhood size is 5, 45.8% of motifs corresponded to the first peak, 65.3% to the top two peaks and 74.5% to the top three peaks. When the neighborhood size is 7, the results are that 46.7% corresponded to the first peaks, 67.0% to the corresponded top two peaks and 77.7% corresponded to the top three peaks. If the neighborhood size is 9, then 46.7% patterns correspond to the first peaks, 65.7% to the top 2 peaks, and 77.3% to the top 3 peaks. If the neighborhood size is 11, then 48.4% of motifs corresponded to the first peaks, 65.3% to the top 2 peaks and 76.0% to the top 3 peaks. Therefore, when the neighborhood size is 7, ARCS could highlight the motif regions with up to 77.7% accuracy corresponding to the top three peaks. In addition, for the three highest peaks of ARCS score for sequences in each family, the precision is  $\sim 35\%$  which means that 35% the peaks (the top three peaks) correspond to a true motif or part of a true motif. In some cases, multiple peaks may correspond to different portions of the same motif.

### 3.2 Performance of LOGOS, AL2CO and ARCS

The aligned protein sequence set PS00702 is used for the comparison. The multiple sequence alignment algorithm correctly aligns the known motif region. Figure 5 gives the LOGOS score of each column of PS00702.

Similar to Smoothed ARCS score, we smooth the Logos score to highlight the conserved regions. That is,

$$\text{Smoothed LOGOS}(i) = \frac{\sum_{0 \leq j \leq \lfloor (w-1)/2 \rfloor} \text{LOGOS}(i \pm j)}{w} \quad (9)$$

In AL2CO paper, it is recommended to use a window size of 3 to smooth the score of AL2CO. To be consistent, we choose the smoothing window size to be 3 for both ARCS and LOGOS too. Figure 6 shows the smoothed LOGOS score of each column of PS00702. Figure 7 illustrates the AL2CO method with the window size 3.

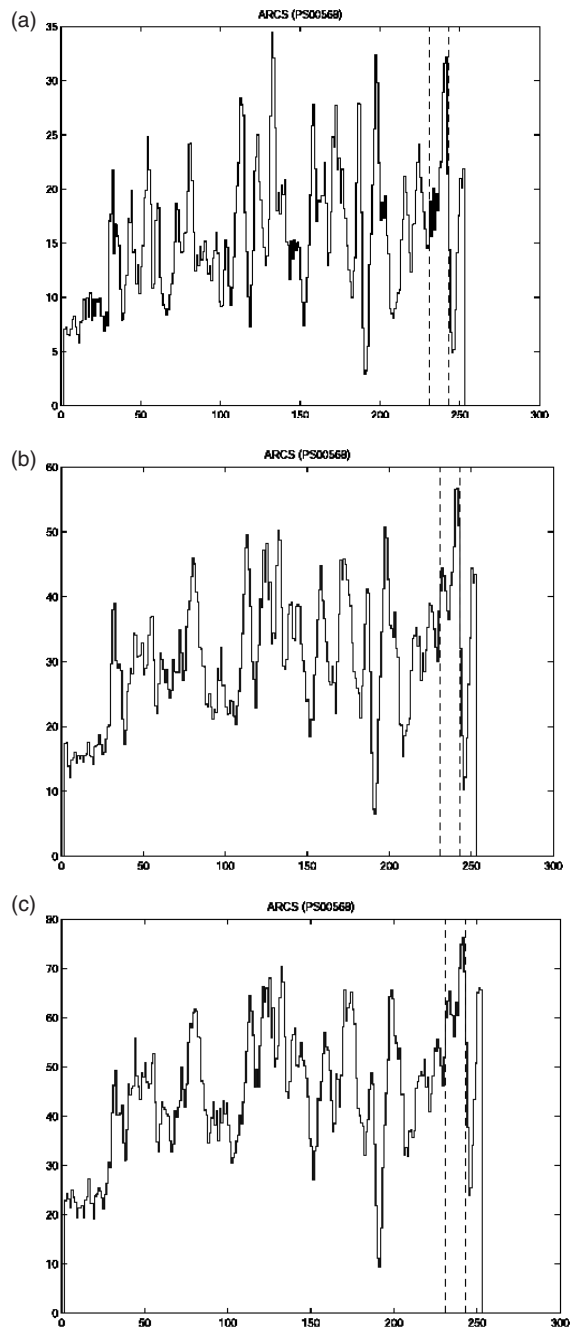


Fig. 4. ARCS score of PS00568 with the same smoothing window size 3 and different neighborhood size. (a) For neighborhood size 3, (b) 5 and (c) 7.

For PS00702, Logos and AL2CO were not able to highlight the motif region clearly; there are a few peaks whose heights are comparable or higher than that of the motif region. In Contrast, with the ARCS method, the highest peak (among a small number of distinct peaks) corresponds to the true motif region (Fig. 3).

*Performance on a large number of datasets.* Table 2 shows the performance of ARCS comparing to LOGOS and AL2CO in terms of the rank of the peaks that corresponds to the motifs on random 533 datasets. To be consistent, we choose the smoothing window

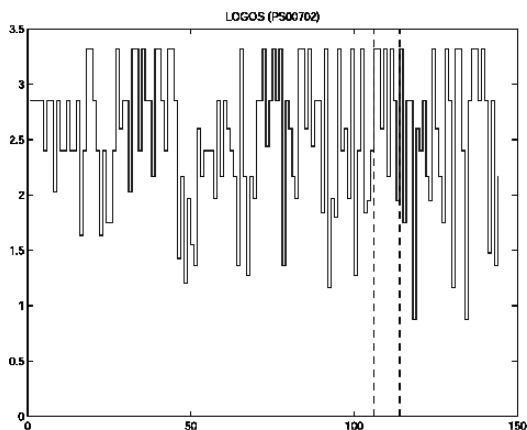


Fig. 5. LOGOS score of PS00702. The motif region is between two dotted lines.

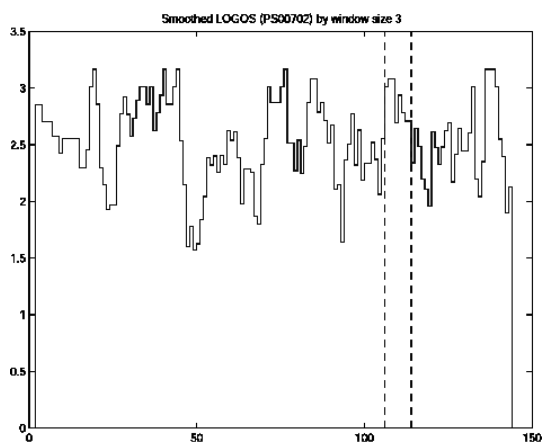


Fig. 6. Smoothed LOGOS score of PS00702 by window size 3. The motif region is between two dotted lines.

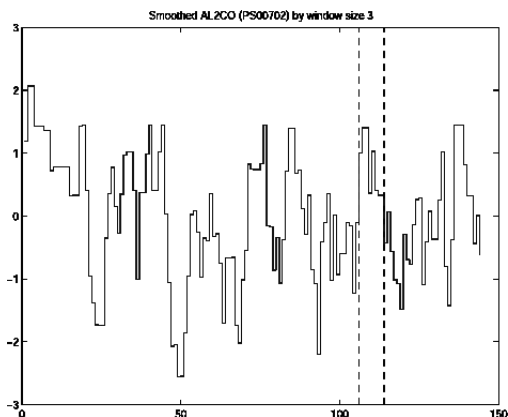


Fig. 7. Smoothed AL2CO score of PS00702 by window size 3.

size to be 3 for ARCS, LOGOS and AL2CO methods. When neighborhood size is 7, ARCS could highlight 46.7% of motifs corresponding to the first peaks, 67.0% to the top 2 peaks, and 77.7% to the top 3 peaks. In contrast, the LOGOS method was able to highlight 35.6% motifs corresponding to the first peak,

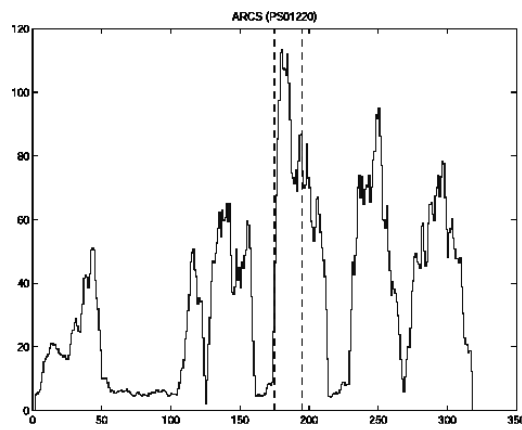


Fig. 8. The ARCS score of PS01220 smoothed by window size 3. The part motif region is between two lines.

52.3% to the top 2 peaks, and 67.2% to the top 3 peaks. AL2CO is 40.7% to the first peak, 60.8% to the top 2 peaks, and 73.2% to the top 3 peaks.

### 3.3 Performance of ARCS on incorrectly aligned sequences

In some datasets, existing multiple sequence alignment tools could not align them ‘correctly’. Only part or none of the motifs is aligned by these tools. For example, the pattern of dataset PS01220 is ‘[FYL]-x-[LVM]-[LIVF]-x-[TIV]-[DC]-P-D-x-P-[SNG]-x(10)-H’. By Clustal W, ‘[LVM]-[LIVF]-x-[TIV]-[DC]-P-D-x-P-[SNG]-x(10)-H’ is aligned. However, the first part motif ‘[FYL]’ is not aligned. In this case, ARCS can find these parts of motifs either. Figure 8 presents the curve of smoothed ARCS score.

A total of 47 patterns are randomly chosen among 176 patterns whose multiple sequences alignments are aligned incorrectly. On these 47 protein families, the first peak of ARCS corresponds to part of motifs up to 40.4% test cases.

### 3.4 Performance in terms of pattern complexity

What we have shown in the previous section is the accuracy of ARCS. It is also important to investigate the sensitivity to certain characteristics of the motif. We measured the sensitivity of ARCS with respect to the motif complexity which is defined as  $1 - \frac{\text{ratio of the number of exact characters in the pattern to the length of the pattern}}$ . Higher complexity means that there are more ambiguous characters in the pattern, thus highlighting true motif regions for the pattern is more difficult. As Figure 9 shows, our method is not sensitive to the motif complexity and it works equally well for very high-complexity cases.

## 4 CONCLUSION

In this paper, we defined a new score scheme, ARCS, that considered column correlation as well as the traditional character similarity measure. We measured the performance of the ARCS method using 533 PROSITE patterns whose sequences were aligned correctly and 47 PROSITE patterns which aligned sequences were incorrectly. In the correctly aligned sequences, ARCS is able to successfully highlight true motif regions up to 77.7%, corresponding to the three highest peaks. Both Logos and AL2CO are not as

**Table 1.** Evaluation ARCS performance with the same smoothing window size 3 and varying neighborhood sizes of 3, 5, 7, 9 and 11 on 533 datasets.

<i>i</i> -th peak	Neighborhood size = 3 ( $T_{mh} = 0.05, T_{ew} = 10$ )	Neighborhood size = 5 ( $T_{mh} = 0.05, T_{ew} = 10$ )	Neighborhood size = 7 ( $T_{mh} = 0.05, T_{ew} = 10$ )	Neighborhood size = 9 ( $T_{mh} = 0.05, T_{ew} = 10$ )	Neighborhood size = 11 ( $T_{mh} = 0.05, T_{ew} = 10$ )
1	214	244	249	249	258
2	106	104	108	101	90
3	60	49	57	62	57
4	36	31	32	36	40
5	35	25	29	24	22
6	15	20	11	18	20
7	8	15	13	13	12
8	14	5	5	5	6
9	5	10	7	6	9
10	8	8	3	3	2
11	6	9	3	2	2
12	5	2	1	2	2
13	5	2	2	2	1
14	8	1	2	3	1
15	1	2	2		1
>15	7	6	9	7	10
Total	533	533	533	533	533

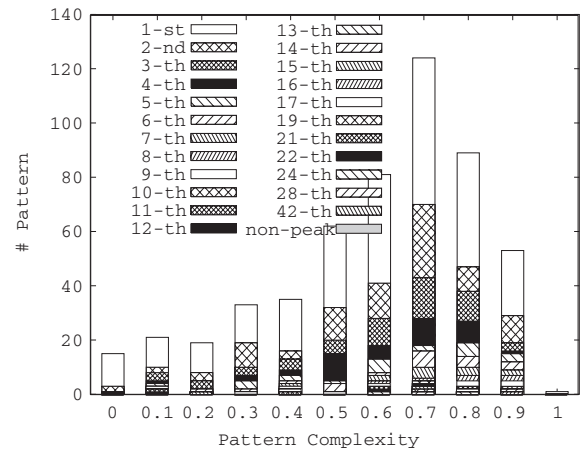
**Table 2.** Evaluation of ARCS with LOGOS and AL2CO in terms of the peak rank

<i>i</i> -th peak	LOGOS ( $T_{mh} = 0.05,$ $T_{ew} = 10$ )	AL2CO ( $T_{mh} = 0.05,$ $T_{ew} = 10$ )	ARCS ( $T_{mh} = 0.05,$ $T_{ew} = 10$ )
1	190	217	249
2	89	107	108
3	79	66	57
4	39	39	32
5	26	33	29
6	20	20	11
7	19	13	13
8	11	6	5
9	10	11	7
10	10	1	3
11	8		3
12	7	3	1
13	6	2	2
14	4	2	2
15			2
>15	15	13	9
Total	533	533	533

ARCS was able to highlight true motif regions in up to 46.7%, while LOGOS is 35.6% and AL2CO is 40.7%.

effective as ARCS. For those incorrectly aligned families, ARCS can still detect part of conserve regions up to 40.4% with the highest peak. We believe that ARCS can be used to help biologists utilize multiple sequence alignments more effectively, i.e. extracting conserved regions and modeling a set of proteins in terms of alignments.

Our work can be extended in many directions. The alignment scoring scheme can be further developed to a *de novo* motif



**Fig. 9.** Motif complexity versus accuracy of our method for  $T_{mh} = 0.05$ ,  $T_{ew} = 10$ , smoothing window size 3 and neighborhood size 11.

discovery algorithm based on the alignment. It will be also interesting to develop an algorithm to find boundaries of conserved regions for given alignment scores.

**ACKNOWLEDGEMENTS**

This is partially supported by National Science Foundation Career DBI-0237901 and INGEN (Indiana Genomics Initiatives) to S.K.

*Conflict of Interest:* none declared.

**REFERENCES**

Bafna, V. et al. (1997) Approximation algorithms for multiple sequence alignment. *Theor. Comput. Sci.*, **182**, 233–244.

- Casari,G. et al. (1995) A method to predict functional residues in proteins. *Nat. Struct. Biol.*, **2**, 171–178.
- Cline,M.S. et al. (2002) Information-theoretic dissection of pairwise contact potentials. *Proteins*, **49**, 7–14.
- Giannella,C. and Robertson,E. (2004) On approximation measures for functional dependencies. *Information Systems*, 483–507.
- Gusfield,D. (1993) Efficient methods for multiple sequence alignment with guaranteed error bounds. *Bull. Math. Biol.*, **55**, 141–154.
- Gusfield,D. (1997) *Algorithms on Strings, trees, and Sequence: Computer Science and Computational Biology*. Cambridge University Press, New York.
- Higgins,D. et al. (1994) CLUSTAL W: improving the sensitivity of progressive-multiple sequence alignment through sequence weighting,position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.
- Landgraf,R. et al. (1999) Analysis of heregulin symmertry by weighted evolutionary tracing. *Protein Eng.*, **12**, 943–951.
- Lee,C. et al. (2002) Multiple sequence alignment using partial order graphs. *Bioinformatics*, **18**, 452–462.
- Li,H. and Fenimore,F.E. (1996) Log-normal distributions in gamma-ray burst time histories. *Astrophys. J.*, **469**, L115–L118.
- Lichtarge,O. et al. (1996) An evolutionary trace method defines binding surfaces common to protein families. *J. Mol. Boil.*, **257**, 342–358.
- Makarova,K.S. and Grishin,N.V. (1999) The Zn-peptidase super-family: functional convergence after evolutionary divergence. *J. Mol. Biol.*, **292**, 11–17.
- Martin,L.C. et al. (2005) Using information theory to search for co-evolving residues in proteins. *Bioinformatics*, **21**, 4116–4124.
- Nicolas,H. et al. (2004) Recent improvements to the PROSITE database. *Nucleic Acids Res.*, **32**, 134–137.
- Notredame,C. et al. (2000) T-Coffee: a novel method for multiple sequence alignments. *J. Mol. Biol.*, **302**, 205–217.
- Ouzounis,C. et al. (1998) Are binding residues conserved? *Pac. Symp. Biocomput.*, 401–412.
- Pei,J. and Grishin,N.V. (2001) AL2CO: calculation of positional conservation in a protein sequence alignment. *Bioinformatics*, **17**, 700–712.
- Sadreyev,R. and Grishin,N. (2003) COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance. *J. Mol. Biol.*, **326**, 317–336.
- Sander,C. and Schneider,R. (1991) Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, **9**, 56–68.
- Schneider,T. and Stephens,R. (1990) Sequence logos: a new way to display consensus sequences. *Nucleic Acids Res.*, 6097–6100.
- Shenkin,P.S. et al. (1991) Information-theoretical entropy as a measure of sequence variability. *Proteins*, **11**, 297–313.
- Smagala,J.A. et al. (2005) Confind: a robust tool for conserved sequence identification. *Bioinformatics*, **21**, 4420–4422.
- Tsai,Y.T. et al. (2004) MuSiC: a tool for multiple sequence alignment with constrains. *Bioinformatics*, **20**, 2309–2311.
- Villar,H.O. and Kauvar,L.M. (1994) Amino acid preferences at protein binding sites. *FEBS Lett.*, **349**, 125–130.
- Zhang,H. et al. (2000) Crystal structure of YbaK protein from *Haemophilus influenzae* (HI1434) at 1.8 Å resolution: functional implications. *Proteins*, **40**, 86–97.