Sequence-Structure Patterns: Discovery and Applications

T. Milledge[1] S. Khuri[2] X. Wei[1] C. Yang[1] G. Zheng[1] G. Narasimhan[1]

[1]Bioinformatics Research Group (BioRG), School of Computer Science, Florida International University
[2]The Dr. John T. Macdonald Foundation Center for Medical Genetics, University of Miami School of Medicine

Abstract

Protein sequence data is being generated at a tremendous rate; however, functional annotation of these proteins is proceeding at a much slower pace. Biologists rely on computational biology and pattern recognition to predict the functionality of proteins. This is based on the fact that proteins that share a similar function often exhibit conserved sequence patterns. Such sequence patterns, or motifs, are derived from multiple sequence alignments and have been collected in databases such as PROSITE, PRINTS, SPAT, and eMOTIF. These patterns help to classify proteins into families where the exact function may or may not be known. Research has shown that these domain signatures often exhibit specific three-dimensional structures. In this paper, we show how starting from a seed sequence pattern from any of the existing sequence pattern databases, and using information from the protein structure databases, it is possible to design biologically meaningful sequence-structure patterns (SSPs). An important by-product of our method to generate sequence-structure patterns is an improved sequence alignment as well as an improved structural alignment of proteins belonging to a family and containing that pattern. Validation was performed by matching the resulting SSPs to domains in the ASTRAL compendium associated with a family or super-family designation in the SCOP database. SSPs generated by this method were frequently either fully specific (no false positives), fully sensitive (no false negatives), or both (diagnostic).

Keywords: Pattern discovery; sequence alignment; structure alignment; sequence-structure patterns.

1. Introduction

New proteins are being sequenced at a tremendous rate. The challenge biologists now face is the automatic, correct, and efficient functional annotation of proteins within these genomes. Laboratory-based research is ongoing, but is slow by definition. Most proteins are annotated using computationaly derived conserved patterns. It has long been known that certain residues within functional domains in proteins are better conserved among homologs than residues elsewhere in the proteins. Databases such as PROSITE [6, 15], eMOTIF [10], PRINTS [1], and SPAT [9] have been created as repositories for sequence patterns that describe and distinguish functional and structural domains in proteins. Kasuya *et al*. [12] systematically investigated the 3D structures of protein fragments whose sequences contain a specific PROSITE pattern. They observed that in a large number of cases, the three-dimensional conformations of the residues from the PROSITE pattern were nearly identical in all the true positives (i.e., proteins belonging to the family and containing the sequence pattern). Furthermore, the corresponding substructures in the false positives were often considerably different from those in the true positives. The main drawback with the approach followed by existing databases to generate sequence patterns is that they base their computations on multiple **sequence** alignments, which are often inaccurate, especially when the sequences exhibit considerable diversity. In this paper, we use both sequence and structure information and show how to automatically construct patterns with a **sequence** component (consisting of a "PROSITE-style" regular expression pattern) and a **structure** component (consisting of a structure template). Because of the use of the structural information, the resulting sequence patterns showed measurably higher discriminatory capability than existing sequence patterns. The patterns described in this paper, which we will refer to as *sequence-structure patterns* (**SSP**s), characterize portions of the three-dimensional protein structure, ranging in size from small pockets to entire domains.

In our method, SSPs are generated by starting from "seed" sequence patterns from PROSITE, eMOTIF, PRINTS, or other sources, and improving them using an iterative alignment of structures from the Protein Data Bank (PDB) [2, 3] and ASTRAL [5] databases, and by using the knowledge of substitution groups [16]. An important by-product of our method

to generate SSPs is an improved sequence and structure alignment of the protein domains that contain the SSP. Our method iteratively improves both the sequence and the structure alignments by using a set of empirically derived equivalence classes (substitution groups) of amino acids. The final alignments are then used to output the final SSP.

We say that a protein has a sequence match with the SSP if it contains the sequence component of the SSP. The SSPs are evaluated with regard to their *specificity* (TP/(TP+FP)) and *sensitivity* (TP/(TP+FN)), where TP is the set of true positive matches, while FP and FN are the sets of false positive and false negative matches with respect to their membership in a SCOP (Structural Classification of Proteins) protein family. The SCOP database is a comprehensive classification of all proteins of known structure [4]. The basic classification unit in SCOP is the domain, a unit of the protein that is either observed isolated in nature or in more than one context in different multi-domain proteins. Proteins are clustered into families if they have either: (a) residue identities of 30% or higher, or (b) lower sequence identities but very similar structures and functions [4]. A related database is the ASTRAL Compendium, which provides sequences and structures for all domains filtered according to percentage sequence similarity [5]. The ASTRAL 40% database (version 1.65) contains a subset of proteins from the PDB database with less than 40% sequence identity to each other, and this database will be referred to as ASTRAL40 in this paper. We also refer to ASTRAL95 and ASTRAL100 (or full PDB) to refer to the corresponding databases with 95% and 100% sequence identity respectively. For our purposes, the ASTRAL40 database was used to generate SSPs for families that were well represented in the PDB and the ASTRAL95 database was used to generate SSPs for protein families with fewer PDB examples. In both cases, the ASTRAL100 database was used for testing.

SSP patterns extend the information provided in PROSITE, SPAT, or eMOTIF databases. PROSITE is based on the assumption that specific regions of a protein are conserved in the amino-acid sequence due to functional properties, such as binding activity. In an attempt to make the pattern as general as possible, the development of shorter patterns has been favored over longer ones [6]. However, "spacer regions" surrounding functional sites also have a well-defined three-dimensional structure [12-14]. By focusing the motif generation process on the functional groups alone, many opportunities are missed for the discovery of other highly conserved, and characteristic, patterns in the sequence and structure of the "non-interacting" regions of the domain. The eMOTIF database, on the other hand, contains sequence patterns without variable gaps. This makes it necessary to look at a number of eMOTIF sequence patterns to be able to completely characterize a variable-length functional domain [10]. Although SPAT is an improvement over PROSITE, it too focuses on sequence alignment information to generate its patterns [9]. Unlike these other methods, our method to generate SSPs departs from this strategy by using both a multiple sequence and a multiple structure alignment to infer the patterns.

Alignment, by either sequence or structure, consists of establishing a correspondence between the residues of the two proteins. Sequence alignment algorithms use a substitution matrix that is based on amino acid type and is independent of position. In a structure alignment, this correspondence is determined based on the 3D coordinates of the residues, not on the amino acid "type". The major difference between the sequence and structure alignments is that the optimization used for sequence alignment is globally convergent, whereas the alignment found by a structural alignment algorithm can depend on the initial equivalences. In sequence alignments, the optimal alignment in one region of the sequence is less sensitive to the optimal alignment in another region. In contrast, structural alignments fail to converge globally because the possible matches for different segments are tightly linked, as they are part of the same rigid 3D structure [7]. Consequently, it is possible to improve a structural alignment by starting with a small number of residues which are known to be structurally related, followed by a realignment of the structure based on new residue equivalences indicated by the alignment. Although structurally variable regions, such as loops, may not align well, in practice, good alignments do exist for essential "core" regions of two similar proteins. These may include the catalytic domain of enzymes, or certain ligand-binding sites. It is known that in these core regions of folded proteins, the packing density can be quite high and that the side chains of distant residues end up in close proximity to each other. In spite of the high packing density, not only do structurally related residues show remarkably good structural alignment, even their side chains exhibit a high degree of conformation. Such residues can be characterized by their biochemical context, or "substitution groups", as described by Wu *et al*. [16]. Based on these descriptions, the substitution groups used by the SSP method are primarily two-residue groups which have a concise biochemical role: [DN] and [EQ] are acid-amide combinations with a similar side chain; [DE] are acidic residues; [KR] are basic residues; [ST] have short hydroxyl side chains; [AS] are small with single carbon side chains and [HY] have polar ring

structures. Two longer groups are [FWY] with aromatic side chains, and [FILMVY] with hydrophobic side chains.

2. The SSP Algorithm

Figure 1 gives a brief description of our algorithm for generating SSPs. It takes as input a "seed" PROSITE-style pattern along with a training set database (in our case, we use ASTRAL40 unless there are not enough structures in it, in which case we use ASTRAL95). It produces as output a SSP, which is a pair <**P**, **T**>, where **P** is a sequence pattern, and **T** is a structure template for the sequence pattern. As mentioned earlier, it also produces sequence and structure alignments of proteins with this SSP.

Step 1 of the algorithm generates candidate proteins that contain the sequence pattern. A "cluster" of structurally-related candidates **L** are chosen for further analysis. One way to pick such a cluster is to measure the pairwise RMSD of the αC atoms of the pattern residues and removing any proteins that deviate from the mean RMSD value by more than a user-defined threshold (RMSD_THRESH). A template structure **T** is then chosen from this list **L**. This can be done by picking the protein with the smallest mean RMSD value for all the proteins in the remaining set. The structure alignment in Step 3 is achieved by structurally aligning each of the remaining proteins to the template using the αC atoms of the pattern residues. Thus Step 3 produces an initial structure alignment of the apo protein structures from **L**. From this structure alignment, a sequence alignment is then created (Step 4) and a subset of this alignment consisting of all ungapped positions (i.e., positions which contain a residue from all proteins in the set) is then considered for the presence of conserved residues (Step 5), i.e., containing a user-defined percentage (SG_PERCENT) of residues from a substitution group. Thus these conserved positions define an intermediate sequence pattern. The proteins are then structurally aligned based on the residues in this new intermediate pattern (Step 6). From this new structure alignment, a new sequence alignment is obtained, which in turn leads to a new sequence pattern, and the process is repeated until no changes in the alignments are observed. This stable intermediate sequence pattern is then used as the seed pattern for a new match of the proteins in the training set database. The process is repeated until no new training set protein sequences are matched by the sequence pattern. Finally, the sequence pattern and the corresponding template are output as components of the output SSP.

SSP ALGORITHM

Input: **(a)** A database of protein structures, and associated protein sequences, **N,**
(b) A PROSITE-style sequence pattern , **P.**

Output: **(a)** Sequence-structure pattern <**P'**, **T**>,
(b) Structure alignment **S** of proteins with pattern **P'**, and
(c) Sequence alignment **Q** of proteins with pattern **P'**.

1. Search for pattern **P** in database **N** to generate a list of candidate proteins **C**.
2. Pick a "cluster" **L** of proteins from **C** that belong to the same SCOP family.
3. Create a structure alignment **S** for **L** using the residues of pattern **P**.
4. Extract sequence alignment **Q** from structure alignment **S**.
5. Identify all positions in sequence alignment **Q** that have residues from a substitution group.
6. If stopping condition is not satisfied, then create a new structure alignment **S** for **L** using the positions identified in Step 6. Then go to Step 5.
7. Construct a PROSITE-style sequence-structure pattern **P'** and template T from the positions in **Q**.
8. Iterate the whole process if new candidates from database **N** are matched.

Fig. 1: The SSP algorithm.

The algorithm was implemented in Perl and in the SwissProt viewer SPDBV script language [8].

3. Results and Conclusions

A comparison of 27 SSPs generated by the above method with PROSITE patterns for the same SCOP family (or superfamily) is available from our website. SSPs, and corresponding PROSITE patterns, for two SCOP families are shown in Table 1. In the case of the highly variable Immunoglobin C1 domain, the specificity of the SSP was 100% with respect to the ASTRAL100 (no false positive matches). The sensitivity of this SSP was also 4.1% higher than for the corresponding PROSITE pattern. For the Zinc Finger family, the specificity of the SSP was marginally lower than for the corresponding PROSITE pattern, however the sensitivity of the SSP was 100% (no false negative matches). Note that since the SSP algorithm uses structural information, SSPs can be generated if there are a reasonable number of hits in the PDB superfamily and the ASTRAL40 database.

We first identified the PROSITE patterns with the highest number of hits in the ASTRAL40 database. The algorithm was then experimentally tested on these protein families. In the case where an SSP for a SCOP family already had a corresponding PROSITE sequence pattern, the PROSITE pattern was improved about 90% of the time with respect to the SSP sequence pattern. This represents an average improvement of specificity of +27.3% and an average improvement of sensitivity of +16.2%. The globins, dehydrogenases, and immunoglobin (V1 and C2) families did not previously have any PROSITE patterns.

Improved sequence alignments, which are a by-product of the SSP method, can also be used for the generation of improved profiles, HMMs and other statistical models. In this regard, SSPs that are fully specific and/or fully sensitive are useful in a number of ways. In the classification of protein families, the existence of a diagnostic SSP provides important verification that the classification is sound. Fully specific SSPs can be used in the annotation of newly sequenced genomes, to attribute both structure and function to a putative gene or open reading frame. This is especially important today with the rise in structure-based functional annotation of proteins, and the number of hypothetical proteins, mainly from archaeal species, which are being crystallized in order to identify their function [11]. The SSPs discovered by our method have been compiled into the SSPsite database and made available at http://www.cs.fiu.edu/sspsite. The Perl and SPDBV script code for our method can be obtained by sending an email request to the corresponding author.

Acknowledgements

GN's research was supported in part by NIH grant P01 DA15027-01.

4. References

[1] T.K. Attwood, *Brief Bioinform*, **3**, 252 (2002).
[2] H.M. Berman, T.N. Bhat, P.E. Bourne, Z. Feng, G. Gilliland, H. Weissig, and J. Westbrook, *Nat Struct Biol*, **7 Suppl**, 957 (2000).
[3] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne, *Nucleic Acids Res*, **28**, 235 (2000).
[4] S.E. Brenner, C. Chothia, T.J.P. Hubbard, and A.G. Murzin, *Methods in Enzymology*, 635 (1996).
[5] S.E. Brenner, P. Koehl, and M. Levitt, *Nucleic Acids Research*, **28**, 254 (2000).
[6] L. Falquet, M. Pagni, P. Bucher, N. Hulo, C.J.A. Sigrist, K. Hofmann, and A. Bairoch, *Nucl. Acids. Res.*, **30**, 235 (2002).
[7] M. Gerstein and M. Levitt, *Protein Science*, **7**, 445 (1998).
[8] N. Guex and M.C. Peitsch, *Electrophoresis*, **18**, 2714 (1997).
[9] R. Hart, A. Royyuru, G. Stolovitzky, and A. Califano, *J. Comput. Biol.*, **7**, 585 (2000).
[10] J.Y. Huang and D.L. Brutlag, *Nucl. Acids. Res.*, **29**, 202 (2001).
[11] J.W. Jung and W. Lee, *J Biochem Mol Biol*, **37**, 28 (2004).
[12] A. Kasuya and J.M. Thornton, *Journal of Molecular Biology*, **286**, 1673 (1999).
[13] K.-Y. Lin, J. Wright, and C. Lim, *Journal of Molecular Biology*, **299**, 537 (2000).
[14] S. Mondal, S.P. Jaishankar, and S. Ramakumar, *Biochemical and Biophysical Research Communications*, **305**, 1078 (2003).
[15] C.J. Sigrist, L. Cerutti, N. Hulo, A. Gattiker, L. Falquet, M. Pagni, A. Bairoch, and P. Bucher, *Brief Bioinform*, **3**, 265 (2002).
[16] T.D. Wu and D.L. Brutlag, *ISMB-96*. 1996.

Protein Family	SCOP Family	Accn. Num.	Pattern	Spec. (TP/TP+FP)	Sens. (TP/TP+FN)
Immunoglobins C1 Set	B.1.1.2	PS00290	[FY]-x-C-x-[VA]-x-H.	97.9% (1095/1119)	71.2% (1095/1537)
		SSP91008	[VP]-X(15,20)-[YVIL]-X(0,1)-C-X(4)-[VLFY]-X(1,3)-[DSP]-X(2,3)-[MALIV]-X(1)-[FILV]-X(1,2)-[FLW]-X(21,28)-[WALFY]-X(5)-[TLV]-X(1)-[ITLFV]-X(7,12)-[HVLFY]-X(1)-C-X(1)-[MAV]-X(1)-[NFYH].	100% (1157/1157)	75.3% (1157/1537)
Zinc Finger	G.37.1.1	PS00028	C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H.	80.5% (95/118)	93.1% (95/102)
		SSP91022	[AHITFVY]-X(1)-C-X(2,5)-C-X(8,12)-[RIMFYL]-X(2)-H-X(3,5)-H.	71.8% (102/142)	100% (102/102)

Table 1: Performance of PROSITE patterns and SSPs for two SCOP families measured using ASTRAL100.