Structure Motif Discovery and Mining the PDB

Inge Jonassen, Ingvar Eidhammer
Dept. of Informatics, University of Bergen
HIB, N5020 Bergen, Norway

Darrell Conklin
ZymoGenetics Inc., 1201 Eastlake Avenue East
Seattle WA, USA 98102

William R. Taylor
National Institute of Medical Research, Mill Hill,
London, UK

Abstract. We describe an algorithm for the automatic discovery of recurring patterns in protein structures. The patterns consist of individual residues having a defined order along the protein's backbone that come close together in the structure and whose spatial conformations are similar. The residues in a pattern need not be close in the protein's sequence. The work described in this paper builds on an earlier reported algorithm for motif discovery. This paper describes a significant improvement of the algorithm which makes it very efficient. The improved efficiency allows us to use it for doing unsupervised learning of patterns occurring in small subsets in a large set of structures, a non-redundant subset of the PDB database of all known protein structures.

1 Introduction

Structural similarities consisting of a few secondary structures or residues can define structurally or functionally important elements of the proteins. The relationships are subtle and do not always appear significant when found in pairwise structure comparisons. Standard approaches for analysis of protein structures builds on pairwise comparisons where the pairwise comparisons are done independently. We have earlier described an approach which allows information from multiple structures to be used simultaneously (Jonassen *et al.*, 1999). In this approach all structures are compared to an external model, a pattern, obliviating the need for all against all pairwise comparisons. The approach was implemented in a program called SPratt. In this paper we describe a more efficient algorithm, named SPratt2, which allows more challenging discovery problems to be tackled.

The new method is able to discover automatically and in an entirely unsupervised fashion patterns shared by as few as two structures in a non-redundant subset of PDB. The method produces large number of patterns compliant with the user-definable constraints, and methods are described for removing redundancy in the output pattern set to facilitate the user's analysis of the output data.

2 Definitions

The local neighbourhood of each residue r in each structure is represented as a string NS_r, called a *neighbourhood string*. The string encodes all residues in the structure that are within a distance of d Angstrom from r (typically $d = 10$), including r itself. The residue r is named the *anchor* of NS_r. The residues are encoded by their amino acid type and the mean coordinates of the residue's side chain atoms. The residue's order in the neighborhood string is defined by their order along the protein's backbone. In this paper when giving examples of neighbourhood strings in this paper we write the single letter amino acid code for each of the residues with the anchor underlined.

We then define a *packing pattern* against which a neighbour string can be matched. A packing pattern consists of a list of elements where each element defines a match set (set of allowed amino acids) and one set of coordinates. Each packing pattern has one unique *anchor element*. We will write a pattern as the string of single letter amino acid codes (enclosed in brackets for elements where more than one amino acid is allowed) underlining the anchor residue.

A neighbour string $r_1 \ldots \underline{r_k} \ldots r_l$ is said to *match* a packing pattern $P = p_1 \ldots p_n$ if it contains a subsequence $r_{i_1} \ldots r_{i_n}$ so that the residues have amino acid types included in the match sets of the corresponding pattern elements and so that the anchor residue of the neighbour string is aligned with the pattern's anchor. For example, ACEWGGTGEA matches the packing pattern CWGT. Also, NS_r is said to *structurally match* P within ϕ if it is possible to superpose the coordinates of NS_r onto the coordinates of P with a RMSd of maximum ϕ. A neighbour string that structurally matches a packing pattern within a threshold ϕ describes an *occurrence* of the pattern. Finally, a pattern which have occurrences in k structures is said to have *support* k.

When presenting discovered patterns a sequence pattern can be given consisting of the residues of the packing pattern separated by spacers whose lengths are determined by the sequence separation of the residues involved in the matches. We use the PROSITE (Hofman *et al.*, 1999) notation. See Section 4 for examples.

3 Algorithms

Given a set of N structures we want to find packing patterns with occurrences in at least k of the structures, i.e., patterns with support at least k. Rather than devising a method for generating all possible packing patterns, the patterns will be generated as generalisations of neighbour strings from the structures. For example, the neighbourhood string

d	average *NS* length	average *NS* length using constraint
10	18	8
12	29	11
14	42	14
20	95	27

Table 1: Average neighbour string lengths for different radi (d-values) and depending on whether the half-sphere constraint (see text) is used.

ACEWGGTGEA can be generalised to a large number of (matching) packing patterns, for example G, GG, WG, and CWGT. If packing patterns are allowed to have amino acid match sets, the amino acids in the neighbourhood string can be generalised to match sets. The packing pattern derived from a neighbourhood string will inherit the neighbourhood string's coordinate sets.

Geometrical constraints are used to limit the lengths of the neighbour strings while keeping in the strings the potentially most interesting neighbour residues. For residue r and a neighbour residue s (within d Ångstrom), it is calculated whether the side chains 'face' eachother by calculating a half sphere in the residue's direction for each of r and s and only including s in the neighbour string of r if s is in r's half sphere and vice versa. Neighbour strings with fewer than 4 elements are discarded. See Table 1 for some statistics on the length of neighbour strings depending on d and use of the half sphere rule.

Starting with one neighbour string (called the probe) a simple depth first search algorithm can be used to find all generalisations of the probe that have occurrences in at least k structures. The simplest generalisation of the probe only contains the probe's anchor and matches all neighbour strings whose anchor has the same amino acid type. This gives us a pattern P (equal to the anchor) with a list of matches M_P. This pattern can be extended by appending a residue a from the probe forming $P \cdot a$. The matches to P are analysed to see if they can be extended to matches of $P \cdot a$, and it is checked whether $P \cdot a$ has sufficient support. If it does, it is again extended in all possible ways by appending elements defined from residues to the right of a in the probe. At any point in this exploration, the patterns can be extended to the left, e.g., a pattern P extended to $a \cdot P$. To avoid analysing the same pattern twice, once a pattern has been extended to the left, all further extensions will be to the left. As the search proceeds all patterns satisfying the constraints given by the user are output and they are postprocessed by separate programs.

For each pattern, the list of matching neighbour strings is stored. When a pattern P is extended to P' ($P' = P \cdot a$ or $P' = a \cdot P$), each match to P is analysed to see if it can be extended to match P'. For $P \cdot a$ each match to P is extended to include the ***first***, if any, a after the residues matching P. The matches to $a \cdot P$ are found in an analogous way. Alternative alignments between the pattern and the neighbour strings are not explored since this could be computationally expensive.

To ensure that any pattern with minimum support potentially can be found in the search, all neighbour strings from the $N - k + 1$ smallest (fewest residues) structures are used as probes. Any pattern with minimum support will have an occurrence in at least one of any subset of $N - k + 1$ structures and the smallest ones are used for efficiency reasons. The search procedure used makes it likely to find the same pattern multiple times since several of the matching neighbour strings may be used as probes. Therefore simple checksums are generated for each identified pattern and when new patterns are found, their checksum is compared to those of all previously discovered patterns before it is output.

In the search, the structural similarity of each match and the pattern is assessed by calculating the distance based RMSd. Distance based RMSd calculation was used in the search to save computations. Matches whose structural similarity is above the threshold are discarded. When patterns are output, the structural similarity of each pair of matches is calculated using superposition based RMSd using McLachlan's algorithm (McLachlan, 1979), and reported together with the a description of the matches.

In total, the algorithm is guaranteed to generate all patterns having minimum support, when the requirement of structural similarity is removed. When structural similarity is required, the heuristic of only including one alignment between a pattern and a matching neighbour string means that, potentially patterns can be discarded because a misalignment caused the structural similarity to be too low.

4 Results

SPratt2 was applied to a non-redundant subset of PDB called culledPDB [1] where the maximum pairwise sequence identity is 30% and only structures with resolution 2.0 or better are included. The set used was generated May 18th 2000 and contained 779 chains, in the following it is referred to as PDB*.

The parameters of SPratt2 was set to let it discover patterns matching at least k chains in PDB*, for k we tried all values between 2 and 20. The radius used was $d = 10$ and the half sphere constraint (see Algorithms) was applied. Furthermore all matches were required to superpose onto the pattern with a (distance based) RMSd of maximum 1.0Å. The computation took between 4 and 7 hours on a Sun Ultra 30 workstation with 512 Megabytes of memory. Figure 1 shows how the running time and the number of produced patterns depend on the value of k.

As Figure 1B shows, the SPratt2 runs produce large numbers of patterns. Semi-manual analysis of some of these have been carried out. For example, for each pattern the classification of the matching structures in the SCOP database (Murzin *et al.*, 1995) was retrieved automatically. It was found that most of the highest scoring patterns match structures from within the same family or superfamily in SCOP. For example, large number of patterns having matches within the immunoglobulin and serine protease families. While this confirms that the algorithm is able to recover known relationships in PDB, we also

[1] see http://www.fccc.edu/research/labs/dunbrack/culledpdb.html

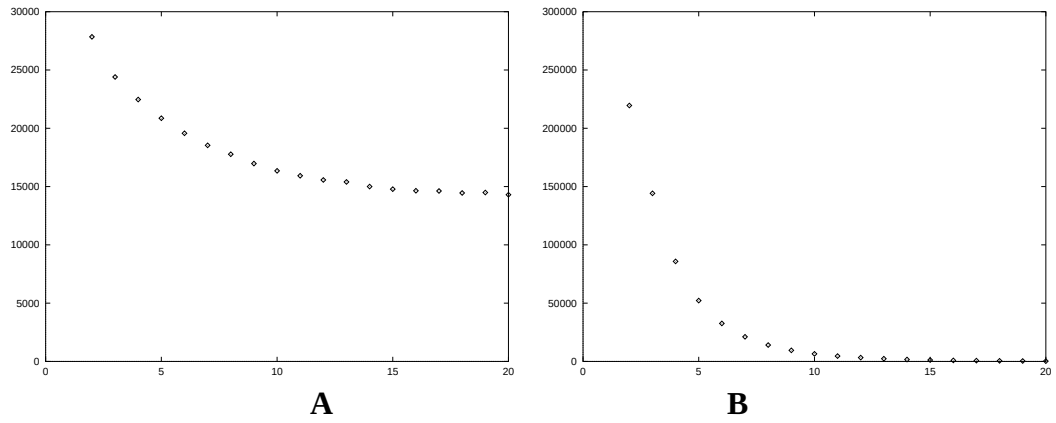

Figure 1: **A**: Running time of SPratt2 (in seconds – vertical axis) is shown when applied to PDB* (see text) with different values of k (minimum support requirement – horizontal axis). We see that for $k = 2$ SPratt2 takes almost 8 hours while for $k = 20$, the run takes around 4 hours. **B**: The number of patterns (vertical axis) produced by SPratt2 for different values of k (horizontal axis).

wanted to see if SPratt2 is able to find relationships between even more remote structures. More comprehensive anlaysis of the produced patterns will be performed and presented elsewhere. Here we give some details about two patterns that span different classes (alpha helical packing pattern) and different folds (cystein scaffold) in SCOP. These were among the patterns having highest scores produced in the run performed with minimum support 5.

4.1 A small cystine scaffold

Small solvent-exposed beta domains are often held into a structural framework by a network of disulfide bonds, without which they would not be stable in solution. These include small proteinase inhibitors, snake toxins, as well as small extracellular binding domains of receptors.

The packing pattern represented by the sequence motif C-x(4,19)-C-x(5,9)-C-x(4,17)-C was discovered and found to match a small two-disulfide framework within several small beta domains of diverse functions (see Table 2). Though the four cystine residues have a variety of spacings between them within the structural occurrences, they are found to superpose within a one Angstrom RMSD. The global topologies of the matching structures fall into two classes. The first class (e.g., 1bte, 3ebx, 9wga) comprises structures known as cyclic cystine knots (Craik *et al.*, 1999). These are four beta strands held together by three disulfide bonds. The packing pattern captures the bonds between strands 1-3 and 204 (strands numbered sequentially from N-terminus). Within this first class is wheat germ agglutinin (9wga), which has four occurrences of the cyclic cystine knot. The algorithm was able to find all four occurrences. The second class (1fle; serine proteinase inhibitor) has a different beta strand topology and the packing pattern connects two strands, and one of these to a loop.

	Pattern			
Protein	C	C	C	C
1clvI	C508	C517	C523	C531
1fleI	C32	C38	C44	C53
1bx7	C6	C11	C17	C22
3ebx	C3	C17	C24	C41
1bteA	C11	C31	C41	C59
9wgaA	C3	C12	C18	C24
9wgaA	C46	C55	C61	C67
9wgaA	C89	C98	C104	C110
9wgaA	C132	C141	C147	C153

Table 2: The matches to the pattern represented by the sequence motif C-x(4,19)-C-x(5,9)-C-x(4,17)-C.

	Pattern			
Protein	V	L	A	A
2dbm	V11	L130	A133	A137
1fua	V91	L164	A167	A171
1gdoA	V131	L141	A144	A148
1dciA	V105	L253	A256	A260
1iow	V143	L163	A166	A170
1qusA	V127	L351	A354	A358
4pgaA	V95	L140	A143	A147
1bw9A	V26	L49	A52	A56
1lam	V248	L339	A342	A346

Table 3: The matches to the pattern represented by the sequence motif V-x(9,223)-L-x(2)-A-x(3)-A.

4.2 Alpha helical packing pattern

Another high scoring pattern found by the algorithm is represented by the motif V-x(9,223)-L-x(2)-A-x(3)-A. In contrast to the flexible cystine motif, 3 of the distances in this pattern are exactly conserved (see Table 3). Inspection of the occurrences of this pattern revealed that the sub-pattern comprising L, A, A is in all occurrences on the buried face of a helix. The side chain of the Valine in the first position of the pattern faces this helix and is on a beta strand (e.g., in 1fwa) or on another helix (e.g., in 2gdm). Within proteins of similar topology, the position of the Valine is not topologically conserved: it occurs on different strands within a 4 strand a/b protein (1bw9), and in an 8 stranded a/b protein (1fua).

5 Discussion

The SPratt2 algorithm presented here together with the previously reported SPratt algorithm (Jonassen *et al.*, 1999) represents a novel approach to discover protein structure motifs. The SPratt2 algorithm is significantly more efficient than the SPratt algorithm which in practice was limited to the anlaysis of relatively small sets of proteins (up to, say, 50) and requiring high support.

The increased efficiency is due to several factors. Firstly, in SPratt the discovery of neighbour string patterns was performed using the tool Pratt (Jonassen *et al.*, 1995; Jonassen, 1997). Pratt takes as input sets of unaligned sequences and discovers patterns of the type used in the PROSITE database (Hofman *et al.*, 1999). In SPratt2 the patterns are effectively describing common subsequences (no restrictions on the sequence distance between residues mathing pattern elements) which results in SPratt2 exploring a smaller solution space. Also, in SPratt2 the search algorithm has been tailored to the particular application while Pratt is a general tool. Secondly, in SPratt2 the structural similarity of each match and the pattern can be assessed and used to reject matches. This was not possible in SPratt since Pratt is given only the neighbour strings themselves. Thirdly, we have introduced the half sphere constraint which reduces the lengths of the neighbour strings to be analysed. Finally, SPratt2 has been implemented as one program and its memory usage has been minimized to fascilitate larger scale analyses.

The efficiency of the SPratt2 algorithm enables mining of the complete set of known protein structures (represented by a non-redundant subset) in an exhaustive and fully automatic manner. In addition to being able to recover known relationships between proteins within families and super-families, it has also discovered packing patterns that occur in diverse folds and topologies. The cystine pattern and the helical packing pattern would be very difficult to induce from sequence information alone. However, in combination with structural information they are revealed from the data in an unsupervised fashion using the algorithm described in this paper.

While encouraging, the results also indicate further exploration, development and refinement of the method. Patterns spanning superfamilies and even fold classes are the most interesting, as these are unlikely to be found by sequence based methods. However,

the two patterns presented here do not appear to preserve topology or even secondary structure. The diverse topologies of the cystine pattern occurrences and the non-conserved secondary structure of the Valine in the helix pattern occurrences clearly illustrate this point. Due to these features, that these patterns cannot be used effectively for structural alignment, as can more specific patterns discovered in a supervised fashion (Jonassen *et al.*, 1999). Furthermore, it is clear that a four amino acid pattern, though conserved, is too short and its connecting regions too degenerate to be used for local tertiary structure prediction. Future research could address these specificity issues. A simple solution to the topology problem is to require preservation of secondary structure context of all elements in a packing pattern. A possible avenue towards the sequence specificity problem is to combine short packing patterns into wider conjunctions of patterns. One might also consider weakening the restriction of absolute residue conservation (considering the use of residue sets) but insisting on much longer patterns spanning a larger volume of 3D space. In conjuction with this, the RMSD tolerance could be relaxed in the hunt for patterns involving more residue context.

The discovery of the cystine pattern is promising, as similar cystine scaffolds occur in proteins of diverse function (Craik *et al.*, 1999; Norton & Pallaghy, 1998) and provide a convenient abstraction for protein fold classification. Future work will include specializing our algorithm to deal specifically with cystine packing motifs.

References

Craik, D., Daly, N., Bond, T., & Waine, C. (1999). Plant cyclotides: a unique family of cyclic and knotted proteins that defines the cyclic cystine knot structural motif. *J. Mol. Biol.* **294** (5), 1327–36.

Hofman, K., Bucher, P., Falquet, L., & Bairoch, A. (1999). The PROSITE database, its status in 1999. *Nucleic Acids Res.* **27**, 215–219.

Jonassen, I. (1997). Efficient discovery of conserved patterns using a pattern graph. *Comput. Appl. Biosci.* **13**, 509–522.

Jonassen, I., Collins, J. F., & Higgins, D. G. (1995). Finding flexible patterns in unaligned protein sequences. *Protein Science,* **4** (8), 1587–1595.

Jonassen, I., Eidhammer, I., & Taylor, W. R. (1999). Discovery of local packing motifs in protein structures. *PROTEINS: Structure Function, and Genetics,* **34**, 206–219.

McLachlan, A. (1979). Gene duplication in the structural evolution of chymotrypsin. *J. Mol. Biol.* **128**, 49–79.

Murzin, A. G., Brenner, S. E., Hubbard, T., & Chothia, C. (1995). SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.* **247**, 536–540.

Norton, R. & Pallaghy, P. (1998). The cystine knot structure of ion channel toxins and related polypeptides. *Toxicon.* **36** (11), 1573.