

Structural similarity search in databases: YAKUSA

Mathilde Carpentier⁽¹⁾, Sophie Brouillet⁽²⁾, Jol Pothier⁽³⁾
Atelier de BioInformatique, Universit Paris 6, 12 Rue Cuvier, 75005 Paris

⁽¹⁾ mathilde@abi.snv.jussieu.fr, ⁽²⁾ sophieb@abi.snv.jussieu.fr, ⁽³⁾ jompo@abi.snv.jussieu.fr

keywords: internal coordinates, structural database scanning, structural similarity, PDB

Introduction

As the number of protein structures in 3D databases increases (21151 PDB entries, Jun. 2003 [3]), further tools for structural analysis are needed, and, particularly, fast searching of structural similarities among proteins become valuable tool. There are three important problems in structure alignment: how to depict protein structures? which space of possible alignments is to be explored? and which structural alignment is the “best” one? Several solutions have been found and combined to align protein structures. There are two major ways to depict protein backbone structures: cartesian coordinates or internal co-ordinates (ϕ , ψ and ω angles [12] or α and τ angles [8]). In order to explore the space of possible alignments, different heuristics are used as Monte Carlo, genetic algorithms... For example, DALI [5]) software decomposes protein structures into hexapeptide distance matrices and uses Monte Carlo to optimize their merging into larger matching substructures. Many others solutions have been found to address these problems and have been implemented in softwares such as CE [13], Prosup [7], Kenobi [15]...

We have devised a software, YAKUSA (Yet Another K-uples Structure Analyser), for structural database scanning with a query protein structure. It searches for the longest common substructures (SHSP: ”Structural High Scoring Pairs”) between a query structure and every structure in the structural database. It relies on internal coordinates (α angles) representation of protein backbones (i.e. a linear sequence) and, as BLAST [2] does, it uses a deterministic finite automaton for pattern matching. It searches for similar sub-structures in proteins and uses a probabilistic score to rank matching sub-structures.

Method

Our goal is to quickly find all the structures of a database that have local structural similarities with a given query structure. As we use a linear description of the structure, the general principle, as in BLAST, is first to find in linear time all small fixed size common patterns (strictly identical or similar) between the query protein and every database protein. Secondly, these patterns are selected and extended to longest structurally similar segments between the two structures, which we call SHSP for ”Structural High Scoring Pairs”. Thirdly, a probabilistic score is computed for all the SHSPs found and this score is used to rank query/database structure pairs. This score is based on a probabilistic mixture transition distribution model (MTD) [11] computed on the α angles occurrences in PDB (MTD allows to model high-order Markov chains with a finite state space).

Structure description

As mentioned before, we use internal coordinates: α and τ angles to describe protein structures. The α angle is the dihedral angle between four consecutive α carbons and τ is the angle between three consecutive α carbons. As τ angles vary lightly around 100° and the distance between two C_α is always around 3.8 \AA , α angles are sufficient to describe accurately protein backbone structures.

We cluster α angles into classes over a mesh. We typically use a 10° mesh, then there are 36 classes of α angles. We represent a class by a numerical symbol (an integer). Thus, we describe the structure as a run of such symbols and it can be considered as a text. Therefore, we can apply any pattern matching algorithm to process this ”structural text”.

Searching SHSPs

The structural similarity is established in four steps, the three first ones being analogous to those used in BLAST: i) build-up of an automaton describing all fixed length patterns identical or similar to those in the query structure (which

we call seeds), ii) search for these patterns in every structure in database, iii) selection and extension of these patterns into SHSPs, iv) selection of compatible SHSPs for each pair query/database proteins.

The automaton built at the first step is the same kind of the one used in Aho-Corasick multi-pattern matching algorithm [1]. It allows to search at the same time for several patterns in time linear with the text length. To build this automaton, we do not use the Aho-Corasick's algorithm but we devised an algorithm taking advantage of query patterns overlapping. This algorithm also allows us to insert at the same time in the automaton, the query strict patterns and all patterns similar to previous ones, allowing non strict structural pattern matching.

After automaton building, each database structure, also encoded into symbols, is scanned linearly, and each symbol activates only one automaton transition. As searched seeds are small (usually 4, 5 residues), they are many of them, and we must merge them and select the longest ones. Then, these merged seeds are extended to SHSPs. A score S_a is defined for a matching pair of α angle segments:

$$S_a = \sum_{i=0}^s T - dc(\alpha_{r+i}, \alpha_{b+i})$$

with s : segment length, T : mean difference between two random α angles in the PDB structures, α_r : angle of index r in the query structure and α_b : angle of index b in the database structure. The mean difference between two angles in the PDB is near 42° . We set T to a default value of 30° in order to have a negative score between two randomly chosen angles. Therefore, the closer the angles and the longer the SHSP, the higher the score. Flanking residue pairs are appended to merged seeds in order to obtain the maximal score for the resulting SHSP.

All SHSPs must not be kept due to overlapping (for example, an α helix segment in the query can be aligned with two others in the database structure). At this step, as there are only few SHSPs (some dozens), all possible SHSPs combinations are generated. The best one according to the sum of its SHSPs score is kept.

Scores and ranking

After these operations, we obtain locally similar structural segments between the query structure and some database structures: the SHSPs. Usually, about 75% of database structures have at least one SHSP with a given query structure. It is then necessary to rank these hits and to keep only the best ones, i.e. the most significant ones. We have two kind of scores: the first one, S_a , is the sum of the SHSP scores (see below) and the second one, S_p , is the sum of the logarithms of α angle probabilities of the segment in the database structure corresponding to the SHSP (least is best). As α angle occurrence at a position is correlated to neighbour angles, we have used a mixture transition distribution model (MTD, [11]) to take this correlation into account to compute probabilities of α angle segments.

We generally use S_p score because canonical secondary structure elements (α helix and β strand) are very regular and frequent. Thus difference between their α angles is low, and their S_a score is very high, which biases the ranking. As they are very frequent and correlated, their occurrence probability is high and they are then discarded according to the S_p .

Nevertheless is the ranking often unsatisfactory with these scores still because of secondary elements. As they are very frequent, several secondary elements can be found even in two completely different structures. In order to distinguish these non relevant structure pairs from far related structure pairs, we determine "spatially" compatible SHSP groups. "Spatially" compatible SHSPs are all quite superimposed when one of them is used to superimpose the two structures. Then, the same scores (S_p and S_a) are computed but only for the best SHSPs group of each structure pair. For non related pairs, the best group contains only one SHSP, so their scores fall down. For far related structures, almost all SHSPs are compatible and scores stay about the same value.

Conclusion

Accuracy of YAKUSA alignments has been tested using several kinds of protein: mainly α as globins and cytochroms P450, mainly β as immunoglobins and pectates liases and others difficult cases. SHSPs found have been compared to results of other software alignments: DALI [5], CE [13]. The protein groups found has been compared to several structural classifications of proteins (SCOP [9], CATH [10], CE database [14], FSSP [6], VAST [4]).

Results show that this local approach for structural alignment is valuable: among the structural database, proteins of the same family are found, even they are far related or difficult to align. The use of "MTD" score improved results: far related proteins are better matched. The main interest of our approach is that it allows to find proteins which resemble to each other only for a local structural block, but not globally. This method of structural alignment is fast (one minute to scan 5000 structures) and accurate. This method will be used to automate protein structure classification. Moreover, the overall database SHSPs will be used to build a database of well specified cores suited for protein threading approaches.

References

- [1] A. Aho and H. Corasick. Efficient string matching: an aid to bibliographic search. *Comm. ACM*, 18(6):333–340, 1975.
- [2] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *J Mol Biol*, 215(3):403–10., 1990.
- [3] F. C. Bernstein, T. F. Koetzle, G. J. Williams, J. Meyer, E. F., M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi. The protein data bank. a computer-based archival file for macromolecular structures. *Eur J Biochem*, 80(2):319–24, 1977. 78043210 0014-2956 Duplicate Publication Journal Article.
- [4] J. F. Gibrat, T. Madej, and S. H. Bryant. Surprising similarities in structure comparison. *Curr Opin Struct Biol*, 6(3):377–85, 1996. 96397940 0959-440x Journal Article Review Review, Tutorial.
- [5] L. Holm and C. Sander. Dali: a network tool for protein structure comparison. *Trends Biochem Sci*, 20(11):478–80, 1995. 96108164 0968-0004 Journal Article.
- [6] L. Holm and C. Sander. Mapping the protein universe. *Science*, 273(5275):595–603., 1996.
- [7] P. Lackner, W. A. Koppensteiner, M. J. Sippl, and F. S. Domingues. Prosup: a refined tool for protein structure alignment. *Protein Eng*, 13(11):745–52, 2000. 21107589 0269-2139 Journal Article.
- [8] M. Levitt. A simplified representation of protein conformations for rapid simulation of protein folding. *J Mol Biol*, 104(1):59–107., 1976.
- [9] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia. Scop: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol*, 247(4):536–40, 1995. 95239730 0022-2836 Journal Article.
- [10] C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton. Cath—a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–108, 1997. 97454794 0969-2126 Journal Article.
- [11] A. Raftery. A model for high-order markov chains. *Journal of the Royal Statistical Society*, B, 47 (3):528–539, 1985.
- [12] G. N. Ramachandran, C. Ramakrishnan, and V. Sesisakharan. Stereochemistry of polypeptide chain configurations. *J Mol Biol*, 7:95–99, 1963.
- [13] I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. *Protein Eng*, 11(9):739–47, 1998. 99010845 0269-2139 Journal Article.
- [14] I. N. Shindyalov and P. E. Bourne. An alternative view of protein fold space. *Proteins*, 38(3):247–60, 2000. 20178325 0887-3585 Journal Article.
- [15] J. D. Szustakowski and Z. Weng. Protein structure alignment using a genetic algorithm. *Proteins*, 38(4):428–40., 2000.