# RECOGNITION OF STRUCTURE SIMILARITIES IN PROTEINS*

**Lin WANG · Yuqing QIU · Jiguang WANG · Xiangsun ZHANG**

Received: 25 July 2008

©2008 Springer Science + Business Media, LLC

**Abstract** Protein fold structure is more conserved than its amino acid sequence and closely associated with biological function, so calculating the similarity of protein structures is a fundamental problem in structural biology and plays a key role in protein fold classification, fold function inference, and protein structure prediction. Large progress has been made in recent years in this field and many methods for considering structural similarity have been proposed, including methods for protein structure comparison, retrieval of protein structures from databases, and ligand binding site comparison. Most of those methods can be available on the World Wide Web, but evaluation of all the methods is still a hard problem. This paper summarizes some popular methods and latest methods for structure similarities, including structure alignment, protein structure retrieval, and ligand binding site alignment.

**Key words** Binding site alignment, circular permutations, flexible alignment, protein structure alignment, protein structure retrieval.

## 1 Introduction

Nearly all proteins have structure similarities to other proteins[1]. An obvious example is that almost all proteins have regular secondary structure elements (SSEs). These similarities must rely on some mechanisms or principles of physics and chemistry or evolutionary relationships, even their combination. Understanding protein structure similarity is central to the post-genomic era[1].

As a fundamental research, many methods and techniques, including structure alignment algorithms, fast protein structure retrieval algorithms, binding site alignment algorithms, have been proposed for evaluating similarities of protein structures (see details in Figure 1).

Structure alignment algorithms consist of pairwise and multiple alignment algorithms. Pairwise structure alignment which is to detect geometric relationship between two structures is a fundamental problem in structural molecular biology. Multiple structure alignment is to identify the conserved structural common core among multiple structures[3]. It can aid in protein structure classification, understanding evolutionary conservation and divergence, and their correlation with sequences, so it carries more information than pairwise structure alignment.

Lin Wang · Yuqing QIU · Jiguang WANG
*Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing* 100190, *China*; *Graduate School of the Chinese Academy of Sciences, Beijing* 100049, *China.* Email: linwang@amss.ac.cn.
Xiangsun ZHANG
*Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing* 100190, *China.*
Email: zxs@amt.ac.cn.

*This research is supported by the National Natural Science Foundation of China under Key Research Grant No. 10631070, and JSPS and NSFC under JSPS-NSFC Collaboration Project under Grant No. 10711140116.

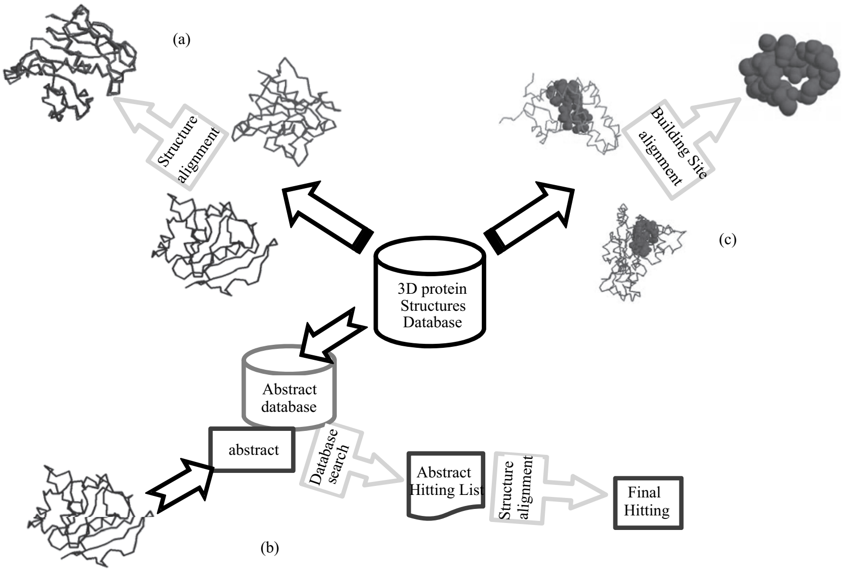

**Figure 1** (a) Structure alignment methods are either distance matrix-based, i.e. comparing distances between corresponding pairs of atoms in the two structures, or transformation-based, i.e. comparing relative positions of the corresponding atoms of two proteins that have been superimposed[11]. The two proteins 1DHFa (blue) and 8DFR (red) are in the same SCOP family[2] and 1DHFa is superimposed upon 8DFR by alignment algorithm. It obviously shows that the two structures are structurally similar. (b) Most structure retrieval algorithms belong to the filter-and-refine paradigm. First all 3D protein structures in the database are compressed to abstracts. Then query structure is compressed in the same way, and is retrieved in abstracts of the database using heuristic methods. After the filter process, finally a detailed structure alignment is performed on the left structures. (c) The binding site alignment algorithms usually consider the structure similarity problem by converting it to a common subgraph isomorphism problem. It shows the binding site (red) similarity of two proteins 1HQCb and 1ZTFa

One of the most important applications of protein structure alignment is to annotate unknown proteins. To do that, we need to search a protein against the protein structure database to find known proteins with similar structures. The methods of protein structure alignment are usually computationally intensive due to the intrinsic complexity of structural alignment. At the same time, the number of known protein 3D structures has been increasing dramatically due to the advances in laboratory technology (such as NMR and X-ray crystallography). So fast protein structure retrieval methods are needed to shorten computational time and decipher the feature of proteins. As reviewed by Aung and Tan[4], most of the fast searching algorithms belong to the filter-and-refine paradigm. All 3D protein structures in the database are compressed to abstracts firstly. The abstract can be sequence of vectors (or symbols) or graph. The query structure is compressed in the same way, and then retrieved in abstracts of the database. Retrieval in abstracts of the database will be much faster than that in 3D structures due to the simple form of abstract. After the filter process, a detailed structure alignment described above is performed on the left structures. Although results of these methods may not be optimal, they can usually get some results not the best but good quickly.

Protein fold space is more continuous than discrete, that is, highly repetitive set of substructures are detected in different folds[5]. The structure similarity of such substructures correlates well with the similarity of functions found between different folds containing these substructures[6] by constructing similarity network of protein folds. Further, Zhang et al.[7] constructed similarity networks of binding sites. Liu et al.[8] constructed a pocket (protein surface) similarity network and Park et al.[9] constructed a binding similarity network of ligands. All these networks are useful in predicting protein function. So recognition of common substructures (the local structural motifs or binding sites) is important for studies in protein function and protein fold changes in evolution[10].

In this paper, we will review some popular methods and recent new methods in identifying structural similarities according to three aspects.

## 2 Protein Structure Alignment

### 2.1 Pairwise Structure Alignment

Given two protein structures $X = \{X_1, X_2, \cdots, X_n\}$ and $Y = \{Y_1, Y_2, \cdots, Y_m\}$, where $X_i$ and $Y_j$ are the 3D coordinates of $i$-th $C_\alpha$ atom of $X$ and $j$-th $C_\alpha$ atom of $Y$. The alignment problem is to find a correspondence between the atoms from different proteins, a transformation (rotation and translation) of protein $X$, and a similarity measure RMSD (Root Mean Square Distance). RMSD vaule is calculated with the following formula:

$$\text{RMSD} = \sqrt{\frac{\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{m} s_{ij}(|A + RX_i - Y_j|^2)}{N}},$$

where $s_{ij}$ is 1 if atom $i$ is aligned with atom $j$, and is 0 otherwise; $A$ and $R$ are rotation and translation. Various techniques can be found based on different score functions to measure the suitability of the correspondence. Refs [3] and [11] both gave excellent reviews about the basic problem description and familiar framework. Here we focus on the detail of the newly published methods, while some classic methods can be found in the above reviews.

#### 2.1.1 Heuristic methods

Kolodny and Linial[12] presented an approximate polynomial-time algorithm to solve structure alignment problem under a score function. The algorithm searches in the space of rotations and exploits dynamic programming to optimize the score function. The complexity of the algorithm is $O(\frac{n^{10}}{\varepsilon^6})$, where $n$ is the length of protein and $\varepsilon$ is an additive error from all optima, so it is rather than a practical method. The authors also showed the difficulties in solving structure alignment based on distance matrices under some scoring functions, i.e., even the $\varepsilon$-approximation solution for this situation, the problem is still NP-hard. So a remarkable thing is that almost all the practical methods for structure alignment are heuristic. These methods are either distance-based by comparing distances between corresponding pairs of atoms in the two structures (e.g., DALI[13], CE[14], SAUCE[15]) or coordinate-based by comparing relative positions of the corresponding atoms of two proteins that have been superimposed (e.g. STRUCTAL[16]). Some other methods combine these two similarity measures (e.g. Matchprot[17], MatAlign[18]). For each category we describe some recently typical algorithms in the following.

Chen and Crippen[15] proposed the SAUCE method that first computes all aligned fragment pairs (AFPs) with the fixed length whose minimal RMSD values are less than a threshold. Then it constructs a network that considers AFPs as nodes, and connects two nodes if the two AFPs are geometrically consistent, the sequence order is maintained, and the fragments do not overlap in sequence in either protein. Two AFPs $i$ and $j$ are geometrically consistent if the sum of distance differences between corresponding pairs of atoms in two structures is less than a threshold, i.e.,

$$\sum_{k=0}^{m-1}\sum_{l=0}^{m-1}\left(d^A_{p_i^A+k,p_j^A+l}-d^B_{p_i^B+k,p_j^B+l}\right)<\varepsilon,$$

where $m$ is the length of fragment; $p_i^A$ and $p_i^B$ ($p_j^A$ and $p_j^B$) are initial atom positions of AFP $i$ ($j$) in protein $A$ and $B$; and $\varepsilon$ is a threshold. Consequently, the method computes cliques of the network to find alignments in core regions. Then alignments in core regions combined with environmental information can be extended to global alignment by dynamic programming.

The STRUCTAL method consists of two iterative steps: 1) for the current transformation of the protein, a correspondence of atoms is obtained using dynamic programming to maximize the STRUCTAL score

$$\text{STRUCTAL}=\sum\frac{20}{1+(\frac{d}{2}.24)^2}-10n_g,$$

where $d$ is the distance between corresponding $C_\alpha$ atoms, $n_g$ the number of gaps in the alignment; 2) this correspondence is used for a best rigid-body superposition with minimal RMSD.

Matchprot has the following three steps: 1) The algorithm computes neighborhoods of all atoms in two structures, then computes alignments between all pairs of neighborhoods from different structures and results in different transformations; 2) The transformations are clustered to get a smaller set of transformations; 3) For each transformation, the optimal alignment is computed between two structures and the algorithm returns the best alignment. The neighborhood has the structure neighborhood and the sequence neighborhood options. The structure neighborhood alignment is operated by common subgraph isomorphism algorithm[19]. The sequence neighborhood alignment is based on the spectral graph matching technique[20]. Aung and Tan[18] proposed MatAlign method that represents two structures as distance matrices, and aligns these matrices by means of two-level dynamic programming in order to find initial alignment. Then it refines the initial alignment iteratively to optimize the scoring function $S = 3 \cdot N/(1 + \text{RMSD})$, where $N$ is the alignment length. To further improve the accuracy of the algorithm, they use heuristic method to get multiple initial alignment seeds and choose the optimal one.

### 2.1.2 Convergent methods

SAMO[21] defines the structure alignment as a multi-objective optimization problem, i.e., maximizing the number of aligned atoms and minimizing their RMSD.

$$\begin{aligned}
\min \quad & \sum_{i=1}^{n_x}\sum_{j=1}^{n_y}s_{ij}(|A+RX_i-Y_j|^2-\lambda^2) && (1)\\
\text{s.t.} \quad & \sum_{i=1}^{n_x}s_{ij}\le 1 \quad \text{for} \quad j=1,2,\cdots,n_y, && (2)\\
& \sum_{j=1}^{n_y}s_{ij}\le 1 \quad \text{for} \quad i=1,2,\cdots,n_x, && (3)\\
& s_{ij}\in\{0,1\}, && (4)
\end{aligned}$$

where $A$ and $R$ are rotation and translation on protein $X$; $n_x$ and $n_y$ are lengths of proteins $X$ and $Y$; $X_i$ and $Y_j$ are the 3D coordinates of $i$-th $C_\alpha$ atom of $X$ and $j$-th $C_\alpha$ atom of $Y$. The variable $s_{ij}$ is 1 if $i$-th atom of $X$ and $j$-th atom of $Y$ are aligned, and is 0 otherwise. Furthermore, it can be decomposed into two subproblems including Weighted Least Square Subproblem

(LSS) and Integer Linear Programming Subproblem (LPS). The algorithm iteratively solves the LSS and LPS until convergence.

Martinez et al.[22] recently proposed the Low Order Value Optimization (LOVO) problem as a framework for the development of convergent structural alignment algorithms, i.e.,

$$f(x) = \max\{f_1(x), f_2(x), \cdots, f_m(x)\}, \tag{5}$$

$$f(x^*) = \max f(x), \tag{6}$$

where each $f_i(x)$ is a score function representing a correspondence between $C_\alpha$ atoms of two proteins, and depends on the transformations of the protein; $m$ is the number of all possible correspondences between $C_\alpha$ atoms of two proteins. So the objective function is to find a correspondence and a transformation to assume the maximum of $f(x)$. Based on the framework, they showed that the step 2 of STRUCTAL algorithm obtaining the rigid-body transformation that minimizes the RMSD for the current correspondence between $C_\alpha$ atoms is not a score-maximizing strategy. Accordingly they developed two algorithms DP-LS and NB-LS which are proved to be convergent. DP-LS iteratively performs two steps as follows: 1) It computes the correspondence with the fixed transformation by dynamic programming like the step 1 of STRUCTAL; 2) Given the correspondence, a single Safeguarded Line Search Newtonian iteration is performed to obtain a new transformation that guarantees a greater score. NB-LS differs with DP-LS in its step 1, that defines a new correspondence without considering both bijection and monotonicity, and is very fast to compute and meanwhile useful for non-sequential alignment.

From the formula (1) of SAMO, we can see that the objective function of SAMO is to minimize RMSD with the fixed correspondence between $C_\alpha$ atoms of two proteins. So SAMO is also in the LOVO framework. Although SAMO, DP-LS, and NB-LS can obtain a convergent solution for the scoring function, the resulting solution may not be globally optimal due to the non-convexity of the protein structure alignment problem[21].

### 2.1.3 Flexible alignment

Protein structures are flexible and undergo structural changes as part of their function[23], so aligning two structures as rigid bodies may miss structure similarity when there is structural distortion in one protein. To address the problem, some methods have been developed. The popular program FATCAT[24] uses a dynamic programming algorithm to connect aligned fragment pairs (AFPs) by combing gaps and twists (or hinges) between consecutive aligned fragment pairs, each with its own score penalty. It has the following iterative formula with at most $t$ twists,

$$S(k) = a(k) + \max\left\{\max_{e^1(m)<b^1(k),e^2(m)<b^2(k)}[S(m) + c(m \to k)],\ 0\right\},\quad \text{s.t.}\ \ T(k) \le t,$$

where $S(k)$ denotes the best score ending at AFP $k$; $a(k)$ is the score of AFP $k$ itself and is related to its length and minimal RMSD; $c(m \to k)$ is the score of introducing a connection between AFP $m$ and AFP $k$ and is related to the compatible content of AFPs $m$ and $k$, the unmatched regions and gaps created by the connection; $T(k)$ is the number of twists required for connecting the chain of AFPs up to AFP $k$ and is equal to $T(m)$+1 if two AFPs are not geometrically compatible.

### 2.1.4 Non-sequential alignment

There are also many non-sequential alignment algorithms, such as SAMO, Matchprot, NB-LS mentioned above, in which the aligned atom pairs do not keep with protein sequence order.

These algorithms are useful in detecting circular permutations that occurred in evolution of proteins, including $C_\alpha$-match (using geometric hashing)[25], GANSTA (using genetic algorithm)[26], CPalign[27]. Circular permutation is a phenomena of a protein that the original $N$ and $C$ termini are linked and new ones are created elsewhere.

Recently CPalign formulated the structural alignment problem as a special case of the maximum-weight independent set problem with the following programming:

$$\max \quad \sum_{\chi \in \Lambda} \sigma(\chi) \cdot x_\chi \tag{7}$$

$$\text{s.t.} \quad \sum_{a_t \in \lambda_a \in \Lambda_a} y_{\chi\lambda_a} \leq 1, \quad \forall a_t \in S_a, \tag{8}$$

$$\sum_{b_t \in \lambda_b \in \Lambda_b} y_{\chi\lambda_b} \leq 1, \quad \forall b_t \in S_b, \tag{9}$$

$$y_{\chi\lambda_a} - x_\chi \geq 0, \quad \forall x \in \Lambda, \tag{10}$$

$$y_{\chi\lambda_b} - x_\chi \geq 0, \quad \forall x \in \Lambda, \tag{11}$$

$$x_\chi,\, y_{\chi\lambda_a},\, y_{\chi\lambda_b} \in \{0,1\}, \quad \forall x \in \Lambda, \tag{12}$$

where $S_a$ and $S_b$ are two protein structures; $\Lambda = \Lambda_a \times \Lambda_b$ is a set of ordered pairs of equal length substructures of $S_a$ and $S_b$; the similarity function $\sigma : \Lambda \to R$ maps each pair of substructures to a positive similarity value; $a_t$ and $b_t$ are atoms of two structures with sequence number $t$, respectively. The variable $x_\chi$ is 1 if the ordered pair $\chi$ is in the final alignment and is 0 otherwise. The variable $y_{\chi\lambda_a}$ ($y_{\chi\lambda_b}$) considers whether the fragment $\lambda_a$ ($\lambda_b$) of $\chi$ is in the final alignment. This problem is NP-hard even for an $\varepsilon$-approximation solution, and is solved approximately by iteratively solving relaxations of the corresponding integer programming problem.

### 2.1.5 Evaluation of the methods

After so many methods have been proposed, a direct question is which one performs best or how they perform? Although Receiver Operating Characteristic (ROC) curves that evaluate how structural alignment algorithms perform compared to gold standard CATH[28] have been comprehensively used, some disadvantages have been reported in [29]. The distinguished problems are that CATH itself is a classification based on structure alignment method SSAP[30] and there are obviously similarities between cross-fold proteins. To correct this issue, Kolodny et al.[29] evaluated alignment methods by directly comparing alignment properties such as alignment length, RMSD and number of gaps, and found there is wide variation in the performance of different methods.

Mayr et al.[31] in another aspect focused on the analysis of the extent of agreement between alignments produced by different pairwise structure comparison algorithms. They showed that occurring of repetitions, indels, circular permutations, and local conformational changes affects the ability of different algorithms to obtain correct alignments.

## 2.2 Multiple Structure Alignment

Many algorithms for pairwise structure alignment can be extended to multiple structure alignment in a progressive or iterative manner, e.g. CE-MC (using Monte Carlo optimization)[32], MATT (allowing local flexibility in intermediate steps)[33], POSA[34], and Vorolign[35]. POSA represents a multiple alignment as a POG (Partial Order Graph) and iteratively performs pairwise alignment of POGs following the guide tree until all structures are merged. The method of alignment of two POGs is similar with that of FATCAT by using dynamic programming.

Vorolign also used two-level dynamic programming which is used in MatAlign. Vorolign represented two protein structures $x$ and $y$ as Voronoi tessellations, and defined the similarity of any two atoms $x_i$ and $y_j$ by their nearest-neighbor sets $N(x_i)$ and $N(y_j)$ which use dynamic programming based on similarity of two atoms $x_{i_k}$ and $y_{j_k}$ in the two nearest-neighbor sets. Then based on the similarity between any two atoms, they use dynamic programming again to obtain alignment between two structures. It is different with MatAlign in its definition of similarity of two atoms $x_i$ and $y_j$. MatAlign defines the similarity of two atoms by using dynamic programming based on distance vectors that correspond to the two atoms. Vorolign exerts multiple alignment by calculating all pairwise alignments and combining them by following a guide tree.

MultiProt[36] processed the input structures simultaneously and outputs both sequential and non-sequential alignment results. Given $m$ structures $S = \{M_1, M_2, \cdots, M_m\}$, it in turn selects $M_i$ as pivot $M_{\text{pivot}}$ for $i = 1$ to $m - 1$. For each pivot, let $S' = S \setminus M_{\text{pivot}}$, it has the following two steps. 1) The algorithm detects all aligned fragments with the pivot structure and then combines all possible of structurally similar fragments between two or more (e.g. $r$) structures, and results in a set of multiple transformations $(T_{i^1}, T_{i^2}, \cdots, T_{i^r})$; 2) Given the set of multiple transformations, the largest structural cores are detected between the aligned structures. At this stage the order of matched atoms can be optionally sequence-order preserved or be sequence-order independent.

## 3 Fast Protein Structure Retrieval

Here we roughly classify fast protein structure retrieval algorithms into two groups according to their methods for representing $3D$ structures.

### 3.1 Graph-Based Methods

Secondary Structure Matching (SSM)[37], a famous algorithm for fast protein structure alignment, represents protein structures as complete graphs of Secondary Structure Elements (SSEs). The SSEs, such as the $\alpha$-helices and $\beta$-sheets, are conserved local segments of protein $3D$ structures. A complex protein structure can be decomposed into several SSEs according to annotation tools like STRIDE[38]. Each SSE is a node and one edge is added between every two SSEs. Furthermore, nodes are labeled by types and lengths of their corresponding SSEs, and edges are labeled by nine-element property vectors describing distance, torsion angle, connectivity, and other biological relationships between corresponding two SSEs. Then the problem of comparing two structures becomes the comparison of two graphs. This is a common subgraph isomorphism problem. Usually this problem is transformed to a maximum clique problem in the node-product graph[39−42]. In SSM, a fast algorithm called CSIA with time complexity $O(m^{n+1}n)$ is applied[19], where $m$ and $n$ are respectively the lengths of compared two proteins. After the initial alignment of SSEs graphs is finished, in the refinement process, atom alignment is performed by some more rigorous techniques. SSM performs well because of its speed and accuracy.

$k$-Clique hashing[42] is another fast protein structure retrieval algorithm based on graph theory. It constructs a graph for each structure with its atoms as nodes, and an edge is added if the distance between two atoms is less than 12 Å. Then the alignment problem is converted to a maximum clique finding problem in the node-product graph. Due to the time complexity for exact algorithm, an approximate algorithm based on geometric hashing technique is used to solve the problem.

### 3.2 Encoding-Based Methods

There are many representations of protein structure as one-dimensional text string such as symbols encoding five-atom-long fragments[43], conformational letters[44], Ramachandran codes[45]. CPSARST (Circular Permutation Search Aided by Ramachandran Sequential Transformation)[45] is the first structural similarity search method for circular permutation. It represents each protein structure as string by using a Ramachandran sequential transformation (RST) algorithm. In the screening stage, the query string is subjected to two rounds of database searches using heuristic method (the second round differs with the first round in its duplication of query string). Results of the two rounds are filtered, i.e., hits with meaningfully improved similarity scores between two rounds are considered as CP candidates, and determine the putative permutation sites. In the refinement stage, each candidate is subjected to two rounds of structure alignments exerted by FAST[46], with and without CP manipulation (link the N and C termini, and break at permutation site), to retrieve CPs more precisely.

Compared with the one-dimensional string representation of protein structure, Konagurthu et al.[47] provided the method TableauSearch using a concise tableau representation of each protein that encodes the relative geometry of secondary structural elements to search for similar folding patterns. TableauSearch utilizes the two-level dynamic programming strategy which is similar to the MatAlign.

## 4 Ligand Binding Site Alignment

To compare the binding sites, several methods have been attempted including substitution matrix-based and graph-based methods. Binkowski et al.[48] use the sequence similarity to evaluate the similarity among protein surfaces via BLOSUM50 amino acid substitution matrix and Smith-Waterman algorithm.

Zhang et al.[7] compared binding sites by considering the physicochemical similarity and geometric similarity between residues. 1) They represent each binding site as a graph. The geometric center of the side chain of each binding residue is a node in the graph, and it is colored in four (PLD-BSSN-I) or eight (PLD-BSSN-II) different colors according to the physicochemical properties of the parent residue. Two nodes are connected by an edge if the distance between them is less than 12Å. 2) Given the graphs $G1$ and $G2$, an associate graph $P$ is constructed, where each node of $P$ is a product of two nodes with identical colors from the two graphs and two nodes of $P$ are connected by an edge if they are geometrically consistent, that is, the difference of the corresponding distances in $G1$ and $G2$ is less than 2Å. 3) The maximal clique of $P$ is detected as the comparison of the binding sites. Park and Kim[9] also compared binding sites via a maximal clique finding algorithm. They encode the binding sites as graphs and construct an associate graph like the method of Zhang but without considering chemical properties of residues. Then they score the aligned residues by BLOSUM62 substitution matrix and a gap penalty is assigned to unmatched residue-pair.

SOIPPA[10] is a powerful tool for the comparison of binding sites for its unnecessary in predefinition of binding sites. It has the following three steps. 1) Each protein structure is represented as Delaunay tessellation of $C_\alpha$ atoms and all $C_\alpha$ atoms characterized with geometric potentials[49]. It should be noted that Geometric potential has different distributions between residues in binding sites and residues not in binding sites. So the two structures are encoded as two node-weighted graphs respectively. In addition, each $C_\alpha$ atom in the sequence is assigned with a profile, i.e., probability distribution and position specific score matrix of 20 amino acids taken from a PSI-BLAST database search. 2) Given these attributes, they construct an associate graph in which each node is a product of two $C_\alpha$ atoms from different encoded protein graphs with the difference of their geometric potentials less than 50, and connect two nodes if they are geometrically consistent evaluated by different information such as distance difference and surface normal difference. 3) For the associate graph they assign a weight to each node using the sequence profiles. After forming the weighted associate graph, a branch-bound algorithm is used to find maximum-weight cliques as the alignments of binding sites.

## 5 Conclusion

Protein structure underlies the function of protein, ranging from transcriptional regulation to signal transduction. With the rapid growth in the number of structures in Protein Data Bank (PDB), there is ever increasing requirement in annotating the function of unknown proteins. Technological innovations such as accurate and efficient methods for computing structure similarities have advanced the understanding in protein fold space and their relationship with function.

Because the history of structure alignment problem is as long as that of sequence alignment problem, there are many algorithms not mentioned in this paper. Among them the methods worthy of mentioning include: Markov transition model of evolution for pairwise alignment[50], geometric hashing-based multiple alignment methods[51−52], projection-based vector methods for retrieval of protein structures[53−54], geometric hashing based[55] and ad hoc-based[56] binding site alignment algorithms, and so on.

The existing pairwise alignment algorithms work well for most instances but still need improvment, especially in some challenging cases, such as the occurring of repetitions, indels, circular permutations, and local conformational changes. For multiple structure alignment problem there are now little satisfactory algorithms and there is no method using the chemical or environmental information to consider the problem, so designing efficient and accurate algorithms is necessary. In addition, there is only one algorithm for circular permutation structure retrieval, so more useful techniques should be introduced.

It has been shown that the local surface structure, i.e., binding site, is more related to function and evolution of protein. Compared with retrieval of protein structures that considering global structure, to our knowledge there is no algorithm to retrieve protein structures that have similar binding sites. In fact, developing such an algorithm is very important for studying fold function and fold changes during evolution. In addition, most of the binding site alignment algorithms are exerted on binding sites which have been extracted from protein structures, so designing algorithms to calculate similarity of binding sites directly on protein structures may be more meaningful.

## References

[1] P. Koehl, Protein structure similarities, *Curr. Opin. Struct. Biol.* 2001, **11**(3): 348–353.

[2] 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.*, 1995, **247**(4): 536–540.

[3] I. Eidhammer, I. Jonassen, and W. R. Taylor, Structure Comparison and Structure Patterns, *J. Comput. Biol.*, 2000, **7**(5): 685–716.

[4] Z. Aung and K. L. Tan, Rapid retrieval of protein structures from databases, *Drug Discov Today*, 2007, **12**(17–18): 732–739.

[5] I. N. Shindyalov and P. E. Bourne, An alternative view of protein fold space, *Proteins*, 2000, **38**(3): 247–260.

[6] I. Friedberg and A. Godzik, Connecting the protein structure universe by using sparse recurring fragments, *Structure*, 2005, **13**(8): 1213–1224.

[7] Z. Zhang and M. G. Grigorov, Similarity networks of protein binding sites, *Proteins: Structure, Function, and Bioinformatics*, 2006, **62**(2): 470–478.

[8] Z. P. Liu, L. Y. Wu, Y. Wang, et al., Predicting gene ontology functions from protein's regional surface structures, *BMC Bioinformatics*, 2007, **8**: 475.

[9] K. Park and D. Kim, Binding similarity network of ligand, *Proteins*, 2008, **71**(2): 960–971.

[10] L. Xie and P. E. Bourne, Detecting evolutionary relationships across existing fold space, using sequence order-independent profile-profile alignments, in *Proc. Natl. Acad. Sci.*, 2008, **105**(14): 5441–5446.

[11] R. Kolodny, D. Petrey, and B. Honig, Protein structure comparison: implications for the nature of 'fold space', and structure and function prediction, *Current Opinion in Structural Biology*, 2006, **16**(3): 393–398.

[12] R. Kolodny and N. Linial, Approximate protein structural alignment in polynomial time, in *Proc. Natl. Acad. Sci.*, 2004, **101**(33): 12201–12206

[13] L. Holm and C. Sander, Protein structure comparison by alignment of distance matrices, *J. Mol. Biol.*, 1993, **233**(1): 123–138.

[14] I. N. Shindyalov and P. E. Bourne, Protein structure alignment by incremental combinatorial extension (CE) of the optimal path, *Protein Eng.*, 1998, **11**(9): 739–747.

[15] Y. Chen and G. M. Crippen, A novel approach to structural alignment using realistic structural and environmental information, *Protein Sci.*, 2005, **14**(12): 2935–2946.

[16] S. Subbiah, D. V. Laurents, and M. Levitt, Structural similarity of DNA-binding domains of bacteriophage repressors and the globin core, *Curr. Biol.*, 1993, **3**(3): 141–148.

[17] S. Bhattacharya, C. Bhattacharyya, and N. R. Chandra, Comparison of protein structures by growing neighborhood alignments, *BMC Bioinformatics*, 2007, **8**: 77.

[18] Z. Aung and K. L. Tan, MatAlign: precise protein structure comparison by matrix alignment, *J. Bioinform. Comput. Biol.*, 2006, **4**(6): 1197–1216.

[19] E. Krissinel and K. Henrick, Common subgraph isomorphism detection by backtracking search, *Software Practice and Experience*, 2004, **34**(6): 591–607.

[20] S. Umeyama, An eigendecomposition approach to weighted graph matching problems, *IEEE Trans. Pattern Anal. Mach. Intell.*, 1988, **10**(5): 695–703.

[21] L. Chen, L. Y. Wu, Y. Wang, et al., Revealing divergent evolution, identifying circular permutations and detecting active-sites by protein structure comparison, *BMC Structural Biology*, 2006, **6**: 18.

[22] L. Martínez, R. Andreani, and J. M. Martínez, Convergent algorithms for protein structural alignment, *BMC Bioinformatics*, 2007, **8**: 306.

[23] D. J. Jacobs, A. J. Rader, L. A. Kuhn, and M. F. Thorpe, Protein flexibility predictions using graph theory, *Proteins Structure Function and Genetics*, 2001, **44**(2): 150–165.

[24] Y. Ye and A. Godzik, Flexible structure alignment by chaining aligned fragment pairs allowing twists, *Bioinformatics*, 2003, **19**(9): 246–255.

[25] O. Bachar, D. Fischer, R. Nussinov, and H. Wolfson, A computer vision based technique for 3-D sequence-independent structural comparison of proteins, *Protein Engineering Design and Selection*, 1993, **6**(3): 279–287.

[26] B. Kolbeck, P. May, T. Schmidt-Goenner, et al., Connectivity independent protein-structure alignment: A hierarchical approach, *BMC Bioinformatics*, 2006, **7**: 510.

[27] J. Dundas, T. A. Binkowski, B. DasGupta, and J. Liang, Topology independent protein structural alignment, *BMC Bioinformatics*, 2007, **8**: 388.

[28] L. H. Greene, T. E. Lewis, S. Addou, et al., The CATH domain structure database: new protocols and classification levels give a more comprehensive resource for exploring evolution, *Nucleic Acids Research*, 2007, **35**(Database issue): 291–297.

[29] R. Kolodny, P. Koehl, and M. Levitt, Comprehensive evaluation of protein structure alignment methods: scoring by geometric measures, *J. Mol. Biol.*, 2005, **346**(4): 1173–1188.

[30] C. A. Orengo and W. R. Taylor, SSAP: Sequential structure alignment program for protein structure comparison, *Methods Enzymol*, 1996, **266**: 617–635.

[31] G. Mayr, F. S. Domingues, and P. Lackner, Comparative analysis of protein structure alignments, *BMC Structural Biology*, 2007, **7**: 50.

[32] C. Guda, S. Lu, E. D. Scheeff, et al., CE-MC: A multiple protein structure alignment server, *Nucleic Acids Research*, 2004, **32**(Web Server Issue): W100.

[33] M. Menke, B. Berger, and L. Cowen, Matt: Local flexibility aids protein multiple structure alignment, *PLoS. Comput. Biol.*, 2008, **4**(1): e10.

[34] Y. Ye and A. Godzik, Multiple flexible structure alignment using partial order graphs, *Bioinformatics*, 2005, **21**(10): 2362–2369.

[35] F. Birzele, J. E. Gewehr, G. Csaba, and R. Zimmer, Vorolign–fast structural alignment using Voronoi contacts, *Bioinformatics*, 2007, **23**(2): e205–e211.

[36] M. Shatsky, R. Nussinov, and H. J. Wolfson, A method for simultaneous alignment of multiple protein structures, *Proteins Structure Function and Bioinformatics*, 2004, **56**(1): 143–156.

[37] E. Krissinel and K. Henrick, Secondary-structure matching, a new tool for fast protein structure alignment in three dimensions, *Acta Crystallogr D Biol. Crystallogr*, 2004, **60**(1): 2256–2268.

[38] D. Frishman and P. Argos, Knowledge-based protein secondary structure assignment, *Proteins*, 1995, **23**(4): 566–579.

[39] J. F. Gibrat, T. Madej, and S. H. Bryant, Surprising similarities in structure comparison, *Curr. Opin. Struct. Biol.*, 1996, **6**(3): 377–385.

[40] H. M. Grindley, P. J. Artymiuk, D. W. Rice, and P. Willett, Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism algorithm, *J. Mol. Biol.*, 1993, **229**(3): 707–721.

[41] I. Koch and T. Lengauer, Detection of distant structural similarities in a set of proteins using a fast graph-based method, in *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 1997, **5**: 167–178.

[42] N. Weskamp, D. Kuhn, E. Hüllermeier, and G. Klebe, Efficient similarity search in protein structure databases by $k$-clique hashing, *Bioinformatics*, 2004, **20**(10): 1522–1526.

[43] J. M. Yang and C. H. Tung, Protein structure database search and evolutionary classification, *Nucl. Acids, Res.*, 2006, **34**(13): 3646–3659.

[44] X. Liu, Y. P. Zhao, and W. M. Zheng, CLEMAPS: Multiple alignment of protein structures based on conformational letters, *Proteins*, 2007, **71**(2): 728–736.

[45] W. C. Lo and P. C. Lyu, CPSARST: An efficient circular permutation search tool applied to the detection of novel protein structural relationships, *Genome Biology*, 2008, **9**(1): R11.

[46] J. Zhu and Z. Weng, FAST: A novel protein structure alignment algorithm, *Proteins*, 2005, **58**(3): 618–627.

[47] A. S. Konagurthu, P. J. Stuckey, and A. M. Lesk, Structural search and retrieval using a tableau representation of protein folding patterns, *Bioinformatics*, 2008, **24**(5): 645–651.

[48] T. A. Binkowski, L. Adamian, and J. Liang, Inferring functional relationships of proteins from local sequence and spatial surface patterns, *J. Mol. Biol.*, 2003, **332**(2): 505–526.

[49] L. Xie and P. E. Bourne, A robust and efficient algorithm for the shape description of protein structures and its application in predicting ligand binding sites, *BMC Bioinformatics*, 2007, **8**(Suppl 4): S9.

[50] T. Kawabata and K. Nishikawa, Protein structure comparison using the Markov transition model of evolution, *Proteins*, 2000, **41**(1): 108–122.

[51] N. Leibowitz, R. Nussinov, and H. J. Wolfson, MUSTA–A general, efficient, automated method for multiple structure alignment and detection of common motifs: application to proteins, *J. Comput. Biol.*, 2001, **8**(2): 93–121.

[52] O. Dror, H. Benyamini, R. Nussinov, and H. Wolfson, MASS: Multiple structural alignment by secondary structures, *Bioinformatics*, 2003, **19**(suppl 1): i95–i104.

[53] E. Zotenko, D. P. O'Leary, and T. M. Przytycka, Secondary structure spatial conformation footprint: A novel method for fast protein structure comparison and classification, *BMC Structural Biology*, 2006, **6**: 12.

[54] S. Bhattacharya, C. Bhattacharyya, and N. R. Chandra, Projections for fast protein structure retrieval, *BMC Bioinformatics*, 2006, **7**(Suppl 5): S5.

[55] A. Shulman-Peleg, R. Nussinov, and H. J. Wolfson, Recognition of functional sites in protein structures, *J. Mol. Biol.*, 2004, **339**(3): 607–633.

[56] B. Y. Chen, V. Y. Fofanov, D. M. Kristensen, et al., Algorithms for structural comparison and statistical analysis of 3d protein motifs, *Biocomputing* 2005: *Proceedings of the Pacific Symposium*, 2005, 334–345.