BIOINFORMATICS ORIGINAL PAPER
Vol. 21 no. 10 2005, pages 2362–2369
doi:10.1093/bioinformatics/bti353

Structural bioinformatics

Multiple flexible structure alignment using partial order graphs

Yuzhen Ye* and Adam Godzik*

Program in Bioinformatics and Systems Biology, The Burnham Institute, 10901 N. Torrey Pines Road, La Jolla, CA 92037, USA

Received on December 31, 2004; revised and accepted on February 23, 2005
Advance Access publication March 3, 2005

ABSTRACT

Motivation: Existing comparisons of protein structures are not able to describe structural divergence and flexibility in the structures being compared because they focus on identifying a common invariant core and ignore parts of the structures outside this core. Understanding the structural divergence and flexibility is critical for studying the evolution of functions and specificities of proteins.

Results: A new method of multiple protein structure alignment, POSA (**P**artial **O**rder **S**tructure **A**lignment), was developed using a partial order graph representation of multiple alignments. POSA has two unique features: (1) identifies and classifies regions that are conserved only in a subset of input structures and (2) allows internal rearrangements in protein structures. POSA outperforms other programs in the cases where structural flexibilities exist and provides new insights by visualizing the mosaic nature of multiple structural alignments. POSA is an ideal tool for studying the variation of protein structures within diverse structural families.

Availability: POSA is freely available for academic users on a Web server at http://fatcat.burnham.org/POSA

Contact: yye@burnham.org; adam@burnham.org

1 INTRODUCTION

Protein structure comparison has a long history in computational biology and it is almost as old as the better-known sequence alignment problem (Philips, 1970). These two problems share many similarities and the former can be viewed as a variant of the latter using a specific scoring function. Comparison of structures is more difficult than sequence comparison because of the non-local similarity score (Kolodny and Linial, 2004); it is not even clear if the optimal solution of structure comparison exists (Godzik, 1996). Both structure and sequence comparison can be pairwise (with two input proteins) or multiple (with more than two input proteins). The multiple comparison problem is more difficult than pairwise comparison even for sequences and it requires using heuristics to find an approximate solution in polynomial time (Gotoh, 1999). Compounded difficulties at both levels (i.e. structure versus sequence and multiple versus pairwise) resulted in only a handful of multiple protein structure alignment algorithms being developed so far. Existing methods differ in both the heuristics used to find the approximate solution and the type of scoring function used, including STAMP (uses a preliminary multiple sequence alignment followed by the optimization of alignment using structure information) (Russell and Barton, 1992), MALECON (searches a library of pairwise alignments for all three-protein alignments, which are progressively expanded to include additional proteins and more spatially equivalent residues) (Ochagavia and Wodak, 2004), CE–MC (builds a progressive alignment by sequentially aligning structures followed by refinement using the Monte Carlo algorithm) (Guda *et al*., 2001), MUSTA (Leibowitz *et al*., 2001b), MASS (Dror *et al*., 2003) and MultiProt (Shatsky *et al*., 2004) (the last three methods use the geometric hashing technique to detect small structurally similar motifs for subsets of input structures).

In contrast to the great variety of methods of building the multiple alignments (both sequence and structure), their presentation is almost always the same: the tabular row–column. This format has many limitations; most importantly it provides very limited information about similarities present only in a subset of proteins being aligned. This limitation is especially severe in structural alignment and it is the main reason why in most multiple structure comparisons the end result is a very small conserved protein core (Matsuo and Bryant, 1999; Leibowitz *et al*., 2001a). On the other hand, to understand the entire picture of protein structure evolution, we may want to analyze not only the core regions but also partially conserved and divergent regions. For this purpose, we need to adopt a representation that can describe nonlinear structure of the multiple alignment by capturing information on partially conserved and unique regions.

Directed graphs (Minieka, 1978), including acyclic and cyclic graphs, have been adopted to describe nonlinear multiple sequence alignments (Lee *et al*., 2002; Raphael *et al*., 2004) and to describe the nonlinear relationship of structure segments in a pairwise sequence alignment method using predicted local structure information (Ye *et al*., 2003). Lee *et al*. (2002) applied directed acyclic graphs (DAGs), known also as partial order graphs (POGs), to develop a sequence multiple alignment method called partial order alignment (POA). Raphael *et al*. (2004) took Lee's approach a step further and developed A-Bruijn alignment to align sequences with repeated and/or shuffled domains using directed graph possibly containing cycles. In this paper we propose to adopt a directed acyclic graph representation of multiple alignment for structural alignments. We formulate multiple protein structure alignment as a process of iterative pairwise alignment of two multiple structure alignments, each represented as a directed acyclic graph (Fig. 1). Constraints of consecutiveness must be obeyed in aligning two POGs, i.e. two nodes (residues) that have no order relationship or have wrong order can never be found in an alignment, as shown in Figure 2b. In this

*To whom correspondence should be addressed.

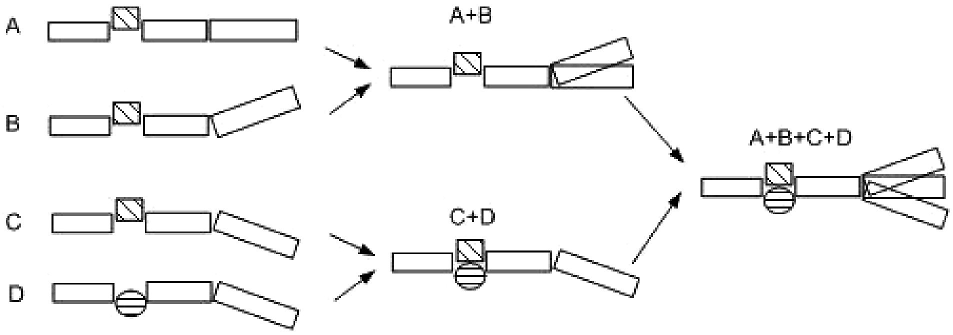

Fig. 1. A schematic demonstration of POSA procedure with four input structures (A, B, C and D). The order of structures to be aligned is given by a guide tree ((**A**,**B**),(**C**,**D**)). At each step, POSA performs alignment of two POGs (see Figure 2 for a demonstration) and in the resulting alignment, the differences of structures are kept as branches, which can be explored in the next step of alignment.

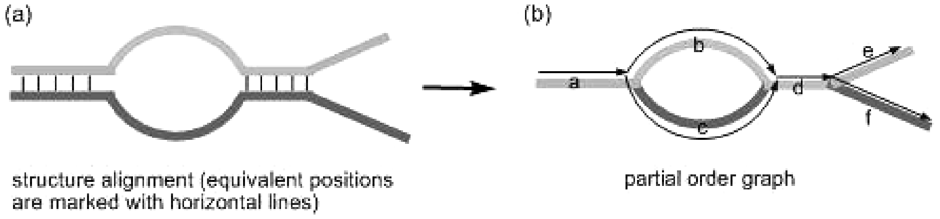
(a)
(b)
a
b
c
d
e
f
structure alignment (equivalent positions are marked with horizontal lines)
partial order graph

Fig. 2. POG representation of multiple structure alignment (only two structures are shown for clarity). (**a**) The structure alignment of two structures and (**b**) the POG representation of the alignment, in which a, b, c, d, e and f are representative residues for each of the regions in which they are located, respectively. It is obvious that residue a is before b and c, and d is after b and c. In contrast, residue b is neither before nor after residue c because b and c cannot be found in the same alignment, and the same situation exists for residues e and f. So in this alignment, some residues have clear sequential order, but some do not; in this sense, the alignment is partially ordered.

sense, the directed acyclic graph is partially ordered. To keep with the naming convention introduced by Lee *et al*., we call our new method partial order structure alignment (POSA).

Protein structure alignment has another complication. Most existing approaches (we call them rigid-body alignments) ask a question: 'What is the largest similar part of two proteins?' Recently, several protein structure alignments reformulated the central question of protein structure comparison to: 'How can one of the structures be rearranged to make it more similar to the other one?' To answer this question, such programs incorporated conformational flexibility in structure comparison (Boutonnet *et al*., 1995; Wriggers and Schulten, 1997; Shatsky *et al*., 2002; Ye and Godzik, 2003). All existing multiple structure comparison programs are based on the standard, rigid-body alignment. In contrast, the method we introduce here (POSA) combines the POA approach and the flexible structure alignment FATCAT (Ye and Godzik, 2003), thus becoming the first algorithm to perform and visualize multiple alignments of protein structures, while accounting for their conformational flexibility. Aided by the visualization tools that we have developed for displaying POA of structures, POSA is able to provide a global picture of the similarity of multiple structure alignment, including the partially conserved regions.

2 METHODS

2.1 Overview of the POSA method

POSA adopts a progressive strategy to build a multiple structure alignment given a set of input structures in the order provided by a guide tree (Grasso and Lee, 2004) (an arbitrary order can be treated as a special tree). A multiple structure alignment is represented as a POG and a single structure can be treated as a special POG containing only one structure (which in this case is completely ordered). POSA iteratively performs pairwise alignment of POGs following the guide tree until all the input structures are aligned together, as shown in Figure 1.

2.2 Partial order representation of structure alignment

We keep information about similarities and differences of the aligned structures by using POG representation of the alignment. A POG (or DAG) is a directed graph that contains no cycles (Minieka, 1978; Lee *et al*., 2002). A multiple alignment can be represented by a POG, in which nodes are the residues (aligned residues are merged) and the sequential order of the residues defines the edges between nodes. In the partial order format, we keep the information of the nonaligned residues (Fig. 1), which is essentially lost in the standard tabular row–column multiple alignment format. For clarity, Figure 2a demonstrates a simple case with only two structures aligned. Aligned positions are merged while positions that are not aligned form the branches in the graph (Fig. 2b). Figure 4 illustrates an actual example of POG generated by POSA.

2.3 Flexible structure comparison

We divide the multiple flexible structure alignment into iterative pairwise alignments of POGs or single structures (which can be treated as a special type of a POG). In POSA we use the same formalism of the flexible structure alignment as in FATCAT, a pairwise flexible structure comparison that we developed recently (Ye and Godzik, 2003). The only, but fundamental, difference is that in FATCAT we align two protein structures, whereas in POSA we also align two structure alignments.

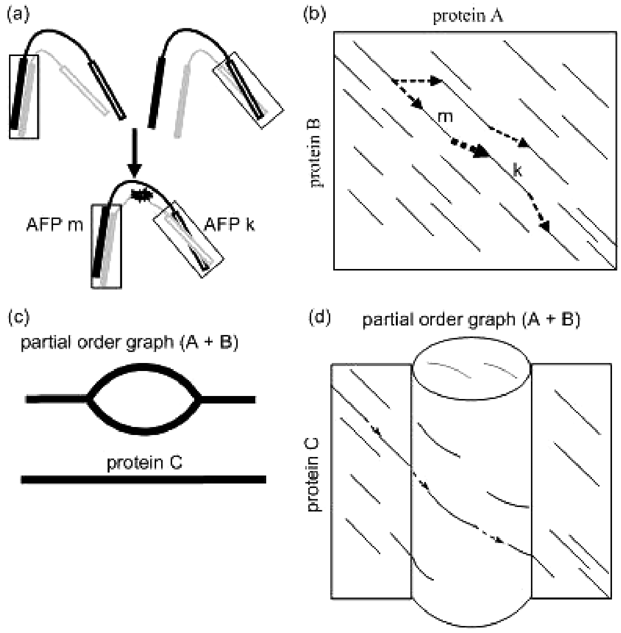

Fig. 3. Flexible structure alignment formulated as chaining AFPs allowing twists. (**a**) AFPs *m* and *k* cannot be found in the same alignment in a rigid-body alignment mode, but they can be in a flexible alignment when a twist is introduced in either of the input structures. (**b**) shows the alignment graph (which is planar) for pairwise flexible structure alignment by chaining AFPs allowing twist, where each line represents an AFP and a chain connecting compatible AFPs corresponds to a possible alignment of structures A and B. POSA follows the same formalism. The only difference is that in POSA all branches have to be explored in the AFP chaining process, in other words, the alignment graph is high-dimensional. (**c**) and (**d**) demonstrate a simple alignment of a POG of two aligned structures (A + B) and structure C by POSA.

2.3.1 Pairwise flexible structure alignment The pairwise flexible structure alignment program FATCAT starts by identifying a list of aligned fragment pairs (AFPs) (Shindyalov and Bourne, 1998), superpositions of two continuous fragments, in the two proteins to be compared (Fig. 3a). Two fragments of length 8 form an AFP if the root mean square deviation (RMSD) of their Cα atoms is less than a certain threshold (3.0 Å in the version available on the Web site) (Ye and Godzik, 2003). The FATCAT structure alignment is formulated as an AFP chaining process (Gusfield, 1999) with connections provided by extensions, gaps or twists (Fig. 3b). In contrast to a standard structural alignment, a rotation/translation (twist) can be introduced between two consecutive AFPs if it results in a substantially better superposition of the structures. FATCAT integrates simple extensions, gaps and twists into a unified scoring function and performs the alignment and hinge detection simultaneously by using dynamic programming. If we denote $S(k)$ as the best score ending at AFP k, it can be calculated from the best ending at previous AFPs that can be connected with AFP k subject to the consecutive constraints (Fig. 3b),

$$S(k) = a(k) + \max_{\substack{e^1(m) < b^1(k) \\ e^2(m) < b^2(k)}} \{S(m) + c(m \rightarrow k), 0\} \qquad (1)$$

$$\text{s.t.} \quad T(k) \leq t$$

where $a(k)$ is the score of AFP k itself; $c(m \rightarrow k)$ is the score of introducing a connection between AFP m and AFP k; $T(k)$ is the number of twists required to connect the chain of AFPs leading up to $S(k)$, calculated by $T(k) = T(m) + t(m \rightarrow k)$, where $t(m \rightarrow k)$ is 1 if a twist is required to connect AFP m and k and 0 if no twist is required. More details of the scoring function can be found in Ye and Godzik (2003).

2.3.2 Multiple pairwise flexible structure alignment POSA is based on iterative pairwise alignment of POGs created at the previous stages. We use the same formalism in aligning two POGs representing alignments of structures as in chaining AFPs in FATCAT. As shown in Figure 3d, in POSA, AFPs and their connections must obey sequential orders of the residues, i.e. two residues that have no sequential order (Fig 2b) can never be found in an AFP or in the chain of AFPs on the higher level. Owing to the presence of branches, the alignment graph of POGs is no longer planar; instead, it is high-dimensional, as shown in Figure 3d. In the example shown in Figure 3d, the searching space of POA of protein C and POG of the alignment of proteins A and B is three-dimensional. Still the best path of AFPs is the optimal alignment of two POGs subject to the constraints discussed above, and finding the optimal alignment can be solved by using dynamic programming (Equation 1). The computation complexity of POSA is $O(L^2N^3)$, where L is the average length of the input structures and N is the number of input structures.

2.3.3 Build a POG of multiple structure alignments considering flexibility As mentioned above we can generate POGs of structure alignments by merging equivalent positions of POGs from an earlier generation, which can represent single structures or previously aligned groups of structures. In the case of structure alignments without twists, all the equivalent positions will be merged; otherwise, we merge only the equivalent positions in the largest block (alignments are divided into $n + 1$ blocks by n twists) and record the information of equivalent positions from the remaining blocks. The reason we do not incorporate the flexibility in constructing POA is that we want to keep the difference of structures as long as possible during the comparison process (one of the major reasons to adopt POG representation of multiple alignments (Lee *et al*., 2002)). However, we have this information on record, and it is used at the end of a POSA process to recover the final alignment with flexibility.

2.4 POSA outputs

POSA reports a POG of the final multiple alignment with all input structures and the superposition of all the structures according to the alignment. In the case when twists are detected while comparing structures, POSA can also report a superposition of all the input structures after changing their coordinates according to the flexible alignment. To compare the performance of POSA with the existing programs (which always focus on the common core), we also report the length of common core and average RMSD (here, defined as the average root mean square of all the pairs of each position for all positions in the common core).

2.5 Guide tree

A small yet important issue that we have not addressed so far is the guide tree that we use to guide the process of multiple structure alignment. The problem of building the phylogenetic tree of proteins based purely on structural information is still open, mostly because despite several efforts (Hall and Barlow, 2003; O'Donoghue and Luthey-Schulten, 2003) no model has yet been proposed for the evolution of protein structures. In this paper, we do not aim to attack this problem; instead, we only want to provide guide trees for practical reasons. Guide trees can be generated in various ways using sequence and/or structure similarity. For the proteins that are very similar to each other in terms of both structure and sequence, trees based on sequence identity are preferable because we have a better understanding of sequence evolution. For distantly homologous structures or nonhomologous but similar structures, building trees of structures based on their structural similarity becomes the only choice because no sequence similarity can be used. Since we are performing multiple structure alignment for proteins with structural similarity, we decided to adopt the latter approach. We first calculate a distance matrix, using the P-value of the similarity of two structures from FATCAT as a measure of distance (Ye and Godzik, 2004), of all pairs for a group of structures to be aligned and then perform a single linkage clustering (Lance and Williams, 1967) on the distance matrix to generate a guide tree (program FATCAT2Tree was implemented for this purpose). The results reported in the Results section all used guide trees built using this approach.

2.6 Implementation

We implemented the POSA algorithm in C++ on a Linux platform with 1.8 GHz CPU. The running time of a POSA process varies from seconds to hours, depending on the number of input structures and their sizes. For example, the running time for comparing three calmodulin structures (details in Results section) is 13 s. We also implemented a public Web server for POSA at http://fatcat.burnham.org/POSA. The POG generated by POSA can be visualized using the dot function from the graphviz package (http://www.graphviz.org) and the superimposed structures are visualized online using the Chime plug-in. The graphs of protein structure superposition in this work were prepared using the molecular graphics software Pymol (http://pymol.sourceforge.net/).

3 RESULTS

We applied POSA to several textbook examples of multiple structural alignments and several structural families that have conformational flexibilities to show the advantages of considering flexibility in multiple structure alignment. Performance of POSA was compared with several alignment programs including MultiProt, CE–MC and MALCON using results from the original papers if available or using the corresponding Websites to calculate the alignments in other cases. The results, described in detail below, show that POSA achieved comparable performances in the examples in which no flexibilities are observed (the first example below compares globin structures). More importantly, POSA achieved better performance in the cases with proteins having conformational flexibility. In all cases, the POSA visualization approach provides a better view of the relationship of structures with high structural divergence. In the following subsections we discuss the details of four examples—comparison of globins, calmodulin-like proteins, tRNA-synthetases and Rossmann fold structures—focusing on the demonstration of the importance of using flexibilities in the structure comparison and the advantages of using POG representation for revealing the mosaic nature of protein structures. Finally, we ran POSA on 399 structure families and compared the results with HOMSTRAD alignments. The comparison shows that POSA-generated alignments are compatible with the HOMSTRAD alignments for the majority of the families, and more importantly POSA-detected flexibility in 27 families in which the flexibility introduced into the alignment is important for improving the alignment quality and for better understanding of the structures.

3.1 Globin family

The globin family has been extensively studied in the literature and is considered to be a relatively easy case for multiple structure comparison. We ran POSA on the example of 15 globin structures as described in Ochagavia and Wodak (2004) and compared the POSA result with the alignments produced by several other methods. POSA detected a common core of 71 positions with an average RMSD of 2.29 Å without introducing any structural flexibility into the alignment. For comparison, MALECON's alignment has 59 residues with an RMSD of 1.73 Å, MALECON$^+$'s alignment has 55 residues with an RMSD of 1.30 Å, the largest alignment reported by MultiProt that includes all 15 input structures has 50 aligned positions with an RMSD of 1.55 Å (note that MultiProt reports all local similarities of different lengths and different input structures), STAMP's alignment has 55 residues with an RMSD of 1.30 Å and CE–MC's alignment has 95 aligned positions with an RMSD of 2.37 Å. In short, as expected, different programs generated different alignments with various lengths and RMSDs, and the results could be summarized by the general rule of 'the longer alignment, the higher the RMSD'. Nevertheless, superpositions of the globin structures according to the alignments from different programs are very similar, showing that the performances of these programs are comparable for this case.

3.2 Calmodulin-like proteins

The advantages of POSA are more visible on the example of comparing calmodulin-like proteins (Crivici and Ikura, 1995). Calmodulin is a Ca^{2+}-binding protein that is a key component of the Ca^{2+} second-messenger system and is involved in the regulation of many biochemical and physiological processes. It has two domains with each domain having a pair of EF-hands (helix–loop–helix Ca^{2+}-binding motif). Each of the two domains may exist in a calcium-free or calcium-bound state, and in addition, domain movements result in an open or closed conformation of the whole protein. Some proteins from this family adopt permanently closed or open conformations. The calmodulin family exemplifies the function of domain movement and the flexibility of structure in the regulation of cell processes.

At first we consider only three calmodulin-like proteins with different conformational states for comparison. These three proteins are troponin C from chicken [an open, classical 'dumbbell' conformation of two pairs of EF-hand (PDB code 1ncx, Fig. 4a)], sarcoplasmic calcium-binding protein from *Branchiostoma lanceolatum* (PDB code 2sas, Fig. 4b; closed conformation in which both domains are pressed to each other and form a compact, globular structure) and EHCABP from *Entamoeba histolytica* (PDB code 1jfj, chain A, Fig. 4c; intermediate, partly open structure). When these three structures are compared by the standard, rigid-body multiple structure alignment programs, only one domain of the structures can be aligned. For instance, the largest alignment detected by MultiProt has 46 positions with an RMSD of 1.70 Å; the core detected by CE–MC has 62 positions with an RMSD of 6.79 Å. In contrast, POSA can get a much longer alignment of these three structures spanning their whole length (alignment length of 132 positions with an RMSD of 2.80 Å) when the flexibilities detected by POSA are incorporated into the superposition (Fig. 4e and h).

Another interesting observation from this POSA comparison is that the sarcoplasmic calcium-binding protein is significantly longer than the other two proteins (Fig. 4h) and the difference is mostly located in one of the EF-hand half domains (highlighted in Fig. 4e), which is 15 residues longer (indicated by an arrow in Fig. 4h). No data on the analysis of this long helix–loop–helix have ever been reported, but this difference is so interesting that it may deserve further exploration.

We further extended the list of input calmodulin-like proteins to include all the domains found in the calmodulin-like fold in the SCOP classification (1.65 release) (Andreeva *et al*., 2004), except for proteins with only one pair of EF-hands. Including one representative from each family resulted in 16 proteins. The interesting result from POSA is that some of these proteins are more similar in the N-terminal domain while others are more similar in the C-terminal domain such that at the end no common core in the sense of rigid-body comparison can be detected. However, after incorporating the flexibilities detected during the POSA comparison into the POG construction and superposition of the structures accordingly, we can recover

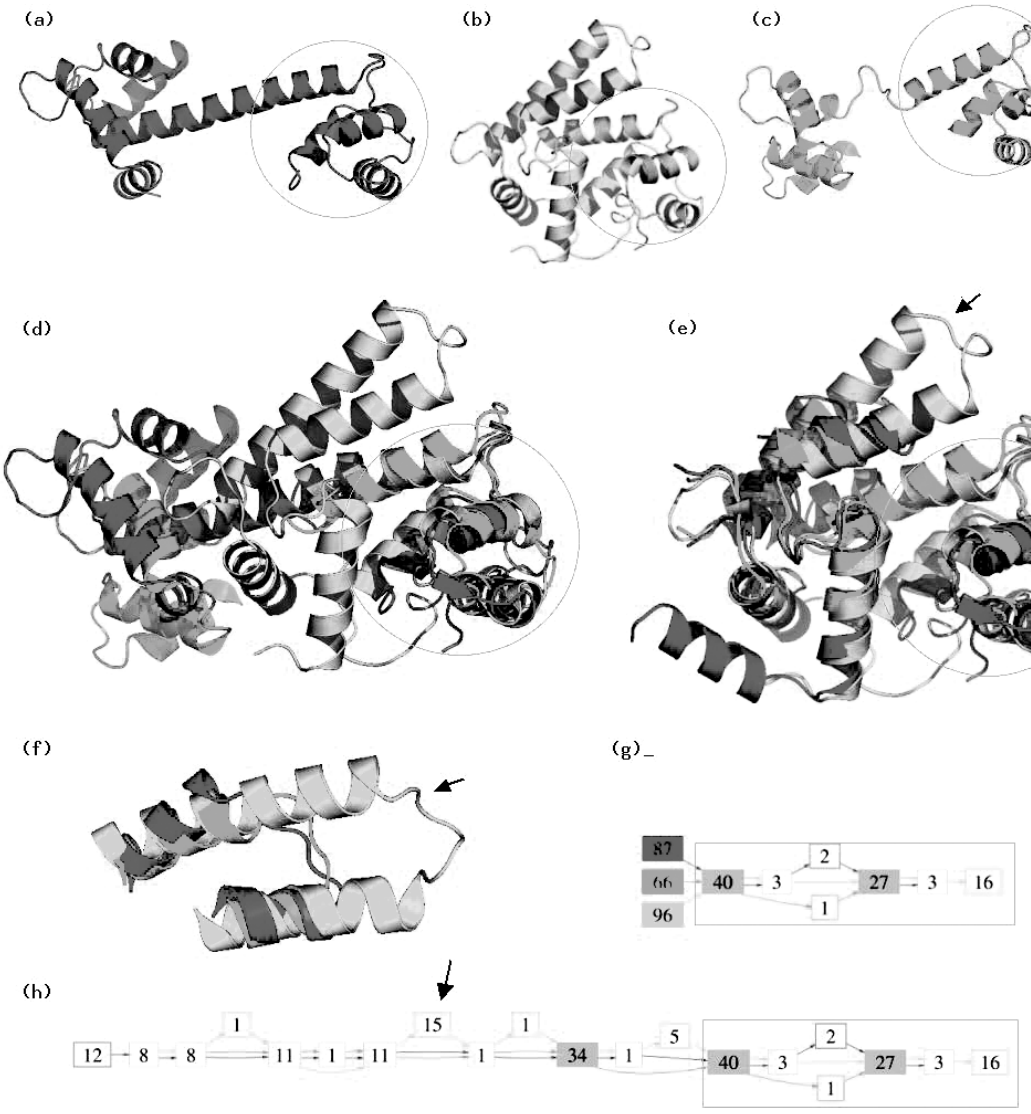

Fig. 4. The POSA results for three calmodulin structures compared. (**a**), (**b**) and (**c**) are the structures of 1ncs, 2sas and 1jfj, respectively. (d)–(h) show the POSA results of these three structures compared, in which 1ncs is shown in red, 2sas in yellow and 1jfj in green. (**d**) and (**g**) are the results of POSA before flexibilities are incorporated, whereas (**e**) and (**h**) are the results of POSA after flexibilities are incorporated in partial graph construction and superposition. (d) and (e) are the superposition of the structures and (g) and (h) are the POGs from POSA. In the POG, each box represents a region shared by proteins (in gray) or unique (in colors following the same color scheme). The numbers in the box represent the number of residues in each region (open boxes for regions with <20 residues and filled boxes otherwise). A structure corresponds to a path in the graph, e.g. 2sas corresponds to the path connecting yellow boxes (unique to 2sas) and gray boxes (also shared by at least one other protein) by yellow arrows, which is 96 → 40 → 3 → 27 → 3 → 16 (each number represents a region). In (d) and (g), only the C-terminal regions of these three structures can be well superimposed (in the gray circles) whereas their N-terminal regions are not. When flexibilities of the structures are considered, as shown in (e) and (h), the superposition of these three structures goes beyond the aligned regions in (d) and (g) and spans the whole length of the structures. (**f**) highlights the difference of the length of an EF-hand among these three structures [indicated by arrows in (e), (f) and (h)].

a common core of 16 input calmodulin-like proteins that includes 70 positions with the aligned residues spanning both N-terminal and C-terminal domains with an average RMSD of 3.27 Å.

3.3 tRNA synthetases

The aminoacyl-tRNA synthetases play a crucial role in protein synthesis by covalently linking an amino acid to the tRNA that bears the corresponding triplet anticodon. These synthetases use induced fit mechanism to achieve high specificity required for accurate protein synthesis; at the same time, the binding of the enzyme with their cognate amino is not very tight for efficient release of product (Torres-Larios *et al*., 2003). Four tRNA-synthetases were studied here including histidyl-tRNA synthetase (PDB code 1adj), prolyl-tRNA synthetase (PDB code 1hc7), threonyl-tRNA synthetase (PDB code 1qf6) and glycyl-tRNA synthetase (PDB code 1ati). All are multiple domain proteins and share two common domains, a catalytic

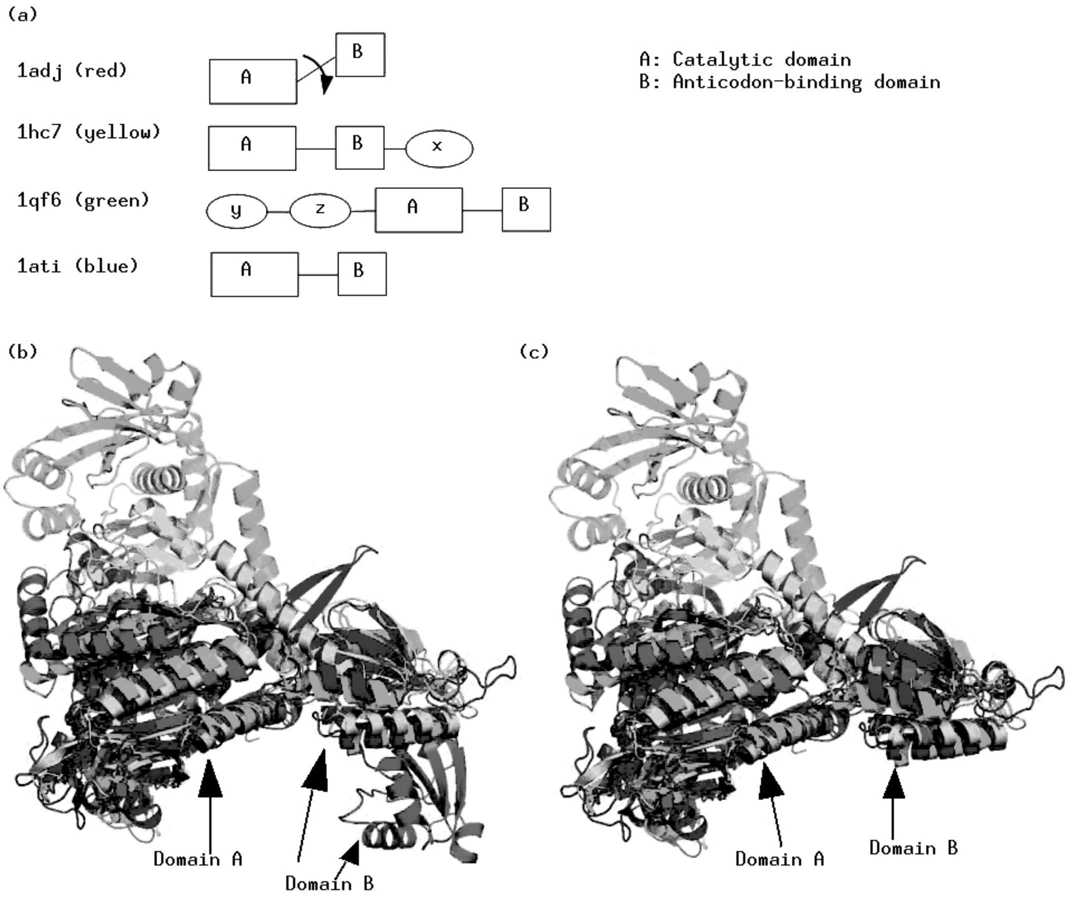

Fig. 5. POSA results comparing four tRNA synthetases. (**a**) shows the domain architecture of the four input synthetases, in which the orientation of B domain relative to A is different in protein 1adj (highlighted by an arrow) as compared with the other three synthetases. (**b**) and (**c**) show the superposition of tRNA synthetases before and after incorporating flexibilities detected by POSA, respectively. (c) shows that these four proteins can be well aligned in both A and B domains when the conformation of 1adj is changed according to POSA alignment.

domain and an anticodon-binding domain (Fig. 5a). This case was studied by MultiProt to show that it can detect the two common domains in the four input structures (Shatsky *et al.*, 2004). MultiProt reports two superpositions of these four structures, focusing on the catalytic domain and the anticodon-binding domain, respectively. POSA also detected the similarities in both domains. The important difference, however, is that POSA aligns both domains in the same picture, allowing us to clearly see the relationship of these two domains and simultaneously see how different the histidyl-tRNA synthetase (1adj) structure is in comparison with the other three. The simultaneous alignment of all four proteins is possible only because of the detection of the conformational change in 1adj at a hinge, and as a result we obtain an alignment spanning the two common domains with a common core of 278 aligned positions and an average RMSD of 2.92 Å (Fig. 5c).

3.4 NAD(P)-binding Rossmann fold

The typical feature of the Rossmann fold is a three-layer sandwich structure with a β-sheet between two layers of α-helix (Rossmann *et al.*, 1975) and sheet order 321456. According to SCOP (1.65 release), this fold consists of one superfamily, which in turn has 10 families. We selected 10 representative structures by using one structure from each family to run POSA analysis. Only a small common core of 48 residues with an average RMSD of 2.92 Å could be detected. The small size of the common core in this fold is also confirmed by MultiProt (Shatsky *et al.*, 2004), which reported an even smaller common core of 32 residues but with a smaller RMSD of 1.30 Å. The POSA result shows that only three β-strands and two α-helices from the three layers are relatively conserved among all the Rossmann fold proteins (Fig. 6c), which is a small part of their overall structures (Fig. 6a and b; in each protein, the regions corresponding to the common core are shown in red while the remaining regions are shown in green). However, when we visualize the superposition of the input structures showing not only the common core (shared by all the input structures), but also partly conserved regions, we can recover the three-layer overall structure much better, as shown in Figure 6d. Comparing this picture with the overall structures of the proteins in this fold (Fig. 6e and two individual structures shown in Fig. 6a and b), the mosaic nature of the proteins in this family is quite clear, suggesting that structural divergence has evolved in this family with the inclusion of additional structural elements to the common core.

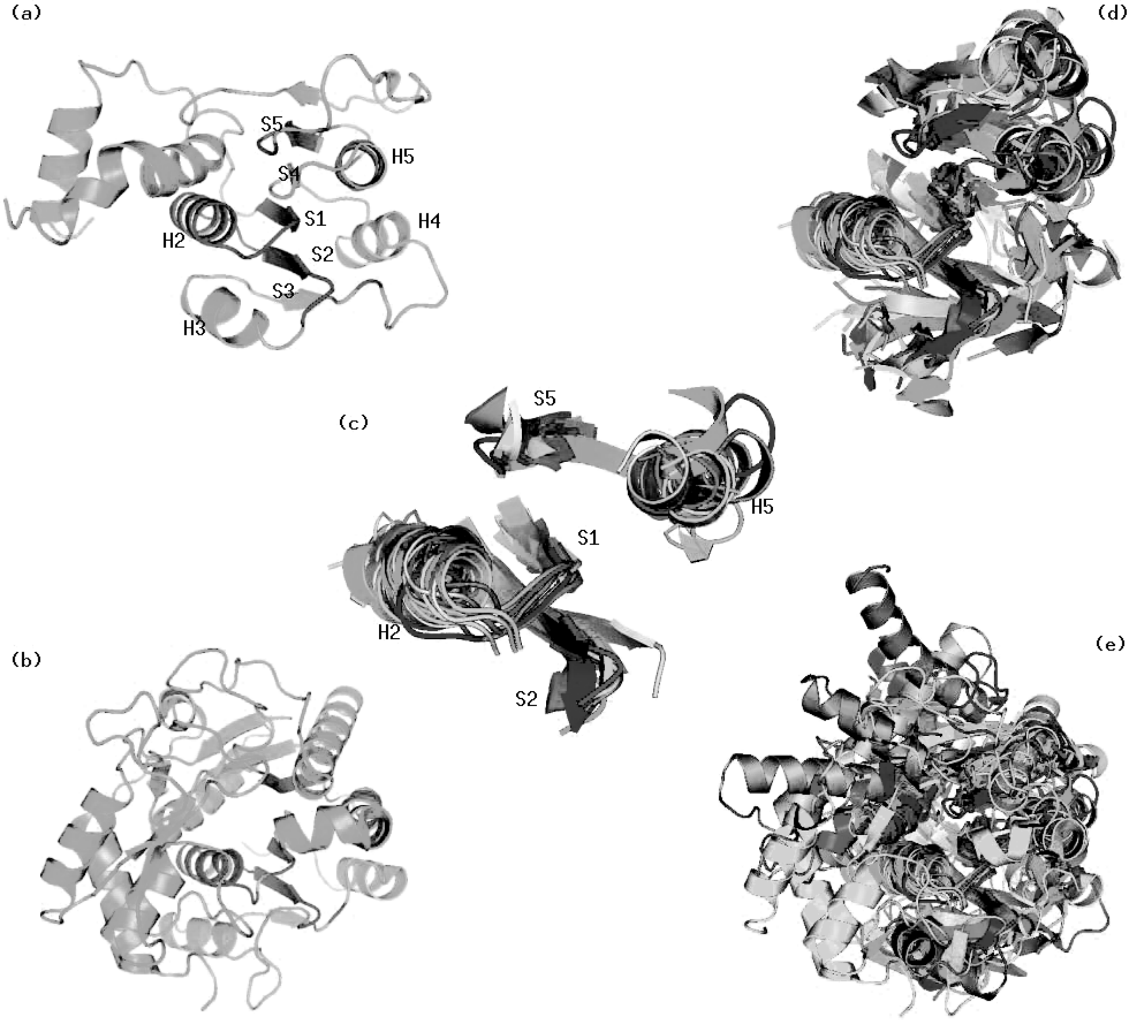

Fig. 6. The POSA results comparing structures of Rossmann fold. (**a**) and (**b**) show the structure of alcohol dehydrogenase-like protein, C-terminal domain (SCOP code d1heta2) and uridine diphosphogalactose-4-epimerase (SCOP code d1ek6a_), respectively. (**c**) shows the common core of Rossmann fold proteins using a representative structure from each family (total 10 structures: d1heta2, d1ek6a_, d1obfo1, d2naca1, d1kyqa1, d2cmd_1, d1np3a2, d1bgva1, d1id1a_ and d1oi7a1). (**d**) shows the superposition of the Rossmann fold structures displaying only regions that can be matched to at least five input structures. (**e**) displays the superposition of Rossmann fold structures, showing the whole input structure.

3.5 POSA performance on 399 structure families

We collected 399 homologous structure families, each with more than two structures based on HOMSTRAD (Dec 15, 2004, version, including 1032 families), a database of protein structure alignments for homologous families (Mizuguchi *et al*., 1998). HOMSTRAD alignments were prepared using the structure alignment programs MNYFIT, STAMP and COMPARER followed by manual examination of individual cases. We ran POSA on this large dataset to test its general performance. The details of POSA results with side-by-side comparison with HOMSTRAD alignments are available at http://fatcat.burnham.org/POSA/POSAvsHOM.html. Overall, POSA-generated alignments are slightly shorter than the HOMSTRAD alignments but result in slightly better superimposition of input structures with smaller RMSDs. The $(\mathrm{CORE}_{\mathrm{POSA}} - \mathrm{CORE}_{\mathrm{HOM}})/\mathrm{CORE}_{\mathrm{HOM}}$ and $(\mathrm{RMSD}_{\mathrm{POSA}} - \mathrm{RMSD}_{\mathrm{HOM}})$ averaged over 399 families are −4% and −0.797 Å, respectively, where $\mathrm{CORE}_{\mathrm{POSA}}$ and $\mathrm{RMSD}_{\mathrm{POSA}}$ ($\mathrm{CORE}_{\mathrm{HOM}}$ and $\mathrm{RMSD}_{\mathrm{HOM}}$) are the size of the common core and the corresponding RMSD of the POSA (HOMSTRAD) alignments. These two values are −3.8% and −0.528 Å, respectively, when excluding 27 structure families with flexibilities detected by POSA. We can conclude that even though POSA was developed aiming to address the flexibility and divergence of protein structures neglected in all previous multiple structure alignment programs, it generally performed well in comparison with HOMSTRAD alignments.

POSA revealed conformational flexibility, an important piece of information for understanding structures, in 27 structure families. Such information is not at all present in HOMSTRAD. In many of these families, POSA generated much-improved superimposition of structures with smaller RMSD by transforming some structures according to the flexibility it detected. For instance, POSA generated an alignment for family ‘rhv’ of virus coat proteins (including

Theiler virus coat protein 1tme, Mengo virus coat protein 2mev, rhinovirus serotype 1 coat protein 1**r1a**, rhinovirus 14 coat protein 4rhv, poliovirus coat protein 2plv and foot-and-mouth disease virus coat protein 1bbt) with a common core of 586 positions and an RMSD of 2.12 Å when incorporating flexibility; the size of the common core dropped to 372 (RMSD 2.45 Å) without incorporating flexibility. In contrast, HOMSTRAD alignment has a common core of 597 but with an extremely large RMSD (21.93 Å). We can expect that real advantages of POSA would be seen in aligning structures not present even in HOMSTRAD or similar databases, because of their inability to detect even a common core for the reason of extreme divergence of structures.

4 CONCLUSIONS

POSA was designed to describe flexibilities of protein structures and to consider both common and divergent regions in multiple structure comparison. By providing a visualization and classification of differences among structures, POSA provides an ideal tool for studying the structural divergence of protein structure families. As we have seen from the examples presented above, it is important to study not only the similarities but also the differences of structures for a better understanding of protein structures in terms of structure–function relationship and the evolution of structures. There is no doubt that new tools, such as POSA, are increasingly important in the age of a rapid growth of the protein structure databases, with most of the growth coming in from solving new structures in large protein families. With the anticipated goal of identifying most new folds by the structural genomics initiative, the questions of analysis of structural divergence in large protein families will surely come to the forefront of structural biology.

ACKNOWLEDGEMENT

This work was supported by NIH grant GM63208 and NSF grant DBI-0349600.

REFERENCES

Andreeva,A. *et al*. (2004) SCOP database in 2004: refinements integrate structure and sequence family data. *Nucleic Acids Res.*, **32**, D226–D229.

Boutonnet,N.S. *et al*. (1995) Optimal protein structure alignments by multiple linkage clustering: application to distantly related proteins. *Protein Eng.*, **8**, 647–662.

Crivici,A. and Ikura,M. (1995) Molecular and structural basis of target recognition by calmodulin. *Annu. Rev. Biophys. Biomol. Struct.*, **24**, 85–116.

Dror,O. *et al*. (2003) Multiple structural alignment by secondary structures: algorithm and applications. *Protein Sci.*, **12**, 2492–2507.

Godzik,A. (1996) The structural alignment between two proteins: is there a unique answer? *Protein Sci.*, **5**, 1325–1338.

Gotoh,O. (1999) Multiple sequence alignment: algorithms and applications. *Adv. Biophys.*, **36**, 159–206.

Grasso,C. and Lee,C. (2004) Combining partial order alignment and progressive alignment increases alignment speed and scalability to very large alignment problems. *Bioinformatics*, **20**, 1546–1556.

Guda,C. *et al*. (2001) A new algorithm for the alignment of multiple protein structures using Monte Carlo optimization. *Pac. Symp. Biocomput.*, **6**, 275–286.

Gusfield,D. (1999) *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*, 2nd ed. Cambridge, NY.

Hall,B.G. and Barlow,M. (2003) Structure-based phylogenies of the serine beta-lactamases. *J. Mol. Evol.*, **57**, 255–260.

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

Lance,G.N. and Williams,W.T. (1967) A general theory of classificatory sorting strategies. *Comp. J.*, **9**, 373–380.

Lee,C. *et al*. (2002) Multiple sequence alignment using partial order graphs. *Bioinformatics*, **18**, 452–464.

Leibowitz,N. *et al*. (2001a) Automated multiple structure alignment and detection of a common substructural motif. *Proteins*, **43**, 235–245.

Leibowitz,N. *et al*. (2001b) MUSTA—a general, efficient, automated method for multiple structure alignment and detection of common motifs: application to proteins. *J. Comput. Biol.*, **8**, 93–121.

Matsuo,Y. and Bryant,S.H. (1999) Identification of homologous core structures. *Proteins*, **35**, 70–79.

Minieka,E. (1978) *Optimization Algorithms for Networks and Graphs*. Industrial Engineering. Marcel Dekkar, NY.

Mizuguchi,K. *et al*. (1998) HOMSTRAD: a database of protein structure alignments for homologous families. *Protein Sci.*, **7**, 2469–2471.

Ochagavia,M.E. and Wodak,S. (2004) Progressive combinatorial algorithm for multiple structural alignments: application to distantly related proteins. *Proteins*, **55**, 436–454.

O'Donoghue,P. and Luthey-Schulten,Z. (2003) On the evolution of structure in aminoacyl-tRNA synthetases. *Microbiol. Mol. Biol. Rev.*, **67**, 550–573.

Philips,D.C. (1970) The development of crystallographic enzymology. *Biochem. Soc. Symp.*, **30**, 11–28.

Raphael,B. *et al*. (2004) A novel method for multiple alignment of sequences with repeated and shuffled elements. *Genome Res.*, **14**, 2336–2346.

Rossmann,M.G. *et al*. (1975) Evolutionary and structural relationships among dehydrogenases. *Enzymes*, **11**, 61–102.

Russell,R.B. and Barton,G.J. (1992) Multiple protein sequence alignment from tertiary structure comparison: assignment of global and residue confidence levels. *Proteins*, **14**, 309–323.

Shatsky,M. *et al*. (2002) Flexible protein alignment and hinge detection. *Proteins*, **48**, 242–256.

Shatsky,M. *et al*. (2004) A method for simultaneous alignment of multiple protein structures. *Proteins*, **56**, 143–156.

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

Torres-Larios,A. *et al*. (2003) Conformational movements and cooperativity upon amino acid, ATP and tRNA binding in threonyl-tRNA synthetase. *J. Mol. Biol.*, **331**, 201–211.

Wriggers,W. and Schulten,K. (1997) Protein domain movements: detection of rigid domains and visualization of hinges in comparisons of atomic coordinates. *Proteins*, **29**, 1–14.

Ye,Y. and Godzik,A. (2003) Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics*, **19**, ii246–ii255.

Ye,Y. and Godzik,A. (2004) Database searching by flexible protein structure alignment. *Protein Sci.*, **13**, 1841–1850.

Ye,Y. *et al*. (2003) A segment alignment approach to protein comparison. *Bioinformatics*, **19**, 742–749.