BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 9 2006, pages 1080–1087
doi:10.1093/bioinformatics/btl046

Structural bioinformatics

Development and validation of a consistency based multiple structure alignment algorithm

Jessica Ebert and Douglas Brutlag*

Program in Biophysics and Department of Biochemistry, Stanford University, Stanford, CA 94305 USA

Received on November 8, 2005; revised on December 20, 2005; accepted on February 6, 2006
Advance Access publication February 10, 2006
Associate Editor: Thomas Lengauer

ABSTRACT

Summary: We introduce an algorithm that uses the information gained from simultaneous consideration of an entire group of related proteins to create multiple structure alignments (MSTAs). Consistency-based alignment (CBA) first harnesses the information contained within regions that are consistently aligned among a set of pairwise superpositions in order to realign pairs of proteins through both global and local refinement methods. It then constructs a multiple alignment that is maximally consistent with the improved pairwise alignments. We validate CBA's alignments by assessing their accuracy in regions where at least two of the aligned structures contain the same conserved sequence motif.

Results: CBA correctly aligns well over 90% of motif residues in superpositions of proteins belonging to the same family or superfamily, and it outperforms a number of previously reported MSTA algorithms.

Availability: CBA is available at http://cba.stanford.edu/ and the source code is freely available at http://brutlag.stanford.edu/software/

Contact: brutlag@stanford.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

INTRODUCTION

Methods for the alignment of multiple proteins, whether guided by sequence, structure or both sources of information, are the cornerstone of many techniques in the biological sciences. A number of current areas of research depend on the ability to identify structurally equivalent residues across a group of proteins, including a variety of fold recognition and prediction techniques (Bystroff and Shao, 2002), homology modeling (Madej *et al*., 1995; Panchenko *et al*., 1999), evolutionary studies of both protein families and entire organisms, and investigations into the relationships among structure, function and sequence (Thornton *et al*., 2000; Todd *et al*., 2001). As factors that contribute to errors and ambiguity often differ between sequence alignment and structure superposition algorithms (Marchler-Bauer *et al*., 2002), multiple structure alignment (MSTA) has a place nearly everywhere multiple sequence alignment (MSA) is used. In particular, since structure is often more conserved than sequence, structural alignment has the potential to be more accurate than sequence alignment below the so-called twilight zone of sequence similarity (Doolittle, 1986). However, relatively few MSTA algorithms exist, and even fewer have been subjected to meaningful and clear validation.

*To whom correspondence should be addressed

Early MSTA efforts often focused on placing all structures in a common frame of reference by aligning each of them to a stationary reference, such as the structure that is in some sense the centroid of the group (Akutsu and Sim, 1999). This approach, however, does little to take advantage of the fact that more information can be obtained from simultaneous consideration of all of the structures than from independent analyses of pairs of proteins. Similar methods instead iterate this process, in each phase aligning the structures to the previous iteration's MSTA, typically by defining the consensus or average structure implied by the multiple alignment (Gerstein and Levitt, 1996; Taylor *et al*., 1994). Progressive algorithms align the most closely related structures first and eventually incorporate the more distantly related ones, and some perform a final optimization step once all structures have been aligned (Russell and Barton, 1992; Sali and Blundell; 1990, Yang and Honig, 2000). Since early iterations examine only a subset of the proteins, they may discard the optimal alignment as suboptimal. Alternatively, it is possible to use pairwise alignments as a starting point for building an MSTA and then determine which residues should be aligned using either structure-based scoring functions (Guda *et al*., 2004) or graph theoretic methods designed to find the best set of residue equivalencies across all structures that do not conflict with one another (Sandelin, 2005). Some recent algorithms make greater use of simultaneous consideration of all structures in the input set, typically by building up a multiple alignment out of aligned structural fragments (Dror *et al*., 2003; Shatsky *et al*., 2004) or using methods such as geometric hashing to identify structural equivalences across a group of proteins (Leibowitz *et al*., 2001).

MSTA algorithms that explicitly or implicitly rely on pairwise superpositions in some way depend on the principle of transitivity, often referred to as consistency, among the pairwise alignments (Gotoh, 1990). Mathematically, the three possible pairwise alignments among three structures are said to be consistent if the residue registrations of two of the alignments predict that of the third. Though a set of pairwise alignments will be fully consistent with one another only in the simplest cases, consistency has successfully been used as a driving force in the multiple alignment of both sequences (Do *et al*., 2005) and structures (Ochagavia and Wodak, 2004). In fact, one recent algorithm for obtaining the residue correspondences necessary for a multiple sequence or structure alignment is based entirely on the premise of 'relaxed transitivity' (Van Walle *et al*., 2003). This method constructs a graph whose nodes are residues and whose edges connect residues that are either aligned in a set of pairwise

alignments or that, if aligned, would create transitive cycles in the graph.

Here, we present CBA (consisteny-based alignment), an MSTA algorithm that uses information from consistent and nearly consistent regions of pairwise alignments to create a MSTA. CBA focuses almost entirely on the problem of determining which residues should be aligned given the relative orientations of the proteins once they have been superimposed. We use consistency first to find residue equivalencies across some or all structures in the input set and then to correct residue registration errors and gaps in the pairwise alignments in order to add additional aligned residues to the MSTA. We also introduce changes to the LOCK 2 pairwise protein superposition algorithm (Shapiro and Brutlag, 2004) that significantly enhance consistency among the alignments that are the starting point for CBA. CBA is a general framework for MSTA consisting of several modules that can be independently modified or even replaced. Simple modifications would also adapt it to perform multiple sequence alignment. Because CBA never uses sequence information, its alignments are well-suited to the study of the relationships among sequence, structure and function.

To date, very few validation techniques have been proposed for MSTA or MSA methods. Alignment algorithms are often validated by comparison with databases containing multiple alignments whose construction has been fully or partially guided by structure, such as HOMSTRAD (Mizuguchi *et al*., 1998), BaliBASE (Thompson *et al*., 1999) and OXBench (Raghava *et al*., 2003), but even alignments that are manually corrected or built entirely by hand are subject to error. Fully or semi-automated databases such as HOMSTRAD require explicit trust in the algorithms used to produce their alignments, thus establishing these methods as *de facto* gold standards. While the HOMSTRAD authors state that alignments are validated by hand, an incorrect automated alignment could easily prove to be a misleading starting point for manual analysis, particularly given the difficulty of visualizing multiple structure superpositions. Comparison of an algorithm with a database such as HOMSTRAD is therefore an implicit comparison with the alignment methods used to produce it, and thus lacks the generality we desire for validation of a new algorithm. Other attempts to quantify alignment quality through the use of simple measurements such as the alignment length and the distances between aligned alpha carbons lack substantive connections to alignment accuracy for anything other than closely related proteins. Though many complex scoring functions have been proposed, even a seemingly reasonable one is nonetheless arbitrary.

We instead validate CBA by examining alignments of protein structures containing the same conserved sequence motifs. Although global sequence alignments of proteins that share only limited sequence similarity are too prone to error to be used as gold standards for MSTA, distantly related proteins may nonetheless contain conserved local sequence motifs that are expected to align to one another. This method avoids the problem of ambiguous or incorrect global sequence alignments in regions of low sequence similarity (<20%) while still retaining the ability to assess alignments of proteins with low overall sequence identity in a fully automated fashion. We apply this benchmark to several previously reported MSTA algorithms and demonstrate that CBA outperforms them for multiple alignments of structures from the same family or superfamily.

METHODS

Validation datasets

We have aligned structures from each family and superfamily in release 1.65 of the Structural Classification of Proteins (SCOP) (Murzin *et al*., 1995) that contains at least three sequences with the same PROSITE pattern (Sigrist *et al*., 2002) or eMOTIF hit (Nevill-Manning *et al*., 1998). These structures come from an ASTRAL subset of the SCOP database created in such a way that no two sequences share >40% sequence identity (Brenner *et al*., 2000). Since not all PROSITE patterns have high specificity, we excluded as too general for our purposes any pattern that recognizes structures from multiple SCOP folds. Only structures from SCOP's all alpha, all beta, alpha + beta and alpha/beta classes were considered.

The 478 family validation sets have an average of 6.3 members, among which the average sequence identity as determined by BLAST (Altschul *et al*., 1990) is ~16.7%, while the 197 superfamily validation sets have an average of 13.4 members whose pairwise sequence identities average 14%. Hence, relatively few pairs of sequences in any given validation set reach the upper limit of 40% sequence identity, and many pairs are within or below the twilight zone of sequence similarity. Among the family and superfamily datasets that contain at least two examples of the same eMOTIF, an average of 4.0 structures share the motif. Since more than one eMOTIF may be built to recognize the same functional region, slightly different versions of the same motif may cover different members of the validation dataset. On average, the number of structures containing any eMOTIF is 4.6 and 5.7 for family and superfamily validation sets, respectively. The average number of structures containing the same PROSITE motif is 2.6 in the family validation datasets and 4.7 in the superfamily datasets. All datasets are available for download in text format at http://fold.stanford.edu/distributions/CBA/validationsets.html

LOCK 2

Though the pairwise structural alignments CBA uses to build a multiple alignment may be created by any structural superposition algorithm, we use an improved version of LOCK 2 by default. LOCK 2 produces an initial superposition by aligning secondary structure elements as previously reported (Shapiro and Brutlag, 2004), and replaces the residue alignment phase developed by Singh and Brutlag (1997) with a dynamic programming algorithm that scores the alignment of a query and target residue pair according to both the distances between their beta carbons and also the angles between the vectors defined by the alpha and beta carbons (Fig. 1). The use of the beta carbon allows LOCK 2 to encode a preference to align residues whose side chains point in the same general direction. Glycine residues' alpha hydrogens are replaced with beta carbons, whose positions are determined using ideal bond lengths and angles.

Assessment of motif alignment accuracy

When two or more structures in a multiple alignment contain the same local sequence motif, we assess the accuracy of the alignment in the motif region using the sequence motif as a gold standard. We sum the number of motif residues correctly aligned over all pairs of structures containing the motif and normalize to the number of residue pairs examined to give the percent of motif residues correctly aligned. If two or more motifs overlap, we count each position only once.

ALGORITHM

Steps 1–3: obtaining an initial multiple superposition

Because pairwise structural alignment algorithms consider only two proteins at a time, they cannot use information regarding the full extent of the variability observed in a given fold to resolve ambiguous regions of the alignment. Both imperfect scoring functions and this rather limited view of fold space lead to registration errors,

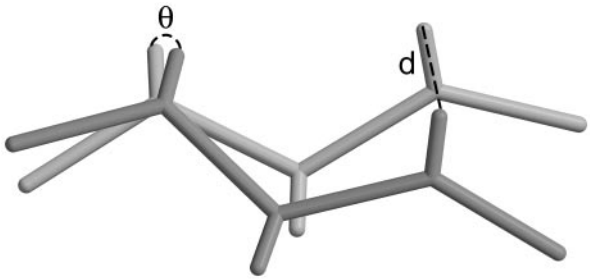

Fig. 1. The LOCK 2 residue alignment scoring function. LOCK 2 scores the alignment of a query and target residue pair by considering both the distance between their beta carbons and the angle between the vectors defined by each protein's C_α and C_β atoms. Beta carbon positions for glycine residues are determined using ideal bond lengths and angles.

and hence the pairwise alignments among a group of proteins tend to contain residue registrations that are not consistent with one another. This problem is exacerbated by the use of rigid body transformation methods, which may be unable to optimize the superposition in all structurally similar regions simultaneously. CBA seeks to resolve inconsistencies in alignments by considering an entire group of structures at once and extracting from them more information than is available from independent analysis of pairs of proteins.

The CBA algorithm consists of seven steps, each of which incorporates additional information into the alignment (Fig. 2). After computing the pairwise alignments among the proteins in the input set (Fig. 2, Step 1), CBA places all of the structures in the input set into a common frame of reference (Fig. 2, Step 2). This is particularly helpful in cases involving structures with repeated subdomains. Given three structures, two with multiple copies of the same subdomain and one with only one copy, LOCK 2 might align the smaller structure to different repeats in the two larger structures, thus yielding completely inconsistent pairwise alignments. While this is not necessarily incorrect, it is inconvenient for analyzing similarities across all three proteins. Following a number of progressive MSTA algorithms, CBA clusters all of the structures in the input set by average linkage using LOCK 2's global alignment scores as measures of similarity. This creates a binary guide tree that we use to progressively transform all structures into the same frame of reference beginning at the bottom of the tree, where clusters contain the most similar structures, and proceeding towards the root node. In the trivial case of a node at the bottom of the tree with only two descendants, we transform one child structure onto the other using the transformation matrix from the pairwise alignment and label the parent node with the resulting superposition. When a node has more than two descendants, we choose one descendant of the left child and one from the right such that the two selected structures have the highest alignment score among all such pairs, and then use their pairwise transformation to add all descendants from the left child to the right child's MSTA.

Since this process may change some pairwise superpositions, we run LOCK 2's residue alignment phase a second time to obtain improved pairwise residue correspondences (Fig. 2, Step 3). Though LOCK 2's residue alignment phase does employ an iterative algorithm in which each cycle uses the previous cycle's residue registration to update the transformation of one protein onto the other, these changes in the overlap between the query and target proteins are minor and serve only to make the correct residue registration more clear. CBA discards the changes to the transformations and retains only the new residue registration, thus maintaining the common frame of reference obtained in Step 2. From this point forward, the positions of the proteins in space remain fixed.

Step 4: obtaining an initial residue registration across all structures

Although the third step of CBA places all proteins in the same frame of reference, this superposition does not directly imply an unambiguous residue registration across all of the structures in any but the simplest of alignment problems. For the sake of clarity, we make a distinction between the superposition of a group of proteins, which defines their relative positions in space, and their residue registration, which specifies the groups of structurally equivalent residues that are aligned to one another. The superposition is fixed from this point forward, and the remaining phases of the algorithm focus on the more difficult problem of determining the residue registration without using sequence information.

The fourth step of CBA applies a fast implementation of the Markov cluster algorithm (MCL) (Van Dongen, 2000) to find the residue registration across all of the proteins that is maximally consistent with the pairwise alignments obtained at the end of Step 3 (Fig. 2, Step 4). One may view this task as the problem of finding clusters in a graph whose nodes are residues and whose edges connect nodes aligned in the pairwise superpositions (Fig. 3). Edge weights, which reflect CBA's confidence that the residue correspondences from the pairwise alignments are correct, are proportional to the LOCK 2 residue alignment scores described in the Methods section. If the pairwise alignments were perfectly consistent, this graph would consist of a series of fully connected clusters, such as the leftmost cluster of Figure 3, and it would have no edges connecting nodes in different clusters.

The MCL algorithm operates under the paradigm that a random walk beginning in a region of the graph that corresponds to a cluster—in this case, a column in the multiple alignment—will visit many of the nodes in that cluster before leaving it for another region. That is, regions of the proteins that are structurally equivalent will give rise to groups of nodes in the graph that are connected to one another by many edges and that have relatively few edges connecting them to other regions of the graph. If the structural similarity in the region is strong, the edges connecting these nodes will furthermore be associated with large weights.

Formally, MCL maximizes the flow through the graph where it is already relatively strong and weakens it where it is already weak, thus strengthening correct residue correspondences, eliminating edges that describe incorrect correspondences, and adding edges between residues not aligned by LOCK 2 when the principle of transitivity suggests that they are structurally equivalent. Each resulting cluster contains equivalent residues across a subset of the proteins in the input set and is the basis for a column in the multiple alignment. Since the MCL method does not enforce sequence order constraints and does not require that each cluster contain only one residue from each protein, CBA must deal with these matters in subsequent steps of the algorithm. In the meantime, one can consider the latter issue in particular as a reflection of uncertainty in the residue registration.

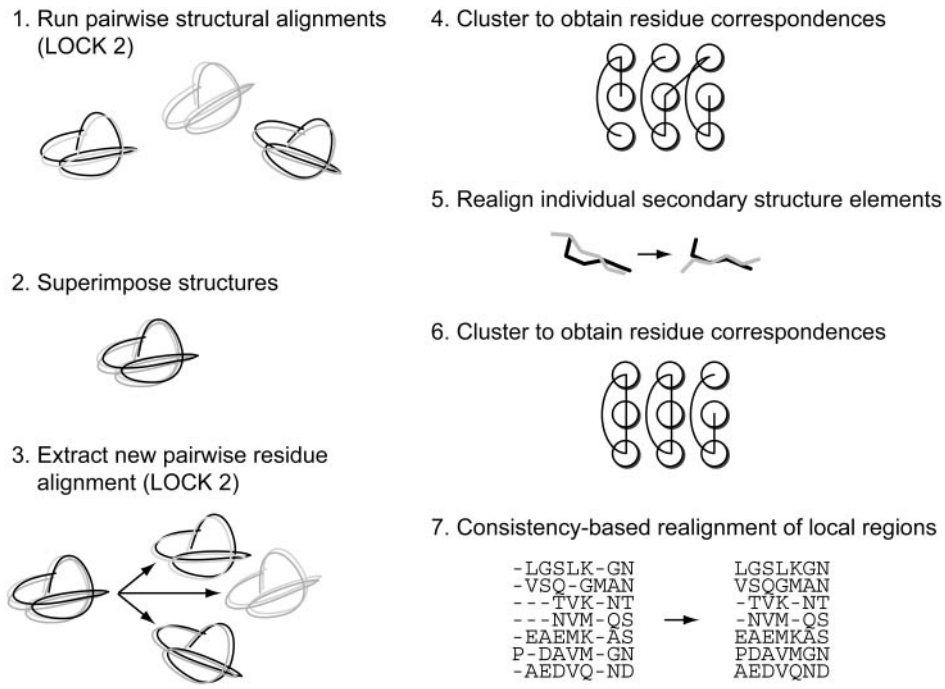

Fig. 2. The CBA algorithm.

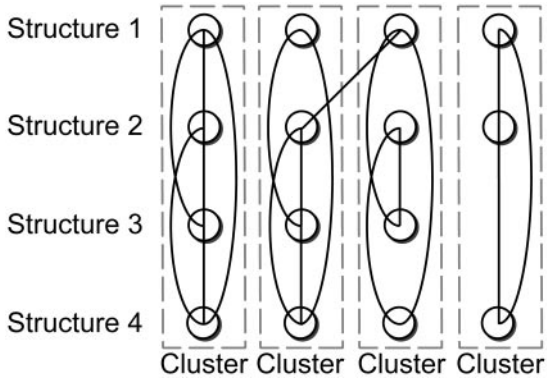

Fig. 3. A graph theoretic approach to determining residue correspondences from pairwise alignments. CBA determines residue correspondences across a group of proteins by constructing a graph in which each node corresponds to a residue from one of the proteins to be aligned. Two nodes are connected by an edge if they are aligned in the pairwise alignment of the structures to which they belong. Thus, no edges exist between nodes of the same protein, and a given node from one protein is connected to at most one node from each additional protein. The leftmost column of nodes in the graph shown here is a connected component, and thus corresponds to a column in the multiple alignment. The two middle columns of nodes contain a registration error that links them together, and the lack of an edge between the nodes from structures three and four in the third column indicates that these residues were left unaligned in the corresponding pairwise alignment. The rightmost column of nodes reflects a deletion in structure 3.

Steps 5 and 6: realignment of secondary structure elements

Even when large portions of two protein structures align well, individual helices and strands may superimpose poorly, thus making identification of residue correspondences difficult. Realigning each pair of superimposed secondary structure elements in isolation circumvents this limitation of rigid body superposition algorithms, which cannot always optimally superimpose all structurally similar regions of two proteins simultaneously (Fig. 2, Step 5). Since more than one registration of two secondary structure elements of the same type is possible, CBA initially superimposes them onto one other using the residue correspondences identified by the MCL algorithm in Step 4. This produces a new superposition that is fully consistent with the current MSTA but that is better suited to LOCK 2's scoring functions than was the overlap of the two secondary structure elements in the global pairwise alignment.

We run the LOCK 2 residue alignment phase on the newly superimposed pair of secondary structure elements to fill in gaps and to correct mistakes in the current multiple alignment's residue registration. We then transfer the resulting residue correspondences to the global pairwise alignment, but discard the changes in the superposition of the two secondary structure elements so that the positions of the proteins in space remain unchanged. After repeating this process for all pairs of aligned secondary structure elements in all pairs of proteins, we run the MCL algorithm on the new pairwise alignments in order to update the MSTA's residue correspondences (Fig. 2, Step 6).

Step 7: realignment of local regions

The final stage of the CBA algorithm seeks to identify and realign small regions of the multiple alignment that could not be resolved in any previous step (Fig. 2, Step 7). Regions to be realigned are derived both from secondary structure elements with gaps and from areas containing MSTA columns established by the MCL algorithm that violate sequence order constraints or that contain more than one residue from the same protein. Each region contains one target

domain whose alignment is suspect, and each secondary structure element or loop may be marked for realignment more than once with different target domains.

CBA realigns the target domain's residues within a particular region by scoring the placement of a residue into a column according to the number of pairwise alignments that agree with the assignment. The assessment of structural similarity is therefore not based on an isolated and possibly misleading local perspective, but is instead derived solely from the global pairwise alignments obtained at the end of the sixth step of CBA. As was the case up to this point in the algorithm, sequence similarity is not considered. Dynamic programming then determines the new registration of the target domain in this region. This process thus uses the consistency among the pairwise alignments to correct local regions of the multiple alignment. After realigning all regions marked as potentially incorrect, we iterate until no further changes are made or for a maximum of 10 iterations.

RESULTS

Within the 391 superfamilies in SCOP version 1.67 containing at least three members in an ASTRAL subset of domains constructed such that no two share >25% sequence identity, LOCK 2's pairwise residue registrations are on average 68.2% consistent. This number is calculated from every triple of aligned residues from each triple of structures within the same superfamily. This high level of consistency among pairwise alignments often allows CBA to align distantly related protein structures. The average consistency within a superfamily increases to 80.4% after obtaining a common frame of reference and recalculating the residue registration in the implied pairwise alignments in Steps 2 and 3. After the realignment of secondary structure elements in Step 5, the average consistency reaches 84.2%. In the 351 families in SCOP containing at least three members in the 25% sequence identity ASTRAL subset, the consistency among the initial LOCK 2 pairwise alignments averages 76.7%. This value increases to 85.1 and 87.7% after Steps 3 and 5, respectively. These increases in consistency occur without a decrease in the number of aligned residues.

Because CBA observes sequence order constraints, we can read off a sequence alignment from the structural superposition. Figure 4a shows a portion of an alignment of the 17 structures in SCOP's globin-like superfamily that are included in the ASTRAL-25 subset; members of four families, including two phycocyanin domains, are included in the superposition. The full alignment (see Supplementary information) required 2.2 min. of CPU time (1.5 min. user time, 42 s system time) on a 3.06 Ghz Intel Xeon processor. Though gaps do occasionally appear in secondary structure elements, many are justified structurally or occur in regions where the correct residue alignment might be considered ambiguous. For example, the superposition of two of the structures, SCOP domains d1itha_ and d1b0b__, shows the insertion of a residue in SCOP domain d1itha_ (asparagine 17) that induces a gap in globin helix A (Fig. 4b). The alignment of d1itha_ and d1b0b__ shown in Figure 4b suggests that it is reasonable to leave either 16L or 17N of d1itha_ unaligned. In cases such as this, CBA tends to align the residue that is most consistently aligned in the pairwise alignments. Many other gaps in helices and strands are induced by insertions of loop residues that do not align to the secondary structure element.

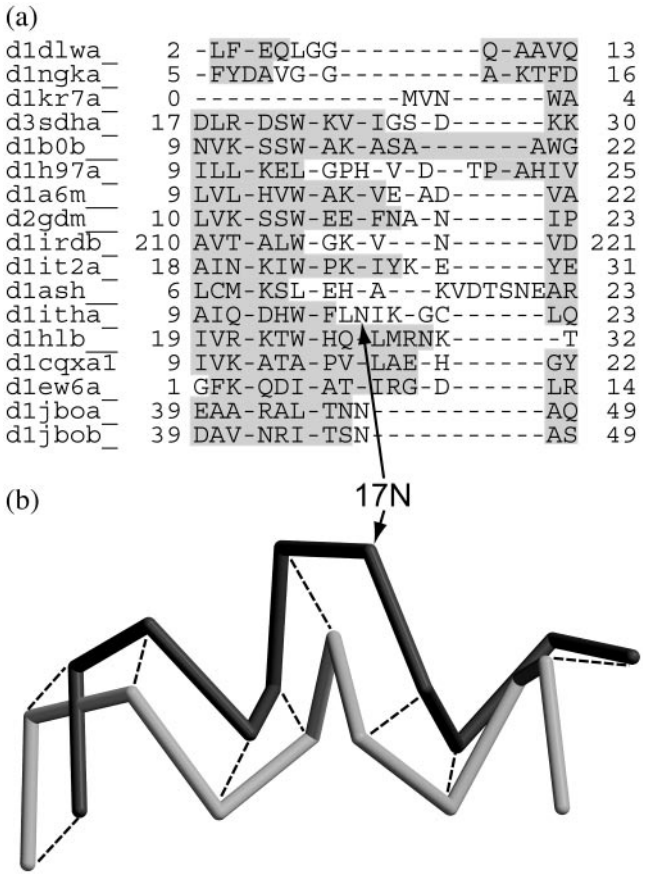

Fig. 4. Structurally reasonable gaps in a superposition of proteins from four globin-like families (**a**) A portion of the CBA alignment of the 17 members of SCOP's globin-like superfamily contained in an ASTRAL-40 subset is shown. The first two structures are protozoan/bacterial hemoglobins (whose A helices are truncated or deleted), the third is a nerve tissue mini-hemoglobin, the next 12 are globins from various species, and the last two are phycocyanins. Helical residues are shaded. The arrow indicates a residue (17N of d1itha_) that induces a gap in helix A. (**b**) The superposition of SCOP domains d1itha_ and d1b0b__, whose relative orientation with respect to d1itha_ is typical of other aligned domains, reveals that the gap induced by the insertion is structurally reasonable.

Validation by comparison to sequence motifs

Since manual analysis and validation of multiple structural alignments can be both prone to error and too time consuming to approach on a large scale, we sought to validate CBA by examining its alignments in regions containing conserved sequence motifs. Even when a set of proteins is so diverse that sequence alignment methods are not reliable enough to be used as a gold standard, a subset of the sequences may still contain the same local sequence motif. If the motifs themselves are reliable, then the alignment accuracy can be assessed in these regions by assuming that residues at the same position of the motif should align to one another. We use both PROSITE patterns (Sigrist *et al*., 2002), which are constructed manually to represent known functional regions, and eMOTIFs (Nevill-Manning *et al*., 1998), which are built automatically from high quality multiple alignments of proteins sharing strong

Table 1. Percent of motif residues correctly aligned by CBA

	Family validation Sets (%)	Superfamily validation Sets (%)	Reduced set of superfamilies[a] (%)
eMOTIFs	96.4	91.6	94.9
PROSITE patterns	97.4	92.6	94.2

[a]Results without considering the P-loop containing nucleoside triphosphate hydrolase superfamily.

sequence similarity. Since CBA never uses sequence information, this is an independent test of its accuracy.

In an ASTRAL subset of SCOP (version 1.65) created such that no two structures have >40% sequence identity, there are 478 families and 153 superfamilies in which at least two structures contain the same PROSITE pattern. Similarly, there are 123 families and 121 superfamilies in which at least two structures contain the same eMOTIF. All members of the SCOP family or superfamily represented in the ASTRAL-40 database are included regardless of whether they contain the sequence motif; this allows for a more difficult test by increasing both the number of structures to be superimposed and the overall structural variability. The average pairwise sequence identity as determined by BLAST within these validation sets is ∼17% for families and 14% for superfamilies.

For each pair of proteins in a validation set that contain a given sequence motif, we count the number of motif positions that CBA correctly aligned to produce a sum of pairs measurement of alignment accuracy. When a motif occurs in more than one location in one or both proteins, we consider only the pair of hits that is best aligned according to this criterion. Summing over all pairs of proteins with the motif yields a measure of the accuracy of the CBA alignments in these regions. CBA correctly aligns 96.4% of eMOTIF residues and 97.4% of PROSITE residues in the family validation sets (Table 1). Interestingly, only 92% of eMOTIF residues were correctly aligned in the LOCK 2 pairwise superpositions used by CBA. Hence, CBA was able to take advantage of the information contained in a set of related structures in order to increase alignment accuracy. In the significantly more challenging superfamily validation sets, CBA correctly aligns 91.6% of eMOTIF residues and 92.6% of PROSITE residues. When an alignment of 135 P-loop containing nucleoside triphosphate hydrolases is excluded from this analysis, these numbers increase to 94.9 and 94.2%, respectively. This procedure examines 17 922 aligned residue pairs in eMOTIF hits and 12 783 pairs in PROSITE patterns within the family validation sets, though there is a great deal of overlap between these two databases. The corresponding numbers for the superfamily validation sets are 18 485 and 14 647, respectively.

Comparison with CEMC, MultiProt and MASS

The use of sequence motifs as a gold standard for alignment accuracy provides an objective benchmark for evaluating MSTA algorithms. Since the accuracy of the alignment in regions containing sequence motifs was assessed only after algorithmic development had concluded, CBA was not overtrained to optimize the alignment of motifs and hence does not have an inherent advantage from this perspective over other algorithms.

Table 2. Comparison of motif alignment accuracies

	CBA Method (a) (%)	CEMC Method (a) (%)	MultiProt Method (a) (%)	MultiProt Method (b) (%)	MASS Method (a) (%)	MASS Method (b) (%)
eMOTIF families	96.2	93.4	75.1	87.7 (77.5)	65.1	80.2 (73.2)
eMOTIF superfamilies	94.8	87.7	47.3	84.9 (51.3)	N/A	61.9 (54.3)
PROSITE families	97.5	95.9	81.7	90.7 (72.9)	72.3	83.2 (73.1)
PROSITE superfamilies	95.4	89.4	55.3	87.8 (57.3)	N/A	60.1 (53.2)

Method (a): Only superpositions that align all domains are considered. A score of zero is given when an algorithm does not produce an alignment that superimposes all domains. Method (b): Superpositions that do not align all domains are permitted. The average percent of structures superimposed in the multiple alignments that achieve the greatest motif alignment accuracies is given in parentheses.

We benchmarked CEMC (Guda *et al*., 2004), MultiProt (Shatsky *et al*., 2004) and MASS (Dror *et al*., 2003) using the same sets of multiple alignment validation sets as reported above for CBA. A minor modification of the CEMC source code was necessary to prevent it from excluding structures that it considered to be too dissimilar to superimpose simultaneously. In some cases, CEMC or MASS did not produce a superposition at all. Because these failures may be more the result of technical issues than of algorithmic inadequacies, we removed these alignments from the validation sets.

Table 2 reports the percent of motif residues correctly aligned for all four multiple alignment algorithms using the sum of pairs score described above for assessing the alignment accuracy in regions containing sequence motifs. Both MASS and MultiProt report more than one possible multiple superposition for a given set of structures, so we count only the solution that yields the highest alignment accuracy with respect to the sequence motifs. This approach is rather lenient, as it does not require MASS and MultiProt to identify the best alignment before the motif alignment accuracy is assessed. Because MASS and MultiProt do not always produce a solution that superimposes all domains, we used two different methods to score their alignments. In the first method [Table 2, method (a)], we consider only alignments that include all domains, and record a score of zero for cases in which the algorithm did not produce a solution meeting this criterion. Since MASS frequently fails to align all structures in the superfamily validation sets, we evaluated it using this scoring method only for the family validation sets.

Though structures that do not contain the motif under consideration were deliberately added to the validation sets to increase their difficulty, we also computed scores for MASS and MultiProt without requiring them to superimpose all domains [Table 2, method (b)]. While this clearly gives MASS and MultiProt a distinct advantage over algorithms that superimpose all structures, they are still outperformed by CBA and CEMC. The average percentage of structures that were aligned in the best performing superposition is given in parentheses in the columns corresponding to method (b) in Table 2.

CBA reaches higher accuracies than the other three algorithms for both the family and superfamily validation sets regardless of the scoring method used, though CEMC does approach CBA's accuracy in the family validation tests. The most striking improvement over the previously reported MSTA methods, however, occurs in the superfamily validation sets. Even when MultiProt and MASS are permitted to exclude an arbitrary number of structures from the alignment [scoring method (b)], CBA's ability to align distantly related structures is clearly superior.

DISCUSSION

CBA provides a general framework for building multiple structure superpositions from pairwise alignments, and it is easily modified for sequence alignment problems by removing steps such as the realignment of individual secondary structure elements that may not be applicable in this case. We address the problem of resolving inconsistencies among pairwise alignments both by finding a common frame of reference for all of the structures and by globally and locally refining the multiple alignment's residue registration. CBA's output consists of the superposition of the proteins and their residue registration.

Though CBA uses LOCK 2 by default to perform pairwise alignments, any alignment algorithm may be substituted. The MCL clustering algorithm, which CBA uses several times to identify residue correspondences across multiple proteins, can similarly be replaced by another clustering algorithm, though we strongly caution against using progressive methods since they tend to propagate mistakes that occur in stages that consider only a subset of the proteins. Finally, the scoring system used in the last stage of CBA to realign small regions of the multiple-sequence alignment implied by the MSTA can be modified, though we have found in practice that more complex scoring functions based on structural measurements increase the algorithm's computational complexity without improving the resulting alignments (data not shown). CBA is designed to use no sequence information in producing a multiple alignment from pairwise alignments, but amino acid substitution scores would be an appropriate addition to the realignment scoring function in the case of MSA.

Assessment of multiple structure alignment's accuracy in regions containing conserved sequence motifs represents a new method for multiple alignment validation. Though it can be argued that comparison with hand-curated, full length sequence or structure alignments is a more thorough approach, restricting the comparison to regions containing conserved sequence motifs provides a more straightforward analysis since this process is less subject to error than is the comparison with global sequence or structural alignments of distantly related proteins. BaliBASE, a database of sequence alignments that are manually corrected using evidence from structural superpositions, recommends that comparisons be restricted to reliably aligned core blocks. Our approach is similar, but the use of sequence motifs is completely automated and does not rely on the ability to construct accurate, global alignments of diverse sequences or structures. Since BaliBASE alignments often contain sequences for which no structures exist, we did not perform a full-scale comparison with CBA.

Inclusion of structures in our validation sets that do not contain the sequence motif under consideration allows for tests of alignment accuracy at varying levels of difficulty with respect to sequence identity, structural similarity and the number of structures in the validation set. Though the alignment accuracy obviously cannot be assessed in structures that do not contain the motif, their presence makes the multiple alignment task more difficult. Despite the fact that we have constructed validation sets from an ASTRAL subset of SCOP domains in which structures may have up to 40% sequence identity, the average pairwise identity within a validation set is in practice <20%. Both the sequence similarity and the structural similarity of the proteins in the validation sets can easily be tuned for different applications.

CBA's highly accurate alignment of the sequence motifs contained within SCOP domains allows us to evaluate the accuracy of the sequence motifs themselves. We find, for example, that a number of misaligned or unaligned eMOTIF residues occur at or near the ends of the motifs, suggesting that the automated procedure used to generate them may occasionally extend a motif beyond its true boundaries with respect to its functional or structural role. Pruning the motifs in regions of poor structural conservation may increase the sensitivity of eMOTIFs, or of motifs from any other source, without decreasing their specificity.

Since PROSITE patterns are constructed manually, we observe misaligned residues at the ends of motifs more rarely than was the case for eMOTIFs. Instead, we have found instances in which a member of a validation set that only partially matched the PROSITE pattern nonetheless superimposed well in the region of the motif. Here, the PROSITE pattern may be too restrictive. Given a family of protein structures, several of which contain the same PROSITE pattern, we can again increase the sensitivity of the pattern without decreasing its specificity by expanding the set of allowed amino acids at one or more positions in order to accommodate the members of the family that superimpose well within the motif region and nearly match the original sequence pattern. Structural information has been added to PROSITE patterns in the past in order to compensate for 'softened' substitution groups that are less restrictive than in the original pattern, but previous methods required a structural comparison between the PROSITE pattern and the target protein and hence were only useful for comparison to proteins of known structure (Jonassen *et al*., 2000).

For example, only two of the three structures in our annexin validation set match the annexin PROSITE pattern, but the third structure (chain A of the first domain of human annexin I) matches the sequence pattern at all but the last of its 53 positions. Its sequence contains a tryptophan at this unmatched position rather than one of the several hydrophobic amino acids specified by the PROSITE pattern. Since CBA accurately aligned this position to the other two structures in the validation set, we created a modified annexin pattern by adding a tryptophan to the last position's substitution group. This new pattern picks up two additional matches in SCOP, both of which occur in a plant annexin (1dk5) and have a tryptophan residue in the last position, without encountering any additional false positives. The use of structural information compensates for the risk that increasing the number of amino acids permitted at a given position will decrease the PROSITE pattern's specificity.

Comparison of CBA with CEMC, MultiProt and MASS indicates that CBA represents a substantial improvement over these previously reported algorithms, particularly in the case of more distantly related structures. CEMC's results in the case of proteins belonging to the same family lag behind CBA only by a relatively

small margin, but CBA clearly outperforms all three algorithms in the superfamily validation sets.

ACKNOWLEDGEMENTS

This work was supported by NIGMS training grant number GM63495.

Conflict of Interest: none declared.

REFERENCES

Akutsu,T. and Sim,K.L. (1999) Protein threading based on multiple protein structure alignment. *Genome Inform. Ser. Workshop Genome Inform.*, **10**, 3–12.

Altschul,S.F. *et al*. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Brenner,S.E. *et al*. (2000) The astral compendium for protein structure and sequence analysis. *Nucleic Acids Res.*, **28**, 254–256.

Bystroff,C. and Shao,Y. (2002) Fully automated *ab initio* protein structure prediction using i-sites, hmmstr and rosetta. *Bioinformatics*, **18** (Suppl 1), S54–S61.

Do,C. *et al*. (2005) Probcons: probabilistic consistency-based multiple sequence alignment. *Genome Res.*, **15**, 330–340.

Doolittle,R. (1986) *Of urfs and orfs: A Primer on How to Analyze Derived Amino Acid Sequences*. University Science Books, Mill Valley, California.

Dror,O. *et al*. (2003) Mass: multiple structural alignment by secondary structures. *Bioinformatics*, **19** (Suppl 1), i95–i104.

Gerstein,M. and Levitt,M. (1996) Using iterative dynamic programming to obtain accurate pairwise and multiple alignments of protein structures. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **4**, 59–67.

Gotoh,O. (1990) Consistency of optimal sequence alignments. *Bull. Math. Biol.*, **52**, 509–525.

Guda,C. *et al*. (2004) Ce-mc: A multiple protein structure alignment server. *Nucleic Acids Res.*, **32**, W100–W103.

Jonassen,I. *et al*. (2000) Searching the protein structure databank with weak sequence patterns and structural constraints. *J. Mol. Biol.*, **304**, 599–619.

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

Madej,T. *et al*. (1995) Threading a database of protein cores. *Proteins*, **23**, 356–369.

Marchler-Bauer,A. *et al*. (2002) Comparison of sequence and structure alignments for protein domains. *Proteins*, **48**, 439–446.

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

Murzin,A.G. *et al*. (1995) Scop: A structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Nevill-Manning,C.G. *et al*. (1998) Highly specific protein sequence motifs for genome analysis. *Proc. Natl Acad. Sci. USA*, **95**, 5865–5871.

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

Panchenko,A. *et al.* (1999) Threading with explicit models for evolutionary conservation of structure and sequence. *Proteins*, **37**, 133–140.

Raghava,G.P. *et al*. (2003) Oxbench: A benchmark for evaluation of protein multiple sequence alignment accuracy. *BMC Bioinformatics*, **4**, 47.

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.

Sali,A. and Blundell,T.L. (1990) Definition of general topological equivalence in protein structures. A procedure involving comparison of properties and relationships through simulated annealing and dynamic programming. *J. Mol. Biol.*, **212**, 403–428.

Sandelin,E. (2005) Extracting multiple structural alignments from pairwise alignments: A comparison of a rigorous and a heuristic approach. *Bioinformatics*, **21**, 1002–1009.

Shapiro,J. and Brutlag,D. (2004) Foldminer: Structural motif discovery using an improved superposition algorithm. *Protein Sci.*, **13**, 278–294.

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

Sigrist,C.J. *et al*. (2002) Prosite: A documented database using patterns and profiles as motif descriptors. *Brief Bioinform*, **3**, 265–274.

Singh,A.P. and Brutlag,D.L. (1997) Hierarchical protein structure superposition using both secondary structure and atomic representations. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **5**, 284–293.

Taylor,W.R. *et al*. (1994) Multiple protein structure alignment. *Protein Sci.*, **3**, 1858–1870.

Thompson,J.D. *et al*. (1999) Balibase: A benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics*, **15**, 87–88.

Thornton,J.M. *et al*. (2000) From structure to function: Approaches and limitations. *Nat. Struct. Biol.*, **7**, 991–994.

Todd,A.E. *et al*. (2001) Evolution of function in protein superfamilies, from a structural perspective. *J. Mol. Biol.*, **307**, 1113–1143.

Van Dongen,S. (2000) *Graph Clustering by Flow Simulation*. University of Utrecht, Amsterdam.

Van Walle,I. *et al*. (2003) Consistency matrices: quantified structure alignments for sets of related proteins. *Proteins*, **51**, 1–9.

Yang,A.S. and Honig,B. (2000) An integrated approach to the analysis and modeling of protein sequences and structures. Iii. A comparative study of sequence conservation in protein structural families using multiple structural alignments. *J. Mol. Biol.*, **301**, 691–711.