PROTEINS: Structure, Function, and Genetics 48:242–256 (2002)

Flexible Protein Alignment and Hinge Detection

Maxim Shatsky,[1] Ruth Nussinov,[2,3,†] and Haim J. Wolfson[1]
[1]*School of Computer Science, Beverly and Raymond Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel*
[2]*Intramural Research Support Program—SAIC, Laboratory of Experimental and Computational Biology, NCI—Frederick, National Cancer Institute, Frederick, Maryland*
[3]*Department of Human Genetics and Molecular Medicine, Sackler Faculty of Medicine, Tel Aviv University, Tel Aviv, Israel*

ABSTRACT **Here we present a novel technique for the alignment of flexible proteins. The method does not require an a priori knowledge of the flexible hinge regions. The FlexProt algorithm simultaneously detects the hinge regions and aligns the rigid subparts of the molecules. Our technique is not sensitive to insertions and deletions. Numerous methods have been developed to solve rigid structural comparisons. Unlike FlexProt, all previously developed methods designed to solve the protein flexible alignment require an a priori knowledge of the hinge regions. The FlexProt method is based on 3-D pattern-matching algorithms combined with graph theoretic techniques. The algorithm is highly efficient. For example, it performs a structural comparison of a pair of proteins with 300 amino acids in about 7 s on a 400-MHz desktop PC. We provide experimental results obtained with this algorithm. First, we flexibly align pairs of proteins taken from the database of motions. These are extended by taking additional proteins from the same SCOP family. Next, we present some of the results obtained from exhaustive all-against-all flexible structural comparisons of 1329 SCOP family representatives. Our results include relatively high-scoring flexible structural alignments between the C-terminal merozoite surface protein vs. tissue factor; class II aminoacyl-tRNA synthase, histocompatibility antigen vs. neonatal FC receptor; tyrosine-protein kinase C-SRC vs. haematopoetic cell kinase (HCK); tyrosine-protein kinase C-SRC vs. titine protein (autoinhibited serine kinase domain); and tissue factor vs. hormone-binding protein. These are illustrated and discussed, showing the capabilities of this structural alignment algorithm, which allows un-predefined hinge-based motions. Proteins 2002; 48:242–256.** © 2002 Wiley-Liss, Inc.*

Key words: structural comparison; hinge bending; flexible structural comparison; efficient algorithm; protein structural comparison

INTRODUCTION

The importance of efficient protein structural comparison tools can hardly be overstated. This is evident from the current existence of a relatively large number of algorithms designed to carry out such a task. Structural comparisons are essential for protein classification, detection of conserved protein folding cores, detection of similarities in functional binding sites, similarities in enzyme mechanisms, evolutionary conservation, construction of nonredundant databases, detection of similarities between domains, and identification of conserved residues. They are further used in fold recognition and homology modeling. Programs for comparisons of protein structures are routinely run, existing in numerous packages.

Despite the relatively large number of structural comparison algorithms, the majority perform *pair-wise rigid structural comparison* (e.g., Refs. 1–13). These have been extensively reviewed (for an excellent recent review, see Eidhammer et al.[14]) and hence will not be reviewed here. In addition, there are few *multiple structure comparison* algorithms. Here, the methods can be roughly classified into two main categories. The first consists of algorithms that accept as input a multiple alignment (derived from either sequence alignment, secondary structure alignment, etc.) and output a refined alignment (e.g., Gerstein and Altmann[15] and Gelfand et al.[16]). The second category includes algorithms that tackle the multiple structure alignment itself. Here, one algorithm[17] assumes that the solution is composed of the solutions to the pair-wise alignment task, extending the pair-wise task by suggesting ways to put the pair-wise solutions into a multiple solution. Pair-wise-based approaches also include the double dynamic programming algorithm of Orengo and Taylor,[2,18,19] which picks a seed pair alignment, and the iterative dynamic programming algorithm of Gerstein and Levitt,[20] which picks a median structure, closest to all other structures in the least-squares sense. All structures are aligned to the median. Recently, we developed a *multiple structure alignment* method that initiates from the atomic coordinates and simultaneously solves the multiple structure alignment and the detection of the conserved common core.[21,22] This algorithm is efficient and independent of the amino acid sequence order.

While the algorithms cited above treat proteins as rigid bodies, and carry out rigid structural comparisons, proteins are flexible molecules. Around their native state,

†Correspondence to: R. Nussinov, NCI-Frederick, Building 469, Room 151, Frederick, MD 21702. E-mail: ruthn@ncifcrf.gov

Received 12 April 2001; Accepted 7 January 2002

there is a range of conformational isomers, with low-energy barriers separating them. Consequently, proteins flip between conformational substates. Depending on the conditions, for example, binding to different ligands or changes in the sequence, the population times of given conformers may change. Movements may reflect small, largely side-chain motions, particularly those on the surface of the protein. Alternatively, they may involve large-scale motions, reflecting hinge-bending movements of the backbone.[23,24] In hinge bending, parts of the molecule rotate with respect to each other as relatively rigid bodies, on a common hinge.

Hence, two proteins that may have similar structures might nevertheless appear different if one is hinge-bent with respect to the other. Owing to the differences in the partner molecule in the crystal, to different ligands (whether sister molecules or other), and to differences in the sequences or in the physical conditions, such a situation is frequently encountered. Similar topologies are likely to have hinge regions at analogous sites. However, the extent of the motions differs. These are reflected in the distribution of the populations and hence in the energy landscape. Rigid body structural comparisons may therefore encounter difficulties.

Few algorithms have been designed to compare flexible molecules. The first is the method of Wriggers and Schulten.[25] This method solves the structural comparison problem allowing protein domain movements. It assumes that the residue alignment task is already solved by a different technique. This may happen, for example, when we are given two structures of the same protein (e.g., in the complexed and free states) or when the residue correspondence is obtained from sequence alignment. The structures are partitioned into disjoint subsets (domains) that can be aligned. Starting from an initial subset (every localized seed of C_α-atoms is considered) and iteratively applying a least-squares procedure,[26] the method tries to enlarge the matching list from the previous iteration by adding a closely located new atom pair, one atom from each structure. The newly obtained 3-D rigid transformation is applied to all remaining atoms. After all domains are detected, the effective rotation axes between the connected domains are calculated.

The second method is based on computer vision techniques.[27] Following the ideas of the geometric hashing[6–8] and the generalized Hough transform[28] algorithms, Verbitsky et al.[27] developed an algorithm for hinge-bending flexible structural comparison. If a hinge motion is suspected at some specific small region of the protein, the molecule can be divided into two rigid parts (domains) separated by the hinge, with the deformation assumed to be confined to their linking hinge region. Thus, the input to the algorithm is two molecules, with the first divided into two rigid parts sharing one common point. The proteins are represented as sets of their C_α-atoms. As common to algorithms based on the geometric hashing paradigm, the method has two steps, *preprocessing* and *recognition*. At the preprocessing step, from every rigid part of the first, flexible, molecule the local shapes are extracted and their shape signatures, which are invariant under rotation and translation, are calculated and stored in a lookup table (a local shape might be a triangle and its side lengths serve as a shape signature). The hinge position is stored relative to a reference frame defined by the local shape. At the recognition step, for every local shape from the second, rigid molecule, its shape signature is calculated. Similar local shapes from the first molecule are extracted from the lookup table. For every such shape a new potential hinge position is calculated and recorded. This position is obtained by applying the transformation to the local shape from the second molecule. This transforms the local shape from the first molecule to the reference frame defined at the hinge location. High-scoring hinge locations are those for which a large enough number of local shapes from the second molecule aligned with the first molecule and placed the hinge at the same position in 3-D space. These solutions are explored to detect pairs of hinge-consistent transformations, which induce nonconflicting alignments. The alignment is nonconflicting when two rigid parts of the first molecule do not overlap. Further details are given in Verbitsky et al.[27] This method was also applied to docking, assuming prior knowledge of the location of potential hinges, either in the ligand or in the receptor.[29,30] Methods originally developed for alignment or docking of flexible molecules[31,32] also require an a priori knowledge of the location of the hinge.

Here we present an algorithm—**FlexProt**—for the alignment of a rigid protein molecule with a flexible one. The method does not require any a priori knowledge regarding the location of the hinge regions in the flexible molecule. The algorithm automatically detects these hinge regions. The input consists of the rigid molecule and some conformation of the flexible protein molecule. The method outputs a list of the best alignments divided according to the number of hinge regions and ranked by the size of the overall alignment. Each flexible alignment contains information regarding the size of the individual aligned rigid domain parts, the root mean square deviation (RMSD) of the alignment, the hinge region positions, the bend and twist angles between any two covalently connected hinge-bent parts, and the extent of interdomain opening, as measured by the distance between the domains in the open conformation compared to the closed. For each candidate solution, this algorithm discovers simultaneously both matching rigid fragments and the connecting hinge regions. The FlexProt algorithm accomplishes a considerably more complex task than the previous flexible alignment methods, where the partitioning of the flexible molecule into rigid parts was required in advance. However, to accomplish this task efficiently it has to exploit the amino acid sequence order and is not sequence order independent as in Verbitsky et al.[27]

Despite the fact that FlexProt automatically detects the hinge regions, the algorithm is as efficient as rigid structure alignment algorithms. Typical run times on proteins with few hundreds of amino acids consisting of several rigid parts are about 7 s. Thus, FlexProt is efficient enough to conduct an all-against-all structural alignment of the Protein Data Bank (PDB[33]).

A preliminary version of this algorithm has been presented recently in the proceedings of the ISMB.[34] The algorithm has been implemented in C++. We conducted numerous experiments on a standard desktop PC (400 MHz, 256 Mb RAM). Below we outline the algorithm and present some of the results we obtained. The first set of results illustrates known domain movements. The second set presents selected results obtained in the exhaustive pair-wise comparisons, demonstrating the power of this method. For comparison we also present some results of random alignments.

ALGORITHM

The goal of the FlexProt flexible structural matching algorithm is to divide the two protein molecules into a minimal number of separate fragments of maximal size, such that the fragments that are matched will be almost congruent. Two fragments are congruent if they have the same number of C_α-atoms and there exists a 3-D rotation and translation that superimposes the corresponding atoms with a small RMSD. The arrangement of the matching fragments should be consistent with their order on the protein chain. Between these are the flexible (hinge) regions. A trivial way to achieve a flexible alignment of maximal size is to allow flexibility between each pair of neighboring C_α-atoms, thus aligning completely the two molecules if they are of the same length. However, our goal is to minimize the number of flexible regions or rigid fragments. A flexible alignment with few hinge regions is more meaningful than one with many. Clearly, these two goals—maximal matching size and minimal number of flexible regions—are conflicting, necessitating a balanced solution. Actually, FlexProt solves the more complex *partial alignment* problem, where only large enough flexible substructures are acceptable in the alignment. Owing to the approximate nature of the matching, and the trade-off between matching accuracy and matching fragments size, our goal is to output a set of (sorted) candidate solutions according to user-imposed accuracy thresholds.

The input to the algorithm are two protein molecules M_1 and M_2, each being represented by the sequence of the 3-D coordinates of its C_α-atoms, $M_1 = v_1,\ldots, v_n$ and $M_2 = w_1,\ldots, w_m$. Assume that (during evolution) molecule M_1 has undergone hinge-bending movements at several locations along its backbone. Further assume that between the flexible joints there are fragments without a significant structural change, although their sequences may differ. The resulting hinge-bent molecule is denoted M_2. Under our assumptions there exists a set of rigid fragments of M_2 that are congruent to the corresponding set of fragments of M_1. The model presented applies not only to different conformations of a given molecule, but to the general case of flexible motif detection between two molecules with different sequences.

In the first step the algorithm detects candidate "big enough" congruent rigid fragments, satisfying the stipulated criteria. We look for equal-size fragment pairs (the minimum fragment length is controlled by a program parameter *MinFragSize* with default value of 12 residues), one from each molecule, such that there exists a 3-D rotation and translation of one of the fragments superimposing it on the other with a small RMSD (e.g., 1.5–3.5 Å). Two tasks are involved in the fragment pair detection step: The first is the *correspondence*, that is, detection of candidate (almost) congruent fragment pairs, which can be superimposed with a small RMSD. Let us call the corresponding atom pairs list a "match list." The correspondence is solved by initiating the alignment with every atom pair (one atom from each molecule) and extending the alignment to the left and right, along the backbone chain, until the RMSD of the best superposition of the two fragments gets larger than a predefined threshold (*MaxRMSD* parameter, whose default is 3 Å in our program). The second task is the *superposition*: Given the corresponding C_α-atom sets (match list) find the rotation and translation which superimposes these sets with a minimal RMSD. It can be shown that both the correspondence and superposition tasks can be done in time, which is linear in the detected fragment size using calculations based on the Schwartz and Sharir[35] algorithm for rigid structure alignment (for mathematical details see the appendix of Shatsky et al.[34]).

Once such corresponding fragments pairs are detected, we seek an optimal subset of these matching pairs, which flexibly aligns M_2 to M_1. This is accomplished by the second step of the algorithm.

In the second step, the rigid fragment pairs are linked together, consistent with the amino acid sequence order. One way to perform such a task is via enumeration over all subsets of the matched fragment pairs, choosing the best scoring ones. This, however, is extremely inefficient. The complexity might grow exponentially with the size of the input. Our solution is based on the efficient *single-source shortest paths* algorithm in a directed weighted acyclic graph (DAG).[36] This solution reduces to detection of shortest paths from a single source in the directed weighted graph, where the vertices represent the congruent fragment pairs and the edges represent the hinge regions.

Once we have a set of congruent fragment pairs, we seek an optimal subset of it that describes a possible alignment of M_2 with M_1 allowing flexibility in M_2 between the fragments. Ideally, this alignment should include a sequence of disjoint congruent fragment pairs in ascending order of the fragment indices i and j from the two molecules. In practice, one should allow a certain overlap of consecutive fragments on the same chain. Our method is close in spirit to the one implemented in FASTA.[37]

Our procedure in this phase of our algorithm is as follows:

1. Represent congruent fragment pairs as vertices of a graph. Thus, a vertex u represents a rigid fragment of M_1 and an (equal length) similar shape fragment of M_2.
2. Join two vertices by a directed edge if the fragment pairs that they represent might be consecutive in the final alignment. This results in an acyclic directed graph.

TABLE I(a). Listing of the Proteins Whose Comparisons are Presented in this Work: Cases are Taken from the Database of Motions, Extended by Taking Additional Proteins from SCOP That Belong to the Same Family.

Protein pair	PDB code	Backbone length	SCOP family
Glutamine-Binding protein	1ggg (chain A)	220	Phosphate-binding protein-like
Glutamine-Binding Protein bound to glutamine	1wdn (chain A)	223	Phosphate-binding protein-like
Glutamine-binding protein	1ggg (chain A)	220	Phosphate-binding protein-like
Histidine-binding protein complexed with histidine	1hpb	238	Phosphate-binding protein-like
Calmodulin complexed with rabbit Skeletal myosin light-chain kinase	2bbm (chain A)	148	Calmodulin-like
Human calmodulin	1cll	144	Calmodulin-like
Calmodulin complexed with rabbit	2bbm (chain A)	148	Calmodulin-like
Skeletal myosin light-chain kinase Troponin C	1top	162	Calmodulin-like
Adenylate kinase isoenzyme-3	2ak3 (chain A)	226	Nucleotide and nucleoside kinases
Adenylate kinase complexed with the inhibitor AP = 5 = A	1ake(chain A)	214	Nucleotide and nucleoside kinases
Adenylate kinase isoenzyme-3	2ak3 (chain A)	226	Nucleotide and nucleoside kinases
UMP/CMP kinase	1uke	193	Nucleotide and nucleoside kinases
Immunoglobulin fab fragment	1mcp (chain L)	220	Immunoglobulin (superfamily)
Immunoglobulin fab fragment	4fab (chain L)	219	Immunoglobulin (superfamily)
Immunoglobulin fab fragment	1mcp (chain L)	220	Immunoglobulin (superfamily)
Murine T-cell antigen receptor	1tcr (chain B)	236	Immunoglobulin (superfamily)
Lactoferrin	1lfh	691	Transferrin
Transferrin	1lfg	691	Transferrin
Transferrin (N-terminal half-molecule)	1tfd	294	Transferrin
Lactoferrin	1lfh	691	Transferrin

The names of both proteins are given, their corresponding PDB file names, the SCOP families to which they belong, and their sizes.

3. Assign weights to the edges, rewarding long matching fragments and penalizing large interfragment gaps as well as large discrepancies in the relative number of gaps in both proteins. We define the weight of an edge $e = (u, v)$ as

$$w(e) = -((l_v + 1) - [\Delta])^2 + \max(|Gap_1|,|Gap_2|) + \|Gap_1| - |Gap_2\|,$$

where Δ is half of the maximal overlapping interval between the vertex fragments (if there is no overlap, Δ is zero). In this weight function we reward quadratically the length, l_v, of the fragments from the second vertex v and introduce a penalty for large gaps (Gap_1/Gap_2 stands for a length of the gap between the fragments of the first/second molecule) between fragments of u and v. The third factor gives priority to edges with a smaller (absolute value) difference between the gaps Gap_1 and Gap_2. Different scoring functions were tried; the presented one gave the most qualified results. The weight is independent of the fragment length of the first vertex.
4. Add a virtual node connecting every other node in the graph with a zero weight edge.
5. Apply the single-source shortest paths algorithm (starting from the virtual node) to the weighted graph.
6. Collect the paths found by the single-source shortest paths algorithm according to the number of vertices in each path, sort candidate solutions according to the total size and RMSD of the fragment alignment, and output the best-scoring potential solutions. Each such solution represents a sequence of consecutive congruent fragment pairs.

In the third step, the algorithm clusters consecutive fragment pairs that have a similar 3-D transformation, even if they are not directly linked. A good example is that of a β-sheet. Assume that M_1 and M_2 have a structurally similar β-sheet. Even so, it is likely that the turn regions connecting the β-strands will not be congruent in M_1 and M_2. The FlexProt algorithm will detect the similarity between the β-strand pairs of the different molecules, yet it might view them as disjoint congruent pairs due to the structural dissimilarity of the turn regions. However, except for the turns, the β-sheet may match with the same 3-D transformation. Therefore, we would like to consider it as a single congruent region pair. Hence, we cluster consecutive congruent pairs, which share an almost similar transformation. Thus, the final *partial alignment* solution is represented by a number of congruent region pairs, each consisting of one or of several congruent fragment pairs.

The theoretical complexity of the algorithm is bounded by $O(n^4)$ (where n is defined as the size of the larger

TABLE I(b). Listing of the Proteins Whose Comparisons are Presented in This Work: Examples Obtained Through Exhaustive Comparisons†

Protein pair	PDB code	Backbone length	SCOP family
C-terminal merozoite surface protein	1b9w(chain A)	89	Merozoite surface protein 1 (MSP-1)
Blood Coagulation factor VIIA	1dan(chain L)	132	EGF-type module
E. coli threonyl-TRNA synthetase	1qf6(chain A)	641	Class II aminoacyl-tRNA synthetase (aaRS)-like/ anticodon-binding domain of Class II aaRS
Histidyl-TRNA synthetase	1adj(chain A)	420	Class II aminoacyl-tRNA synthetase(aaRS)-like/ anticodon-binding domain of Class II aaRS
Histocompatibility antigen	2clr(chain A)	275	MHC antigen-recognition domain/C1 set domains (antibody constant domain-like)
Neonatal FC receptor	3fru(chain A)	269	MHC antigen-recognition domain/C1 set domains (antibody constant domain-like)
Human tyrosine-protein kinase C-SRC	1fmk	437	SH3-SH2-KINASE domains
Haematopoetic cell kinase (HCK)	1qcf(chain A)	449	SH3-SH2-KINASE domains
Human tyrosine-protein kinase C-SRC	1fmk	437	Tyrosine kinase
Titine protein	1tki	321	Serine/threonine kinases
Tissue factor (TF)	1a21(chain A)	194	Fibronectin type III
Growth hormone-binding protein	1hwg(chain C)	191	Fibronectin type III

†Family representatives are taken from each family in SCOP and an all-against-all flexible structural comparison is carried out. The names of both proteins are given, their corresponding PDB file names, the SCOP families to which they belong, and their sizes.

molecule); however, in practice its performance is much faster. Comparison of a pair of flexible structures of about 300 amino acids each takes approximately 7 s on a standard desktop PC (400 MHz, 256 Mb RAM). A more detailed description of the method is given in Shatsky.[38]

RESULTS

Here we present three sets of results. In the first, we applied FlexProt to some of the cases that have been usefully collected and catalogued in Gerstein's database of motions.[23,24] This database contains proteins known to undergo hinge-bending motions. The database contains three classes of cases, involving fragment, domain, and subunit movements. In each category, for each case, the molecule pairs, triplets, or more are essentially identical; however, they have been crystallized (or solved by NMR) in different conformations. In running FlexProt on these examples, we assumed no a priori knowledge of the hinge positions. The only input is the C_α-atom coordinates. However, because, in these examples the residue correspondence problem is trivial (each residue corresponds to itself in the aligned protein), they do not demonstrate adequately the full power of our algorithm. The power of FlexProt is in its ability to solve simultaneously both the residue correspondence problem and hinge region detection in proteins. Therefore, we supplement these examples by more difficult ones where the aligned proteins are not the same. This is achieved by replacing one of the original proteins with a different one from the same SCOP family.[39] In the second set of results, we present a more demanding task. Here we give some of the hinge-bending cases uncovered through exhaustive PDB database comparisons. For these cases, our procedure is to take one protein from each of the SCOP families and exhaustively compare it against all other representatives (here the first protein in the SCOP family list). A total of 1329 family representatives were compared in this analysis. Table I(a) lists the proteins used in the first set of results and Table II(a) enumerates the output obtained by FlexProt for these cases. Tables I(b) and II(b) list selected results from the exhaustive all-against-all runs. The third set of results contains examples of random comparisons of unrelated proteins. This was done to illustrate the performance of the algorithm on random pairs of proteins, both belonging to the same SCOP class and to different classes.

We assess the relative movements of the rigid fragments (i.e., the domains), separated by the flexible regions through the bending and twisting angles, and the domain distances, following Maiorov and Abagyan.[40] The bending angle between two domains is defined by three points: the centroid of the first domain, the centroid of the flexible region (i.e., the region between the two domains), and the centroid of the second domain. The twist angle is defined by four points: the centroid of the first domain, the boundary of the first domain with the flexible region (the boundary point is defined as the coordinates of the C_α-atom belonging to the residue that connects the domain with the flexible region), the boundary of the second domain with the flexible region, and the centroid of the second domain. The distance between two domains is defined by the distance between the domain centroids. The distance between the domain boundaries with the flexible region (connecting these two domains) is also computed by FlexProt. The angles and distances are illustrated in Figure 1 and tabulated in Tables II(a) and (b).

TABLE II(a). Results Obtained by FlexProt for the Proteins listed in Table I: Cases Taken from the Database of Motions, Extended by Taking Additional Proteins from SCOP That Belong to the Same Family (Corresponding Figures are 2–5)†

Protein pair	Backbone length	Number of flexible regions	Match list size	Matched rigid fragments	Total RMSD (Å)
1wdn(A)	223	2	218	(5–87)− −(88–180)− −(181–222)	
1ggg(A)	220			(5–87)−{−34.4,−24.7,−5.8}−(88–180)−{−8.6,8.6,−0.9}−(181–222)	0.94
1hpb	238	2	220	(7–91)− −**[92–184]**− −(190–234)	
1ggg(A)	220			(5–89)−{−28.5,13.0,−6.1}−**[90–179]**−{−28.9,−48.4,−0.5}−(180–224)	2.34
2bbm(A)	148	1	139	(4–77)− −(78–142)	2.22
1cll	144			(4–77)−{−84.1,−34.6,−14.8}−(78–142)	
2bbm(A)	148	3	147	(2–25)− **h−1** −(26–59)− **h−2** −(60–76)− **h−3** −(77–148)	2.40
1top	162			(12–35)−{46.5,52.3,5.4}−(36–69)−{40.8,41.3,7.3}−(70–86)−{−98.6,20.9,−13.5}−(90–161)	
1ake(A)	214	2	200	(1–115)− −(118–157)− −**[158–207]**	
2ak3(A)	226			(6–120)−{−45.7,85.6,−11.5}−(121–160)−{−25.0,45.1,−5.2}−**[161–205]**	2.44
2ak3(A)	226	2	182	(1–109)− **h−4** −(113–127)− **h−5** −(159–216)	2.9
1uke	193			(2–110)−{14.2,−18.2,0.5}−(120–134)−{−25.7,−10.9,−1.9}−(137–194)	
1mcp (L)	220	1	218	(2–110)− −(111–219)	1.93
4fab (L)	219			(1–109)−{16.5,−91.4,2.6}−(110–218)	
1mcp (L)	220	1	212	**[1–112]**− **h−6** −**[114–220]**	2.36
1tcr (B)	236			**[1–116]−{34.9,−16.9,8.0}−[121–246]**	
1lfh	691	2	691	(1–84)− −(85–244)− −(245–691)	1.41
1lfg	691			(1–84)−{16.1,29.8,7.1}−(85–244)−{37.7,18.9,5.0}−(245–691)	
1tfd	294	2	291	**[6–89]**− **h−7** −**[90–244]**− **h−8** −**[245–304]**	1.98
1lfh	691			**[6–86]**−{17.3,48.5,−6.7}−**[87–248]**−{−29.5,−33.0,−6.3}−**[249–309]**	

†The table lists the PDB file names, the backbone lengths, the number of flexible regions between the matched fragments, the match list size (*i.e.*, the number of corresponding C_α pairs), the matched rigid fragments, and the actual matched fragments. The difference between the distances prior and following the domain movements are listed inside parentheses between the fragments showing these hinge-bending motions. Flexible region identifier, **h-i**, represents the marked region in the corresponding figure.

TABLE II(b). Results Obtained by FlexProt for the Proteins Listed in Table I: Examples Obtained Through Exhaustive Comparisons (Corresponding Figures are 6–11)†

Protein pair	Back-bone length	Number of flexible regions	Match list size	Matched rigid fragments	Total RMSD (Å)
1b9w(A)	89	1	75	(8–36)− **h−10** −**[42–89]**	2.78
1dan(L)	132			(50–78)−{−91.4,35.9,17.2}−**[84–129]**	
1qf6(A)	641	1	323	**[256–494]**− **h−11** −(532–633)	4.43
1adj(A)	420			**[2–296]**−{−47.9,−20.3,−10.3}−(320–421)	
2clr(A)	275	2	253	**[1–86]**− **h−12** −**[92–168]**− **h−13** −**[169–261]**	2.71
3fru(A)	269			**[3–87]−{4.5,−2.8,1.9}−[88–165]−{−2.9,31.2,−3.0}−[166–256]**	
1fmk	437	2	424	**[82–138]**− **h−14** −(139–251)− **h−15** −**[253–521]**	1.25
1qcf(A)	449			**[82–138]**−{−0.6,0.9,−1.5}−(139–251)−{8.3,15.9,0.0}−**[252–521]**	
1fmk	437	2	231	(266–327)− **h−16** −(328–424)− **h−17** −**[431–516]**	3.28
1tki	321			(24–85)−{−9.8,12.0,−1.1}−(86–168)−{6.9,56.8,−2.5}−**[188–276]**	
1a21(A)	194	4	163	(5–25)− **h−18** −**[31–105]**− **h−19** −(106–126)− **h−20**	2.75
1hwg(C)	191			(32–52)−{4.8,13.8,1.4}−**[63–130]**−{18.1,0.7,−5.9}−(135–155)−{−77.3,14.0,−11.6}− −(140–157)− **h−21** −(163–208) −(162–179)−{−10.1,−11.3,−3.7}−(187–232)	

†The table lists the PDB file names, the backbone lengths, the number of flexible regions between the matched fragments, the match list size (*i.e.*, the number of corresponding C_α pairs), the matched rigid fragments, and the actual matched fragments. The difference between the distances prior and following the domain movements are listed inside parentheses between the fragments showing these hinge-bending motions. Flexible region identifier, **h-i**, represents the marked region in the corresponding figure.

The only thresholds used in the comparisons is that two rigid fragments can be matched if their length is above *MinFragSize* (default is 12 residues) and the RMSD of the alignment is less than *MaxRMSD* value (default is 3 Å). There are no limitations on angles or distances between the rigid "domains." In some cases the default values were changed to obtain a better alignment.

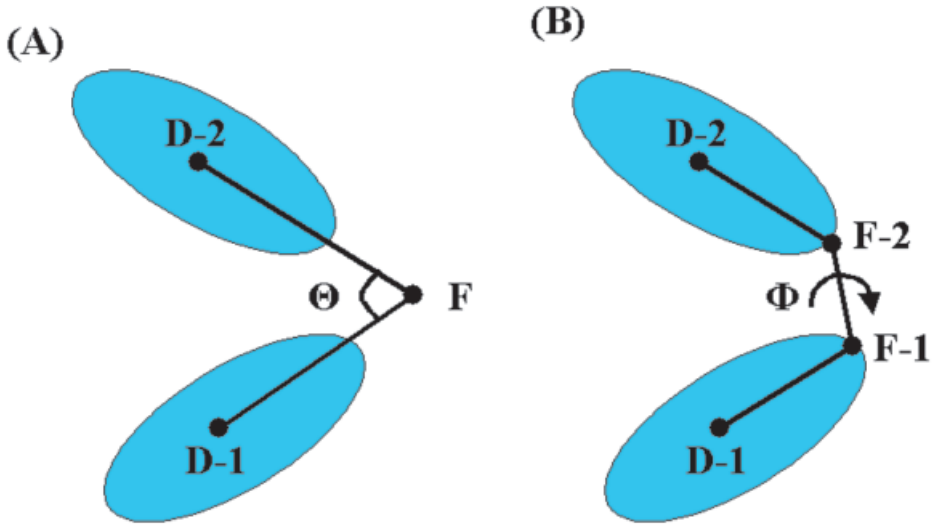

Fig. 1. Description of the relative domain positions.[40] Points **D-1** and **D-2** are the domain centroids. (**A**). Point **F** is the centroid of the interdomain region (flexible region). The angle is defined as the bending angle. (**B**). Points **F-1** and **F-2** are the domain boundaries with the flexible region that connects them. The angle is defined as the domain twisting angle. The distance between the domains is defined by the two distances: |*D*-1, *D*-2| and |*F*-1, *F*-2|.

The figures of these experiments were prepared using InsightII software.[42] Each flexible region, depicted in the figure, has its unique identifier, marked by **h-i**, which can be found in Tables II(a) and (b) in the corresponding row. Consider an example from Table II(b):

Protein pair	Backbone length	Number of flexible regions	Match list size	Matched rigid fragments	Total RMSD
1qf6(A)	641	1	323	[**256–494**]− **h−11** −(532–633)	4.43
1adj(A)	420			[**2–296**]−{−47.9,−20.3,−10.3}−(320–421)	

This entry says that protein 1qf6(chain A) is aligned with 1adj(chain A). The length of 1qf6(A) is 641 amino acids and the length of 1adj(A) is 420 amino acids. The flexible alignment contains one flexible region (and thus two rigid parts). The total size of the flexible alignment is 323 amino acids. The next column displays the alignment itself. The fragments typed in bold and enclosed in square brackets represent a cluster of consecutive fragments having the same 3-D transformation. Thus, the cluster from residue 256 to residue 494 of the first protein is matched with the cluster **[2–296]** of the second molecule (notice that due to inner gaps between the fragments the length of the clusters might not be of equal size). This is the first rigid part. Then (on the second line) follow values of the bending and twisting angles, and the domain distance difference prior and following the domain movement. The flexible region is denoted in this case as **h-11** (first line). This is done to allow easy detection in the attached figure (if available). Then, the second pair of matched rigid fragments (532–633) and (320–421) of the first and second molecule appear correspondingly. The last column lists the total RMSD of the flexible alignment, 4.43 Å. This RMSD score is computed after aligning (transforming) all matched fragments and is based only on the aligned C_α-atoms. C_α-atoms that are not from the aligned fragments are not considered in this RMSD calculation.

Extended Comparisons of Known Domain Motions

Glutamine-binding protein

We compared the glutamine-binding protein (PDB: 1ggg, chain A) in the ligand-free form with the glutamine-bound complex (PDB: 1wdn, chain A), picked from the database of motions. Two hinge regions were detected, at residues 87–88 and 180–181. The RMSD of the total matching set is 0.94 Å. To extend the analysis, we compared 1ggg with a member of the same (phosphate-binding protein-like) family, as classified in SCOP, a histidine-binding protein complexed with histidine (1hpb). FlexProt detected four similar fragments, two with similar transformations although separated by a turn located at residues 132–135 of the 1hpb. These resulted in three matched clusters with total RMSD of 2.34 Å [Table II(a)].

Calmodulin

Calmodulin (CaM) is a C_α^{2+}-binding protein, involved in a wide range of cellular C_α^{2+}-dependent signaling pathways. Human calmodulin (1cll) and calmodulin complexed with rabbit skeletal myosin light-chain kinase (2bbm,

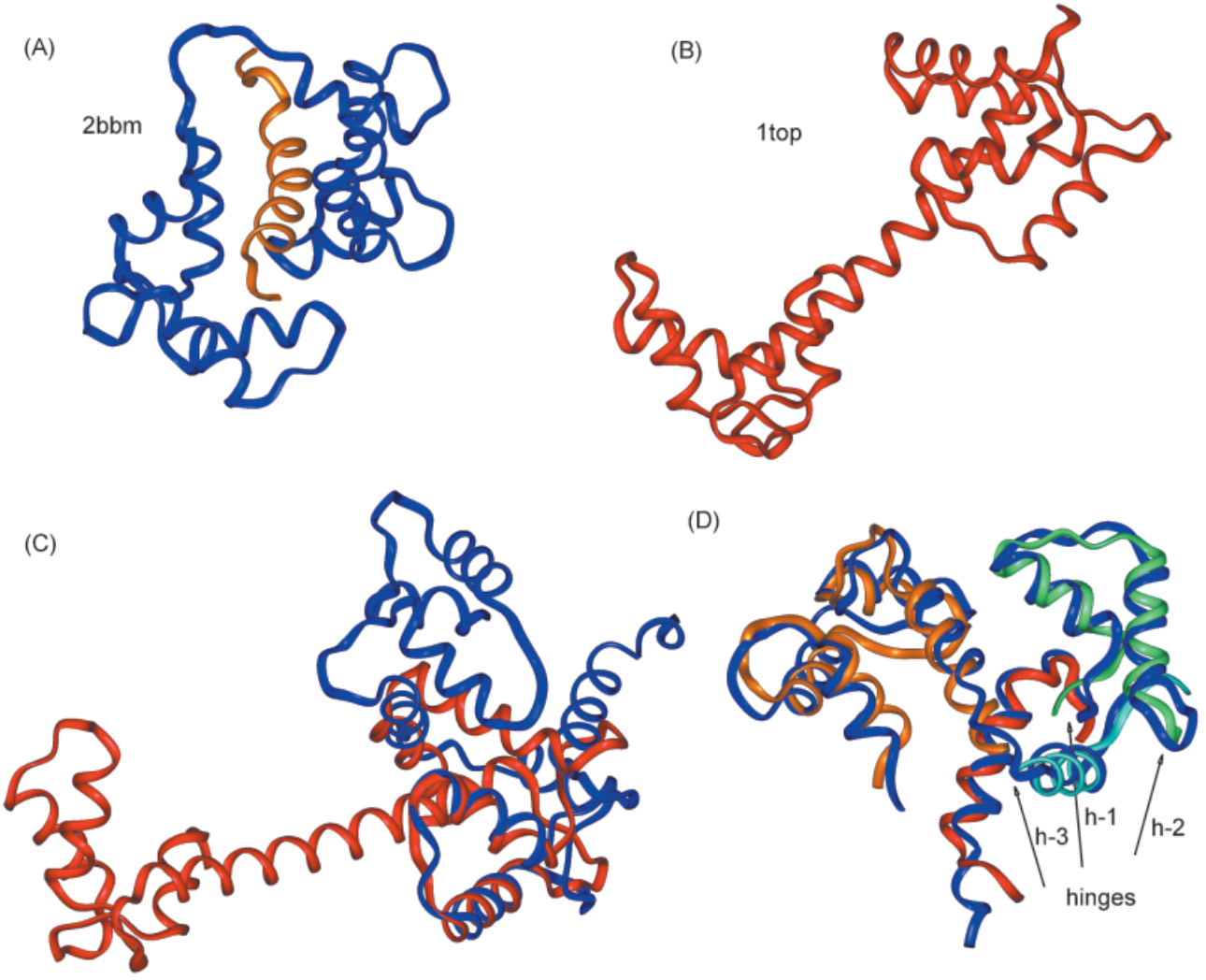

Fig. 2. Results obtained from the application of FlexProt automated structural comparison algorithm to known cases of hinge-bent molecules. No knowledge of the hinge-bending sites is assumed. The examples were taken from the Database of Motions, supplemented by an application of FlexProt to another protein from the same family, as classified in SCOP. The cases are enumerated in Table I(a) and the results tabulated in Table II(a). Calmodulin (PDB: 2bbm, chain A) compared with Troponin C protein (1top). (**A**). Calmodulin complexed with rabbit skeletal myosin light-chain kinase (2bbm). (**B**). Troponin C protein (1top). (**C**). Rigid structural alignment of the two molecules (2bbm; blue). (**D**). Superposition of 1top on 2bbm (blue) with respect to the three flexible regions found by the program. The resulting structural alignment is almost complete.

chain A) have been taken from the database of motions. FlexProt detected a hinge region at the α-helix connecting the two domains, at residues 77,78. SCOP classifies Troponin C (1top) as similar to calmodulin. Comparison of 2bbm and 1top yields four similar rigid fragments separated by three hinge regions. See Figure 2 and Table II(a).

Adenylate kinase

Comparison of adenylate kinase isoenzyme-3 (2ak3, chain A) with adenylate kinase complexed with inhibitor (1ake, chain A) detected four similar regions, with the last two with similar transformations and thus clustered to one group. The first flexible region is between residues 120 and 121 of 2ak3 (115 and 118, 1ake) and the second between 160 and 161 of 2ak3 (157 and 158, 1ake). FlexProt compared 2ak3 with UMP/CMP kinase (1uke), another member of the nucleotide and nucleoside kinases family in SCOP. Three similar regions were detected. 2ak3 contains a small rudiment zinc finger subdomain at residues 125–161, between two α-helices. Instead of the zinc finger subdomain, 1uke has only a small loop (residues 134–137) between these α-helices. FlexProt detected the hinge region almost exactly at the location of this small subdomain. The total flexible alignment size is 182 residues and the RMSD is 2.9 Å [Fig. 3, Table II(a)].

Immunoglobulin (fab elbow joint)

We tested our program on an fab elbow joint-like motion,[41] comparing an immunoglobulin fab fragment (1mcp, chain L) with that of 4fab (chain L). FlexProt detected a hinge region at residues 110–111 of 1mcp-L (109–110 of 4fab-L) with a total RMSD of 1.93 Å. A murine T-cell antigen receptor (1tcr, chain B) that also belongs to an antibody variable domain-like family in SCOP was compared to 1mcp, finding two clusters that represented two

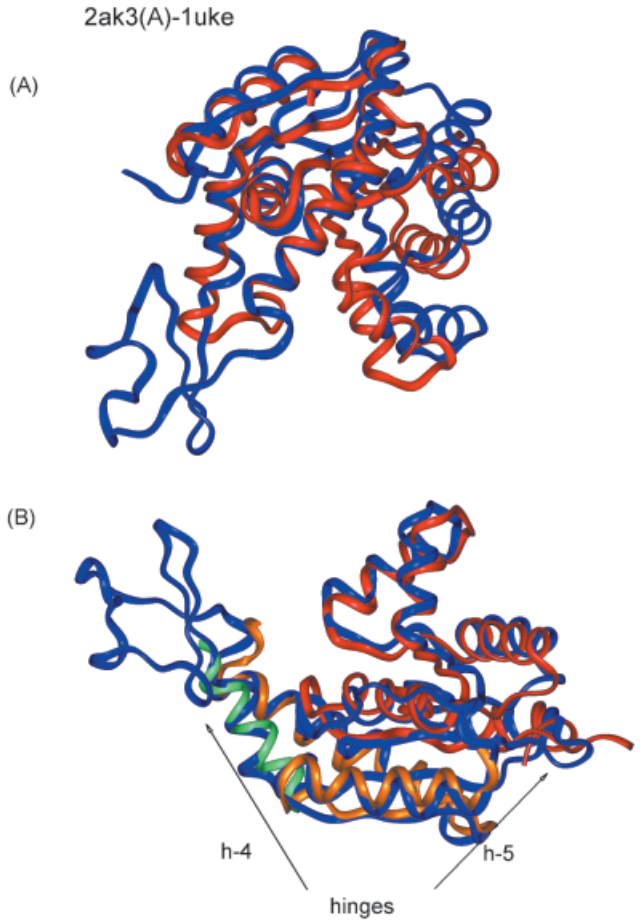

Fig. 3. Adenylate kinase (2ak3, chain A) compared with UMP/CMP kinase (1uke). (**A**). Rigid structural alignment of the two molecules. Note that the lower parts of the molecules are misaligned (2ak3; blue). (**B**). The flexible alignment with respect to the two flexible regions found by the program (2ak3; blue). For further details, see the legend to Figure 2.

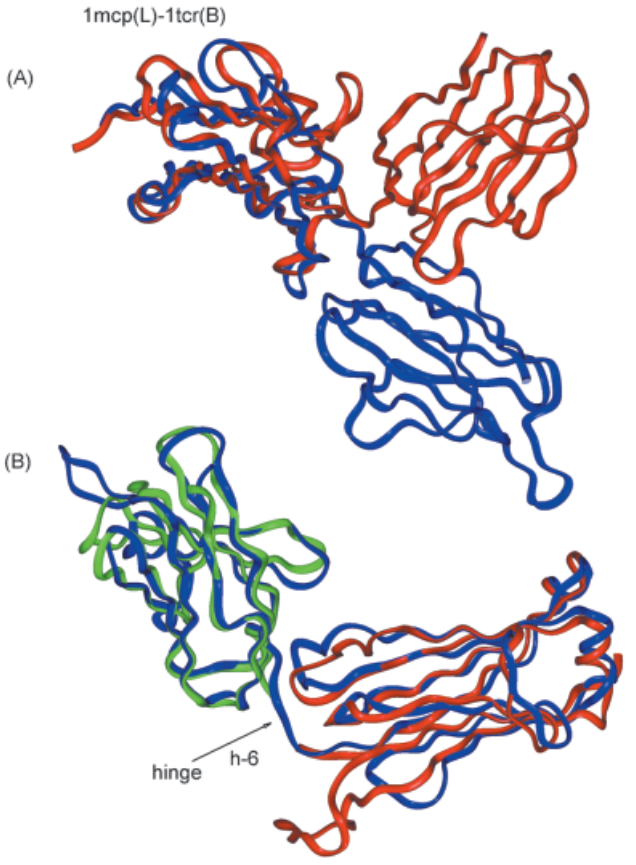

Fig. 4. Immunoglobulin fab fragment (1mcp, chain L) matched with murine T-cell antigen receptor (1tcr, chain B). (**A**). The best rigid superposition between the two molecules (1mcp; blue). (**B**). The maximal alignment with one hinge region detected by the program, with 1tcr superimposed on 1mcp (1mcp; blue). For further details, see the legend to Figure 2.

domains separated by a flexible region with a total RMSD of 2.36 Å [Table II(b), Fig. 4].

Lactoferrin

Lactoferrin is an iron transport protein, folded as two lobes, N and C, each composed of two domains. Binding iron with the N lobe involves a domain motion with two hinges. The rotational bonds are located at the β-strands linking the domains. An example of this motion is given by Gerstein and Krebs[24] for two lactoferrin proteins 1lfh and 1lfg. FlexProt has detected two hinges at residues 84 and 244, consistent with the database. We further compared the N-terminal of the human lactoferrin (1lfh) with the N-terminal of transferrin (rabbit, 1tfd). In SCOP these proteins belong to the same transferrin family. Hinges were detected almost at the same positions as found in the comparison of 1lfh and 1lfg, at residues 86 and 248 of the 1lfh. The RMSD of the superposition is 1.98 Å [Table II(a), Fig. 5]. In this experiment the default value (3 Å) of *MaxRMSD* parameter was changed to 3.1 Å. With the default value the algorithm introduced two additional hinges, although the correspondence of C_α-atoms is almost the same in both cases.

Similarities Detected by Exhaustive Pairwise Comparison of SCOP-Family Representatives

C-terminal merozoite surface protein vs. blood coagulation factor VIIA

An interesting structural similarity was detected between the C-terminal merozoite surface protein (1b9w, chain A) and blood coagulation factor VIIA (1dan, chain L). FlexProt automatically aligned 1dan-L onto 1b9w-A with just one hinge region. The size of the total alignment is 75 residues. The size of 1b9w-A is 89 residues; 1dan-L is 132. 1b9w-A is a C-terminal merozoite surface protein from *Plasmodium cynomolgi*. 1dan-L is blood coagulation factor VIIA protein from the complex of human blood coagulation factor VIIA with human recombinant soluble tissue factor. According to the SCOP classification, both 1b9w-A and 1dan-L belong to the EGF/laminin superfamily, but at the higher level they belong to different families. 1b9w-A belongs to the merozoite surface protein 1 family while 1dan-L belongs to the EGF-type module family. There is no significant sequence similarity between 1b9w-A and 1dan-L. One hinge region was detected at residues 78–84 of the 1dan-L protein. The total RMSD is 2.78 Å [Table II(b), Fig. 6].

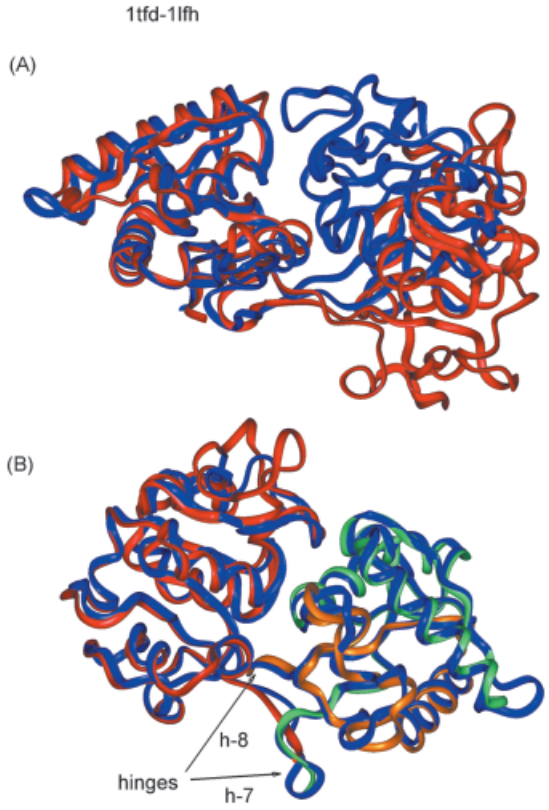

Fig. 5. The N-terminal of lactoferrin (1lfh) aligned with N-terminal of transferrin (1tfd). (**A**). The largest rigid alignment (1tfd; blue). Note that on the right side of the figure the proteins are not aligned. (**B**). The flexible alignment with two detected hypothesized hinges (1tfd; blue). For further details, see the legend to Figure 2.

Class II aminoacyl-tRNA synthetase (aaRS)-like

FlexProt has detected a flexible similarity between the structure of *Escherichia coli* threonyl-tRNA synthetase complexed with its cognate tRNA (1qf6, chain A) and histidyl-tRNA synthetase complexed with histidine (1adj, chain A). There is no significant sequence similarity between these two proteins. According to the SCOP classification both proteins have two structurally similar domains. The first domain, Class II aminoacyl-tRNA synthetase (aaRS)-like catalytic domain, consists of residues 243–532 of 1qf6 and residues 2–325 of 1adj. The second domain, anticodon-binding domain of Class II aaRS, consists of residues 533–642 of 1qf6 and residues 326–421 of 1adj. In this experiment we were interested in detecting the interdomain motion of these two proteins rather than concentrating on the inner domain properties. With default parameter values the algorithm aligned these molecules with six hinge regions, where five hinge regions were placed inside the first domain and one between the domains. Changing the default value (3 Å) of *MaxRMSD* parameter to 5 Å and the minimum length of the congruent rigid fragment pairs (*MinFragSize* parameter) to 45 C_α-atoms (default is 12) allowed FlexProt to detect this interdomain motion with a bending angle of about 48° [Table II(b), Fig. 7].

Histocompatibility antigen vs. neonatal FC receptor

According to SCOP, the histocompatibility antigen (2clr) and the neonatal FC receptor (3fru) have two structurally similar domains. In both molecules chain *A* was considered. The sequence identity of these two proteins is 27%. FlexProt has detected the domain motion for this protein pair. According to the SCOP classification, the first domain, residues 1–178 of 3fru and 1–181 of 2clr, belongs to the MHC antigen-recognition domain. The second domain, residues 179–269 of 3fru and 182–275 of 2clr, belongs to the C1 set domains (antibody constant domain-like). The interdomain motion, found by FlexProt, has a bending angle of about 3°. However, not only the interdomain motion was detected for these two proteins but also a motion inside the first domain was detected by FlexProt. The hinge regions were located at residue 87 of 3fru and at residues 86–92 of 2clr. The bending angle of this motion is about 4.5°. The RMSD of the total alignment (three rigid fragment pairs were aligned) is 2.71 Å [Table II(b), Fig. 8]. The default value (3 Å) of *MaxRMSD* parameter was changed to 3.5 Å, which resulted in the reduction of hinge number from four to two hinges.

Tyrosine-protein kinase C-src vs. haematopoetic cell kinase (HCK)

In this example a large-scale interdomain motion is detected. Two proteins, human tyrosine-protein kinase C-src (1fmk) and haematopoetic cell kinase (HCK, 1qcf), have three common domains, SH3–SH2–kinase. The first domain, SH3, resides between residues 82–145 of 1fmk (80–145 of 1qcf). The second domain, SH2, is located between 146–248 of 1fmk (146–248 of 1qcf). The third domain, tyrosine kinase, is between residues 249–533 of 1fmk (249–531 of 1qcf). FlexProt successfully detected the three domains separated by flexible motion. The hinges were detected at residues 138 and 251. The bending angle between domain SH3 and SH2 is about 0.63° and the bending angle between domain SH2 and tyrosine kinase is about 8.37°. The RMSD of the total flexible alignment is 1.25 Å. To compare the rigid alignment against the result of the FlexProt algorithm see Figure 9 and Table II(b). The default value (3 Å) of *MaxRMSD* parameter was changed to 1.5 Å. With default value, due to small interdomain movements, the algorithm places two hinges at different places, although the size of the alignment is almost the same.

Tyrosine-protein kinase C-src vs. titine protein (autoinhibited serine kinase domain of the giant muscle)

The interdomain motion of the tyrosine-protein kinase C-src (1fmk) was already described above. In this example we investigate the tyrosine kinase intradomain motion. We compare the tyrosine kinase domain of 1fmk with titine protein (autoinhibited serine kinase domain of the giant muscle, 1tki). According to SCOP both structures belong to the same superfamily of protein kinase-like (PK-like). However, 1fmk belongs to the tyrosine kinase family and 1tki is in the serine/threonine kinases family. FlexProt detected the intradomain motion with two flex-

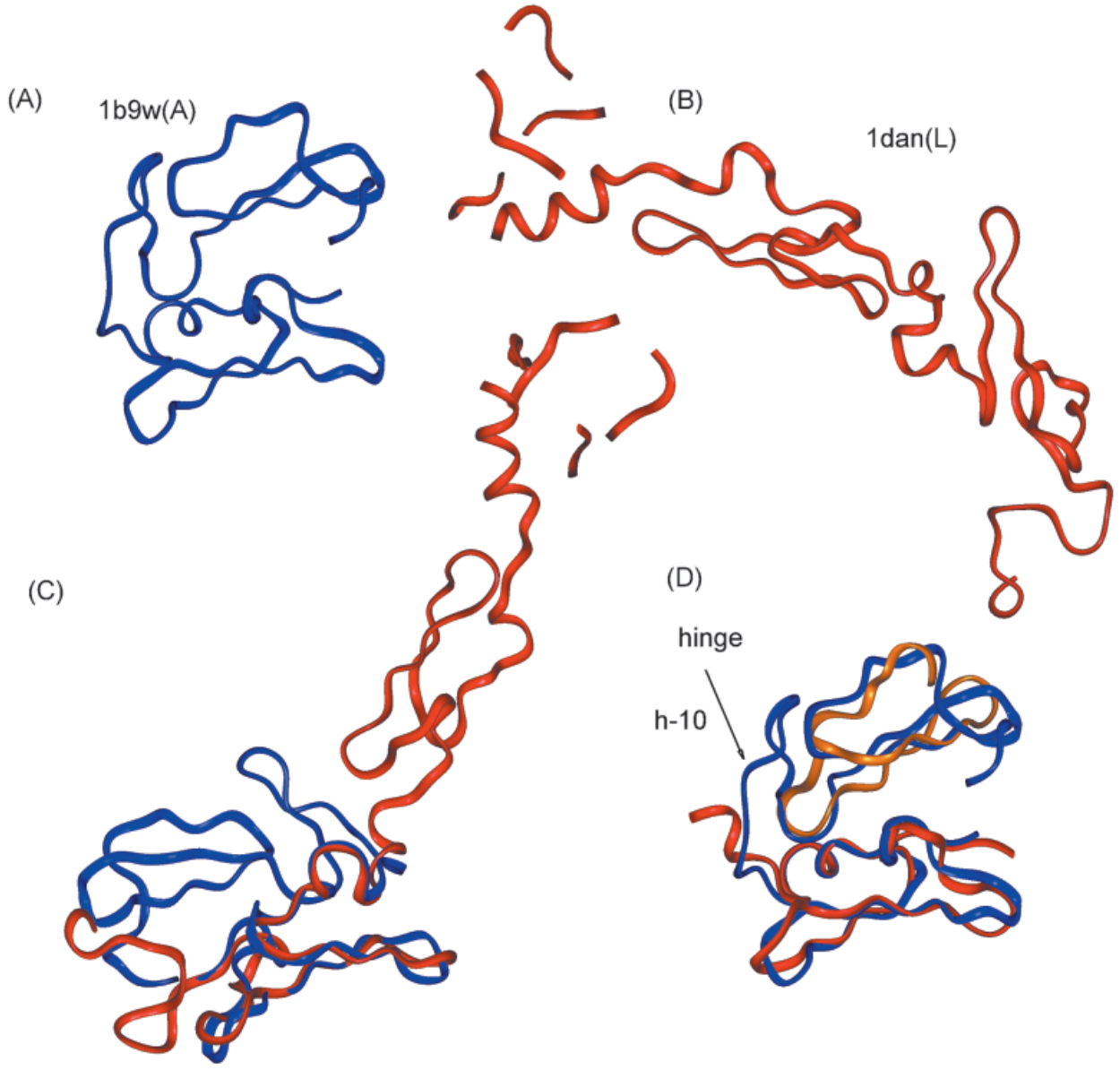

Fig. 6. Results obtained through exhaustive pair-wise database application of FlexProt. The cases are enumerated in Table I(b) and the results tabulated in Table II(b). C-terminal merozoite surface protein (1b9w-A) compared with blood coagulation factor VIIA (1dan-L). The sequence similarity between these two proteins is small. (**A**). The structure of 1dan-L. (**B**). 1b9w-A. (**C**). Best rigid superimposition between the two molecules (1b9w; blue). (**D**). Maximal alignment with one hinge region as detected by FlexProt (1b9w; blue).

ible regions, where the three rigid fragments are of sizes 62, 83, and 86 amino acids (231 in total). 1tki consists of 321 amino acids and the tyrosine kinase domain of the 1fmk has 437 amino acids. The RMSD of the flexible alignment is 3.28 Å [Table II(b), Fig. 10]. In this experiment the default value (3 Å) of *MaxRMSD* parameter was changed to 3.5 Å. Upon applying the default value the algorithm introduced an additional hinge region.

Tissue factor (TF) from rabbit vs. growth hormone-binding protein

According to SCOP both proteins, 1a21 (tissue factor (TF), chain A) and 1hwg (growth hormone-binding protein, chain C), belong to the fibronectin type III protein family. The proteins have two domains, each composed of two β-sheets. The sequence similarity between both proteins is minor. Upon applying FlexProt with the default parameters the algorithm detected solution with six hinge regions, with a total of 166 amino acids and RMSD of 2.72 Å. Three rigid regions were very short, less than 15 amino acids. Thus, we decided to enlarge *MinFragSize* parameter to 15 amino acids (default is 12). In this case FlexProt detected four hinges, although the total length of the solution was slightly shorter (163 amino acids). The first hinge was placed inside the first domain (1hwg, residues 53–62), while the second hinge is found between the domains (1hwg, residues 131–134). The other two hinges were placed inside the second domain of the proteins (1hwg,

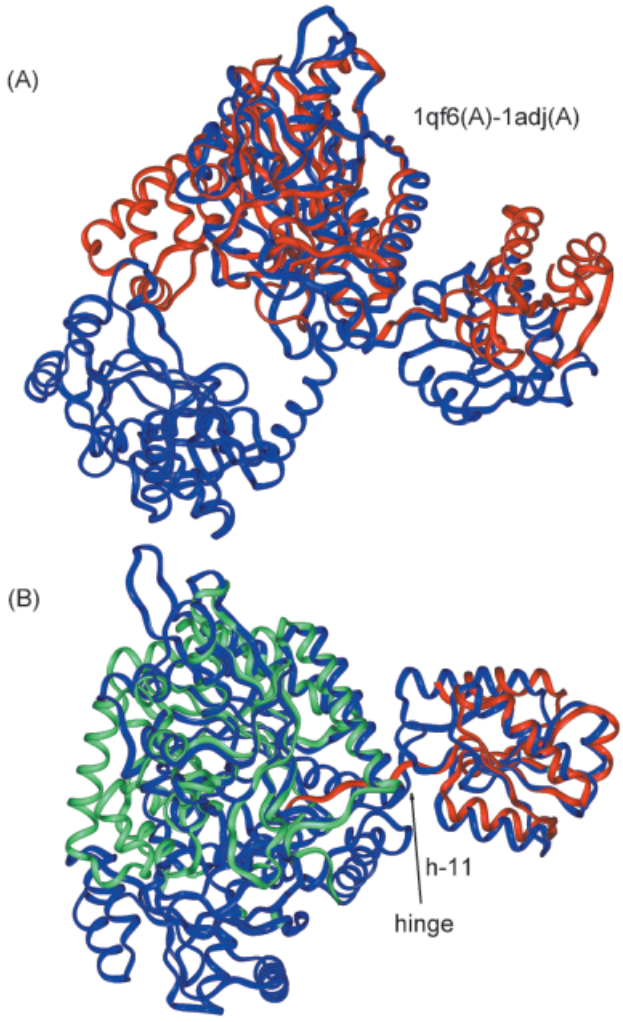

Fig. 7. *E. coli* threonyl-tRNA synthetase (complexed with its cognate tRNA) (1qf6, chain A) vs. histidyl-tRNA synthetase (complexed with histidine, 1adj, chain A). (**A**). Rigid alignment with respect to the catalytic domain class II aminoacyl-tRNA synthetase (aaRS)-like (1qf6; blue). The second domain, anticodon-binding domain of Class II aaRS, is not aligned. (**B**). Flexible alignment (1qf6; blue). Both domains are aligned. The domain bending is about 48°. For further details see the legend to Figure 6.

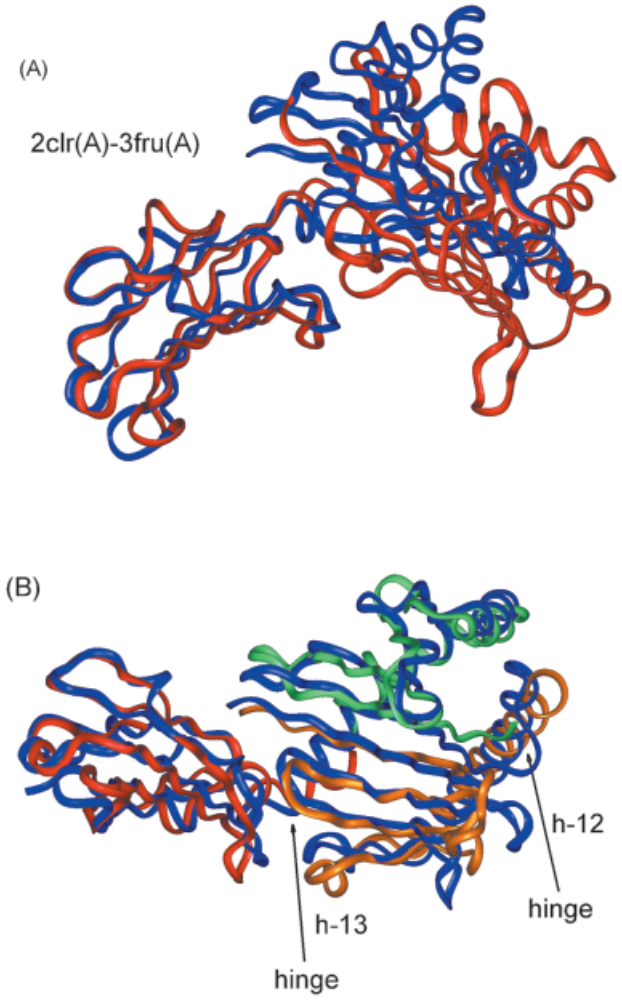

Fig. 8. Histocompatibility antigen (2clr, chain A) and neonatal FC receptor (3fru, chain A) have two structurally similar domains. (**A**). Rigid structural alignment (2clr; blue). Only one domain, on the left side, is aligned. (**B**). Flexible alignment with two hinges (2clr; blue). One hinge is detected between the domains, while the second hinge is located inside the MHC antigen-recognition domain. For further details see the legend to Figure 6.

residues 156–161 and residues 180–186). The RMSD of the flexible alignment is 2.75 Å [Table II(b), Fig. 11].

Random Comparisons

We experimented with several comparisons of random pairs of proteins to illustrate the algorithm's behavior in such cases. As expected, when comparing two proteins of the all-α class a hinge was introduced in every loop separating a pair of helices. Even the comparison of an all-α vs. all-β protein resulted in a 78 amino-acid long alignment with five intermediate hinge regions. The results of these alignments demonstrate that to be significant a flexible alignment should not only be long enough but also should contain at least a pair of consecutive secondary structure elements (helices, strands, loops), which are aligned without intermediate hinges.

All-α proteins: 2fha vs. 1hlb

In this experiment we considered two proteins from different folds of the class all-α proteins (according to the SCOP classification[39]). The first protein, (Apo)ferritin 2fha, belongs to "fold: ferritin-like (core: 4 helices; bundle, closed, left-handed twist; 1 crossover connection)." The second protein, hemoglobin 1hlb, belongs to "fold: globin-like (core: 6 helices; folded leaf, partly opened)." The proteins have length of 172 and 157 amino acids accordingly and the topology of α-helices is different. The algorithm detected a flexible solution with six hinge regions. Six α-helices of 1hlb were aligned (one was split into two) with four α-helices of 2fha. The length of the total alignment is 133 amino acids.

All-α protein vs. all-β proteins: 1hlb vs. 1f5w-A

From the previous experiment we took 1hlb protein and compared it with coxsackie virus and adenovirus receptor

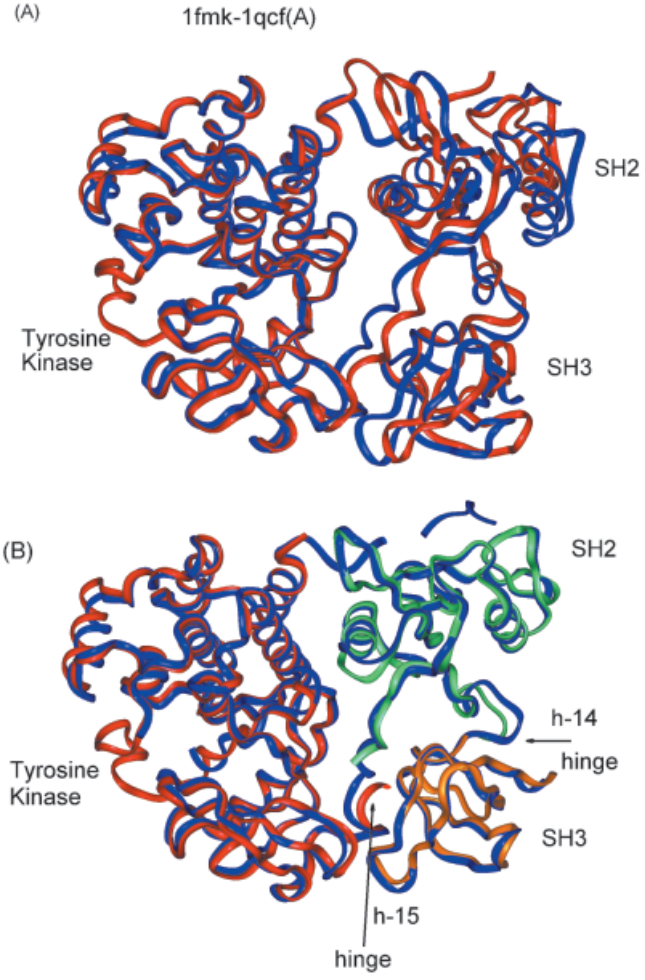

Fig. 9. Human tyrosine-protein kinase C-SRC (1fmk) and 1qcf, haematopoetic cell kinase (HCK). The interdomain motion between three domains, SH3-SH2-KINASE, is detected by FlexProt. (**A**). Rigid structural alignment of both molecules (1fmk; blue). Only the tyrosine kinase domain is aligned; however, the other two domains are not matched. (**B**). Two hinges detected by the algorithm align the three domains with an RMSD 1.25 Å (1fmk; blue). For further details see the legend to Figure 6.

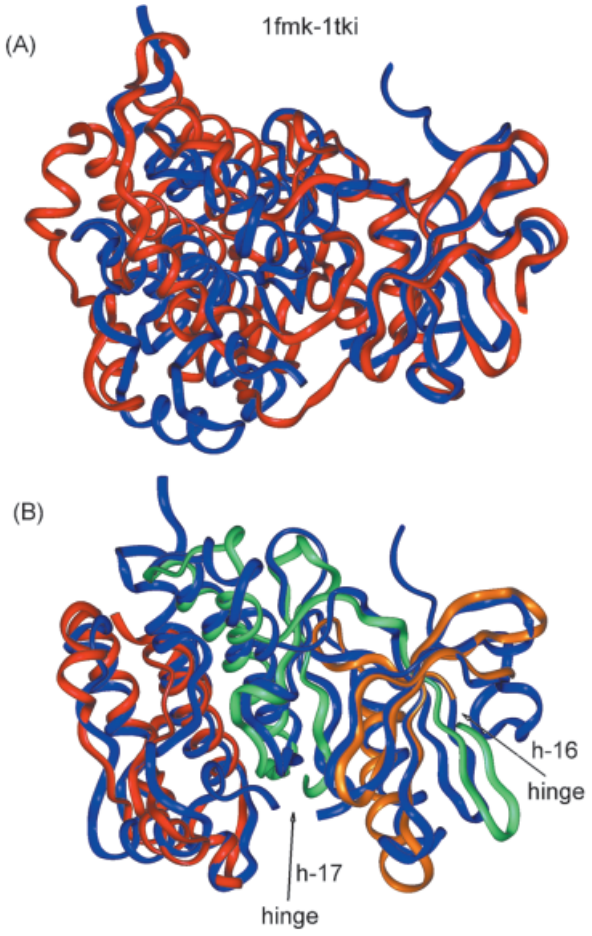

Fig. 10. Intradomain motion between human tyrosine-protein kinase C-SRC, tyrosine kinase domain (1fmk, residues 249-533) and 1tki, titine protein (autoinhibited serine kinase domain of the giant muscle). (**A**). Rigid structural alignment of both structures (1fmk; blue). Only the right part is aligned. (**B**). Flexible alignment with two hinges (1fmk; blue). The RMSD is 3.28 Å. For further details see the legend to Figure 6.

(Car) (domain 1) 1f5w (chain A). 1f5w belongs to the fold "immunoglobulin-like β-sandwich (sandwich; 7 strands in 2 sheets; greek-key)" from the class all-β proteins. Not surprisingly, the algorithm detected a very "bad" alignment. The largest rigid fragment pair contained 16 amino acids. The total length of the alignment was 78 amino acids (the length of 1hlb and 1f5w-A is 157 and 124 amino acids) and the number of hinge regions was five.

DISCUSSION AND CONCLUSIONS

Here we have presented a powerful tool for automatically comparing protein molecules, allowing domain or subdomain motions without an a priori knowledge neither of the flexible regions between the domains, subdomains, or other structurally preserved fragments nor of the residue correspondence. We have further presented a range of results, illustrating its capabilities.

Hinge-bending motions are critical for protein function. The sites of domain motion appear to largely correlate with protein-binding sites. Unlike other types of flexibility (frequently referred to as surface plasticity), hinge-bending motions can bring about larger movements, and thus may relate to opening/closing of the binding sites. As proteins (always) function via binding, the capability of efficiently finding the sites of hinge-bending flexibility is important for the identification of binding sites and hence for alteration of these sites and inhibitor/drug design. The motion detected need not be large. For our purpose, so long as it is consistent between protein structures involving the same, or homologous molecules, it indicates that the motion is there and most likely conserved for its function. Being able to assess the extent of the motions may be advantageous, suggesting designing larger drugs filling the larger volumes upon the opening of the domains. Extensive utilization of such a tool will yield statistics likely to aid in understanding of the nature of the flexible regions and of interdomain interface interactions.

Throughout our extensive comparisons, the algorithm has presented its robustness. Obviously, there is a trade-off between the tightness of the required structural fit and

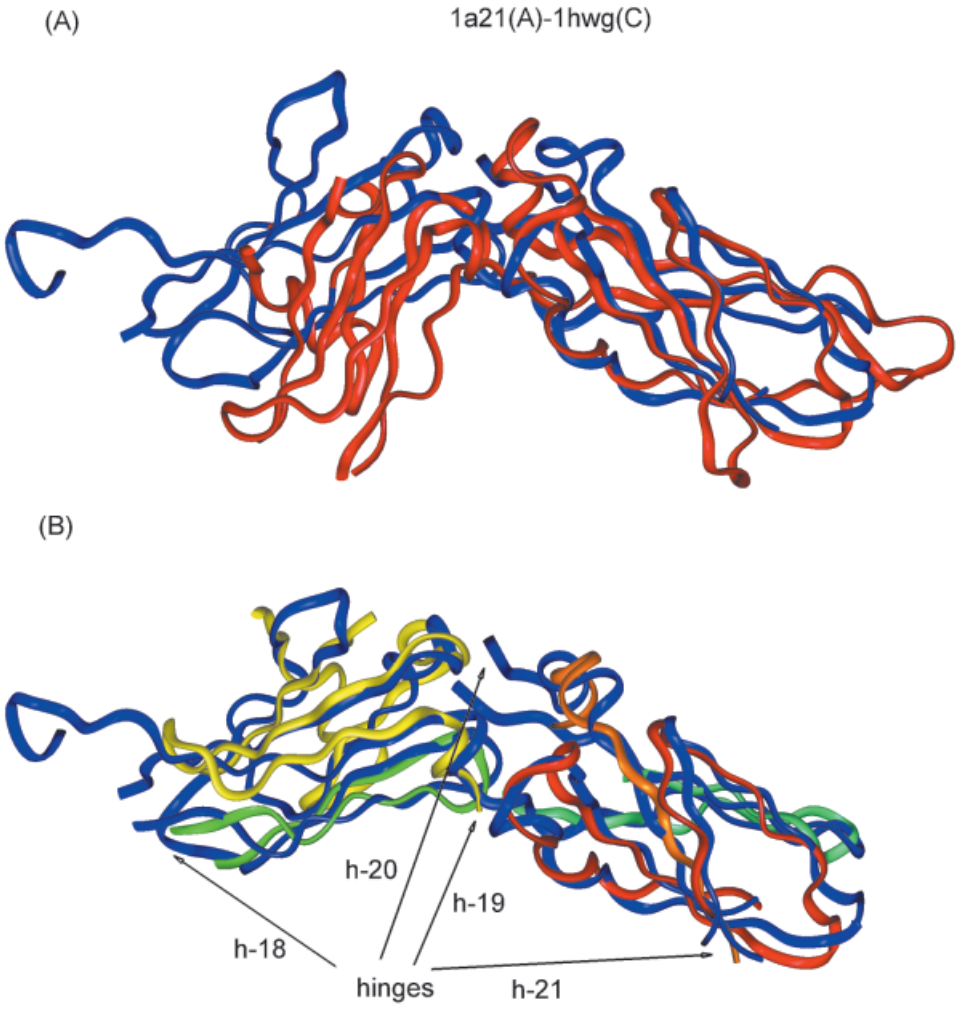

Fig. 11. Tissue factor (TF), (1a21, chain A), is compared to growth hormone-binding protein (1hwg, chain C). (**A**). Rigid structural alignment (1a21; blue). Only one domain is aligned. (**B**). Flexible alignment with four hinges (1a21; blue). The RMSD of the alignment is 2.75 Å. For further details see the legend to Figure 6.

the length of the detected flexible alignment as well as the number of hinges introduced by the algorithm. In some cases a relatively small increase in the *MaxRMSD* parameter resulted in the elimination of several hinges and the elongation of rigidly matching regions. These illustrate the fact that interactive application of the FlexProt algorithm by a knowledgeable user may result in improved results. The speed of the algorithms allows such interactive application. It is also obvious that the required tightness of fit can change depending on the problem at hand. Thus, we preferred to keep the maximal allowed RMSD between a pair of rigidly matching fragments as a user-defined parameter that could be dynamically adjusted. The results of FlexProt illustrate compatibility with the SCOP classification, which has been carried out using both automated procedures and manual inspection. This suggests that FlexProt may be used for structural classification.

The algorithm can work at different resolutions of structural analyses. FlexProt is capable of comparing similar structures, for example, NMR models of the same protein, as well as highly diverged proteins having (partially) similar 3-D structures. It can be utilized in extensive comparisons of proteins and other macromolecules in the search for common domains and flexible joints. Our method is efficient and hence applicable to large-scale database applications. For average size proteins, consisting of about 300 residues, FlexProt takes approximately 7 s on a standard desktop PC (400-MHz PentiumII processor with 256-MB internal memory). The FlexProt algorithm is available at http://bioinfo3d.cs.tau.ac.il/FlexProt/.

ACKNOWLEDGMENTS

The authors thank Meir Fuchs, Ram Nathaniel, and Zipora Fligelman for valuable discussions and for contribution of software to this project. They thank Dr. J.V. Maizel for discussions and encouragement. The research of R. Nussinov and H.J. Wolfson in Israel has been supported in part by the Center of Excellence in Geometric Computing and its Applications funded by the Israel Science Foundation (administered by the Israel Academy of Sciences). The research of H.J.W. is partially supported by the Hermann Minkowski-Minerva Center for Geometry at Tel Aviv

University. This project has been funded in whole or in part with Federal funds from the National Cancer Institute, National Institutes of Health, under Contract NO1-CO-56000. The content of this publication does not necessarily reflect the view or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organization imply endorsement by the U.S. Government.

REFERENCES

1. Taylor WR, Orengo A. Protein structure alignment. J Mol Biol 1998;208:1–22.
2. Orengo CA, Taylor WR. SSAP: Sequential structure alignment program for protein structure comparison. Meth Enzymol 1996;266: 617–635.
3. Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J Mol Biol 1970;48:443–453.
4. Sali A, Blundell TL. The 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 1998;212:403–428.
5. Holm L, Sander C. Searching protein structure databases has come of age. Proteins 1994;19:165–173.
6. Nussinov R, Wolfson HJ. Efficient detection of three-dimensional motifs in biological macromolecules by computer vision techniques. Proc Natl Acad Sci USA 1991;88:10495–10499.
7. Bachar O, Fischer D, Nussinov R, Wolfson HJ. A computer vision based technique for 3-D sequence independent structural comparison. Protein Eng 1993;6:279–288.
8. Fischer D, Tsai CJ, Nussinov R, Wolfson HJ. A 3-D sequence-independent representation of the protein databank. Protein Eng 1995;8:981–997.
9. Brint AT, Willet P. Pharmacophoric pattern matching in files of three-dimensional chemical structures: comparison of geometric searching algorithms. J Mol Graphics 1987;5:49–56.
10. Brint AT, Willet P. Algorithms for the identification of three-dimensional maximal common substructures. J Chem Inf Comput Sci 1987;27:152–158.
11. Smellie A, Crippen G, Richards WG. Fast drug receptor mapping by site-directed distances: A novel method for predicting new pharmacological leads. J Chem Inf Comput Sci 1991;31:386–394.
12. Stockman G. Object recognition and localization via pose clustering. J Comp Vision Graphics Image Process 1987;40:361–387.
13. Johnson MS, Overington JP, Blundell TL. Alignment and searching for common protein folds using a databank of structural templates. J Mol Biol 1993;231:735–752.
14. Eidhammer I, Jonassen I, Taylor WR. Structure comparison and structure pattern. J Comput Biol 2001;7:685–716.
15. Gerstein MB, Altmann RB. A structurally invariant core for the globins. Comp Appl Biosci (CABIOS) 1995;11:633–644.
16. Gelfand I, Kister A, Kulikowski C, Stoyanov O. Geometric invariant core for the V_L and V_H domains of immunoglobulin molecules. Protein Eng 1998;11:1015–1025.
17. Akutsu T, Halldorsson MM. On the approximation of largest common subtree and largest common point sets. Lect Notes Comp Sci 1994;834:405–4135.
18. Taylor WR, Flores TP, Orengo CA. Multiple protein structure alignment. Protein Sci 1994;3:2358–2365.
19. Orengo CA. CORA-toplogical fingerprints for protein structural families. Protein Sci 1999;8:699–715.
20. Gerstein MB, Levitt M. Using iterative dynamic programming to obtain accurate pairwise and multiple alignments if protein structures. In Proceedings of the International Conference on Intelligent Systems in Molecular Biology (ISMB). Menlo Park, CA: AAAI Press; 1996. p 59–67.
21. Leibowitz N, Fligelman Z, Nussinov R, Wolfson HJ. An automated multiple structure alignment and detection of a common substructural motif. Proteins 2001;43:235–245.
22. Leibowitz N, Nussinov R, Wolfson HJ. MUSTA: a general, efficient automated method for multiple structure alignment and detection of a common motif. J Comp Biol 2001;8:93–121.
23. Gerstein M, Lesk AM, Chothia C. Structural mechanisms for domain movements in proteins. Biochemistry 1994;33:6739–6749.
24. Gerstein M, Krebs W. A database of macromolecular motions. Nucleic Acids Res 1998;26:4280–4290. http://bioinfo.mbb.yale.edu/MolMovDB/.
25. Wriggers W, Schulten K. Protein domain movements: Detection of rigid domains and visualization of hinges in comparisons of atomic coordinates. Proteins 1997;29:1–14.
26. Kabsch W. A solution for the best rotation to relate two sets of vectors. Acta Crystal 1976;A32:922–923.
27. Verbitsky G, Nussinov R, Wolfson HJ. Structural comparison allowing hinge bending, swiveling motions. Proteins 1999;33:232–254.
28. Wolfson HJ. Generalizing the generalized Hough transform. Pattern Recog Lett 1991;12:565–573.
29. Sandak B, Wolfson HJ, Nussinov R. Flexible docking allowing induced fit in proteins: insight from open to closed conformational isomers. Proteins 1998;32:159–174.
30. Sandak B, Nussinov R, Wolfson HJ. A method for biomolecular structural recognition and docking allowing conformational flexibility. J Comput Biol 1999;5:631–654.
31. Rarey M, Kramer B, Lengauer T, Klebe G. A fast flexible docking method using incremental construction algorithm. J Mol Biol 1996;261:470–489.
32. Rigoutsos I, Platt D, Califano A. Flexible 3D-substructure matching and novel conformer derivation in very large databases of 3D-molecular information. Yorktown Heights, NY: T.J. Watson Research Center, IBM Research Division; 1996.
33. Bernstein FC, Koetzle TF, Williams GJB, Meyer EF Jr, Brice MD, Rodgers JR, Kennard O, Shimanouchi T, Tasumi M. The Protein Data Bank: A computer-based archival file for macromolecular structures. J Mol Biol 1997;112:535–542.
34. Shatsky M, Fligelman Z, Nussinov R, Wolfson H. Alignment of flexible protein structures. In Altman et al., Editorial Proceedings of the 8th International Conference on Intelligent Systems in Molecular Biology (ISMB). Menlo Park, CA: AAAI Press; 2000. p 329–343.
35. Schwartz JT, Sharir M. Identification of partially obscured objects in two dimensions by matching of noisy characteristic curves. Int J Robotics Res 1987;6:29–44.
36. Cormen TH, Leiserson CE, Rivest RL. Introduction to algorithms. Cambridge, MA: MIT Press; 1990. chap. 25.4.
37. Gusfield D. Algorithms on strings, trees, and sequences: computer science and computational biology. Cambridge, MA: Cambridge University Press; 1997.
38. Shatsky M. Alignment of flexible protein structures. Tel Aviv: Computer Science Department, Tel Aviv University; 2001. M.Sc. thesis.
39. Murzin AG, Brenner SE, Hubbard T, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J Mol Biol 1995;247:536–540.
40. Maiorov V, Abagyan R. A new method for modeling large-scale rearrangements of protein domains. Proteins 1997;27:410–424.
41. Lesk AM, Chothia C. Elbow motion in the immunoglobulins involves a molecular ball and socket joint. Nature 1988;335:188–190.
42. InsightII user guide, October 1995. San Diego, CA: MSI; 1995.