PROTEINS: Structure, Function, and Genetics 35:338–352 (1999)

Identification of Structural Domains in Proteins by a Graph Heuristic

Lorenz Wernisch,[1] Marcel Hunting,[2] and Shoshana J. Wodak[1,3]*
[1]*EMBL Outstation, European Bioinformatics Institute,Wellcome Trust Genome Campus, Hinxton, Cambridge, United Kingdom*
[2]*Institute for Applied Mathematics, University of Twente, Twente, The Netherlands*
[3]*Unité de Conformation de Macromolécules Biologiques, Université Libre de Bruxelles, Bruxelles, Belgium*

ABSTRACT **A novel automatic procedure for identifying domains from protein atomic coordinates is presented. The procedure, termed STRUDL (STRUctural Domain Limits), does not take into account information on secondary structures and handles any number of domains made up of contiguous or non-contiguous chain segments. The core algorithm uses the Kernighan-Lin graph heuristic to partition the protein into residue sets which display minimum interactions between them. These interactions are deduced from the weighted Voronoi diagram. The generated partitions are accepted or rejected on the basis of optimized criteria, representing basic expected physical properties of structural domains. The graph heuristic approach is shown to be very effective, it approximates closely the exact solution provided by a branch and bound algorithm for a number of test proteins. In addition, the overall performance of STRUDL is assessed on a set of 787 representative proteins from the Protein Data Bank by comparison to domain definitions in the CATH protein classification. The domains assigned by STRUDL agree with the CATH assignments in at least 81% of the tested proteins. This result is comparable to that obtained previously using PUU (Holm and Sander, Proteins 1994;9:256–268), the only other available algorithm designed to identify domains with any number of non-contiguous chain segments. A detailed discussion of the structures for which our assignments differ from those in CATH brings to light some clear inconsistencies between the concept of structural domains based on minimizing inter-domain interactions and that of delimiting structural motifs that represent acceptable folding topologies or architectures. Considering both concepts as complementary and combining them in a layered approach might be the way forward. Proteins 1999;35:338–352.** © 1999 Wiley-Liss, Inc.

Key words: contact area; Voronoi polyhedra; branch and bound; structural motifs; protein architecture

INTRODUCTION

The issue of subdividing protein molecules into structural and functional units has come up repeatedly over the last 25 years. In the early days, the main emphasis was on identifying structural units capable of folding independently and of being stable on their own.[1] Some attempts were made to link such units with exon/intron boundaries[2] and thereby with evolutionary processes. In recent years, with the number of known protein structures reaching over 10,000, the major incentive has been to devise automatic methods for identifying domains that can form the basis for a consistent protein structure classification.[3–5]

A number of algorithms for identifying structural domains from the atomic coordinates have been proposed and domains have been discussed by many authors.[3–16] With very few exceptions all the proposed algorithms have been based on the original concepts of Wetlaufer[1] and Richardson,[17] which entail that the interactions within domains are stronger than between domains. In most methods these interactions are modeled by counting interatomic contacts, and domain limits are defined by identifying groups of residues such that the number of contacts between the groups is minimized.

The latter operation represents an optimization problem in which a large number of possible partitions of the protein 3D structure must be searched for those which fulfill the above criterion. Since an exhaustive search for such partitions is impossible, nearly all available algorithms take into account the order of the residues along the sequence, and systematically split the protein into contiguous segments along the polypeptide chain. Wodak and Janin[18] considered partitions that cut the polypeptide only once (1-cut) or at most twice (2-cut). The former yields two domains, each composed of a single contiguous segment, whereas the latter produces a contiguous domain and a non-contiguous one comprising two non-contiguous segments. A more recent procedure starts with a hierarchy of recursive 1-cuts along the chain, and subsequently assembles strongly interacting segment, thereby defining multi-segment domains.[3] The underlying basic partitions are however still produced by artificial 1-cuts. Other approaches involve partitioning the chain into three or at most four contiguous segments,[15] as additional chain cuts

*Correspondence to: Shoshana J. Wodak, EMBL Outstation, European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, United Kingdom. E-mail: shosh@ebi.ac.uk
Received 17 September 1998; Accepted 7 January 1999

become prohibitively costly to compute. Such methods are thus not general enough to correctly identify domains in the small yet interesting fraction of proteins containing domains made up of more than four non-contiguous segments.[14]

The only approach to date that tackles the partition problem in its full generality is that of Holm and Sander.[14] This approach uses a principal component analysis of a modified atomic contact matrix to find a partition with a low number of contacts, independently of the order of the residues along the sequence. But except for the minimizing properties of the eigenvector components, the connection between the computed partition and the problem of minimizing the number of contacts between the generated parts cannot be demonstrated.

Here we propose an algorithm for partitioning a protein 3D structure into sets of residues such that the interactions between the sets is minimum. This task may be viewed as a graph partitioning problem, which is however known to be NP-hard.[19] Fortunately, the type of graphs representing residue interactions in proteins is amenable to analysis by a graph heuristic, which approximates the exact solution closely. This heuristic is the Kernighan-Lin algorithm,[20] modified here to handle partitions into sets of different sizes. The generated partitions produce an arbitrary number of cuts in the chain without any reference to the order of residues along the sequence. The effectiveness of our algorithm is demonstrated by comparing the results for several proteins with the exact solution produced by a branch and bound algorithm.[21]

Another novel aspect of our approach concerns the way in which the interactions between residues are measured. We use the contact area between atoms defined as the area of intersection of the van der Waals sphere around each atom and the faces of its weighted Voronoi polyhedron.[22] The same type of polyhedron is commonly used to partition space between atoms in a molecule and compute their volumes.[23–27] This contact area is related to the buried surface area, which was previously used to define domain limits[18] based on the observation that it was more robust than counting atomic contacts due to its lower sensitivity to distance thresholds. The contact area is, however, superior to the buried area for domain analysis and other applications which involve manipulations of contact matrices and graphs, owing to its additivity and symmetry.

Using the contact-area measure and the Kernighan-Lin heuristic, our procedure identifies the partition with minimum contact area. The identified partition is then accepted or rejected on the basis of a set of additional criteria, describing various expected properties of structural domains and inter-domain interfaces. When a partition is accepted, the entire procedure is repeated recursively on each of the generated substructures until no further splits are authorized. This recursive approach successfully handles proteins composed of any number of domains. Additional criteria similar to those used here have been relied upon by many other domain definition programs for evaluating proposed partitions,[28] because the landscape of all residue contact measures used to date is inherently noisy. However, a systematic evaluation of their effectiveness has hitherto not been performed. In this work a statistical analysis of the performance of a total of ten such criteria is carried out and the degree of correlation between them is determined. The criteria that yield the best results are then used throughout the study.

The performance of our procedure for identifying STRUctural Domain Limits (STRUDL) is assessed by applying it to a set of 787 representative protein chains from the PDB,[29] and the results are compared with the domain definitions derived recently by Jones et al.[16] used as the basis for the CATH domain structure classification.[30] For over half of these chains (438) Jones at al.[16] assigned the domains automatically using the consensus definition by three established methods: PUU,[14] DETECTIVE,[5] and DOMAK.[15] For the remaining chains, for which consistent assignments could not be made automatically, domains were assigned manually.

This comparison shows that STRUDL performs as well as the best of the above- mentioned three methods although in contrast to these methods it uses no information on secondary structures. For 81% of the representative proteins the computed domain limits coincide closely with the CATH definitions. The remaining structures for which our assignments differ from those in CATH are grouped in categories and discussed in detail. This provides valuable insight into the nature of the problems any geometric approach encounters, and on the role protein topology and architecture might play in domain definition.

METHODS

Contact Area as an Interaction Measure

The interactions between substructures are evaluated using the contact area obtained from the *weighted Voronoi diagram*[22] (also known as the "radical planes" method[27]). Each atom is represented by its accessible sphere of radius $R = R_{vdw} + 1.4$ Å, the van der Waals radius increased by the radius of a spherical probe representing a water molecule (see legend of Figure 1 for radii values). In the diagram a Voronoi cell is defined for each atom of the protein as the smallest polyhedron created by the set of planes perpendicular to the lines connecting the atom's center to those of its neighbors, and positioned at the intersection of the accessible sphere of the atom and those of its neighbors (Fig. 1a, b). Polyhedra defined by this or similar procedures are commonly used to define atomic volumes in proteins.[23–27,31] Each of the polygonal faces of these polyhedra separates a cell atom from a neighboring one. The *contact area* with a neighboring atom is then defined as the area of the intersection of such face and the accessible sphere of the cell atom (Fig. 1a).

The advantage of the Voronoi-based contact-area measure over the buried or molecular surface areas used earlier[18] is the symmetry and additivity of the areas: the contact area between atom *A* and atom *B* is the same as that between *B* and *A* (such symmetry does not hold for

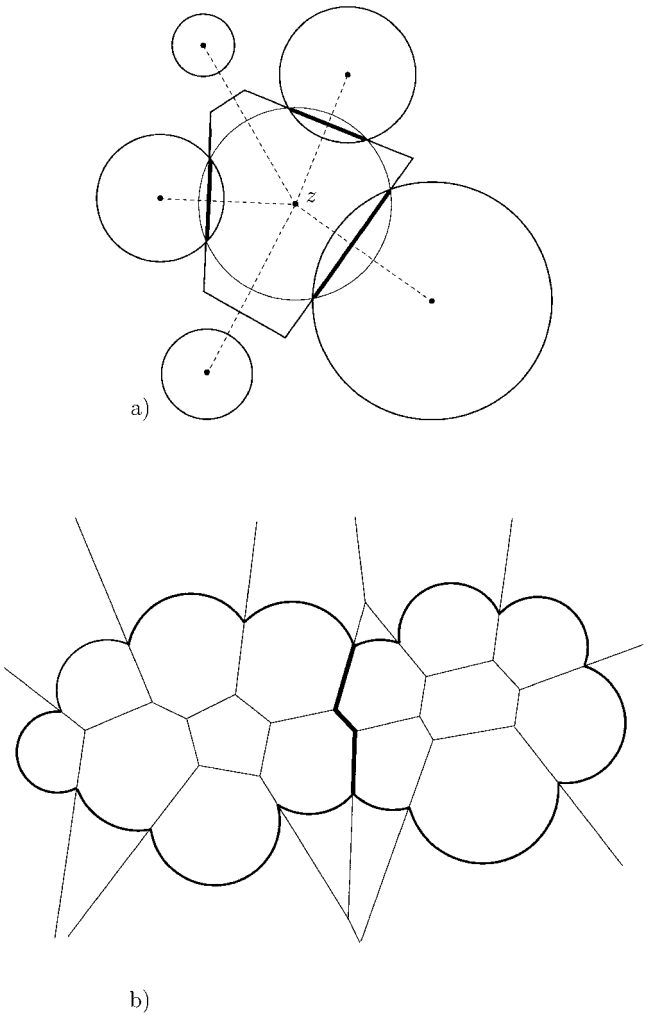


Fig. 1. Use of the Voronoi cell in computing the contact area between groups of atoms. **a)** A section through the Voronoi cell of an atom. This cell is the smallest polyhedron created by the set of planes perpendicular to and cutting the lines connecting the center of atom *z* to those of its neighbors. The contact area between the atom and each of its neighbors is defined as the area of intersection (bold) between the face of the Voronoi cell and the accessible sphere of the atom. The radius of the accessible sphere is the van der Waals radius of the atom increased by 1.4 Å, the radius of the probe representing the water molecule. The van der Waals radii used in this work are: 1.9 Å (C), 1.4 Å (O), 1.6 Å (N), 1.8 Å (S), 1.6 Å (P). Note that the set of van der Waals radii chosen has little influence on the domain partitions found by STRUDL. **b)** Voronoi interface between two groups of atoms. A Voronoi diagram of the solvent excluded volume of a collection of atoms, modeled by their solvent accessible spheres. The Voronoi interface between two groups of atoms is indicated in bold. The thin lines correspond to the intersections of the Voronoi cells with the plane of the figure. The cells of surface atoms have curious shapes and are often unbound (with infinite volumes).

buried surface areas). Furthermore, to obtain the contact area between two groups of atoms, one adds up the contact areas of all the pairs of contacting atoms from the two groups (Fig. 1b). For a given pair of residues *i* and *j* this summation yields the inter-residue contact area c_{ij}. By considering all residue pairs in a given protein structure, the residue contact area matrix $[c_{ij}]$ is obtained.

Searching for Substructures With Minimum Contact Area

Next we tackle the problem of partitioning the residues of a given structure into two groups *U* and *V*, irrespective of chain connectivity, such that the size of *U* equals a target size *k* and the contact area $c(U, V)$ between the groups is minimized, yielding the minimum contact area $c_{\min}(k)$. Since this problem is NP-hard[19] an exact solution cannot be readily computed, but efficient heuristic procedures can be an attractive alternative. Here we use the Kernighan-Lin heuristic for graphs[20] and show that it is capable of adequately approximating the exact solution. The graph heuristic works as follows (Fig. 2):

1. Starting with a subset *U* consisting of one randomly chosen residue, other residues are added to this subset such that the contact between *U* and its complement *V* is increased minimally at each addition, until *U* has reached the target size *k*.
2. A series of exchanges is performed, each exchange moving one residue from *U* to *V* and one from *V* to *U* (Fig. 3). The pair of exchanged residues is chosen so that the contact between residue sets *U* and *V* is minimally increased (or maximally decreased, if possible) after the exchange. The value of the contact area between *U* and *V* is recorded at each exchange step, and each residue leaving *U* is flagged. Exchanges are made until *V* contains only flagged residues.
3. The recorded contact area values are searched for the minimum value, and the corresponding *U*, *V* partition is taken as an approximation to the minimum contact area partition.
4. Starting from this partition, steps (2) and (3) are repeated until no further reduction in contact area is achieved (usually after about three iterations).

This procedure, being heuristic in nature, is not guaranteed to yield the overall minimum, but the likelihood of finding it increases with the number of trials. Our tests show that, in order to achieve successful partitioning, the random selection of the starting residue and the ensuing Kernighan-Lin procedure needs to be repeated about six times. This point is discussed further in connection with the robustness of the algorithm.

Minimum Contact Profiles

To identify domains, for which the limits and size are not known in advance, the partitioning procedure described above must be repeated for all relevant values of the domain size *k*, which range from 1 to *N*/2, where *N* is the total number of residues in the protein. Plotting the minimum contact areas found by the heuristic for all values of *k* produces the *minimum contact profile*. Such profile, computed for a variant of p-hydroxy benzoate hydroxylase mutant (PDB code 1dob), a 394-residue protein composed of two non-contiguous domains, is illustrated in Figure 4a.

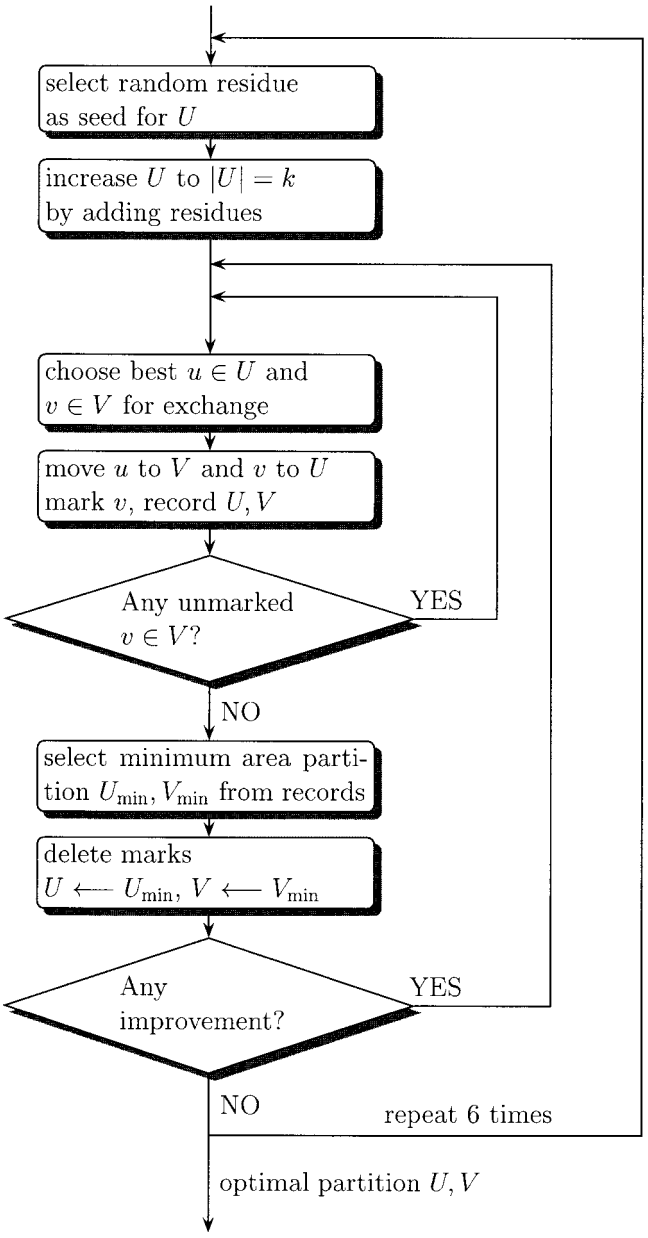


Fig. 2. An overview of the major steps in the STRUDL algorithm as described in the Methods section.

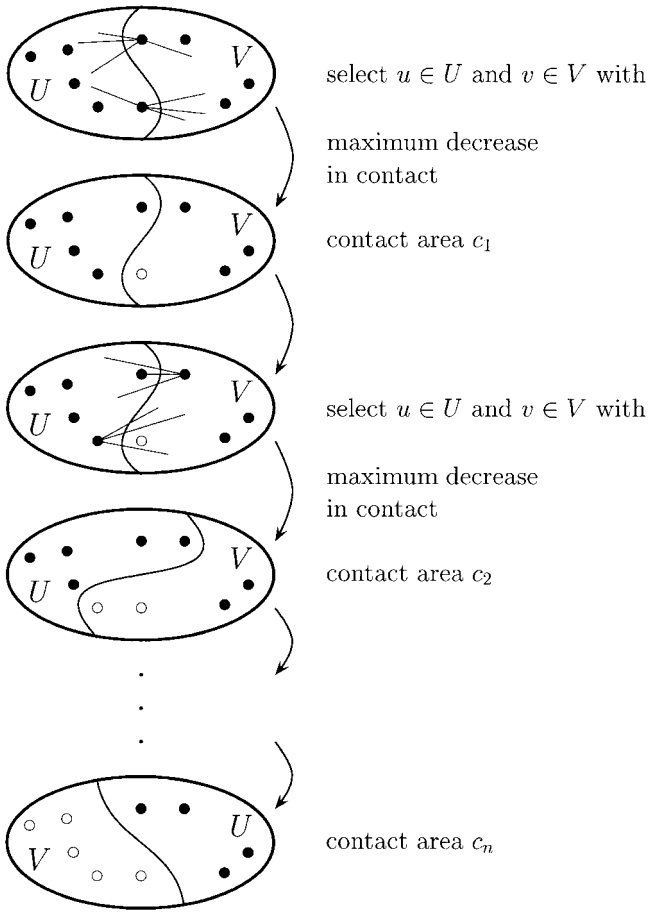


Fig. 3. The residue exchange procedure in STRUDL. Residues $u \in U$ and $v \in V$ (filled circles) are selected so as to produce a maximal decrease or, if that is not possible, a minimal increase in the contact area between U and V upon exchange. Once moved to V, residue u is flagged (empty circle), and can hence not be moved back to U. The exchange procedure stops when V contains only flagged residues. Among all partitions with contact area c_i, $i = 0, \ldots, n$, the one with minimum contact area $c_{\min}$ is selected.

Robustness

Figure 4a shows that large deviations can occur between contact areas for neighboring values of k, as is the case for k values between 50–150, and above 180. Conversely, partitions of different size k can in principle have the same or similar contact areas. One may thus wonder to what extent partitions with similar k and contact area values differ from each other.

To investigate these issues, individual partitions in Figure 4a are compared by computing the number of residues ΔN, which differ in assignment between each partition of size k and the global minimum partition ($k_{\min} = 172$). The results, displayed in Figure 4b, cluster essentially along two parallel lines with unit slope. The lower line, which contains most of the points, comprises solutions lying on the *lower envelope* in Figure 4a. It intercepts the abscissa at $k_{\min} = 172$, implying that the partitions whose size k is close to that of the global minimum are very similar to the minimum partition, an indication that the heuristic is robust in this regard.

The upper line corresponds to solutions with contact areas above the envelope in Figure 4a. They are suboptimal mirror solutions, schematized by the hatched region of domain 1 in the inset of Figure 4b, in which the heuristic sometimes gets trapped. But these solutions can be readily avoided by focusing on those near the lower envelope of Figure 4a. This envelope has the highest probability of containing the optimal solution and is therefore a useful approximation to the minimum contact area profile. Having confirmed that the behavior illustrated here is also displayed in other test cases, this envelope is

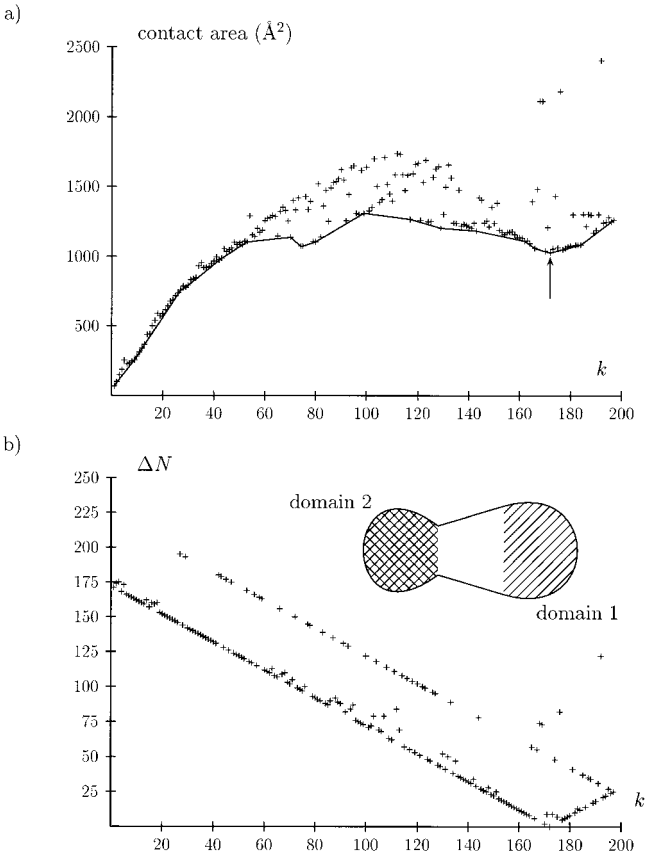


Fig. 4. Robustness of the heuristic procedure. Solutions provided by the heuristic procedure for p-hydroxy benzoate hydroxylase mutant (PDB code 1dob) for partition sizes k = 1, . . . , N/2, where N is the total number of residues. **a)** The contact area values found by the graph heuristic. A lower envelope tracing the points of locally minimal solutions is also shown. The best solution at k_{min} = 172 is indicated by an arrow. **b)** The number of non-equivalent residues N between each partition of size *k* and the optimal partition. The lower line corresponds to gradual deviations from the optimal partition as *k* takes on values different from the optimal size of 172. The parallel upper line corresponds to a series of local mirror solutions in which the graph heuristic gets trapped. Only a few erratic partitions correspond to neither of these two obvious solutions. The inset displays the relation between the mirror solution (singly-hatched region contained in domain 1) and the optimal solution (crosshatched region of domain 2) found by the graph heuristic.

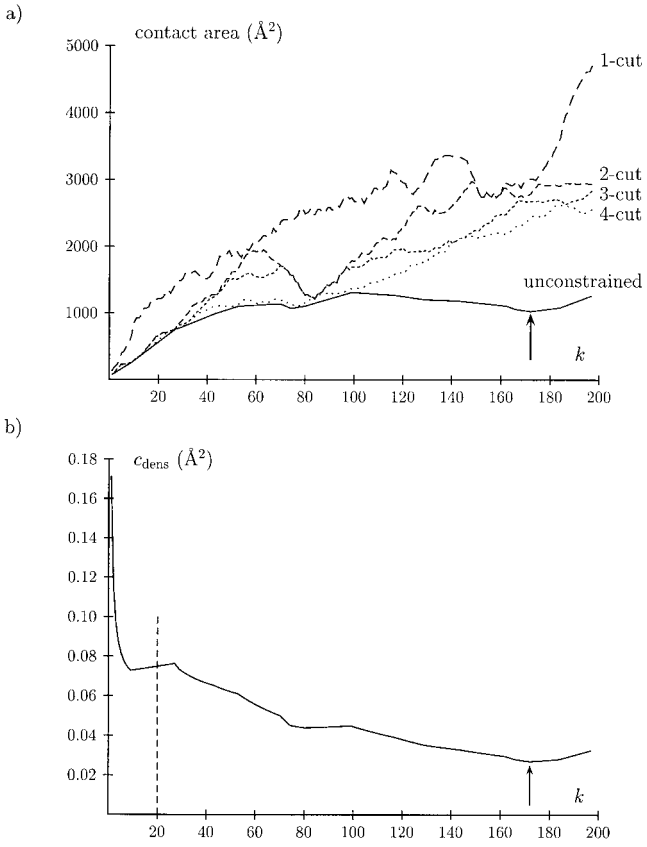


Fig. 5. Minimum contact and minimum contact density profiles. **a)** The minimum contact area profiles computed by the heuristic procedure for p-hydroxy benzoate hydroxylase mutant (PDB code 1dob). Shown are the profiles with no constraints on the number of allowed chain cuts (——), and four other *p*-cut profiles, where *p* is the number of allowed chain cuts (see text). Those are the 1-cut (— — — —), 2-cut (– – – –), 3-cut (-----), 4-cut (·····)profiles. The contact areas (in Å²) are plotted as a function of *k*, the number of residues in the smallest substructure *U*. The global minimum of the contact area (arrow) can only be located in the unconstrained profile, illustrating the advantage of STRUDL over other procedures, in which the number of allowed chain cuts is fixed. The largest value of *k*, which is the maximum size of *U*, equals *N*/2, where *N* is the total number of residues. **b)** Minimum contact density profile for the same protein. The plotted c_{dens} values equal the values of the minimum contact profile divided by $k(N - k)$, $k = 1 \ldots N/2$ (see text).

used throughout this study to approximate the minimum contact profile.

Generality With Regard to Chain Cuts

Most existing domain definition methods make use of the order of the residues along the sequence. They consider only those partitions into two groups of residues with minimum contact, which involve a fixed number of cuts *p* of the polypeptide chain, generally ranging from 1 to 4, and the chain segments produced by these cuts are then assigned to *U* and *V* alternatingly.

p-cut profiles can be obtained by a systematic search letting the size *k* of *U* vary from 1 to $N/2$ and imposing a restriction on the number of allowed cuts *p*. Such profiles, computed with values for *p* ranging from 1 to 4, are shown in Figure 5a alongside the unconstrained minimum-contact profile. The displayed profiles are for 1dob, whose domains comprise a total of six different chain segments involving five cuts. Inspection of the various curves clearly indicates that the *p*-cut profiles converge to the unconstrained profile as the number of allowed chain cuts *p* increases, demonstrating that the latter profile is a generalization of the *p*-cut profiles in which all restrictions on *p* are removed. Note that the computing time becomes prohibitively large as the value of *p* exceeds 4.

Minimum Contact Density Profiles

Having computed the profiles, one needs to identify from them the size *k* that yields the optimal domain partition. Searching for a sufficiently deep local minimum in the profile is a possible approach.[18] The profile of 1dob features a clear minimum at k_{min} = 172 (Fig. 5a), which yields the

correct solution. In general however, the situation is not as clear-cut. Local minima in profiles of some multi-domain proteins can be very shallow, whereas profiles of single-domain proteins can feature some local minima. Moreover, for small k values, which correspond to unbalanced partitions (one small and one large substructure), the contact area is trivially small.

To avoid these pitfalls, we consider the *contact-area density* c_{dens} in place of the raw contact area:[3,14]

$$c_{\text{dens}}(U, V) = \frac{c(U, V)}{|U||V|}$$

where $|U|$ and $|V|$ are the numbers of residues in U and V and $c(U, V)$ is the contact area between U and V defined above. $c_{\text{dens}}(U, V)$ represents the average contact area for all $|U|\,|V|$ possible contacts between the residues in U and V.[3] The *minimum-contact density profile* is readily obtained from the contact profile by dividing the contact area at each k by $k(N - k)$. The domain definition algorithm then searches for the global minimum in this density profile. However, even after this normalization the global minimum in some proteins still occur at trivially small values of k. To avoid such trivial minima, k values below a certain threshold (0.05 N) are not considered.

Figure 5b shows the minimum contact density profile for 1dob. The global minimum in this profile also occurs at $k_{\text{min}} = 172$, corresponding to the expected solution. The density profiles of single-domain proteins may also feature global minima. But the corresponding density value is usually significantly higher.

Criteria for Evaluating Proposed Domain Partitions

Once the global minimum is identified in the minimum contact density profile, a decision must be taken to either accept or reject the corresponding partition, with a rejection corresponding to classifying the structure as a single-domain protein. An obvious criterion on which to base such decision is the actual value of the contact area density c_{dens}. If this value is below a given threshold, the partition is accepted, otherwise it is rejected. But this simple measure is not always reliable. Following other authors we therefore consider additional criteria, representing expected properties of domains and of the interfaces between them.[18,28] A total of nine different parameters, in addition to the actual values of c_{dens}, were selected from a much larger set on the basis of a preliminary evaluation. These parameters are investigated in order to determine their effectiveness as criteria for evaluating partitions proposed by our algorithm.

They comprise two measures of the normalized contact area between putative substructures. The first is closely related to $c_{\text{dens}}(U, V)$, only the product $|U|\,|V|$ in the denominator is replaced by a minimum:

$$c_{\text{min}}(U, V) = \frac{c(U, V)}{\min(|U|, |V|)}.$$

The other relates the contact area $c(U, V)$ between U and V to the sum of inter-residue contact areas in the entire protein:

$$c_{\text{prop}}(U, V) = \frac{c(U, V)}{\Sigma_{\text{all } i,j}\, c_{ij}}$$

where the sum is over all inter-residue contacts $c_{i,j}$.

In addition, we use a measure of compactness, derived from the average inter-residue contact area within a subset U of residues:

$$b_{\text{tot}}(U) = \frac{1}{|U|} \sum_{i,j \in U} c_{ij}$$

where c_{ij}, $i, j \in U$, are the inter-residue contact areas within the subset U. Then $b_{tot} = b_{tot}(P)$ is the average inter-residue contact area in the entire protein P. From the above quantity we compute the mean of the average inter-residue contact areas in U and V:

$$b_{\text{mean}}(U, V) = \tfrac{1}{2}(b_{\text{tot}}(U) + b_{\text{tot}}(V)).$$

Profiles of multi-domain proteins tend to be flat or to have local minima, whereas profiles of single-domain proteins tend to feature monotonically increasing profiles with no local minima (data not shown). In an attempt to capture this tendency we consider three additional parameters.

1. p_{avg}, the average slope of the profile.
2. p_{prop}, the proportion of k sizes with positive slope. For example, the minimum contact profile of 1dob (Fig. 5a) has a negative slope for $k = 70, \ldots, 74$ and $k = 99, \ldots, 171$, consequently $p_{\text{prop}} = 1 - (5 + 73) = 197 = 0.604$.
3. d, the depth of the contact density profile at the minimum, defined as the difference between the value at the minimum and at the highest peak encountered to the left of the minimum, that is for all k values smaller than that of the minimum.

Lastly, we also consider n_{total}, the total number of residues in the protein, and n_{opt}, the size of the smaller group of residues in a minimum contact density partition.

The Procedure for Identifying STRUctural Domain Limits (STRUDL)

Our automatic procedure for identifying structural domains in a given protein comprises the following steps:

1. The Kernighan-Lin heuristic is applied to produce the minimum contact area density profile.
2. The decision to cut the protein into two domains or leave it uncut is made on the basis of a subset of the additional criteria defined above, which has been optimized on a training set of 192 proteins, as will be described below.

3. Since the number of domains into which a protein can be partitioned is not known in advance, (1) and (2) are repeated on each of the identified substructures in turn. When the algorithm refrains from partitioning a given substructure further, this substructure is taken to represent a domain. Otherwise the substructure is partitioned further, and the algorithm is applied to the resulting moieties independently, and so on. This generates a binary tree, whose leaves form the domains of the protein.
4. Steps (1) to (3) yield rather satisfactory partitions most of the time, which rarely contain short chain segments that meander from one domain to another. However, to completely avoid such trivial trimming, all chain segments with fewer than 14 residues are merged with their surrounding segments (the procedure removes smaller segments first, and when the segments are of equal size they are removed in the direction from the N- to the C-terminus). When this editing procedure results in a p-cut, with $p \leq 3$, the corresponding p-cut profile is computed and a partition is derived in a similar manner as for the minimum contact area profile (this usually takes two or three seconds even for large structures).

The entire algorithm is implemented mainly in C++, wrapped in a pearl script that administrates files and parallelizes treatment of a list of structures. All running times mentioned are for an SGI Power Challenge with R10000 processors at 194 Mhz. For example, the domains of a protein of 300 residues are computed in about 30 seconds.

Data Sets

Two protein data sets are used in this study. One, the test set, comprises 787 protein structures from the PDB[29] selected by Jones et al.[16] as the representatives of the different fold families in the CATH protein classification (Orengo and Jones, personal communication, 1998). Of this protein set, 524 structures have been classified as single-domain and 263 as multi-domain proteins and their domain limits derived.[16] The domains were identified by combining consistent definitions obtained by three different automatic procedures, and in cases of inconsistent results, by human intervention. These definitions are used here as the reference, against which the performance of our algorithm is compared.

The second set comprises 192 structures distinct from those in the test set, and selected from the PDB (the PDB codes are given in the legend of Table II). This group of proteins is used as training set for the derivation of the optimal set of additional criteria used to assess the proposed partitions. The selected proteins consist of 132 single-domain and 60 multi-domain structures with different folds and secondary structures. The domain assignments for these structures are taken from the CATH classification, where they were derived using the semi-automatic procedure described above.

TABLE I. Comparison of the Results of the Kernighan-Lin Heuristic with Those of the Branch and Bound Algorithm†

PDB code	size k of U	B & B area (Å^2)	time (sec)	KL area (Å^2)
labk (211)	96	279.29	48.9	279.29
	97	273.95	43.3	273.95
	98	271.73	34.6	271.73
	99	281.30	112.0	282.52
	100	277.19	35.0	277.19
1gky (187)	44	247.72	65.1	247.72
	45	217.26	12.7	217.26
	46	217.26	13.0	217.26
	47	262.70	58.3	262.70
	48	262.20	39.8	262.20
1gss (209)	92	721.40	1007.1	721.40
	93	716.34	957.5	716.34
	94	710.51	905.1	710.51
	95	716.14	1032.1	790.61
	96	720.14	987.4	720.14
1hilA (217)	102	200.42	162.5	200.42
	103	174.64	164.9	174.64
	104	134.39	13.5	134.39
	105	158.39	77.0	158.39
	106	162.53	58.8	162.53
1ppfE (218)	87	842.69	7770.0	902.57
	88	857.10	8044.7	904.00
	89	875.63	6136.3	875.63
	90	882.91	8155.9	904.07
	91	892.65	8409.2	907.42

†The leftmost column lists the PDB code and the total number of residues in the protein (in parentheses). The second column gives the size k (number of residues) of the smaller subset U of the corresponding partition (see text for details). The B & B area is the exact minimum contact area computed by the branch and bound algorithm. The KL area is the approximation of this area computed by the Kernighan-Lin heuristic. For about 4 out of the 5 listed entries the heuristic procedure is shown to yield the optimum. Both the values of the minimum contact areas listed in columns 3 and 5, and the corresponding domain limits (not shown) are identical. The bottom entry, leukocyte elastade (1ppfE), is a compact protein, for which both algorithms encounter difficulties in proposing a partition. In such case the heuristic procedure is seen to approximate less accurately the exact solution found by the branch and bound method, for all tested sizes except size $k = 89$, where it finds the exact solution nevertheless.

Assessing the Quality of the Heuristic Approximation

The quality of the approximation to the minimum contact area computed by the Kernighan-Lin heuristic is assessed by comparing the results with those of a branch-and-bound algorithm[32] which yields the exact solution. Details of the branch and bound implementation of the domain application can be found in Hunting,[32] and will not be discussed here. Suffice it to mention that the upper bounds for the pruning step were obtained from the solution provided by the Kernighan-Lin algorithm, whereas the lower bounds were computed from a Lagrangean dual of a suitable integer programming formulation of the partitioning problem.

Results of the comparison are given in Table I. Unfortunately, the exact algorithm, with running times between

TABLE II. Pairwise Correlations Between the Parameters for Evaluating Proposed Domain Partitions[†]

par.	c_{dens}	c_{min}	c_{prop}	b_{tot}	b_{mean}	p_{avg}	p_{prop}	d	n_{opt}
c_{dens}	1.00								
c_{min}	0.77	1.00							
c_{prop}	0.83	0.83	1.00						
b_{tot}	−0.32	−0.10	−0.08	1.00					
b_{mean}	−0.69	−0.59	−0.53	0.84	1.00				
p_{avg}	0.67	0.77	0.78	0.10	−0.36	1.00			
p_{prop}	0.21	0.46	0.44	0.12	−0.16	0.63	1.00		
d	−0.37	−0.55	−0.50	0.08	0.37	−0.62	−0.69	1.00	
n_{opt}	−0.59	−0.58	−0.43	0.47	0.69	−0.51	−0.15	0.43	1.00
n_{total}	−0.62	−0.53	−0.52	0.42	0.62	−0.49	−0.03	0.30	0.92

[†]The listed parameters embody expected basic physical properties of the domains in relation to the protein, or of the minimum contact area profiles. Their formal definitions are given in Methods. The listed parameters were evaluated for U, V partitions corresponding to global minima in the contact area and the contact area density profiles, computed by the graph heuristic for the proteins in our training set. These values were used to calculate the listed pairwise correlation coefficients. The training set used in this work comprises the following 192 chains from the PDB, given here by their PDB code[29] and, in case of chains, with the chain identifier appended: 115l, 129l, 154l, 182l, 1acbI, 1acx, 1ama, 1asoA, 1ats, 1avhA, 1aznA, 1bcd, 1brcI 1bti, 1bvh, 1cbx, 1choE, 1cif, 1clm, 1comC, 1crj, 1csw, 1cty, 1cue, 1cvo, 1cwdL 1cziE, 1dmyA, 1dud, 1dxu, 1eaf, 1ebdC, 1eco, 1edd, 1epi, 1fdd, 1ffe, 1frd, 1fssA 1fxi, 1gamB, 1gdi, 1gdk, 1glh, 1goc, 1gpb, 1grc, 1hbiB, 1hcc, 1heq, 1hge, 1hilA 1hom, 1horB, 1iad, 1irn, 1kum, 1l07, 1l53, 1lap, 1lh7, 1lhi, 1lld, 1loeB, 1lte 1ltgD, 1lz3, 1mamH, 1mbd, 1mgsA, 1mntB, 1moa, 1mygA, 1nccN, 1nsb, 1ouf, 1ovaC 1pafA, 1pbx, 1pha, 1phg, 1phh, 1pi2, 1pla, 1pnc, 1pnt, 1ppn, 1ra2, 1rat, 1rbp, 1rn4 1rnb, 1rnh, 1rop, 1rve, 1s01, 1sbt, 1sgt, 1sha, 1snc, 1spa, 1stp, 1tfg, 1tho, 1trb 1tro, 1ttb, 1ula, 1xia, 1xib, 1yat, 1ycc, 1ymc, 2abh, 2abk, 2ace, 2ada, 2alp, 2aza 2c2c, 2cbe, 2cdv, 2cpkE, 2cts, 2ctx, 2cwg, 2cyp, 2dri, 2end, 2fb4, 2fbjH, 2fx2 2gcr, 2gn5, 2gst, 2had, 2hpr, 2hsd, 2lig, 2liv, 2madL, 2mcm, 2mev1, 2mgf, 2mhr 2ms2, 2npx, 2omf, 2pab, 2pf1, 2pf2, 2pgd, 2phh, 2plv1, 2ren, 2sga, 2snv, 2sodB 2vaaA, 3adk, 3b5c, 3cla, 3cox, 3ebx, 3gap, 3gbp, 3hsc, 3lzm, 3phv, 3rp2, 3rubL 3sc2A, 3sod, 4bp2, 4enl, 4fgf, 4icd, 4rcrH, 4rhv, 4ts1A, 5cna, 5fbp, 5rub, 6acn 6at1, 6gstA, 8abp, 8adh, 8cat, 8tlnE, 9rubB, 9wga.

13–8,400 seconds, is too slow for practical purposes. In comparison, the modified Kernighan-Lin algorithm takes no more than 0.2 seconds in all the shown cases, and often finds the exact minimum or approximates it within an error of at most 5%. The bottom entry in Table I is for human leukocyte elastase (PDB code 1ppfE), a protein with no obvious domain structure, making it difficult for both algorithms to find optimal partitions.

Cross Validation Procedure

The choice of the best combination of criteria for deciding whether to accept or reject a proposed partition is made by identifying those which yield the best performance of our algorithm on proteins in the training set. To make a realistic assessment of this performance using the training set alone, a cross validation scheme based on the bootstrap method[33] was used.

A bootstrap sample is a random sample of size n drawn with replacement from the training set (that is, some structures might appear several times in the random sample and some not at all). The optimal threshold of a given parameter is calculated for the bootstrap sample and then applied to all the structures of the training set that are not in the sample. This procedure is repeated until the error estimate ϵ_0 converges, with

$$\epsilon_0 = \frac{1}{n}\sum_{i=1}^{n}\frac{Q_i}{B_i}.$$

where B_i is the number of samples not containing protein i, and where for Q_i of these B_i samples protein i was misclassified as single- or multi-domain protein. Note that the concept of misclassification is relative to a chosen reference, the CATH classification in our case.

RESULTS

Optimized Criteria For Evaluating Domain Partitions

The first step of our study has been to derive the optimal set of criteria, from the total of ten considered parameters, for accepting or rejecting a partition proposed by the minimum contact area analysis. Since such set should in principle comprise complementary, as opposed to redundant, parameters, the correlation between the ten parameters was analyzed as follows: The contact area, and contact-area density profiles were generated for our training set of 192 proteins (see Methods) and U, V partitions corresponding to the global minimum of the contact density profiles were recorded. For these partitions, the values of all ten parameters and their pairwise correlation coefficients were computed (Table II).

Not surprisingly, Table II reveals a high correlation between the contact measures c_{dens}, c_{min}, and c_{prop}, between the burial measures b_{tot} and b_{mean}, and between the size parameters n_{total} and n_{opt}. There is also some correlation between the profile parameters p_{avg}, p_{prop}, and d, though not as high, but with the direction of the depth d opposing those of p_{avg} and p_{prop}. The n_{total} row shows the parameters that increase (positive correlation) or decrease (negative correlation) with the size of the protein, with only the proportion of positive slopes p_{prop} exhibiting no correlation

TABLE III. Performance of Parameter Couples for Evaluating Proposed Domain Partitions†

par.	c_{dens}	d	c_{min}	b_{mean}	p_{avg}	n_{opt}
d	11.5/15.0					
	13.7					
c_{min}	12.0/15.0	13.0/16.5				
	14.6	19.4				
b_{mean}	9.9/13.1	9.9/13.1	10.4/14.0			
	13.6	16.1	14.3			
p_{avg}	11.5/14.2	13.0/16.6	13.5/16.0	10.9/14.1		
	14.9	19.1	18.2	13.2		
n_{opt}	11.5/15.6	10.4/13.0	11.0/14.4	13.5/16.1	13.0/17.8	
	14.5	15.9	15.7	17.0	16.9	
c_{prop}	12.0/15.5	13.0/16.4	13.5/16.6	9.9/12.9	13.5/16.3	11.5/15.5
	14.9	19.1	21.2	12.7	20.3	13.9

†This table summarizes the results obtained when the thresholds of two parameters for evaluating partitions proposed by the heuristic algorithm are optimized concomitantly. The six analyzed parameters are those which yielded less than 20 percent error when used individually. Error magnitudes are given in percent of the proteins incorrectly classified as single- or multi-domain, as compared to the CATH domain assignment. Each matrix entry lists (from left to right and top to bottom) the apparent, estimated, and actual errors, respectively (see text). The selection of suitable parameters for the overall algorithm (c_{prop} combined with b_{mean}) is based on the estimated errors shown here.

with size. The low correlation of b_{tot} with the contact density parameters and of b_{mean} with the group of profile parameters should also be noted.

Next, we assess the effectiveness of the ten parameters in classifying the proteins in our test set into one of two categories: single-domain (no cut), or multi-domain (at least one cut). This is done in two steps. In a first step, individual parameters are assessed as classifiers, by assigning a threshold to each parameter and classifying proteins with a parameter value below or above this threshold in either category, as appropriate. The *classification error* is defined as the percentage of proteins incorrectly classified as single- or multi-domain structures, using as reference the CATH domain assignments.[16,30] The threshold value for the considered parameter is then chosen so as to minimize the error for the protein set in question. In a second step, the parameters yielding classification errors below 20% are considered in pairs and the thresholds of all pairwise combinations are optimized.

Table III lists the error magnitudes for all the considered parameter pairs. These magnitudes are given as percent of the proteins that are incorrectly classified as single- or multi-domain. For each parameter pair, three errors are listed: the apparent error, defined as the error on the training set itself, the estimated error, calculated using the bootstrap procedure, and the actual error on the 787 proteins of our test set. We find that b_{mean}, does well in combination with other parameters, whereas c_{dens} and d do better when used individually (data not shown). b_{mean} has the same apparent error when applied in conjunction with c_{dens}, c_{prop} and the depth d, but the estimated real error is smallest when combining it with c_{prop}. The corresponding demarcation line in Figure 6 is positioned so as to maximize the segregation between the single- and multi-domain proteins of our test set, yielding a total of 19 misclassified proteins (9.8% error). Note however, that this pair of optimized thresholds reduced the estimated real error only slightly (12.9%) relative to that of the best performing single parameters (13.8% for c_{dens}). On the other hand, it is reassuring to see that in the test set of 787 proteins the derived thresholds for the same pair of parameters also yield the minimum actual error (12.7%), while using the best single parameter threshold yields a

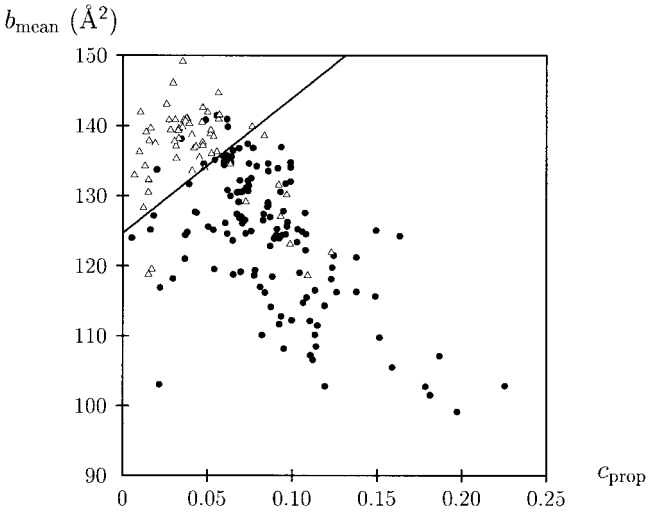


Fig. 6. Threshold optimization for a pair of parameters used to evaluate proposed domain partitions. This figure displays the plot of the mean burial b_{mean} versus the contact area ratio c_{prop}. Both quantities are evaluated for domain partitions corresponding to the global minimum in the contact-area density profiles computed by STRUDL for the proteins of our training set. b_{mean} is the average interresidue contact area of a given substructure; c_{prop} is the ratio of the contact area between the putative domains to the sum of the inter-residue contact areas in the entire protein. The straight line optimally separating the single- (filled circles) from the multi-domain proteins (empty triangles) is shown. This optimal separation entails however 19 errors—proteins classified in the wrong category—out of the total of 192 proteins in the considered set.

TABLE IV. Performance of the Voronoi-Based Contact Areas Versus the Number of Inter-Atomic Contacts†

	Error			Error	
Parameter	Area	Contacts	Parameter	Area	Contacts
c_{dens}	14.2	19.0	n_{opt}	18.0	17.0
d	18.3	18.0	c_{prop}	22.0	23.0
c_{min}	19.7	24.1	p_{prop}	28.6	23.6
b_{mean}	15.5	17.3	n_{total}	18.6	18.6
p_{avg}	18.9	27.9	b_{tot}	26.0	32.7

†All errors are given in percent of incorrectly classified proteins as single- or multi-domain, in the test set of 787 chains, with the thresholds of the listed parameters also optimized on the same test set. The area column lists the errors obtained using a contact matrix based on the Voronoi interface area. The contacts column lists the errors computed for the matrix based on the number of interatomic contacts at an 8 Å distance.

larger error (14.9%). Thus it seems that in a realistic situation some improvement is obtained by combining two criteria instead of using only one.

We also tested the use of combined thresholds of three parameters. But this turned out not to be useful. For example, combining parameters such as p_{avg}, c_{prop} and b_{mean}, reduced the apparent misclassification error to 9.4%. But the bootstrap test yielded an error estimate of 14.0%, indicating that the low apparent error is probably the result of overfitting.

Based on this analysis, we use optimized thresholds of the globularity measure b_{mean} and the cut density c_{prop} to evaluate each *U* and *V* partition proposed on the basis of the global minimum of the contact-area density profile. As shown above, these parameters are only weakly correlated, and hence represent complementary criteria. When the values of both parameters fall in the region of multdomain proteins in Figure 6, the protein is cut and each of the parts is processed further. Otherwise, it is classified as single-domain.

Voronoi Contact Areas Versus Inter-Atomic Contacts

In order to determine if the use of contact areas actually confers any advantage with regard to domain definitions, we re-derived the best threshold values for the ten considered parameters from the test set of 787 proteins itself, using as contact measure the number of atom pairs within an 8 Å distance limit. The corresponding classification errors were then computed for the test set, and compared to those with the Voronoi-based contact measure. The results, shown in Table IV, confirm that our interface area measure is superior to inter-atomic distances. We see indeed that significantly lower values of the minimum actual errors are obtained using the Voronoi interface areas (14.2% and 15.5%) than with inter-atomic distances (17.0% and 17.3%). These conclusions are also maintained for different choices of distance limits.

Performance of STRUDL

In order to evaluate the performance of STRUDL, we applied it to the test set of 787 representative protein chains from the PDB, for which domain assignments were made by Jones et al.[16] and used as a basis for the domain structure classification CATH.[30] The agreement between the assignments produced by STRUDL and the CATH assignments was then evaluated. An assignment was rated as correct when the number of domains was the same as in CATH and more than 85% of the residues were assigned to the same domain by both procedures. The same criterion was applied by Jones et al.[16] in comparing the performance of three automatic domain assignment methods to the crystallographers' assignments, compiled from the literature.

According to this criterion STRUDL yields correct assignments for 635 chains (80.7%). This performance is comparable to the best result obtained using PUU (80%), by Jones et al.,[16] and can be regarded as very satisfactory, considering that unlike PUU our method does not take into account information on secondary structure. In the following, we discuss the 152 remaining cases for which our results are at odds with the CATH assignments. These cases are grouped into four main categories in order to permit a systematic evaluation of the origins of the discrepancies. In the first category, STRUDL assigns fewer domains than in CATH, but the identified domain boundaries, if any, are similar to those in CATH. In such cases, the domain assignment by STRUDL is coarser than that in CATH, and the corresponding chain is classified as *Undercut*. This is the largest category, with 70 proteins, representing 46% of the cases.

The second category comprises 30 protein chains (19.7%), for which our algorithm assigns more domains than in CATH, and hence yields a finer partition of the structure. This category is termed *Overcut*.

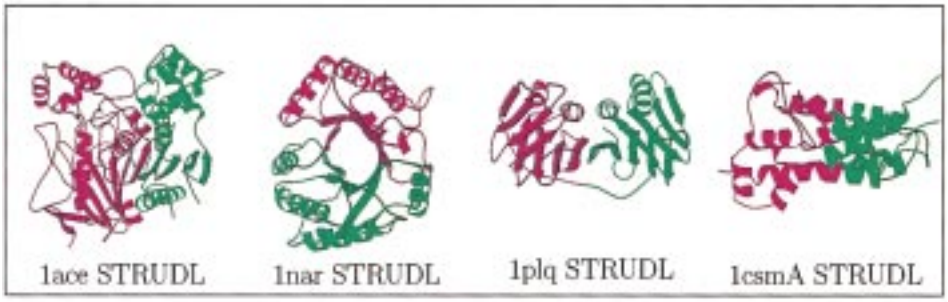
The third category corresponds to the more complex cases where the boundaries differ considerably in manners that cannot be described in terms of coarser or finer partitions. These cases, which number 14 (9.2%) in total, are assigned to the category *Complex*. In this category we also include chains with a large number of assigned domains, some of them representing subdivisions and others groupings of the corresponding CATH domains.

In the remaining 21 structures (13.8%) the domain assignments computed by our method, though not rated as correct by our criterion, seem nevertheless very satisfactory or even superior to those in CATH on the basis of visual inspection. These structures form the fourth category, termed *Debatable* assignments. The domain assignments made by STRUDL for all the proteins considered in this study, as well as a detailed comparison with the CATH assignments in the 152 protein chains, where these assignments differ from ours, are provided on the internet (http://www.ebi.ac.uk/strudl).

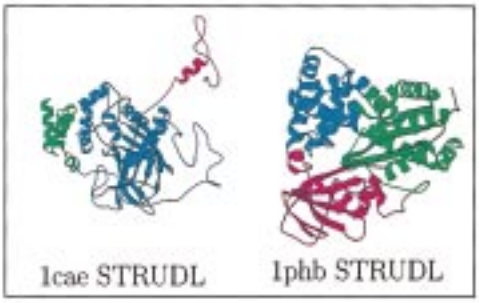
Undercut

The Undercut category (Fig.7a) is the largest of the four categories of misassigned proteins, where our algorithm defines fewer domains than in CATH. In the majority of the cases, STRUDL leaves the protein uncut, whereas CATH splits it approximately in half. The overlap between

a) Undercut



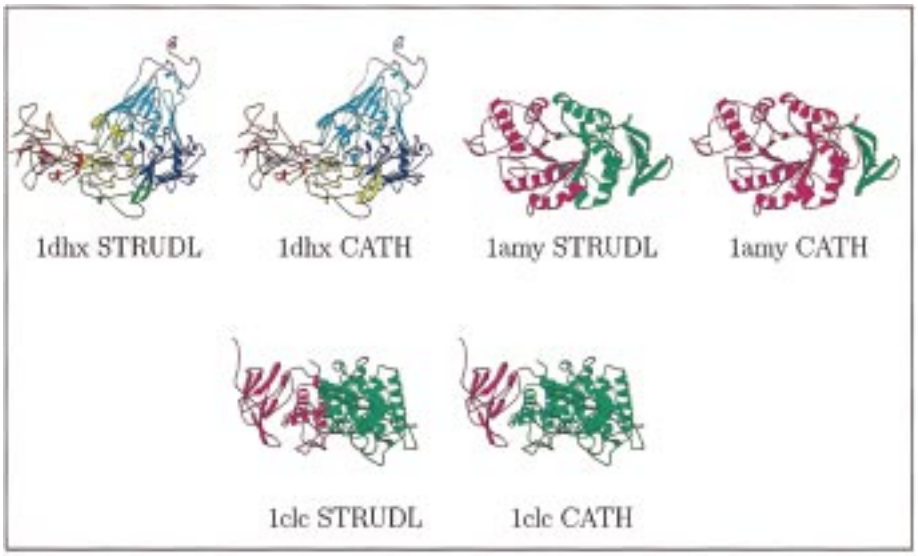
b) Overcut with single architecture



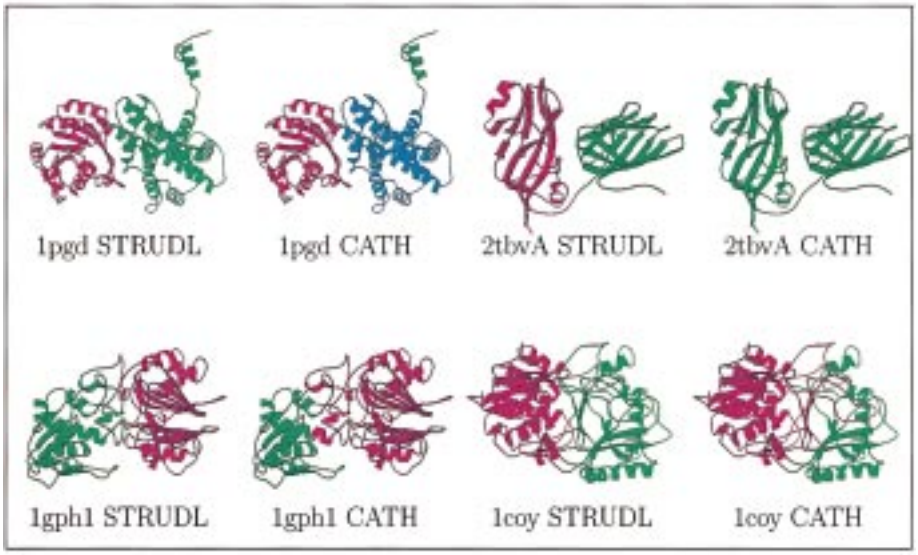
c) Overcut with appendices and multiple architecture

Fig. 7. Examples of proteins for which the STRUDL domain assignments differ from those in CATH (cartoons prepared by MOLSCRIPT[36]). **a)** STRUDL assignments for the Undercut category, in which STRUDL assigns fewer domains than in CATH. The chain A of Cathepsin B (PDB code 1cpjA) is a typical example of proteins in this category. **b)** STRUDL assignments for the category Overcut with single architecture, which groups cases where STRUDL splits certain architectures or secondary structure constellations which CATH regards as single domains. The shown examples in this category are Torpedo californica acetylcholinesterase monomer (1ace); the lant seed protein narbonin (1nar); the eukaryotic DNA polymerase processivity factor PCNA (1plq); chorismate mutase, chain A (1csm). **c)** STRUDL assignments for the category Overcut with appendices and multiple architecture. This category groups cases that may be considered as featuring more than one domain on the basis of simple geometric considerations, and which STRUDL, but not CATH, splits accordingly. A subset of the proteins in this category, such as Proteus mirabilis catalase (1cae), contain decorations, which are given domain status by STRUDL. In others, like camphor 5-monoxygenase (1phb), STRUDL identifies two α-β-domains and an α-domain, whereas in CATH the protein is assigned as a single-domain with an α-non-bundle topology. **d)** The category Complex, comprising large structures with many domains of different sizes and a complex architecture, where both the number of domains and their limits, or just the domain limits, as produced by STRUDL do not agree with those in CATH. Examples given are those of the adenovirus hexon protein (1dhx), the endogluconase (1clc) and the α-amylase from barley (1amy). For each protein we show both the CATH (left) and STRUDL (right) assignments. **e)** The category Debatable includes cases where STRUDL seems to come up with more sensible assignments than those in CATH. Some of the discrepancies may be due to simple "slips" in the CATH assignments, which have been or will be corrected in the near future. Shown examples of proteins in this category are: 6-phosphogluconate dehydrogenase (1pgd), tomato bushy stunt virus coat protein, subunit A (2tbv), glutamine phosphoribosylpyrophospate amidotransferase, chain 1 (1gph), and dehydroisoandrosterone (1coy). For each protein, both the CATH (left) and STRUDL (right) assignments are depicted.

the two assignments, defined as the percentage of residues assigned to the same domain, is thus about 50%. A typical example is cathepsin chain A (1cpjA), for which CATH defines two domains with different topologies, one consisting of an $\alpha\beta$-roll, and the other of a mainly-α-non-bundle. But our analysis indicates that interaction between these



d) Complex



e) Debatable

Figure 7. (Continued.)

domains, and hence their interface area, is not significantly weaker than for other partitions tested by our algorithm, which therefore abstains from cutting the chain.

This cannot be avoided for some of the proteins considered as multi-domain in CATH, even with parameters optimized to maximize the agreement with the CATH assignments, as is clearly evident from the scatter plot of Figure 6.

A simple adjustment of these parameters leads to the correct domain assignments for the majority of the structures in this category (data not shown). This, however, produces incorrect assignments for proteins in other categories, illustrating the difficulty in obtaining a unique set of parameters that consistently improves the results throughout.

Overcut

The proteins in this category, for which our algorithm identifies more domains than in CATH, have been subdivided into two groups. The first group *Overcut with single architecture* (Fig. 7b) includes the cases where STRUDL splits certain architectures or secondary structure constellations, which CATH regards as single domains. For example, our algorithm splits β-sheets, as in the case of the Torpedo californica acetylcholinesterase monomer, or architectures like TIM-barrels (plant seed protein narbonin, 1nar) and αβ-boxes (eucaryotic DNA polymerase processivity factor PCNA, 1plq). The latter cases are notoriously difficult to handle by algorithms that consider contact strength or surface area. The difficulty arises from the

modular nature of these structures—tandem repeats of βαβ units in the TIM barrels, a clear duplication of the αβ motif in the box—and the fact that the interactions between the substructures are not of equal strength throughout. The weaker interactions then form a fault line along which the algorithm splits the structure. While this may not seem acceptable for the TIM barrels, it may make good sense for the two-fold symmetric αβ-box motif, whose asymmetric units could make quite acceptable domains. On the other hand, splitting the protein orthogonally to the axes of α helices, as done by our algorithm in the α-bundle protein chorismate mutase (1csm, chain A), disrupts what clearly appears to be a single entity of protein architecture, and is therefore harder to justify.

This type of problem has been encountered by all other domain assignment procedures based on the evaluation of contact strength, and the typical attempt to tackle the problem has been to use secondary structure information and imposing rules that prevent splitting across β-sheets and helices.

The second group *Overcut with appendices and multiple architecture* (Fig. 7c) contains structures which may be considered as featuring more than one domain on the basis of simple geometric considerations. These structures often contain decorations which, when large enough, are easily identified by STRUDL as independent domains. In some cases, such as Proteus mirabilis catalase (1cae), a protruding short terminal a-helical fragment, as well as a group of a-helices, are given domain status. In others, however, STRUDL singles out substructures which appear to adopt different fold architectures. For example, in camphor 5-monoxygenase (1phb), it identifies two α-β-domains and an α-domain, whereas in CATH this protein is assigned as a single domain with an α-non-bundle topology. Since the discrepancies with the reference assignments involve decorations and appendices, the overlap with the STRUDL assignments is in general higher (70%).

Complex

The cases in the Complex category correspond to large structures with many domains of different sizes (Fig. 7d). In the Adenovirus hexon protein 1dhx both the number of domains and their limits, as produced by STRUDL, do not agree with those in CATH. In other cases STRUDL and CATH agree on the number of domains, but assign completely different domain boundaries. In many of the latter cases the proteins feature either modular, or complex architectures, like the TIM-barrel a-amylate from barley (1amy) or the α-non-bundle endogluconase (1clc), respectively, into which are inserted smaller structural units, often consisting of contiguous chain segments with a completely different architecture. Such insertions usually form decorations, which are tightly associated with the main architectural motif, and can therefore not be readily recognized as completely separate units by our algorithm.

Debatable

In this last collection of cases STRUDL seems to come up with assignments at least as sensible as those in CATH (Fig. 7e). Some of the corresponding structures contain small decorations to which CATH sometimes gives domain status, 6-phospho gluconate dehydrogenase (1pgd), just as STRUDL has done for some of the other cases discussed above. In other instances, some very clear two-domain structures, such as that of tomato bushy stunt coat protein subunit A (2tbv, chain A), are defined as a single domain by CATH, probably due to simple oversight (which might be subsequently rectified).

In a third type of cases, the solution found by STRUDL for structures like glutamine phosphoribosylpyrophospate amidotransferase (1gph, chain 1) seems to be preferable, since it involves fewer chain cuts than the CATH assignment.

In addition, there are these debatable cases, where small intervening structural elements can be assigned to different adjacent domains. A case in point is that of cholesterol oxidase (1coy), where it may indeed be a matter of discussion to which of its two domains—the αββα-sandwich domain, or the αβ-sandwich domain—the single intervening β-sheet belongs.

Finally, we find that in the majority of the cases in this category (16 out of 21), the overlap between the STRUDL and CATH assignments is quite high ($>$ 75%).

DISCUSSION

In this study we presented a novel procedure for identifying structural domains in proteins, which uses a graph heuristic to find substructures with minimum contact area between them, irrespective of chain connectivity. We showed that it yields results in good agreement with the consensus domain assignments used in the CATH classification for 81% of 787 representative protein chains. When the cases in the Debatable assignments category are counted as correct in addition, STRUDL is seen to yield acceptable domain definition for about 83% of the analyzed proteins completely automatically. This is a good performance, which is at least comparable to that obtained by Jones et al.[16] using PUU,[14] the only other available algorithm designed to identify domains composed of any number of non-contiguous chain segments. Unlike PUU, however, and unlike most other automatic domain definition programs, ours uses no information on secondary structure. This information is used by the other programs to prevent splitting β-sheets and α-helices. It is a first step towards preserving the integrity of clearly defined motifs of protein architecture, and improving the correspondence between such motifs and the defined domains. But it has the disadvantage of introducing additional rules for defining acceptable partitions (for example, splitting across a β-hairpin is acceptable, whereas splitting across a β-sheet with three strands or more, is not). The well documented inconsistencies between secondary structure assignments produced by different methods[34] may also be a problem, as they could in turn influence the domain assignments.[16]

Our procedure avoids these problems by not using information on secondary structures, while at the same time achieving, overall, a good correspondence with architectural motifs. We believe that this is due to three main

characteristics, which single out our procedure from other available methods: 1) it uses the Voronoi-based contact area to measure the strength of the interaction between groups of residues. This measure is shown here to be superior to counting atomic contacts, as nearly all other methods do; 2) it applies graph heuristics to closely approximate the exact solution to the minimum contact area partition of a given 3D structure; 3) the additional criteria it uses for accepting or rejecting identified partitions have been optimized using a rigorous statistical approach.

We tried nonetheless to find out if the correspondence between the assignments provided by STRUDL and the architectural motifs defined in CATH could be further improved by taking into account secondary structure information. To this end we gave higher weights to contact areas between residues belonging to the same β-sheet, so as to discourage the algorithm from splitting sheets. Though this improved the correspondence for some proteins, the overall performance of STRUDL, as judged by our criteria, remained unchanged. This suggests that our procedure, as presented in this work, has been optimized in a self-consistent manner, so that adding new parameters, without a more comprehensive optimization approach which considers all the parameters concomitantly, may not be effective.

The study of Jones et al.[16] highlights the differences between three available domain assignment methods by showing that consensus definitions by these methods could be obtained for only a little more than half (55.7%) of the 787 analyzed chains. This has been interpreted as resulting primarily from the inherent difficulties of automatic procedures in coping with the natural complexity of domain arrangements in proteins. Similar opinions have been expressed by other authors.[3]

So far manual assignments have been the only remedy, which many authors use to complement or validate their automatic domain definition methods. But with the growing number of determined protein structures, ways must be found to reduce human intervention as much as possible. It has been suggested that higher reliability of automatic domain assignment methods could be achieved by combining the results of several of domain assignment programs, much like the consensus approach of Jones et al.,[16] but more important still, by supplementing them by knowledge-based "post-processing" to take into account key parameters, which are not considered in the automatic procedure. Strictly speaking, the additional criteria used here to accept or reject proposed partitions could be considered as post processing, in which knowledge about the physical properties of domains is exploited. But the most appropriate set of criteria to use could very well depend on the purpose for which the domain assignments are made.

If the goal is to use the assigned domains as the basis for structural classification, as for example in CATH, then descriptions of protein topology and architecture must be introduced. Our analysis has shown indeed that in many of the cases where our assignments were at odds with the CATH definitions, particularly those in the Overcut category, there seemed to be a clear inconsistency between the concept of structural domains based on minimizing interdomain interactions and that of delimiting structural motifs, which represent acceptable folding topologies or architectures.

The problem of combining the two concepts in an automatic approach is that general procedures for defining and identifying acceptable folding topologies have yet to be designed. In the meantime, a practical alternative—suggested by other authors[16]—has been to verify that a newly assigned domain corresponds to an already known folding motif. This may well take care of the majority of the cases, especially once the structural databases become populated with enough examples of all the possible folding motifs. Until then, however, it harbors the potential danger of biasing the domain assignments towards what is already known.

Lastly, it may be worthwhile noting that algorithms such as STRUDL, which implement a carefully optimized and general procedure for identifying structural units on the basis of physical interactions, may have useful applications in analyses of protein stability and folding, more along the concepts pioneered by Wetlaufer[1] and others. Since STRUDL identifies domains irrespective of chain connectivity or residue order along the sequence, it could also be used to analyze domains comprising several protein chains, such as those that may result from domain swapping in some multimeric proteins.[35]

ACKNOWLEDGMENTS

We are indebted to Susan Jones, Christine Orengo, and Janet Thornton for providing their domain assignments for the representative proteins in computer readable form. We furthermore acknowledge fruitful discussion with Ulrich Faigle, Walter Kern, and with Geoff Barton. Riccardo Valente and Mark Wooding are thanked for their help with the EBI computer systems. Note that a web interface to STRUDL as well as additional material is provided at http://www.ebi.ac.uk/strudl.

REFERENCES

1. Wetlaufer DB. Nucleation, rapid folding, and globular intrachain regions in proteins. Proc Natl Acad Sci 1973;70:697–701.
2. Go M. Correlation of DNA exonic regions with protein structural units in hemoglobin. Nature 1981;291:90–92.
3. Islam SA, Luo J, Sternberg MJE. Identification and analysis of domains in proteins. Protein Eng 1995;8:513–525.
4. Sowdhamini R, Blundell TL. An automatic method involving cluster analysis of secondary structures for the identification of domains in proteins. Protein Sci 1995;4:506–520.
5. Swindells MB. A procedure for detecting structural domains in proteins. Protein Sci 1995;4:103–112.
6. Rossman M, Rao S. Comparison of supersecondary structures in proteins. J Mol Biol 1973;76:241–256.
7. Rossmann MG, Liljas A. Recognition of structural domains in globular proteins. J Mol Biol 1974;85:177–181.
8. Levitt M, Chothia C. Structural patterns in globular proteins. Nature 1976;261:552.
9. Rose GD. Hierarchic organization of domains in globular proteins. J Mol Biol 1979;134:447–470.
10. Sander C. Physical criteria for folding units of globular proteins. In: Balaban M, editor. Structural aspects of recognition and assembly in biological macromolecules, Vol I: Proteins and protein

complexes, fibrous proteins. Jerusalem: Alpha Press; 1981. p 183–195.
11. Janin J, Chothia C. Domains in proteins—definitions, location, and structural principles. Meth Enzymol 1985;115:420–430.
12. Zehfus MH, Rose GD. Compact units in proteins. Biochemistry 1986;25:5759– 5756.
13. Kikuchi T, N'emethy G, Scheraga HA. Prediction of the location of structural domains in globular proteins. J Protein Chem 1988;88: 427–471.
14. Holm L, Sander C. Parser for protein folding units. Proteins 1994;19:256–268.
15. Siddiqui AS, Barton GJ. Continuous and discontinuous domains: an algorithm for the automatic generation of reliable protein domain definitions. Protein Sci 1995;4:872–884.
16. Jones S, Stewart M, Michie A, Swindells MB, Orengo C, Thornton JM. Domain assignment for protein structures using a consensus approach: Characterisation and analysis. Protein Sci, 1998;7:233–242.
17. Richardson JS. The anatomy and taxonomy of protein structure. Adv Protein Chem 1981;34:246–253.
18. Wodak J, Janin J. Location of structural domains in proteins. Biochemistry 1981;20:6544–6552. 340 p.
19. Garey MR, Johnson DS. Computers and intractability, a guide to the theory of NP-completeness. San Francisco:W.H. Freeman; 1979. 340 p.
20. Kernighan BW, Lin S. An efficient heuristic procedure for partitioning graphs. Bell Systems Technical J 1970;49:291–307.
21. Nemhauser GL, Wolsey LA. Integer and combinatorial optimization. New York: John Wiley & Sons; 1988. 763 p.
22. Edelsbrunner H. Algorithms in combinatorial geometry. Berlin Heidelberg: Springer; 1987. 423 p.
23. Richards FM. The interpretation of protein structures: total volumes, group volume distributions and packing density. J Mol Biol 1974;82:1–14.
24. Gerstein M, Tsai J, Levitt M. The volumes of atoms on the protein surface calculated from simulations, using Voronoi polyhedra. J Mol Biol 1995; 249:955–966.
25. Harpaz Y, Gerstein M, Chothia C. Volume changes on protein folding. Structure 1994;2:611–649.
26. Pontius J, Richell J, Wodak SJ. Deviations from standard volumes as a quality measure for protein crystal structures. J Mol Biol 1996;264:121–136.
27. Gellatly BJ, Finney JL. Calculation of protein volumes: an alternative to the Voronoi procedure. J Mol Biol 1982;161:305–322.
28. Rashin AA. Location of domains in globular proteins. Nature 1981;291:85–86.
29. Bernstein F, Koetzle T, Williams G et al. The protein data bank: a computer based archival file for macromolecular structures. J Mol Biol 1977;112:535–542.
30. Orengo CA, Michie AD, Jones S, Jones DT, Swindells MB, Thornton JM. CATH—A hierarchic classification of protein domain structures. Structure 1997;5:1093–1108.
31. Richards FM. Calculations of molecular volumes and areas for structures of known geometry. Meth Enzymol 1985;115:440–464.
32. Hunting M. Relaxation techniques for discrete optimization problems. University of Twente, Enschede, Netherlands: Institute for Applied Mathematics; 1998. 150 p.
33. Efron B, Tibshirani R. An introduction to the bootstrap. New York: Chapman and Hall; 1993. 436 p.
34. Colloch N, Etchebest C, Thoreau E, Henrissat B, Mornon JP. Comparison of 3 algorithms for the assignment of secondary structure in proteins—the advantages of a consensus assignment. Protein Eng 1993;6:377–382.
35. Schleinegger MP, Bennet NJ, Eisenberg D.. Oligomer formation by 3D domain swapping: a model for protein assembly and misassembly. Adv Protein Chem 1997;50:61–122.
36. Kraulis PJ. MOLSCRIPT—a program to produce both detailed and schematic plots of protein structures. J Appl Crystallogr 1991;24:946–950.