Protein Structure Comparison Using Bipartite Graph Matching and Its Application to Protein Structure Classification*

William R. Taylor‡

A measure of protein structure similarity is calculated from the matching of pairs of secondary structure elements between two proteins. The interaction of each pair was estimated from their axial line segments and combined with other geometric features to produce an optimal discrimination between intrafamily and interfamily relationships. The matching used a fast bipartite graph-matching algorithm that avoids the computational complexity of searching for the full subgraph isomorphism between the two sets of interactions. The main algorithm used was the "stable marriage" algorithm, which works on the ranked "preferences" of one interaction for another. The method takes 1/10 of a second for a typical comparison making it suitable as a fast pre-filter for slower, more exhaustive approaches. An application to protein structure classification is described. ***Molecular & Cellular Proteomics 1:334–339, 2002.***

The problem of comparing the three-dimensional structure of proteins has received considerable attention over the years (for reviews see Ref. 1 or Ref. 2) and has been approached from a variety of viewpoints, most of which involve some simplification of the structure. Like the threading problem, structure comparison is difficult (3), and so most of the computational methods used are based on heuristics (4, 5).

The level of description of protein structure at which the greatest simplification can be achieved with the least loss of important topological information is when secondary structure elements (SSEs)[1] are represented as line segments. This gives an order-of-magnitude reduction in the volume of data, which in some algorithms can lead to significant increases in speed (6, 7). With this level of description it is practical to represent the interactions between secondary structures as a graph and to apply graph-theoretic methods to find common substructures in proteins (8–10). The principal method used in these approaches is to identify subgraph isomorphisms between the two secondary structure interaction graphs from each protein. This is a reasonably difficult problem, especially when the desired solution is to find the maximal common subgraph. Most methods use domain heuristics to guide a branch-and-bound search based on the algorithms of Ullmann (11) or Bron and Kerbosch (12).

In this work, the matching of protein SSEs is approached using the simpler class of bipartite graph-matching algorithms. These are designed to find an optimal pairing-up of SSEs, but do not attempt to constrain these in a coherent network as is required in the isomorphism algorithms. Despite this limitation, it will be demonstrated that good solutions can be found that, because of their speed, will prove most useful as a prefilter on the selection of proteins to be compared by a more exhaustive comparison method.

MATERIALS AND METHODS

Line Segment Overlap

The degree of interaction between two line segments would be expected to be greatest for long segments that run close together and least for remote end-to-end juxtapositions. A geometric measure that captures these features is the degree of overlap between the segments. This can be quantified by the length of the region over which the two segments can be connected by a series of lines with end points equidistant from the contact normal[2] (see Fig. 1). However, the overlap length must be modified by how closely the two line segments lie together, with close lines attaining a higher interaction score. A simple reciprocal of the approach distance was considered, but this

From the Division of Mathematical Biology, National Institute for Medical Research, The Ridgeway, Mill Hill, London NW7 1AA, United Kingdom

Received, March 1, 2002, and in revised form, March 4, 2002
Published, MCP Papers in Press, March 4, 2002, DOI 10.1074/mcp.T200001-MCP200

[1] The abbreviations used are: SSE, secondary structure element; IOA, inverse overlap area; SAC, summed area change; SMA, stable marriage algorithm; FN, false-negative(s); FP, false-positive(s).

[2] The contact normal between two extended lines is the (unique) line that is perpendicular to both. For two line segments, $a \rightarrow b$ and $c \rightarrow d$, running in the directions $x = b - a$ and $y = d - c$, the direction of their contact normal is $z = x \times y$. A matrix (M) can then be formed from x, y, and z (referred to as the basis vectors) as $M_x = x$, $M_y = y$, and $M_z = z$. If the line segments are joined by $e = d - a$, then the end points of the contact normal (g and h) can be found from the components of the basis vectors needed to get from a to d via the contact normal. That is, $e = fM$, where f contains the required coefficients and M contains the basis vectors. The components of f can be solved for as $f = eW$ (where W is the inverse of M) and then obtaining the displacements along each line as $x' = f_x \cdot x$ and $y' = f_y \cdot y$, giving $g = a + x'$ and $h = d + y'$. Note: $\times$ indicates the vector (or cross-) product whereas $\cdot$ is a simple product (and not the dot product). Note also that the ends of the contact normal need not lie within either line segment. The situation of exactly parallel line segments does not arise in natural proteins but can be encountered in artificial (idealized) models where the situation is treated separately.

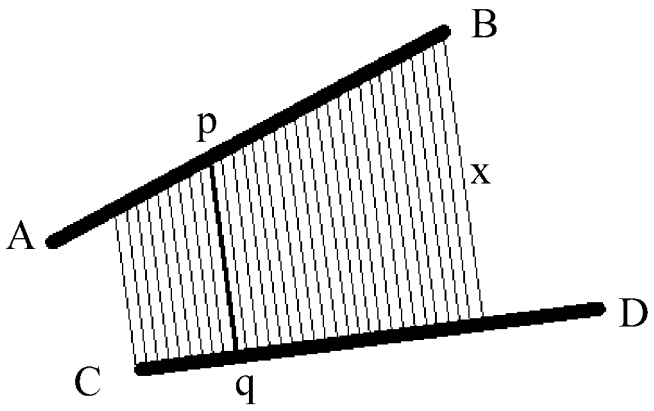

FIG. 1. **Line segment overlap measure.** Two line segments corresponding to secondary structure elements are shown ($A \rightarrow B$ and $C \rightarrow D$) as *thick lines*, with their mutually perpendicular connecting line (p and q) shown at medium thickness. This may lie outside one or both of the line segments. A series of *fine lines* cover the span in which the line segments overlap, the end points of which are equidistant from their corresponding ends of the mutual perpendicular. A measure of interaction is calculated from this as a summation of the lengths (x) of these lines as specified in Equation 1.

was found to give too great a score to close lines, and instead a Gaussian function was used. This was modified further by setting a base level below which the score was set to 1 and above which the Gaussian damping was applied, as follows in Equation 1.

$$a = \begin{cases} 1, & x \le b \\ \exp\left(-\dfrac{(x-b)^2}{d^2}\right), & x > b \end{cases} \qquad \text{(Eq. 1)}$$

In this equation, x is the distance between two points that lie equidistant from the end points of the contact normal between the two lines. The parameters b and d are the distance cutoff (b) beyond which the Gaussian decay is applied with a damping factor determined by d. The values of b (for base) and d (for decay) will be adjusted below.

The interaction of the line segments was measured by summing the inverted distance (a) over a series of lines that have end points on the two line segments equidistant from the contact normal of the lines (Fig. 1). The set of lines have a separation of 0.1 Å giving typically 100 lines summed for an average interaction. The summed measure will be referred to below as the inverse overlap area (IOA).

Solvent-accessible Surface Area Changes

The interactions of the segments of protein structure corresponding to the SSEs were measured using the change in the solvent-accessible surface area observed when each segment was removed from the intact structure. This approach follows earlier studies (13, 14) but used the Definition of Secondary Structure in Proteins program (15) to calculate the solvent areas rather than the original program of Lee and Richards (16).

The accessible surface area (summed over each residue) was calculated first using the intact protein (length N) giving a set of residue areas $\{C_1 \ldots C_N\}$. The two segments were then removed in turn, and the areas were recalculated, giving two further sets of residue areas $\{A_1 \ldots A_N\}$ and $\{B_1 \ldots B_N\}$ (in which the residue numbering of intact protein is retained). If the first segment runs from $a_n \ldots a_c$ and the second runs from $b_n \ldots b_c$, then the combined effect of removing each segment on the other (summed area change (SAC)) can be found as follows in Equation 2.

$$\text{SAC} = \sum_{i=a_n}^{a_c}(A_i - C_i) + \sum_{i=b_n}^{b_c}(B_i - C_i) \qquad \text{(Eq. 2)}$$

Note that (within the error of the area calculation) the areas of the intact protein (C_i) are always less than those after removal of a segment. In addition, the linear segments were always separated by at least one residue to avoid covalent bonded surfaces being exposed and counted.

Calibrating Segment Packing

Continuous Packing Classes—Each segment was characterized only by the length/residue (rise) along the segment axis (17). This can be extended to pairs of segments as $R_{ij} = r_i + r_j$, where r is the rise along the axis of the segment for segments i and j. Plotting the combined rise of both segments against their interaction strength (Fig. 2*a*) shows that the $\alpha\alpha$, $\beta\beta$, and $\beta\alpha$ classes remain sufficiently distinct and that little information is lost by reducing the pair of values to one.

Initial Adjustment—The solvent-accessible SACs were calculated for each pair of segments as described under "Materials and Methods" for a sample of 300 proteins, all of which were free of any errors reported by the DSSP program (such as missing atoms or chain breaks). The IOA was also calculated for each segment pair, using an initial estimate of $b = 5$ and $d^2 = 40$ and plotted against the corresponding SAC value. When broken down into the different packing types ($\alpha\alpha$, $\beta\beta$, and $\beta\alpha$), it was clear that the $\alpha\alpha$ class has less SAC/IOA relative to the $\beta\beta$ class and that the latter also has a large number of IOA interactions with little or no SACs. Rather than find an uneasy compromise between α and β types it was thought better to make the IOA parameters b and d become functions of the segment types, giving the $\alpha\alpha$ type both a longer flat region and a slower decay than the $\beta\beta$ type (with the $\beta\alpha$ intermediate). This was done by setting $b = d = p - R_{ij}$, where p becomes the new parameter to be adjusted, and R is the joint rise of both segments i and j, as calculated above. The rise along the α-helix is 1.5, and it is 3.1 along a β-sheet ($R_{\alpha\alpha} = 3.0$, and $R_{\beta\beta} = 6.2$) making a value around 10 a suitable estimate for p. Although this reformulation helped with the problems outlined above, it did not eliminate completely the larger average SAC/IOA ratio for $\alpha\alpha$ packing. This was then corrected by applying a small explicit multiplying factor to the IOA values of $1 + 1/R_{ij}$. This increases the $\alpha\alpha$ IOAs by 14% relative to the $\beta\beta$ values, giving the improved correspondence plotted in Fig. 2*b*.

Matching Segment Packing

The previous section developed a geometric measure of line segment interaction that gives a reasonable estimate of true interaction of the full atomic coordinates of the segments (as measured by solvent-accessible surface area). In the forthcoming section it is investigated whether this measure will provide a useful basis for the comparison of protein structures when represented as line segments.

Secondary Structure Packing Plot—For two segments, i and j, their combined rise (R_{ij}) can be plotted against their interaction strength as estimated by the IOA measure. This gives a very quick visualization of the type of protein in terms of its secondary structure packing (or architecture). Furthermore, the comparison of two of these plots can give a rough measure of the similarity of the packing between two proteins. Without resolving the specific identity (or sequence order) of the SSEs, it is possible to match up similar interactions. For example, a small $\beta\alpha$ protein (5nul) with two helices above a five-stranded sheet and three helices below will have three $\alpha\alpha$ interactions and four $\beta\beta$

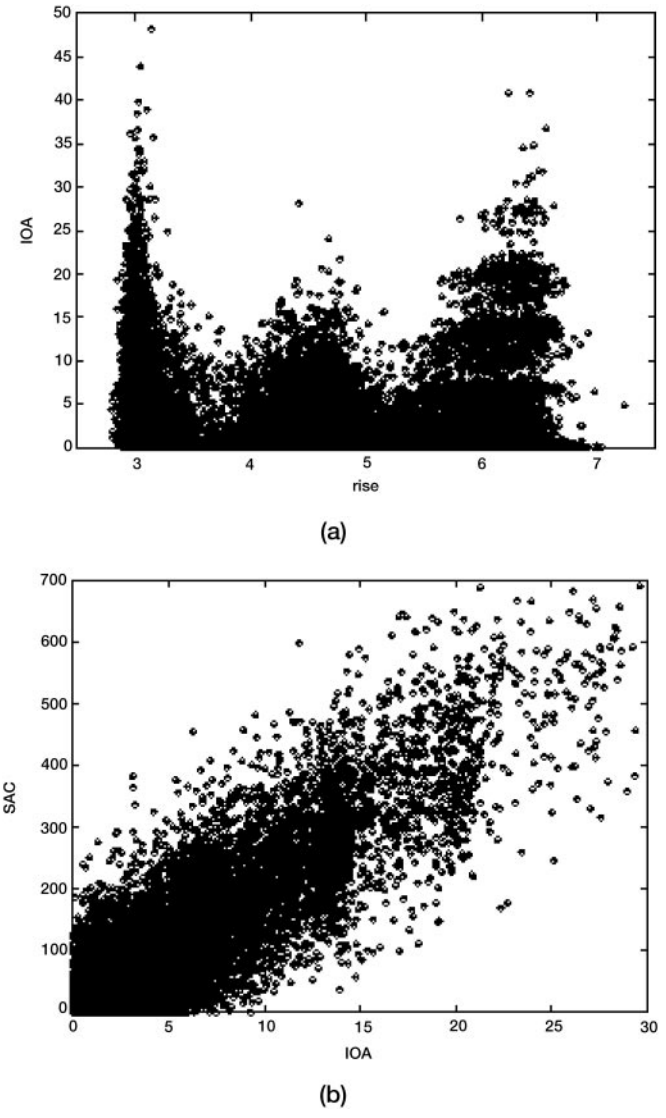

FIG. 2. *a*, the interaction of pairs of SSE lines (*IOA*) is plotted against their combined helical rise (3.0 = $\alpha\alpha$, 4.6 = $\beta\alpha$, and 6.2 = $\beta\beta$). The stratification in the $\beta\beta$ *peak* results from the addition of pairs of residues to bonded strands and not to separation in the sheet. *b*, the SAC seen on extracting the SSEs is plotted against the normalized line segment overlap area (*IOA*).

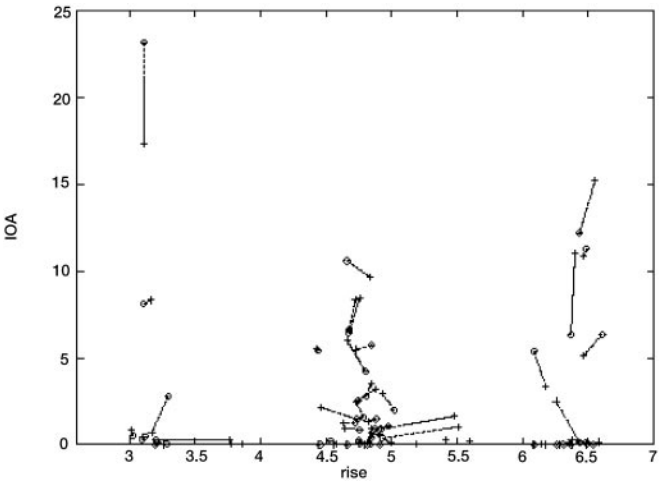

FIG. 3. **SSE packing interaction plot.** The combined rise of the two secondary structures (*x* axis) is plotted against their interaction as measured by the IOA overlap measure (*y* axis). Data are shown for two small $\beta\alpha$ proteins 5nul (*diamonds*) and 1fx1 (*addition symbols*). Corresponding interactions are linked by a *dashed line*. $\alpha\alpha$ interactions lie close to rise = 3, and three corresponding pairs can be seen (strong, medium, and weak).

interactions plus various $\beta\alpha$ interactions. When plotted with a structurally equivalent protein (1fx1), corresponding interactions are apparent to the eye (Fig. 3).

Matching Packing Interactions—To measure the similarity in Fig. 3, an algorithm is required that can match up the corresponding points. If a distance is defined among all points between the two proteins, then the task can be viewed as a bipartite graph-matching problem. The continuous range of distances between points complicates the situation, but an optimal solution can still be found in cubic order time using the Hungarian algorithm (Association for Computing Machinery algorithm 548) (18). However, the size of the graph increases with the square of the number of SSEs, and as this could easily rise over 1000, a cubic order algorithm might prove to be slow. An alternative algorithm was therefore considered, called the stable marriage algorithm (SMA), that operates on the ranked preferences (or match score) of each element with the others (19). If the preferences are found in ranked order, then the execution time of the algorithm has a linear increase with the number of points.

Trials on matrices of random numbers with various sizes and distributions of values suggested that the SMA generally found a solution around 5% less optimal than the Hungarian algorithm. Given the rough nature of the comparison, this was not considered to be an unacceptable result, and with its more efficient time dependence, the SMA algorithm was adopted in the studies below. Indeed, on a limited number of trials on real data, the SMA performed almost equally as well as the Hungarian algorithm (see below).

Interaction Match Score—The matching algorithms discussed above require a metric between points in the packing plot. This should have the properties that strong interactions are paired up and that SSEs of unlike type are not matched. If the interaction (IOA) between two SSEs, *i* and *j*, in protein *A* is designated as I^A_{ij}, and their combined rise values ($r_i + r_j$) is R^A_{ij}, then an equivalent pair (*m* and *n*) in another protein, *B* will have values I^B_{mn} *and* R^B_{mn}. To ensure that both interactions are significant, the match of these pairs of SSEs was quantified by the product of their individual interactions weighted by *W* as follows.

$$P = (1 + I^A_{ij} \cdot I^B_{nm} \cdot W)^{\frac{1}{2}} \quad \text{(Eq. 3)}$$

The difference in secondary structure types ($r = R^A_{ij} - R^B_{mn}$) should lead to a weaker match as the difference increases. This inversion was implemented using a Gaussian switch function of the form, shown in Equation 4,

$$s = \exp(-R^2 \cdot w_s) \quad \text{(Eq. 4)}$$

such that when there is no difference, then the value of *s* is 1. The factor w_s (which corresponds to the standard deviation in the normal distribution bell curve) modifies the effect of the transform and will be optimized below.

Two further geometric quantities that were readily available from the calculation of the overlap areas were the distance between the lines (specifically, the closest approach of the line segments) and the

angle between them.[3] The difference in these quantities was inverted as above, giving a distance score d and an angle score a, each with their associated weighting factor, w_d and w_a, respectively. Similarly the difference in packing was also considered in the form of a similar score p and weight w_p.

These quantities were all combined in an overall score (S) as shown in Equation 5.

$$S = P \cdot s \cdot d \cdot a \cdot p \qquad \text{(Eq. 5)}$$

Thus, if the secondary structure types, distances, angles, and packing are a close match, then the value of S will approach P (the combined interaction), but if any one quantity should differ markedly, then the value of S will fall toward zero.

The value of S for each pair of interactions compared between two proteins provides the preferences on which the stable marriage algorithm operates and the resulting sum of the S values over all matched pairs of interactions gives the measure of similarity between the two proteins (referred to below as the score).

Data Selection—As described above, the method embodies four adjustable parameters, along with the weight W on the combined packing. It is desirable to optimize these for typical proteins. This was done by selecting a set of true relationships and a set of false relationships and varying the parameters to optimize the separation between the scores obtained on each set.

The set of true relationships was taken as all pairwise relationships between proteins defined as belonging to a family in the HOMSTRAD data bank (20). Similarly, false relationships were taken as the interfamily pairs in the HOMSTRAD data bank. As there are many more interfamily pairs, these were limited to the same number of intrafamily pairs, giving slightly over 5000 in each set.

Dividing True from False—Both from theory and trials, the size of the score was expected, and found, to be in rough proportion to the square of the number of matched pairs, and its square root was plotted correspondingly as a function of the number of pairs. When matching elements in objects of different size, the number of matched pairs is often taken as the normalizing quantity. In the current application, as in sequence alignment, this would be the size of the smaller protein. However, with the current method, when matching a small protein against a large one, the former is almost sure to find a better match for its components as the size of the latter increases. To model this aspect, the geometric mean of the lengths was taken, which retains the property that the expected score is zero if one protein has zero components and increases slowly (as a function of the square root of the difference) as one protein grows larger than the other.

On this plot, the line was then found that divided optimally the true scores from the false score. This was determined as the sum of the number of true scores lying below the dividing line and the number of false scores lying above the line. This quantity is often referred to as the sum of false-negatives and false-positives or FN + FP. For computational efficiency, this sum was not minimized as this would involve a search, dealing with multiple minima. Instead, the minimum sum was found when FN = FP (±1), which is more easily located and is referred to below as the balanced sum.

RESULTS

Algorithm Execution Times—The execution time for the core matching algorithm was measured by running the program over the 5000 protein pairs in the true data set, first as described above and then with the matching algorithm short-circuited (immediately returning a zero score). Taking the difference in times thus allows all the housekeeping functions of the program to be ignored.

For the stable marriage algorithm the elapsed computer time on an otherwise quiet 733-MHz Pentium III processor was 8 min for the 4000 comparisons (0.12 s per comparison). This includes filling the score matrices and their sorting (into ranked order). The time for the Hungarian algorithm was just four times greater, but comparison of the scores found by the two algorithms revealed that there was almost no difference, suggesting that the stable marriage algorithm performs better on real data compared with the trials with random number fields described above.

Parameter Optimization—The fast execution time of the SMA allowed the parameter space to be explored extensively. The weight on the packing match (W in Equation 3) was varied from 0 to 0.5 by intervals of 0.1, whereas the Gaussian factor on the packing difference (w_p in Equation 4) was varied from 1/20 to 20 in rough doubling intervals (0.05, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0, and 20.0). For the best values found with these combinations, all the other parameters were raised or lowered by a factor of two, and the process was repeated until the minimum value in packing parameter space was not lowered by displacement of any of the other parameters. The minimum solution was found when $W = 0.2$ and $w_p = 10$, lying in a smooth shallow basin (Table I). At this point, the other parameters were $w_s = 2$, $w_a = 5$, and $w_d = 0.05$ for secondary structure, angle matching, and distance matching, respectively. The low weight on the distances reflects their redundancy with the packing measure. These results were generated using the signed angles (incorporating chirality) and were slightly better than those generated using the unsigned angle (data not shown).

TABLE I
Parameter optimization

The balanced sum of errors (FP + FN when FP = FN = minimum) is tabulated for different values of the weights on the total packing (W) and the difference in packing (w) for secondary structure interactions. Each FP is a pair of proteins from different HOMSTRAD families that score above the cutoff whereas an FN is a family pair that lies below the cutoff. The cutoff is the line that divides the 10,000 data points such that FP = FN = minimum. The value 155 in bold is the minimum.

W	0.0	0.1	0.2	0.3	0.4	0.5
w						
20.0	178	170	162	165	165	166
10.0	172	158	**155**	158	163	166
5.0	164	160	158	160	164	167
2.0	166	161	159	164	166	172
1.0	166	160	160	166	173	174
0.50	170	166	166	171	177	181
0.20	176	170	176	180	191	197
0.10	177	178	186	192	194	198
0.05	182	194	193	200	200	203

[3] Two types of angles were considered. The first was the simple (unsigned) angle between the two lines as calculated from their scalar (dot) product in the range $0 \rightarrow \pi$. The second had the angle signed by the chirality of the pair as lines right = $0 \rightarrow \pi$ and left = $0 \rightarrow -\pi$. The difference in the latter angles was calculated as $a - b$ or $2\pi - a - b$ when $a - b > \pi$.

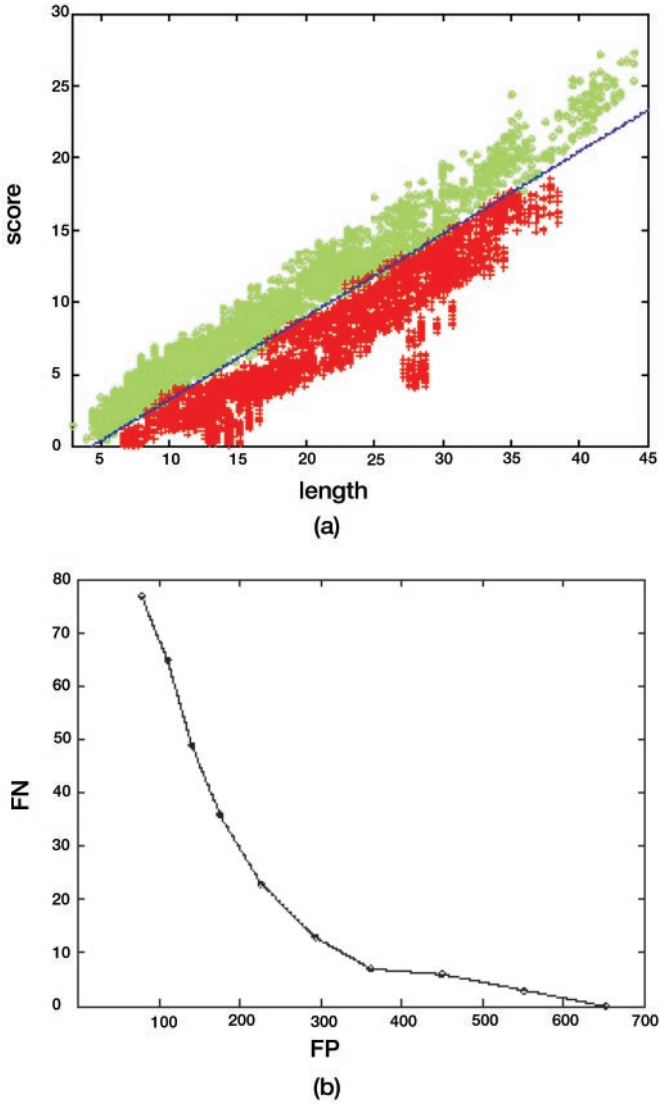

FIG. 4. **Best true/false separation.** *a*, geometric mean of the lengths of pairs of proteins (*length*) plotted against the square root of their comparison score (*score*). Pairs within a HOMSTRAD family (true) are plotted as *green diamonds* whereas pairs in different families (false) are plotted as *red addition symbols*. The best dividing line (*blue*) is $0.5745 \times x - 2.5$ leaving only 155 misclassified points out of over 10,000. *b*, starting from the point in *a* where FP = 77 and FN = 78, the dividing line is lowered (in steps of 0.1), and the FP and FN values are plotted until FN = 0.

Analysis of the Results—The balanced sum at the optimum point was 155, being the total number of misclassified points out of 10,320 (1.5% error). The data that gave this result are plotted in Fig. 4*a*, along with the best dividing line.

As one of the potential uses of the method is a pre-filter to select pairs of proteins for further analysis, it is of interest to plot the number of FP against FN while lowering the dividing line (Fig. 4*b*). It can be seen that 99.9% accuracy (specificity) can be attained with only 10% error (in FP), whereas to avoid completely the loss of true pairs results in the admission of

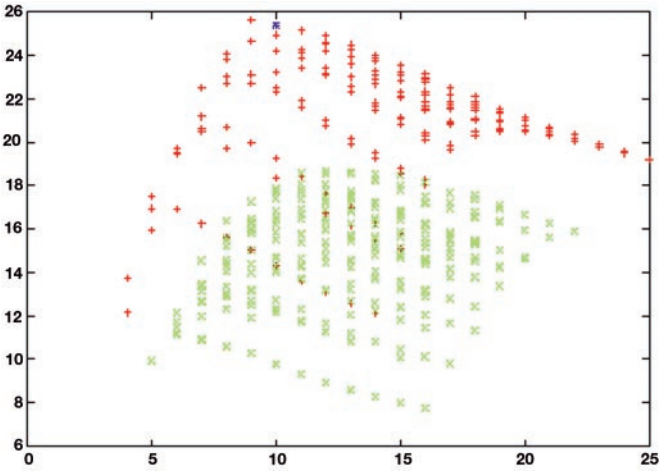

FIG. 5. **Match of chemotaxis-Y against the ideal forms.** The native structure of the chemotaxis-Y protein (PDB code 3chy) was reduced to a stick model and matched against the idealized stick structures (forms) representing the three-layer ($\alpha\beta\alpha$) class (*red diamonds*) and the four-layer ($\alpha\beta\beta\alpha$) class (*green crosses*). The score of the match is plotted against the number of secondary structures in the form (N), normalized for size by $\sqrt{N}$. The largest form to be matched had six helices above and below a 13-stranded sheet (6–13-6; $n = 25$), but in practice such forms that are clearly much bigger than the native probe can be avoided. The form corresponding to the native structure (2–5-3) is marked as a *blue asterisk*. The smaller form that scores slightly higher is the 2–5-2 form, corresponding to the core of the protein.

650 false-positives (1/8 of all false pairs). Although this might seem large, it is just over 1 min of calculation time.

Protein Structure Classification—One of the clear applications of a fast structure comparison method is the pairwise comparison of all known structures (21). This results in lists of similarities that can be ordered partially into a tree structure. However, because so many of the similarities are effectively random, the data do not conform to a metric space, and the resulting trees (or other visualizations) do not provide a unique overall representation of the data. This problem has been approached recently in a different way by using idealized protein structure representations (stick models) that can be arranged in something like the periodic table of elements (22). These ideal protein forms, however, must accommodate many arrangements of SSEs and are therefore very numerous (currently over 12,000) and still require a fast-matching algorithm. Their comparison with native protein stick structures is suited ideally to the current SMA method as, initially, the sequential order of the SSEs is not considered (23).

Each native protein structure (reduced to sticks) was matched against a collection of idealized forms, and the score obtained from the current algorithm was plotted against the number of sticks in the match. The results of this application are shown for the small $\beta\alpha$ chemotaxis-Y protein (PDB code 3chy) in Fig. 5. The native protein is matched against the set of forms with three secondary structure layers (α-helices packed on either side of a β-sheet) and the set with four layers

(α-helices packed on either side of a β-sandwich). The former class to which the native protein corresponds is clearly favored, and the ideal form corresponding to the native structure that has two helices above and three helices below a five-stranded sheet (2–5-3) scores second highest.

The top scoring forms were then passed to a more conventional alignment algorithm (24), and a root mean square deviation was calculated. When applied to a non-redundant sample drawn from the Protein Data Bank, the designation of the best fitting form (*e.g.* 2–5-3 for 3chy) provides an automatic classification for each protein structure (22).

CONCLUSIONS

The simple bipartite graph matching employed here has been shown capable of achieving a good separation of true intrafamily protein relationships from (false) interfamily relationships across a wide range of structural types and degrees of relatedness. The square root of the scores scales linearly with the geometric mean of the number of SSEs in a pair of proteins, allowing a simple function to be used to separate true and false matches.

The algorithm has been applied to the comparison of the PDB against a large number of idealized structures (forms) (22) and should also be useful as a pre-filter on more conventional large structure comparison problems. It will prove to be particularly suitable when these involve fake or decoy structures or large numbers of automatically generated models (25), because its focus at the SSE level makes it insensitive to conformational detail. A further application is to incorporate the algorithm into an existing heuristic algorithm to improve the selection of potentially equivalent SSEs that could then be refined at the more detailed residue level (24).

Acknowledgments—Darrell Conklin and Kjell Petersen are thanked for useful discussion, and Alex May and Jaap Heringa are thanked for comments on the manuscript.

* The costs of publication of this article were defrayed in part by the payment of page charges. This article must therefore be hereby marked "*advertisement*" in accordance with 18 U.S.C. Section 1734 solely to indicate this fact.

‡ To whom correspondence should be addressed. Tel.: 44-02089138552; Fax: 44-02089138545; E-mail: wtaylor@nimr.mrc.ac.uk.

REFERENCES

1. Brown, N. P., Orengo, C. A., and Taylor, W. R. (1996) A protein structure comparison methodology. *Comput. Chem.* **20,** 359–380
2. Eidhammer, I., Jonassen, I., and Taylor, W. R. (2000) Structure comparison and structure patterns. *J. Comput. Biol.* **7,** 658–716
3. Lathrop, R. H. (1994) The protein threading problem with sequence amino acid interaction preferences is NP-complete. *Protein Eng.* **7,** 1059–1068
4. Šali, A., and Blundell, T. L. (1990) Definition of general topological equivalence in protein structures: a procedure involving comparison of properties and relationship through simulated annealing and dynamic programming. *J. Mol. Biol.* **212,** 403–428
5. Taylor, W. R., and Orengo, C. A. (1989) Protein structure alignment. *J. Mol. Biol.* **208,** 1–22
6. Orengo, C. A., Brown, N. P., and Taylor, W. R. (1992) Fast protein structure comparison for databank searching. *Proteins Struct. Funct. Genet.* **14,** 139–167
7. Gibrat, J. F., Madej, T., Spouge, J. L., and Bryant, S. H. (1997) The VAST protein structure comparison method. *Biophys. J.* **72,** Meeting Proceedings 298
8. Mitchell, E. M., Artymiuk, P. J., Rice, D. W., and Willett, P. (1989) Use of techniques derived from graph theory to compare secondary structure motifs in proteins. *J. Mol. Biol.* **212,** 151–166
9. Grindley, H. M., Artymiuk, P. J., Rice, D. W., and Willett, P. (1993) Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism algorithm. *J. Mol. Biol.* **229,** 707–721
10. Koch, I., Kaden, F., and Selbig, J. (1992) Analysis of protein sheet topologies by graph theoretical methods. *Proteins* **12,** 314–323
11. Ullmann, J. R. (1976) An algorithm for subgraph isomorphism. *J. Assoc. Comput. Machinery* **23,** 31–42
12. Bron, C., and Kerbosch, J. (1973) Algorithm 457: finding all cliques of an undirected graph. *Comm. Assoc. Comput. Machinery* **16,** 575–577
13. Richmond, T. J., and Richards, F. M. (1978) Packing of α-helices: geometrical constraints and contact areas. *J. Mol. Biol.* **119,** 537–555
14. Cohen, F. E., Sternberg, M. J. E., and Taylor, W. R. (1981) Analysis of the tertiary structure of protein β-sheet sandwiches. *J. Mol. Biol.* **148,** 253–272
15. Kabsch, W., and Sander, C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* **22,** 2577–2637
16. Lee, B. K., and Richards, F. M. (1971) The interpretation of protein structures: estimation of static accessibility. *J. Mol. Biol.* **55,** 379–400
17. Taylor, W. R. (2001) Defining linear segments in protein structure. *J. Mol. Biol.* **310,** 1135–1150
18. Kuhn, H. W. (1995) The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **2,** 88–97
19. Sedgewick, R. (1990) *Algorithms in C.* Addison-Wesley, New York, p. 495
20. Mizuguchi, K., Deane, C. M., Blundell, T. L., and Overington, J. P. (1998) HOMSTRAD: a database of protein structure alignments for homologous families. *Protein Sci.* **7,** 2469–2471
21. Holm, L., and Sander, C. (1997) Dali/FSSP classification of three-dimensional protein folds. *Nucleic Acids Res.* **25,** 231–234
22. Taylor, W. R. (2002) A periodic table for protein structure. *Nature* **416,** 657–660
23. Taylor, W. R. (2000) Searching for the ideal forms of proteins. *Biochem. Soc. Trans.* **28,** 264–269
24. Taylor, W. R. (2002) in *Bioinformatics and Genome Analysis* (Seidel, H., Mewes, H.-W., and Weiss, B., eds) Vol. 38, pp. 133–148, Springer-Verlag, Berlin/Heidelberg, Ernst Schering Research Foundation Workshop
25. Taylor, W. R., May, A. C. W., Brown, N. P., and Aszódi, A. (2001) Protein structure: geometry, topology and classification. *Rep. Prog. Phys.* **64,** 517–590