A Study on Tools and Algorithms for 3-D Protein Structures Alignment and Comparison

Ying-Hung Lin Hsun-Chang Chang Yaw-Ling Lin

Department of Comput. Sci. and Info. Management, Providence University,
200 Chung Chi Road, Shalu, Taichung County, Taiwan 433.
{g9234020,hcchang}@cs.pu.edu.tw,yllin@pu.edu.tw

Abstract

Since protein structure is well conserved over evolutionary time, it therefore provides the opportunity to recognize homology that is undetectable by sequence comparison, and it represents a powerful means of discovering functions. In addition, the three-dimensional structure of a protein can yield direct insight into its molecular mechanism. Currently, there are several techniques available in attempting to find the optimal alignment of shared structural motifs between two proteins.

In this paper, we propose novel distance/similarity measurements and algorithms for pairwise alignment of protein structures. Methods of locating suitable isometric transformations of one structure, and align it to the other are addressed. Our methods allow sequence gaps of any length, reversal of chain direction, and free topological connectivity of aligned segments. Sequential connectivity can be imposed as an option. The method is fully automatic to identify structural resemblances and common structural cores accurately and sensitively, even in the presence of geometrical distortions.

Keywords: Bioinformatics, Structural genomics, Algorithms, Structure alignments and comparisons.

1 Introduction

The three dimensional structure of proteins is highly conserved during evolution [4]. Proteins are constructed by one or more polypeptide chains that fold into complicated 3D structures. In order to recognize the function of proteins, we can obtain insights by means of structures comparison. Detection of proteins with a similar fold can suggest a common ancestor, and often a similar function [6]. Comparison of 3D structures makes it possible to establish distant relationships, even between protein families distinct in terms of sequence comparison alone. This is why structural alignment of proteins increases our understanding of more distant evolutionary relationships [3]. The link between structural classification and sequence families enables us to study functions of various folds, or whole proteins.

Protein structure alignment techniques have grown increasingly important as a means to quantitatively compare and classify all known protein structures. The number of structures in the Protein Data Bank is currently (as of Aug 2004) more than 26,711 [2]. One of the primary goals of structural alignment programs is to quantitatively measure the level of structural similarity between all pairs of known protein structures. This data can provide several meaningful insights into the nature of protein structures and their functional mechanisms. For instance, the comparison of all structures against each other can show relationships, both functional and structural, between proteins that were previously not known to be related [11].

In addition, structure based distance measures are critical to constructing accurate phylogenies of proteins and classifying structures into families that share similar folds or motifs. Identifying these shared structural motifs using structural alignment techniques can provide significant insight into the functional mechanisms of the protein family. There have been several methods proposed to compare protein structures and measure the degree of structural similarity between them. These methods have been based on alignment of secondary structure elements as well as alignment of intra and inter-molecular atomic distances [1, 8, 10].

Dynamic programming techniques for 1D-base sequence comparisons such as Needleman-Wunsch algorithm [16] and Smith-Waterman algorithm [21] are usually applied to find a structural alignment for two 3D protein chain structures using various heuristics.

There have been several methods proposed to compare protein structures and measure the degree of structural similarity between them. The basic idea is , first, rapid identification of pair alignments of secondary structure elements, clustering them into groups, and scoring the best substructure alignment. The first one methods (VAST) is based on continuous distribution of domains in the fold space. Second method FSSP/DALI provides two levels of description - a coarse-grained one and one with a fine-grained resolution. Third method CATH provides the complete PDB fold classification by domains and links to other sources of information. The last two methods (CE and LGscore2) are based on a different idea. They focus on the local geometry rather than global features such as orientation of secondary structures and overall topology (as in the case of VAST or DALI) [5, 9, 12, 17, 20].

In this paper, our objective is to calculate the significance of score (rmsd) between spatial arrangements of Cα atom of protein backbone that are not necessarily adjacent in sequence. We use a idea of matching of Cα atom between them. To find the best match by the continuously perturb. Finally, we can obtain a lower score (rmsd).

2 Method

In this paper, we propose a novel distance / similarity measurement and algorithm for pairwise alignment of protein structures. We first propose a novel superposition distance measurement between two given structure, and then describe an algorithm in calculating the similarity. Secondly, methods of locating suitable isometric transformations of one structure, and align it to the other are addressed. In our methods of finding suitable isometric transformation, we use Monte Carlo procedure to pick up suitable initial setting. Our method allows sequence gaps of any length, reversal of chain direction, and free topological connectivity of aligned segments. Sequential connectivity can be imposed as an option. The method is fully automatic and identifies structural resemblances and common structural cores accurately and sensitively, even in the presence of geometrical distortions.

2.1 Protein (molecular) structure distances, similarities, and scoring functions

We briefly explain the idea of the smallest root mean squared deviation (rmsd). The idea is to align atom vectors of the two given (molecular) structures, and use the common least averaged squared errors as a measurement of differences between these two (paired) sequences. The rmsd fitting is a kind of least-squares fitting method for two sequences of points, and was developed by several persons independently [18].

Let $P = \{p_1, \ldots, p_n\}$ and $Q = \{q_1, \ldots, q_n\}$ be two sequences of points. We assume that P is translated so that its centroid $(\frac{1}{n}\sum_{k=1}^{n} p_k)$ is at the origin. We also assume that Q is translated in the same way. For each point or Vector x, $(x)_i (i = 1, 2, 3)$ denotes the i-th (X,Y,Z) coordinate value of x, and $\|x\|$ denotes the length of x. Let $d(P, Q, R, \mathbf{a}) = \sqrt{\frac{1}{n}\sum_{k=1}^{n} \|Rp_k + \mathbf{a} - q_k\|^2}$ where R is a rotation matrix and $\mathbf{a}$ is a translation vector. Then, the *rmsd* value $d(P, Q)$ between P and Q is defined by $d(P, Q) = \min_{R,\mathbf{a}} d(P, Q, R, \mathbf{a})$.

$d(P, Q, R, \mathbf{a})$ is minimized when $\mathbf{a} = 0$ and $R = (A^t A)^{\frac{1}{2}} A^{-1}$ where the matrix $A = (A_{ij})\; i, j = 1, 2, 3$ is given by $A_{ij} = \sum_{k=1}^{n} (p_k)_i (q_k)_j$, $A^{\frac{1}{2}} = B$ means $BB = A$, and $\mathbf{o}$ denotes the zero vector [19]. Thus, $d(P, Q)$, R and $\mathbf{a}$ can be computed in $O(n)$ time, where $O(f(n))$ time means that the computation time is at most $C \cdot f(n)$ for some constant C.

Note that there must be an atom-pairing scheme before one can do the *rmsd* computation. The first atom of the first selection is compared to the first atom of the second selection, fifth to fifth, and so on. Usually, most existed protein alignment algorithms use *rmsd* to calculate the averaged squared different distances between Cα atoms of two protein backbones. Through *rmsd*, we can find the similarity between two protein structures. The *rmsd* algorithm is used by VAST,CE, and many other packages as the final refined measurement step. The trick, though, is how these algorithms to identify the suitable paired atoms selected from the two given structural elements.

One reasonable way of defining the distance measurement (similarity) between two given structures is to find the best *rmsd* that fits between the smaller protein and a subset of the larger protein with the same number of residues. For example, if the smaller protein has n residues, the optimal alignment is defined by finding a subset (subsequence) of n residues in the larger structure such that the (minimum) rmsd between the smaller structure and the subsequence of the larger structure is minimized. The main difficulty of this method is that there are exponential ways to choose n residues in the larger structure. Furthermore, since the n chosen residues are unordered, it may be necessary to exploit as many as $n!$ different permutations before a correct rmsd alignment (with minimum deviation) is found.

In this paper, we propose a more computationally feasible solution for the similarity measurement. We use the geometric projective method by projecting the 3D atoms into three orthogonal 2D image planes, namely, the xy-plane (equation: $z = 0$), xz-plane ($y = 0$), and yz-plane ($x = 0$). In the following, we define the set difference measure of two set of points $P = \{p_1, \ldots, p_n\}$ and $Q = \{q_1, \ldots, q_m\}$ on the xy-plane, the other two plane measurements follow accordingly. Without loss of generality we will assume that $n \leq m$. Intuitively, the set P represents a smaller molecular structure while Q represents a larger structure.

```
$\Delta(P,Q)$.
Input: Two set of points $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}; n < m$
Output: The superposition distance between $P$ and $Q$.
for each $p$ in $P$, $q$ in $Q$, do $w(p) \leftarrow 1; w(q) \leftarrow 1; w \leftarrow 0$;     $\triangleright$ The canceled total weights.
    for each point $q$ in $Q$ do
        while $t \neq 0$ do
            Let $p$ in $P$ be nearest point of $q$, such that $d_i \leq d(p,q) < d_{i+1}$;
            $t \leftarrow min\{w_i; w(p), w(q)\}$;
            $w \leftarrow w + t; w(p) \leftarrow w(p) - t; w(q) \leftarrow w(q) - t$;
            Remove $p$ from $P$ if $w(p) = 0$;
Return $m + n - 2w$ as the superposition distance, $\Delta(P,Q)$
```

Figure 1: The algorithm for computing the superposition distance $\Delta(P,Q)$.

```
ALIGN$(P, Q \cdot R_0)$
Input: Two set of points $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}; n < m$ $R_0$:
       an initial rotation transformation of $P$.
Output: a good isometric transformation $T$ for aligning structures $P$ and $Q$.
 1  Translate points of $P$ and $Q$ such that each of their mass centers after
    the translation becomes the origin point $(0,0,0)$.
 2  Rotate: $P \leftarrow R_0 \circ P; T \leftarrow R_0 \circ$ "the translation";
 3  $w \leftarrow \Delta(P,Q)$;
 4  Repeat the following 5 to 9 until the superposition converges.
 5  For each one of the three different orthogonal planes rotations, find the
    best rotation angle (pivoted at the center of the mass).
 6  Let $\theta_x$ be the good angle that rotates $P$ around the $x$-axis (i.e., along
    the $yz$-plane) such that superposition distance $\Delta(R(\theta_x) \circ P, Q)$ after
    the rotation is sufficiently smaller than $\Delta(P,Q)$. The other two angles
    $\theta_y$ and $\theta_z$ is also defined similarly.
 7  $\theta \leftarrow$ arg min $\{\Delta(R(\theta_x) \circ P, Q), \Delta(R(\theta_y) \circ P, Q), \Delta(R(\theta_z) \circ P, Q)\}$;
 8  $P \leftarrow R(\theta) \circ P; T \leftarrow R(\theta) \circ T$;
 9  Repeat 5 to 9 if $\Delta(R(\theta_z) \circ P, Q)$ is smaller than $\Delta(P,Q)$; otherwise,
    exit to 10.
10  Return the transformation $T$ and superposition distance,$\Delta(P,Q)$.
```

Figure 2: Finding a good isometric transformation T to align structures P and Q.

2.2 Finding a suitable rigid transformation for matching structures

A point q in Q is *canceled* by a point p in P if they are relatively closed to each other. Specifically, we choose some suitable parameter k and define a list (an array) of *canceling distances* $[d_0, d_1, \ldots, d_k]$ and a list of *canceled fractions* $[w_0, w_1, \ldots, w_k]$ such that if two molecular with distances between d_i to d_{i+1} $(d_i \leq d < d_{i+1})$ then the collided molecular is canceled by a fraction of w_i. It follows that the difference between P and Q on the xy-plane, denoted by $\Delta_z(P,Q)$, is the total remained *un-canceled* molecular weights after the cancellation of points in P and Q. The notation of $\Delta_y(P,Q)$ and $\Delta_x(P,Q)$ follow accordingly. Finally, the superposition distance between P and Q is just the t-norm distance defined by:

$$\Delta(P,Q) = [\Delta_x(P,Q)^t + \Delta_y(P,Q)^t + \Delta_z(P,Q)^t]^{\frac{1}{t}}$$

for some suitable chosen parameter t. Here we propose an algorithm for computing the superposition distance $\Delta(P,Q)$ in the Figure 1.

Also, since many molecular biology researchers prefer using the similarity or scoring function in measuring the relationship between two molecular structures, here we mention that there is a general way of converting the distance function $\Delta(P,Q)$ can be made by defining two adjustable parametric constants a and b such that the scoring (similarity) function

$$S(P,Q) = \frac{a}{b + \Delta(P,Q)}$$

Note that a smaller distance of $\Delta(P,Q)$ results in a higher score S(P,Q), while a larger distance causing a lower score, and vice versus.

With the similarity/difference function, $\Delta(P,Q)$, at hand, we need a method to iteratively find a good superposition, by using 3D isometric transformation (rotation + translation), of two structures, such that the

```
PERTURB($P, \theta, t$)
Input: A set of points $P = \{p_1, p_2, \ldots, p_n\}$;$\theta = (0..2\pi)$;
    the rotation axis $t \in \{x|y|z\}$.
Output: The set of points after rotation, and the rotation matrix defined by $(\theta, t)$.
R ← the matrix rotating $\theta$ angle along the $t$-axis ; $P' \leftarrow \emptyset$
for each $p$ in $P$ do $P' \leftarrow P' \cup \{R \circ p\}$
return $(P', R)$.
```

Figure 3: The algorithm for perturbing the set of points P.

resulting structures have sufficiently low $\Delta(P, Q)$ difference. Here we propose an algorithm for performing suitable isometric transformation T between structures P and Q such that the resulting superposition distance $\Delta(T \circ P, Q)$ is sufficiently low; the algorithm is shown in Figure 2.

This iterative algorithm is seeded with an initial superposition that is based on translating both mass centers to the origin point and an initial rotation transformation R_0. A custom-based initially seeded position is also possible. We briefly explain possible heuristics for finding a good rotation angle θ_x. The method of finding θ_y and θ_z also follows symmetrically. Note that the projected points of P to the yz-plane are just n 2D points. One possibility is to find the two least-squares regression lines for both P and Q and align (rotate) points of P accordingly. Another possibility is to use the exponential jumping method. Given a minimum angle $\theta_{\min}$, a maximum angle $\theta_{\max}$, and a positive real ratio r the algorithm will try all possible $r^k\theta_{\min}$ for all nonnegative integer k until $r^k\theta_{\min} > \theta_{\max}$. The algorithm then picks an angle that minimizes $\Delta(R(\theta_x) \circ P, Q)$.

To avoid that an ill-chosen initial transformation might lead to a local maximal solution and miss some better alignment, we use Monte Carlo procedure to pick up several different initial settings of initial rotation R_0's such that better results might be chosen. That is,

$R^* = \arg\min_R \{$ALIGN$(P, Q \cdot R)|$ several randomly picked rotation $R\}$;

ALIGN(P, Q) =ALIGN$(P, Q \cdot R^*)$

Finally, we can do the final refinement by utilizing the RMSD procedure to fine-tune the final result. Let T^* be the isometric transformation obtaining ALIGN$(P, Q \cdot R^*)$, $P' = T \circ P$, and Q being translated to Q' such that the mass center of Q' is located at the origin. We construct a weighed graph $G = (V, E)$ with V being labelled with points of P' and Q', and each (p, q) in E being weighted some scoring function of the Euclidean 3D distance, for example, $w(p, q) = a/(b + \|p, q\|)$ for some parameters a and b. We then solve the weighted maximum matching problem [7] to obtain the best matching of P' and Q'. After the matched pairings, we perturb and refine the final alignment by applying the algorithm MB-ALIGN and PERTURB to obtain lower *rmsd*, the algorithm is show in Figure 3 and Figure 4.

Note that MB-ALIGN uses PERTURB to rotate the Cα atoms of a protein, and performs minimum weighted bipartite matching to find good choices of atoms pairing between two structures before performing the refinement RMSD. First, the algorithm MB-ALIGN constructs a set S containing a set of configurations of the points set. Atoms of the structure are rotated along each axle (x-axis, y-axis, or z-axis) to several set of groups of set points; each rotated group is called a *seed*, s; the set of seeds is denoted by the set S. Note that each s in S is a structure containing a real value rms$[s]$ and a rotation matrix mat$[s]$. The algorithm then perturb each s in S by rotating six directions by the PERTURB procedure. For each perturbed seed, MB-ALIGN finds the minimum bipartite matching, MBM, to decide the points pairing between point sets. Once it observes an improved seed s (smaller rms$[s]$), the seed is then put back to S. The algorithm stops either when a sufficiently small rms$[s]$ is observed or when no further improvement is possible.

3 Preliminary Experiments and Result

To validate our method, we have implemented the algorithms as several independent C programs to perform experiments. Given a set of points, P, we use programs to rotate and translate them into another set P'. We then use our independent system (not knowing how the original set was perturbed) to find the best structural alignment between P and P'. In implementing our system, we adapt the LEDA [15] package system to perform the minimum weighted bipartite matching. The algorithm of maximum bipartite matching is implemented by Dijkstra's algorithm and heuristic method. In the worst case, the time complexity of this algorithm is $O(n(m + n \log n))$ [15]. Furthermore, we make use of the open source licensed software, ProFit [13], to calculate root mean square deviation. ProFit is designed to be the ultimate protein least squares fitting program, there are now some 1300 registered users around the world. It has many features including flexible specification of fitting zones and atoms, calculation of *rms* over different zones or atoms, and *rms*-by-residue calculation. Fitting is implemented by using the McLachlan algorithm [14].

```
MB-ALIGN$(P, Q, F, \tau)$.
Input: Two set of points $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}; n < m$
    The threshold $\tau$ is a real number, and a flag $F$ is integer.
Output: a sufficiently low rmsd and its corresponding rotation matrix $R$.
$\theta_{\text{Max}} = \frac{2\pi}{10}$ and create a empty structure queue S.
        ▷ Each $s$ in $S$ contains a real number rms$[s]$ and a rotation matrix mat$[s]$.
Create two array $P'$and $R$.        ▷ $P'$ is a temp set of points, and $R$ is a matrix.
for $i \leftarrow 1$ to 10 do
    for each $t$ in $\{x, y, z\}$ do        ▷ Placing seeds by rotation along $x$-axis, $y$-axis and $z$-axis.
        $(P', R) \leftarrow$ PERTURB$(P, i \cdot \theta_{\text{Max}}, t)$; $L \leftarrow$ MBM$(P', Q)$; rms$[s] \leftarrow$ RMSD$(L)$;
        mat$[s] \leftarrow R$; $S \leftarrow S \cup \{s\}$;
$C \leftarrow 0$        ▷ Initializing the counter.
while $C \leq F$ do
    for each $s$ in $S$ do
        Let $r$ be a real value picked uniformly random from $(0..1)$.
        Let $\theta_{\text{Adj}} = \theta_{\text{Max}}\cdot$ rms$[s]$ / rms$[Adj]$        ▷ The rms$[Adj]$ is determined by empiricism.
        for each $t$ in $\{x, y, z\}$ do
            $(P', R) \leftarrow$ PERTURB$(P\circ$ mat$[s], r \cdot \theta_{\text{Adj}}, t)$; $L \leftarrow$MBM$(P', Q)$;
            if rms$[s] >$ RMSD$(L)$ then rms$[s] \leftarrow$ RMSD$(L)$; mat$[s] \leftarrow R \circ \text{mat}[s]$;
            $(P', R) \leftarrow$ PERTURB$(P\circ$ mat$[s], r \cdot -\theta_{\text{Adj}}, t)$; $L \leftarrow$MBM$(P', Q)$;
            if rms$[s] >$ RMSD$(L)$ then rms$[s] \leftarrow$ RMSD$(L)$; mat$[s] \leftarrow R \circ \text{mat}[s]$;
        $C \leftarrow C + 6$
        if rms$[s] \leq \tau$ then return rms$[s]$ and mat$[s]$.
return the minimum rms$[s]$ and mat$[s]$ from $S$.

MBM$(P, Q)$        ▷ Finding the minimum minimum bipartite matching of two points sets.
Input: Two set of points $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}; n < m$
Output: The minimum bipartite matching of $P$ and $Q$, encoded in the list $L$.

RMSD$(L)$        ▷ Finding the minimum root mean squared deviation of two ordered sets of points.
Input: An ordered list L.
Output: The minimum root mean square deviation of $L$.
```

Figure 4: Aligning two sets of atoms with low *rmsd* by pairing points according to the maximum bipartite matching measurement.

We perform our experiments as the following. First, a points set, P, of size varying from 50 to 1,000 are randomly generated as the tested case. The point set P is then rotated and translated randomly to another set Q. The idea is then to use our structure alignment system to find the suitable reversed transformation so that the resulting *rmsd* $\simeq 0$ or at least sufficiently small.

To fine-tune the structure alignment system, several experiments have been done. For example, to figure out a better Monte Carlo strategy in perturbing the seeds, we adapt two slightly different approaches. One is that all seeds in S shares one single random dice (r), while the other is to let each seed having its own private (local) dice. Note that the rotation angle will be adjusted in accordance with the rms$[s]$ value. The experimental results is shown in Figure 5. Furthermore, we also compare the differences of the performance of the system when the number of seeds in consideration are varied; the average required rotation numbers under different seeding conditions is illustrated by the table shown in Figure 6.

References

[1] D.W. Barakat and P.M. Dean. Molecular structure matching by simulated annealing, iii. the incorporation of null correspondences into the matching problem. *J. Comp. Aided Mol. Design.*, 5:107–117, 1991.

[2] F.C. Bernstein, T.F. Koetzle, Williams G.J.B., Meyer E.F.Jr., M.D. Brice, J.R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi. The protein data bank: A computer based archival file for macromolecular structure. *J. Mol. Biol.*, 112:535–542, 1997.

[3] J.M. Bujnicki. Phylogeny of the restriction endonuclease-like superfamily inferred from comparison of protein structures. *J Mol Evol.*, 50:38–44, 2000.

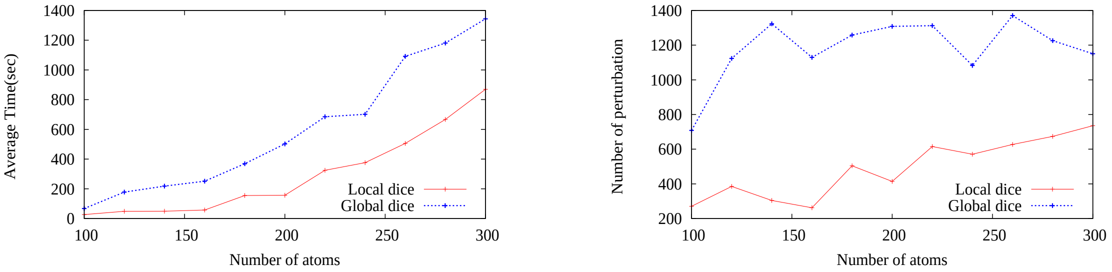

Figure 5: The average execution time and number of perturbations for alignment of two structures.

Seeds #	*12*	*18*	*24*	*30*	*36*	*42*	*48*
rotation #	832.60	759.20	751.80	873.20	979.00	1078.73	970.40
deviation	517.34	346.44	474.89	520.81	636.49	588.52	582.77

Figure 6: The average rotation numbers under different numbers of seeds for the moderate-sized (154 Cα atoms) sperm whale myoglobin F46V n-butyl isocyanide with Protein Data Bank (PDB) code 101M.

[4] C. Chothia and A.M. Lesk. The relation between the divergence of sequence and structure in proteins. *EMBO J.*, 5:823–826, 1986.

[5] S. Cristobal, A. Zemla, D. Fischer, L. Rychlewski, and A. Elofsson. A study of quality measures for protein threading models. *BMC Bioinformatics*, 2:5, 2001.

[6] S. Dietmann and L. Holm. Identification of homology in protein structure classification. *Nature Struct. Biol.*, 8:953–957, 2001.

[7] Z. Galil. Efficient algorithms for finding maximum matching in graphs. *ACM Computing Surveys*, 18:1:23–38, 1986.

[8] M. Gerstein and M. Levitt. Using iterative dynamic programming to obtain accurate pair-wise and multiple alignments of protein structures. *In Proc. Fourth Int. Conf. on Intell. Sys. for Mol. Biol.* Menlo Park, CA: AAAI Press, pp 59-67, 1996.

[9] J.F. Gibrat, T. Madej, and S.H. Bryant. Surprising similarities in structure comparison. *Curr Opin Struct Biol.*, 6:377–385, 1996.

[10] L. Holm and C. Sander. Protein structure comparison by alignment of distance matrices. *J. Mol. Biol.*, 233:123–138, 1993a.

[11] L. Holm and C. Sander. Structural alignment of globins, phycocyanins, and colicin. *FEBS Lett.*, 315:301–306, 1993b.

[12] L. Holm and C. Sander. Touring protein fold space with dali/fssp. *Nucleic Acids Res.*, 26:316–319, 1998.

[13] A.C.R. Martin. `http://www.bioinf.org.uk/software /profit/`.

[14] A.D. McLachlan. Rapid comparison of protein structres. *Acta Cryst*, A38:871–873, 1982.

[15] K. Mehlhorn and St. Naher. *The LEDA Platform of Combinatorial and Geometric Computing*. Cambridge University Press, 1999.

[16] S.B. Needleman and C.D. Wunsch. A general method applicable to the seach for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, 48:443–453, 1970.

[17] C.A. Orengo, A.D. Michie, S. Jones, D.T. Jones, M.B. Swindells, and J.M. Thornton. Cath - a hierarchical classification of protein domain structures. *Structure*, 5:1093–1108, 1997.

[18] S.T. Rao and Rossmann M.G. comparison of super-secondary structures in proteins. *J. Molecular Biology*, 76:241–256, 1973.

[19] J.T. Schwartz and M. Sharir. Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. *Int. J. Robotics Research*, 6:29–44, 1987.

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

[21] T.F. Smith and M.S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195–197, 1970.