Recursive Dynamic Programming for Adaptive Sequence and Structure Alignment

Ralf Thiele, Ralf Zimmer, Thomas Lengauer

GMD SCAI, Schloß Birlinghoven, P.O. Box 1316
D 53754 Sankt Augustin, Germany
Phone: +49 2241 14 2302, 2818, 2777 Fax: +49 2241 14 2656
email: {Ralf.Thiele, Ralf.Zimmer, Thomas.Lengauer}@gmd.de

Keywords:Recursive algorithm, protein threading, sequence alignment, structural alignment, adaptive combinatorial optimization, multiple alignment

Abstract

We propose a new alignment procedure that is capable of aligning protein sequences and structures in a unified manner. Recursive dynamic programming (RDP) is a hierarchical method which, on each level of the hierarchy, identifies locally optimal solutions and assembles them into partial alignments of sequences and/or structures. In contrast to classical dynamic programming, RDP can also handle alignment problems that use objective functions not obeying the principle of prefix optimality, e.g. scoring schemes derived from energy potentials of mean force. For such alignment problems, RDP aims at computing solutions that are near-optimal with respect to the involved cost function and biologically meaningful at the same time. Towards this goal, RDP maintains a dynamic balance between different factors governing alignment fitness such as evolutionary relationships and structural preferences. As in the RDP method gaps are not scored explicitly, the problematic assignment of gap cost parameters is circumvented. In order to evaluate the RDP approach we analyse whether known and accepted multiple alignments based on structural information can be reproduced with the RDP method. For this purpose, we consider the family of ferredoxins as our prime example. Our experiments show that, if properly tuned, the RDP method can outperform methods based on classical sequence alignment algorithms as well as methods that take purely structural information into account.

Introduction

Alignments are one–to–one mappings of elements of two or more structured sets of objects, such as sequences of amino acids, physico-chemical profile vectors, or 3D–coordinates of atoms. Dynamic programming algorithms are capable of finding optimal alignments, if the objective function (or scoring scheme) meets the principle of prefix–optimality. This means that optimal alignments of subsequences (or structures) can be extended to optimal alignments of the original sequences (or structures). Classical scoring models for sequence alignment meet this requirement.

This research is supported by the German Ministry for Research and Technology (BMBF) under grant number 413-4001-01 IB 301 A/1.

We are interested in threading a protein sequence into a protein structure. *Threading* or *sequence–structure alignment* is a version of the alignment problem that maps a sequence onto a structure and estimates the fitness of the resulting conformation. In order to be accurate in this process, we have to account for spatial interactions between residues or their atoms, explicitly. Incorporating such interactions leads to scoring schemes that violate the principle of prefix–optimality. This problem can be circumvented by encoding the structural information on the environment of each sequence position in the form of one–dimensional profiles (Bowie, Lüthy, & Eisenberg 1991). The advantage of such a coarse description is that standard dynamic–programming algorithms can be used. A more precise description of the structure entails an at least two–dimensional representation, e.g., a contact matrix, describing interacting residues. As an approximation, Jones, Taylor, & Thornton (1992) use a two–level dynamic–programming scheme (1989) for doing sequence–structure alignment. On the first level their approach takes pairwise interactions weighted by virtual energy functions (e.g., Sippl (1990)) into account, which rate the distance dependent preference of residue pairs forming interactions. Lathrop & Smith (1994) describe a branch&bound method for the optimal solution of the full fragment threading problem with such pairwise interaction scores.

For a formally defined version of the threading problem, Lathrop (1994) proved that the problem is NP–hard, if we allow for variable–length gaps in the alignment and model nonlocal effects with pairwise interaction terms in the cost function. All such formal definitions of the threading problem incorporate a large set of parameters whose setting makes or breaks the method. As a rule, these parameters entail a sizeable error margin, because they are derived by statistics over the sequence and structure databases. Therefore, finding the exact global optimum has to stand back behind the goal of making the method adaptive to the delicate signals that have to be found in the biological data and robust to the statistical inaccuracies that pop up along the way. For this reason, we chose to develop a method that intelligently adapts to the changing priorities with which different fitness factors contribute to the overall cost of an alignment. Two examples for

such fitness factors are evolutionary sequence homology and structural preferences, e.g., potentials of mean force.

The hierarchical RDP method presented here computes near–optimal solutions with respect to the involved cost function by focusing on highly conserved regions with highest priority and considering matching/contacting regions on lower levels of the hierarchy. The procedure starts by identifiying sequence segments entailing most significant matches in the current problem instance. By removing these matched segments from the current sequences, the problem is split into disjoint subproblems. In order to avoid severe mistakes caused by mismatched alignment portions computed early on—which can easily happen, due to lack of information—the method allows for handling sets of a limited number of alternative subalignments on each level of the hierarchy. This recursive divide&conquer procedure terminates once the match score reaches the overall noise level. Therefore, only a limited region of the solution space, characterized by near–optimal alignment scores in significant regions of the alignment, is searched. By leaving the gaps as the rest, which cannot be aligned at all, the RDP method has the additional advantage that gaps need not be scored explicitly. Scoring gaps is quite problematic, because gap terms are not included in most scoring potentials. The RDP method has been implemented as part of the ToPLign system (**To**olbox for **P**rotein **Align**ment) (Mevissen *et al.* 1994) developed at GMD.

Our first experiments on biological data show that this effective limitation of the search through the solution space does not prevent our method from coming up with biologically meaningful solutions of the threading problem. Even more, based on a mix of sequence and structure information the RDP method is able to align sequences on which both exclusively sequence–based methods as well as exclusively structure–based methods fail. To our understanding, the essential element of this performance is that we trade the goal of exact optimization with an adaptivity of the algorithm to the varying significance of different criteria contributing to alignment fitness instead of optimizing only one fitness criteria.

The paper is organized as follows: The first section puts the different versions of alignment problems into a uniform formal setting. The following section describes the general RDP method. The third section discusses variants of letting sequence and structure information guide the alignment process. As a running example, we discuss ferredoxins, which are proteins containing iron–sulfur ([Fe–S]) clusters, but otherwise are quite diverse in sequence and structure. Because of their importance in basic metabolic pathways and of their ubiquitous occurrence in a number of organisms including algae, plants, and archaebacteria ferredoxins are quite old and have undergone substantial evolutionary changes in sequence and structure. These changes, on the one hand, and a striking sequence and structural conservation in connection with the coordination of the [Fe–S]–cluster as well as the protein's electron transfer function, on the other hand, make this protein family an interesting object for all kinds of alignments. The last section interprets some RDP alignments of ferredoxins and presents some preliminary analysis of RDP alignments by comparing them to alignments from different sequence and structure alignment databases.

Formal background

For the context of this paper, we define an alignment formally as a homomorphism f between two ordered or otherwise structured sets A and B. This setup includes sequence alignment, as well as sequence–structure alignment (threading) and structure comparison. A *structured set* (e.g., a sequence, a tree, a graph) is a set of *elements* (residues, letters, vertices) with a binary relation on the elements. An *alignment* is a homomorphism, e.g., a relation–preserving partial injective mapping between two structured sets. In this terminology, sequence–structure alignment is a homomorphism between two protein sequences, whose elements are ordered by their position in the sequence. In addition, the image sequence has an associated protein structure, which is represented by an additional relation indicating spatial adjacency between residues. By the homomorphism, this relation is imposed on the pre–image sequence. An example for such a homomorphism between a sequence A and a structure B is shown in figure 1. Thus, the alignment problem can be formalized as follows:

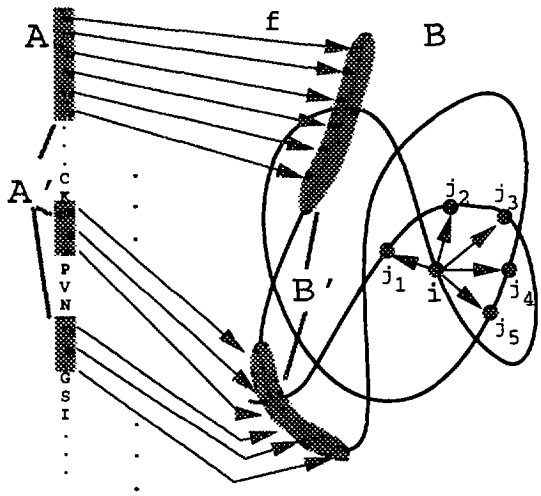


Figure 1: A sequence-to-structure alignment. The arrows emanating from residue i in the structure B denote interactions to spatially close residues in B.

Given two sets A and B with relations $R_A \subseteq A \times A$ and $R_B \subseteq B \times B$,
find an injective, partial homomorphism $f : A \rightarrow B$ ($f : A' \rightarrow B'$ bijective),
which optimizes a scoring function $\phi(f, A, B)$.

The scoring function ϕ is usually specified as a sum of the following functions on subsets:

$$\phi(f, A, B) = \mathrm{M}(f, A', B') + \mathrm{I}(f, A \setminus A') + \mathrm{D}(f, B \setminus B')$$

where M scores the matched parts $A' = \textit{pre-image}(f)$ and $B' = \textit{image}(f)$, I scores *insertions*, i.e., elements in $A \setminus A'$, and D scores *deletions*, i.e., elements in $B \setminus B'$. In the context of biological applications the I and D parts are usually realized as affine cost functions $gi + ge * k$, where the gap–insertion penalty gi is accounted for every gap, regardless of the length k, and the gap–extension penalty ge is charged for every insertion and deletion. The exact definition of the M–term depends on the specific kind of problem to be dealt with. For classical sequence alignment, this term reduces to the Dayhoff–matrix (Dayhoff 1978), for threading, it consists of distance–dependent energy terms (Sippl 1990), for structure comparison it involves the rms distances of superposed elements.

Recursive dynamic programming

The main idea of the alignment approach proposed in this paper is to recursively decompose the problem into smaller subproblems. In order to do so, the algorithm uses a method for computing a reliable partial solution of the alignment problem. In a first step, we use classical local alignment algorithms based on dynamic programming, here. But, in general, we can use any algorithm for this purpose that computes a convincing subsolution. We call such an algorithm an *oracle*. The solution computed by the oracle splits the alignment problem into disjoint subproblems, each one pertaining to a portion of the problem that has not been matched yet by the oracle. We apply the oracle recursively to each of these subproblems, in order to extend the matched regions in the alignment. This process terminates if the oracle finds no more significant matches, i.e., subalignments whose scores rise above the overall noise level.

In order to avoid the known drawbacks arising from restricting the attention to optimal solutions and to circumvent the inaccuracies involved in the choice of parameters (e.g., pseudo–energies rather than real energies), we collect sets of suboptimal but near–optimal solutions for each of the subproblems under consideration. The size of such sets can be tuned to a suitable but limited value.

Figure 2 sketches the recursive decomposition of a sequence–alignment problem into subproblems and the assembly of the alignment from the solutions for these subproblems. For clarity, we show only one solution for each subproblem.

Each recursive refinement step of the RDP procedure expands the partial information currently available about the final mapping f to regions of the alignment problem that have not been matched yet. In the case of threading, the interactions between already aligned regions in the structure B and the rest of the structure can be analyzed in order to find additional

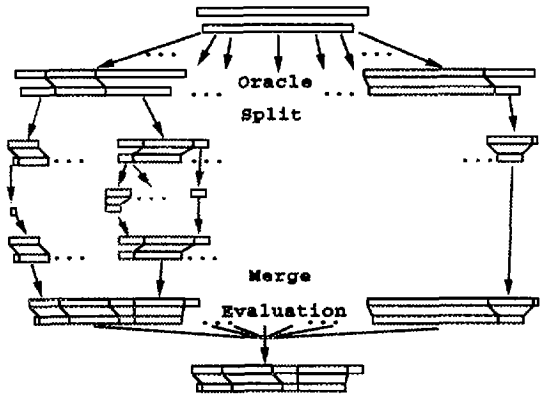


Figure 2: Diagrammatic sketch of the RDP method.

appropriate matches with the sequence A. Using the partial alignment in its reverse direction, we carry out this analysis inside the sequence A, rather than in B. This is a more accurate model than the frozen approximation (Godzik, Kolinski, & Skolnick 1992), which uses the structural setup of B for this purpose and does not take into account that residues of A are already mapped onto the interaction partner sites by f.

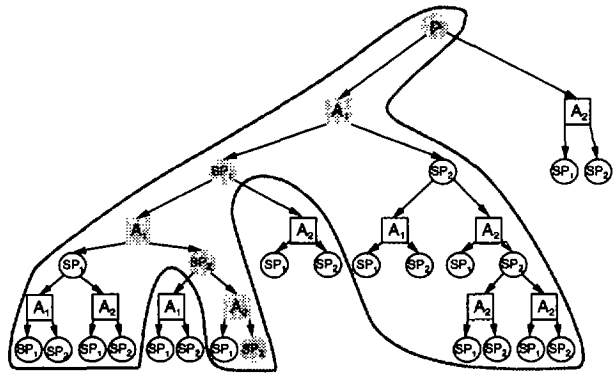


Figure 3: AND–OR tree representing the set of solutions. A subsolution for node SP_2 on the bottom level may depend on all of the encircled part of the tree.

The RDP procedure constructs a AND–OR–like solution tree as depicted in figure 3. The nodes P and SP_i (OR nodes) in the tree represent subproblems to be solved. The nodes A_i (AND nodes) represent solutions for the problem represented by their father node. As alternative (optimal or suboptimal) solutions are allowed for each subproblem, there are multiple subsolutions A_i connected via edges to their respective father node SP_i or P.

In general, the outcome of the method depends on the order in which the subproblems are solved, e.g., for threading a subsolution for node SP_2 on the bottom level in figure 3 may depend on the encircled part of the tree via spatial interactions in the structure. Thus the order of evaluation of the subproblems is an object of algorithm tuning. As usual for branch&bound methods, open subproblems are maintained in a priority queue. The criterion by which this queue is sorted determines the order of evaluation, e.g., breadth–first, or by decreasing size of the subproblem, or by increas-

ing expected score for a solution of the subproblem. A more sophisticated approach is to order the subproblems according to some appropriate notion of significance, to reevaluate this notion regularly, and to reorder the queue accordingly.

The implementation of the *top–down* recursive refinement phase depends on the type of information that is processed (e.g., pure sequence information, local structure, or interactions between residues) and on the partial solutions already computed on the same level and on upper levels. There are several possible levels of sophistication here: The simplest concept is to ignore dependencies on brother or ancestor subsolutions altogether. This option is chosen if the alignment is computed on the basis of the classical scoring schemes for sequence alignments.

On the other hand, the optimization of a pairwise interaction score necessarily implies dependencies on positions distributed all over the alignment. There are two types of such dependencies: *Backward* dependencies are depedencies between the solution of the currently solved subproblem and solutions of previously solved subproblems. In order to take backward dependencies into account, the algorithm needs to maintain not only the score but all subsolutions realizing this score. In a standard dynamic–programming setting this requirement exceeds available resources in time and space. *Forward* dependencies exist between the solution of the current subproblem and its extensions towards the overall solution. These dependencies may render currently suboptimal solutions optimal lateron in the algorithm and vice versa. Thus, forward dependencies not only pose even more serve problems w.r.t. resource demands but may even destroy the acyclicity of the dependency graph between the subproblems. By constructing the alignment not from the left or right ends of the sequences but from those segments for which significant and reliable alignments can be derived, the RDP method heuristically circumvents the problems occuring from backward as well as from forward dependencies. RDP considers backward dependencies only on reliable solutions which are artificially restricted in number via the number of allowed alternatives and the number of returned solutions after evaluation and ranking. Whether a dependency is backward or forward depends on the order of evaluation of the subproblems. By computing the significant parts of the alignment with highest priority forward dependencies only exist to subproblems, which pertain to less significant regions of the alignment. Therefore, the RDP method limits the mistakes made by neglecting forward dependencies to be within a certain range.

Different versions of the RDP procedure essentially differ in the kind of oracle that they use. In the following, we concentrate on oracles that also take structural information into account. For this purpose, we consider the following three types of oracle which differ in the degree to which structural information is exploited. For illustration, refer to Figure 4. This

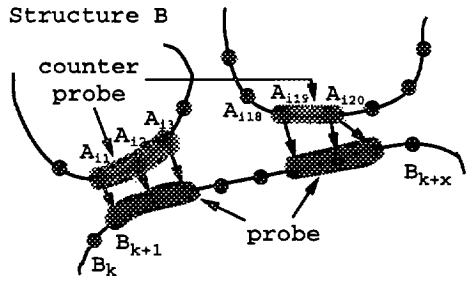


Figure 4: Probe and inverse probe alignment

figure shows a section of a protein structure B onto which already a few parts of sequence A (e.g., all nodes labelled with a A_i) have been mapped. The pairwise interactions within the structure are depicted by edges between the interacting residues. For clarity, we direct the edges from the residues in B that have been matched already to the residues in B that have not been matched yet. On the next level of the RDP procedure, we want to match the residues that are adjacent to already matched residues in B to appropriate regions in A. In the figure, these are residues $(B_{k+1}, \ldots, B_{k+3})$ and $(B_{k+x-3}, \ldots, B_{k+x-1})$. We call these residues a *probe* P_B. The oracles differ in the process, with which they find regions in A that match the probe P_B. The already matched residues in B that give rise to the probe P_B are called the *counterprobe*. In Figure 4, the counterprobe is composed of the residues $(A_{i1}, \ldots, A_{i3})$ and (A_{i19}, A_{i20}).

Oracle 1. Sequence probe: Find a match for the probe P_B on the basis of some kind of classical sequence alignment. This oracle does not evaluate any structure fitness and resides entirely on evolutionary similiarity but focusses selectively on residues interacting with more highly conserved regions.

Oracle 2. Structure probe: Find a match for the probe P_B based on the structural fitness to the counterprobe. We use interaction potentials in order to evaluate this fitness. This oracle does not consider any sequence homology measure.

In general we expect both sequence homology and structural fitness to contribute to the alignment score. Therefore, we consider the following oracle:

Oracle 3. Mixed probe: Find a match based on both sequence homology (sequence probe) and structure fitness (structure probe) with appropriate weights. Dynamically change the weights during the algorithm.

It is an essential factor of the efficiency of the RDP procedure that all of the above oracles can be computed using the extensions of the classical dynamic programming method implemented in ToPLign.

A tour through Delphi

In this section, we report on preliminary experiments that tell us what kind of information we should appropriately make available to the oracle. We discuss the different versions of alignment for different applications separately.

Sequence alignment

In order to be able to discriminate more between different sequence alignments than the typical Dayhoff–type score functions (Dayhoff 1978) are able to do, we use the concept of a *path–contour map*. The path–contour map is a matrix representation of the solution space for an alignment problem. Entry (i, j) in the matrix contains the score of the optimal alignment of the two sequences A and B that matches position A_i to position B_j. The path–contour map can be efficiently computed for all kinds of global and local alignments in time $O(mn)$ (Mevissen *et al.* 1994).

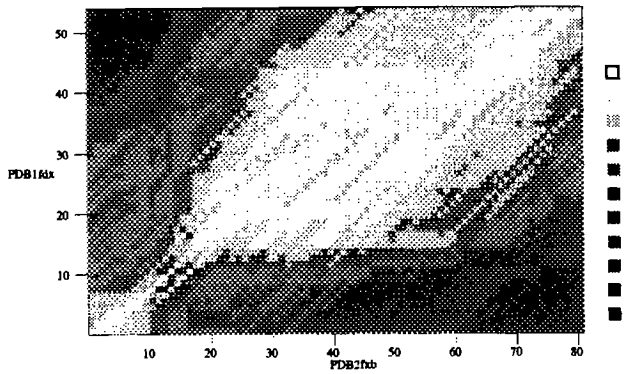


Figure 5: Global sequence alignment (2FXB vs. 1FDX)

A path–contour map can conveniently be visualized by colouring the matrix w.r.t. score (see Figures 5,8, and 9). The set of optimal alignments forms a set of ridges and/or plateaus coloured white in the path–contour map. The colour of a position in the path–contour map gets darker with decreasing score of the alignment containing the respective match. Therefore, near–optimal alignments form the light regions around the optimal alignments.

The path–contour map gives us information not only on suboptimal alignments but also on the reliability of different regions in the alignment (Mevissen *et al.* 1994; Mevissen & Vingron 1995). This information is vital to the oracles, especially to those that are used in the upper levels of the RDP procedure. Intuitively, the reliability of an alignment position is correlated with the gradient that occurs close to this positions in the path–contour map.

Sequence–structure alignment

For sequence–structure alignment (or threading) the estimate of fitness is mostly based on some kind of pseudo–energy function. It is quite tricky to derive meaningful pseudo–energy functions and many proposals have been made in the literature (Bryant & Lawrence 1993; Hubbard 1994; Maiorov & Crippen 1992; Miyazawa & Jernigan 1985; Sippl 1990). Most of these energy potentials of mean force include terms that evaluate a specific interaction in the given structure. For instance, the interaction between the positions i and j_1 of the structure B in figure 1 is scored according to the residues of the sequence A that are mapped onto the positions i and j_1 via the homomorphism f. We can make two general observations on

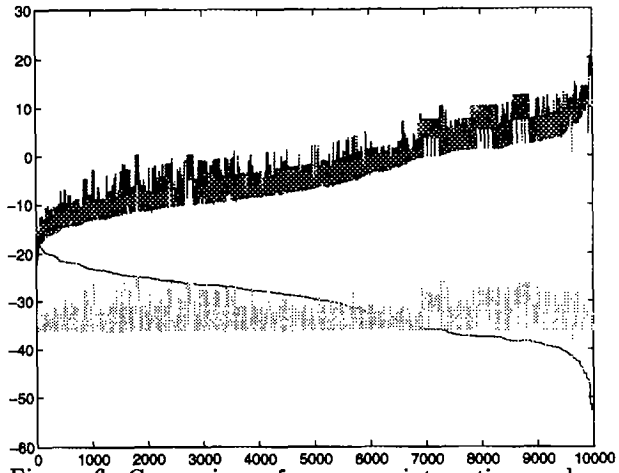


Figure 6: Comparison of sequence, interaction, and combined score for 10000 optimal and suboptimal alignments of 1FDX with the ferredoxin 2FXB from BACILLUS $THERMOPROTEOLYTICUS (sorted w.r.t. decreasing interaction energy)

the role played by pseudo–energies in the evaluation of sequence–to–structure fitness. The first observation is illustrated by Figure 6. This figure shows the overall pseudo–energies of 10000 threadings of the ferredoxin 2FXB into the ferredoxin 1FDX, sorted in decreasing order (black curve). All of these threadings are, by definition, near–optimal w.r.t. their sequence–alignment score. In the figure, this score is shown in light–gray. A mixed score amounting to the sum of the sequence score and the pseudo–energy of the alignment is shown in dark–gray. The figure shows that pseudo–energies are able to discriminate between the large number of near–optimal and thereby mostly similar alignments. The challenge is to tune the energy potential to optimizing structural fitness within the space of near–optimal sequence alignments.

The second observation is that energy values cannot only be used as a tool for discriminating between different alignments globally, but rather that we can use the potentials in order to single out local regions in the structure, which dominate the sequence–to–structure fitness. Figure 7 presents two renderings of the energy distribution for the threading of 2FXB into 1FDX. Part (a) of the figure shows a map, in which entry (i, j) represents the energy contribution (Russell & Barton 1994) of residue j in 2FXB when it is singly mapped onto residue i in 1FDX (frozen approximation). Entry (i, j) of the map in part (b) of the figure represents the difference between that energy value and the energy contribution of the original residue i in 1FDX. In all three diagrams the cysteine residues coordinating the [4Fe-4S] clusters of the ferredoxins show up dramatically. Such clear signals can be analyzed in a preprocessing step of the RDP procedure and be used by the oracle to rate the reliability of subsolutions containing

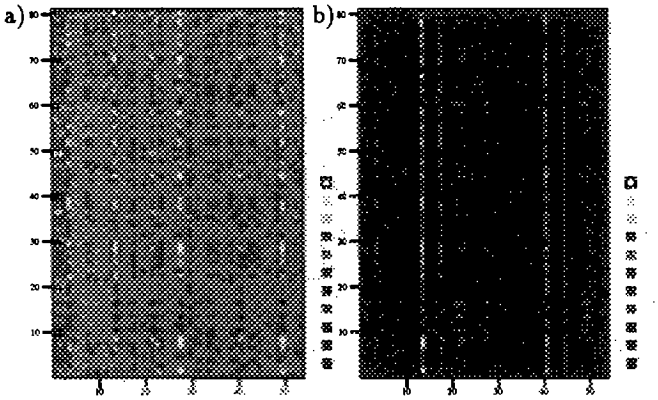


Figure 7: 'Energy' plots of positional contributions (a) of single positions, (b) in comparison to the native sequence

the correctly mapped cysteines.

Structure comparison with RDP

A common approach to comparing protein structures is to superpose their 3D–coordinates and to compute the root–mean–square (rms) deviation of the assigned residues or atoms. In order to compute the superposition efficiently, the matching of residues or atoms is assumed to be given, for instance, via a sequence alignment. However, the structure comparison can only be as good as the sequence alignment which forms its basis.

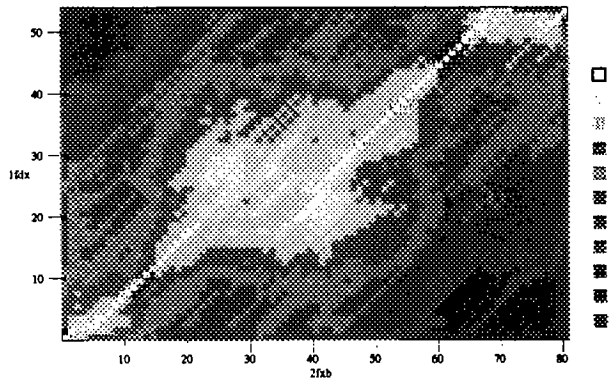


Figure 8: Global ϕ/ψ angle alignment (2FXB vs. 1FDX)

A statistical analysis of the sets of protein structures and sequences (Orengo 1994) shows that there are many proteins with remarkably low sequence similarity that nevertheless share the same overall structure. For such proteins, it is very hard to produce a meaningful sequence alignment as a starting point for a structure comparison and it would be better to derive a structural alignment directly. Here, we present two methods of detecting information on structural similarity of local substructures which can be exploited by the RDP method.

First, we can compare protein structures as sequences of rotation– and translation–invariant dihedral ϕ/ψ angles. Local alignments of sequences of ϕ/ψ angles can be used as oracle for the RDP method in order to compute structure–alignments which figure out local structural similarities between structures. In the ToPLign tool environment (Mevissen *et al.* 1994) such sequences can be aligned via dynamic programming just like sequences of amino–acid identifiers and the RDP method can be guided by information from the associated path–contour maps similarly as in sequence alignment. Figures 5, 8, and 9 contrast the effects of aligning sequences of ϕ/ψ angles derived from the structures versus aligning the sequences of the amino acids themselves. The amino–acid sequence alignment uses Dayhoff homology scores with standard gap penalties, whereas the ϕ/ψ alignment uses the rms deviation of the angles of the corresponding amino acids with appropriate gap penalties. The figures show differences as well as points in common among the two kinds of alignments, which can be the basis for reliability criteria used by the oracle.

Second, we can superpose all short fragments of fixed length of the first structure with all fragments of the same length in the second structure via standard fragment superposition methods (McLachlan 1982). Such a preprocessing step produces a list of similar fragments sorted by their rms deviation. Part (c) of Figure 9 shows a rms–distance matrix for superpositions of fragment of length 8 taken from 1FDX and 2FXB. Now, the oracle of the RDP method reduces to efficient table–lookups in this list and the RDP procedure combines the fragments into structure alignments in such a way that all mutually consistent segments of two given structures, which are structurally similar, are covered by the alignment. If we like, the method guarantees that the matched parts are still compatible with the sequential order prescribed by the sequence of the proteins.

Multiple alignments

The RDP method can also be applied to multiple alignments. Using the method, we can first identify consistent local matches in the pairwise alignments and than split up all participating sequences according to such alignments. Doing so, sequence patterns often show up with much more significance than with usual SOP or tree alignment. The result is an increased reliability and efficiency of the method. Figure 10 shows that the first consistent match drastically reduces the search space left over for the recursive procedure. For instance, in the plots of the second row only the non–black regions of the matrices have to be searched by the algorithm, after the first reliable segment of the alignment has been found.

Experimental results

Figure 11 shows an HSSP-alignment (Sander & Schneider 1991) of 36 homologous ferredoxin sequences that can be derived almost identically via a dynamic–programming–based tree alignment. It is an interesting observation that, despite of the unquestionable multiple alignment of the sequences and the assumed functional conservation with a [Fe–S] cluster (and supposedly similar structure), almost all positions have a

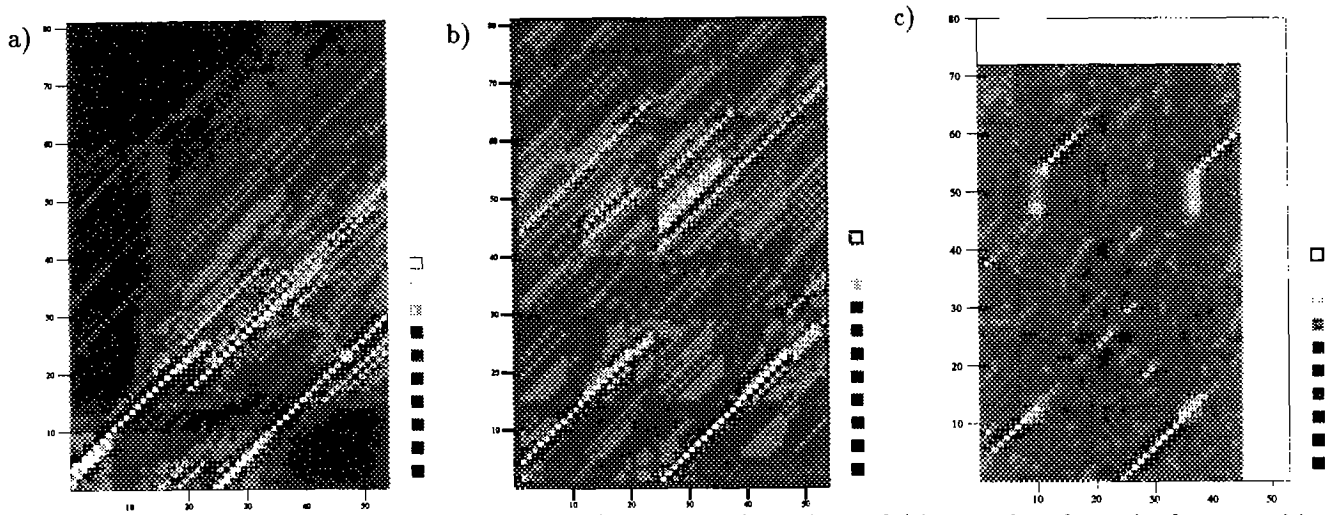


Figure 9: Local alignment (a) based on sequences, (b) based on ϕ/ψ angles, and (c) rms values for optimal superpositions of fragments of length 8, starting at position i in 1FDX and j in 2FXB

high degree of variability with 18 positions containing 8 and more (up to 14) different amino acids.

```
Number      :.....10.....20.....30.....40.....50....
1FDX.PDB_01 :ATVIWDSCIACGACKPECPVWIQQGSIYAIDADSCIDCGSCASVCPVGAPNPED
            :==================================================
P00198..._02 :ATVIWEACISCGACDPECPVDAISDSRTYVIDADTCIDCGACAGVCPVDAP----
P00195..._03 :ATKIADSCVSCGACASECPVWAISDSIFVIDADTCIDCGNCANVCPVGAPVQE-
P00196..._04 :AFVIWDSCVSCGACAGECPVSAITDTQFVIDADTCIDCGNCANVCPVGAPNQE-
P00197..._05 :ATKITDGCINCGACEPECPVEAISESVRVIDADKCIDCGACANTCPVDA-----
P00194..._06 :ATKIEETCISCGACAAECPVWAIEDTIFVVNADTCIDCGNCANVCPVGAPVAE-
P22846..._07 :ATKILDTCVSCGACAAECPVDAISDTQFVIDADTCIDCGNCANVCPVGAPVQE-
P07508..._08 :ATFITDACISCGACESECPVSPISPSVYVIDADACIECGACANVCPVDAPQQK-
P14073..._09 :ATKITDECIACGSCADQCPVEAISGSIYEIDEALCTDCGACADQCPVEAIVPED
P00200..._10 :AHIITDECISCGACAAECPVEAIHEGTYEVDADTCIDCGACEAVCPTGAVKAE-
P06123..._11 :-------ECTVCGDCKPVCPTGSIQGGIYVIDADSCNECADCLGVCPVD------
P00205..._12 :ALYITEECTYCGACEPECPTNAISAEIYVIDAAGCTECVGCAAVCPAE------
P00201..._13 :-HVISDECVKCGACASTCPTGAIEGETKYVVTDSCIDCGACEAVCPTGAISAE-
P24496..._14 :-YVIAEPCVDDKACIEECPVDCIYGGRHLYIHPDECVDCGACEPVCPVEAIYYED
P00204..._15 :ALYITEECTYCGACEPECPVTAISAGIYVIDAMTCNECAGCVAVCPAE------
P00206..._16 :AHRITEECTYCAACEPECPVWAISAEIYIVDESVCTDCEGCVAVCPVD------
P00215..._17 :-YVIAEPCVDVKACIEECPVDCIYERRLYIHPDECVDCGACEPVCPVEAIYYED
P13279..._18 :-YVIAQPCVDVKACIEECPVDCIYERSLYIHPDECVDCGACEPVCPVEAIFYED
P03941..._19 :PFVITSPCIGKKACVETCPVDAIHEGQYYIDPDLCIDCAACEPVCPVNAIYQEE
P14939..._20 :-------ECTVCGDCEPVCPTGSIRDDHYAIEADSCNECTDCLGVCPVD------
P00218..._21 :-----DCCIADGACNDVCPVNLYEWNLDPVRESDCIFCHACESVCPVRA-----
P12712..._22 :--IIASQCTQCGACEFECPRGAVNGEKYVIDPTKCNECKGCASVCPVSNT----
P18082..._23 :-YVVTDNCIACKDCVEVCPVDCFYEGTLVIHPDECIDCGVCEPECPADAIRPDT
P03942..._24 :PHVICQPCIGVKSCVEVCPVECIYGDQFYIHPEKCIDCGACVPACPVNAIYPEE
P06253..._25 :-----DKCIACEVCVRVCPINALPQLKNYSIDFGVCIFCGNCVEYCPTK------
P00208..._26 :ALRITDQCINCNVCQPECPNGAISDETYYIEPSLCTECVGCVEVCPVD------
P00202..._27 :ATVNADECSGCGTCVDECPNDAIEKGIAVVDNDECVECGACEEACPNQAIKVEE
1FD2.PDB_28 :AFVVTDNCIKCKDCVEVCPVDCFYENFLVIHPDECIDCALCEPECPAQAIFSED
P00213..._29 :-FVVTDNCIKCKDCVEVCPVDCFYENFLVIHPDECIDCALCEPECPAQAIFSED
P08811..._30 :-FVVTDNCIKCKDCVEVCPVDCFYENFLVIHPDECIDCALCEPECPAQAIFSED
P00219..._31 :-----DLCIADGSCITACPVNVFQKKADPVNQQACIFCHACVNVCPVAAIDYKP
P23481..._32 :----------PCLQVCPVNAISQRDDAIWESLCIGCKLCAVVCPFGAISASG
P00211..._33 :--VDSDKCIGCGECVDVCPVEVYEGKAVPVWEEECLGCESCIEVCPQNA-----
P18776..._34 :AYVLSISCNHCEACTKVCPSQAEHKRPFVVVDEDVCIGCRYCHHACPYGAPQYNE
P08813..._35 :--VDTDKCTGDGECVDVCPVEVYEGKAVPVDEEECLGCESCVEVCEAGAITYEE
            :==================================================
# different :368686C26B5671AB5116677898E8726789154188169612783SB766
dissimilar  :
consensus   :              C   CP            C C C   C
# majority  :******9**7***** ******** 69************** 9**********
majority    :ATVITDECISCGACV2ECPVDAIS2SIYVIDADECIDCGAC2PVCPVGAI--E-
abs.majority:   I D CI CGAC  ECPV AI     VID D CIDCG C  VCPV A   E-
conserved   :4123130920533911399512311001253231933921920493315211 33
variability :577676906946608860048758899874676806607807840068478866
entropy     :2443436036233055200263345564413435033044035200351355 33
score       :| || | || |||C   CP|          || | C||C  C  |C|| |   |
reliability :888877787788887779975555556668887788897787789888877788
```

Figure 11: Multiple alignment of 36 ferredoxins

In a multiple tree alignment the signals of the few conserved residues are amplified by the stepwise extension of the alignment starting with closely related sequences and then extending to more unrelated sequences. Pairwise sequence alignment is not adaptive to conserved regions, in this sense. Therefore, ferredoxins are quite challenging examples for deriving meaningful pairwise alignments without taking information beyond the pairwise sequence homology into account.

We have performed the following two initial experiments with the RDP method: First, we have analyzed the agreement of sequence alignments computed by the RDP method with other alignments reported in the literature. Second, we have tested the capability of RDP to detect structural issues.

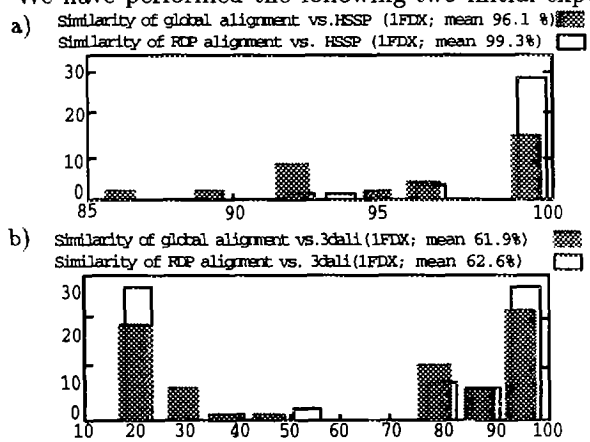
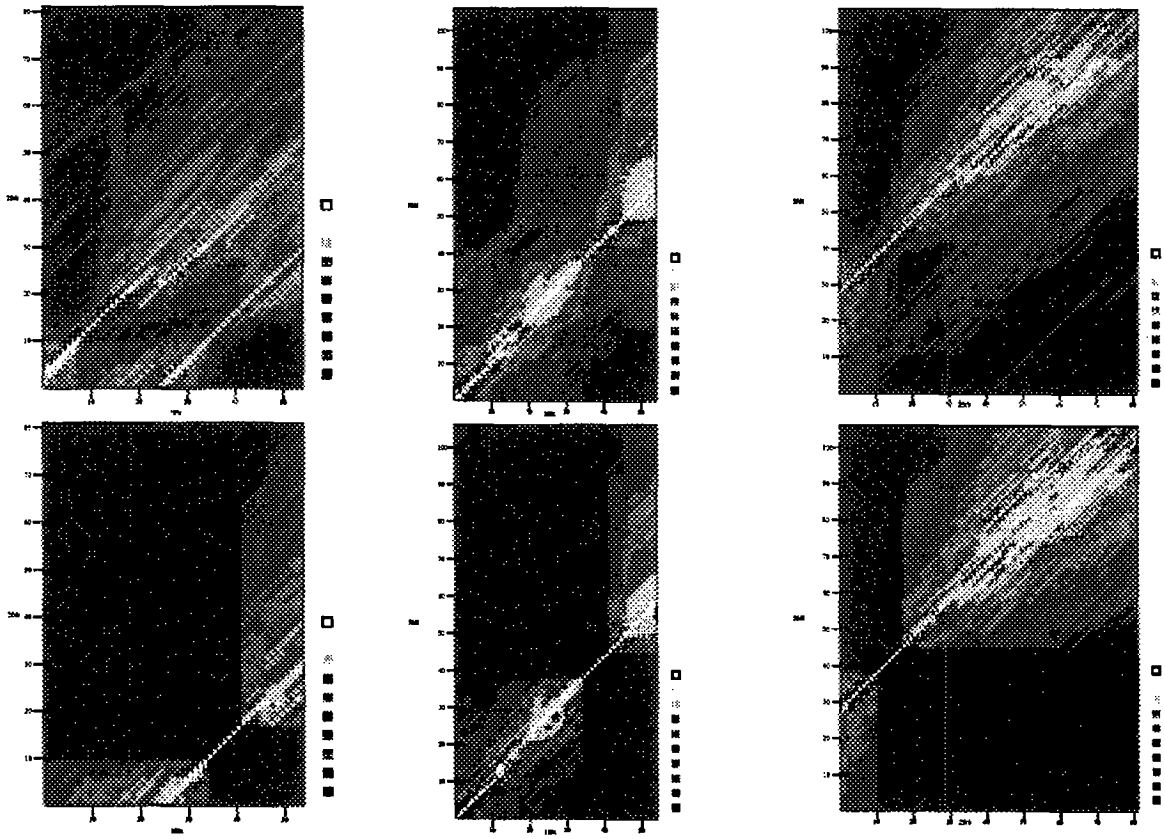


Figure 12: Capability of reproducing (a) HSSP– and (b) 3Dali–alignments for ferredoxins

Let us turn to the former experiment. Here, we take the HSSP and 3Dali alignments of ferredoxins as the standard of truth. For the aim of comparison we decompose the given multiple alignments into pairwise alignments. A score for the similarity of two pairwise alignments can be computed elegantly using the ToPLign (Mevissen *et al.* 1994) *profile alignment* of alignments. To be precise, we align the two alignments such that the number of column–wise agreements between them is maximized. Here, an agreement is counted if the matched columns are identical in both alignments. Figure 12 compares the percentage of agreement between HSSP (part a)) and 3Dali (part b)) alignments, on the one hand, and standard



```
1FDX     :AYVINDSCIAC--GACKPECPVN-IIQGSIY-AIDADSCIDCGSCASVCPVGAPNPED.....................
5FD1     :AFVVTDNCIKCKYTDCVEVCPVDCFYEGPNFLVIHPDECIDCALCEPECPAQAIFSEDEVP.................
2FXB     :.............................PKYTIVDKETCIACGACGAAAPDIYDYDEDGIAYVTLDDNQGIVEVPDI
consensus:.............................          CI C  C    P      ED.....................
```

Figure 10: Path–contour matrices of all pairwise sequence alignments of 3 ferredoxins before and after fixing the best scoring compatible region (matching the cluster of 2FXB onto the 2nd clusters of 1FDX / 5FD1) and the corresponding alignment.

global alignment (filled bars) and RDP (unfilled bars), on the other hand.

The results show that RDP outperforms classical global alignment both with respect to the number of exactly recreated alignments and with respect to the average number of identically reproduced matches. The results are much more striking for HSSP alignments than for 3Dali alignments. As discussed below in more detail, the single [Fe–S] cluster of ferredoxins with only one cluster can be structurally mapped on both [Fe–S] clusters of ferredoxins containing two clusters. This ambiguity is reflected in part (b) of figure 12 by alignments having an agreement with the 3Dali alignment less than 50 %.

The quality of the RDP method, as it is exhibited by these experiments, is especially striking in view of the fact that the RDP method does not compute optimal solutions and the power of structural information cannot be put to full use in pure sequence alignment. We believe that the method is superior because of its power of adaptively focussing on significant regions of the alignment.

We now turn to the second experiment. Figure 13 shows several sequence– and structure–based alignments of 1FDX, which has 54 residues and two [4Fe–4S] clusters and 2FXB with 81 residues and only one [4Fe–4S] cluster. In 1FDX, the two clusters are coordinated via four cysteine residues at positions 8, 11, 14, and 45, for the first cluster, and at positions 18, 35, 38, and 41, for the second cluster. The single cluster in 2FXB is coordinated via the four cysteine residues at positions 11, 14, 17 and 61. The four cysteine residues that participate in the coordination of each of the two clusters in 1FDX occur in the motif CxxCxxCxxxCP, which is characteristic for ferredoxins. The last cysteine residue occurs only in the ferredoxins that have two [4Fe–4S] clusters, i.e., only in 1FDX but not in 2FXB. The parts of the protein containing the cluster are sequentially and structurally highly conserved. However, because of the small difference in the motifs defining the cluster–coordinating sites, both global and local and both sequence and structure alignment methods are trapped: They search for and find the motif CxxCxxCxxxCP. The global, local, and 3Dali alignments correctly match the first sequence motif CxxCxxCxxxCP, but then fail to align the second part with the fourth cysteine coordinating the cluster (45 in 1FDX and 61 in 2FXB) or match it incorrectly.

```
Global alignment computed with ToPLign:
1FDX:A-YVI--NDSCIACGACKPECP-----------------VNIIQGSIYAIDA-----DSC-IDCGSCASVCPVGAPNPED
2FXB:PKYTIVDKETCIACGACGAAAPDIYDYDEDGIAYVTLDDNQGIVEVPDILIDDMMDAFEGCPTDSIKVADEPFDGDPNKFE
Local alignment computed with ToPLign:
1FDX:...AYVINDSCIACGACKPECPVNIIQGSIYAIDADSCIDCGSCASVCPVGAPNPED.....................................
2FXB:PKYTIVDKETCIACGACGAAAP------DIYDYDEDG.....................IAYVTLDDNQGIVEVPDILIDDMMDAFEGCP...
Structure alignment from 3Dali:
1FDX:---AYVINDSCIACGACKPECP-VN--IIQG-SIYAIDADSCIDCGSCASVCPVGAPNPED-------------------------------...
1fxb:PKYTIVDKETCIACGACGAAAPDIYDYDEDGI-----------------------------AYVTLDDNQGIVEVPDILIDDMMDAFEGCP...
Structure alignment from GBF:
1FDX:AYVINDSCIACGACKPECPVNIIQGSIYAIDADSCIDCGSCASVCPVGAPNPED-------------------------------------...
2FXB:PKY----------afegcptdsik---TIVDKETCIACGACGAAAP-DIYDYDEDGIAYVTLDDNQGIVEVPDILIDDMMDafegcptdsik...
Sequence Alignment computed with RDP:
1FDX:--AYVIND--SCIACGACKPECPVNIIQGSIYAIDADSCI-----DCGSCASV--------------CPVGAPNPED-----------
2FXB:PK-YTIVDKETCIACGACGAAAP------DIYDYDEDGIAYVTLDDNQGIVEVPDILIDDMMDAFEGCPTDSIKVADEPFDGDPNKFE
Structure alignment computed with RDP:
1FDX:--AYVIN-DSCIACGACKPECPV-NIIQG---SIYAIDADSCIDC-----------------GSCASVCPVGAPNPED--------------
2FXB:PKYTIVDKETCIACGACGAAA--PDIYDYDEDGIAYVTL------DDNQGIVEVPDILIDDMMDAFEGCPTDSIK---VADEPFDGDPNKFE
```

Figure 13: Sequence and Structure alignments for 1FDX and 2FXB

The GBF structure alignment (Lessel & Schomburg 1994) optimizes the structural superposition ignoring the sequence order. The resulting alignment identifies a whole cluster by mapping the CxxCxxCxxxxP motif of 2FXB onto the second CxxCxxCxxxCP motif of 1FDX and also correctly identfies the region around the fourth cysteine in the first motif. This results in 43 superposed C_α atoms at a rms distance of 0.96 Å. However, for comparative modeling, this alignment could be quite misleading because it is not in accordance with the sequence order.

Both the sequence– and structure–based RDP alignment correctly find the structurally significant matches (see Figure 13). Furthermore, the order of the alignments respects the sequence order, in contrast to the purely structural alignment. The RDP alignment maps the CxxCxxCxxxxP motif of 2FXB onto the first CxxCxxCxxxCP motif of 1FDX and the fourth cysteine 45 of 1FDX correctly onto cysteine 61 of 2FXB. Even the long segment that is not aligned in the structure–based RDP alignment is consistent with the structural properties, because the helix in this region is longer in 2FXB than in 1FDX. This fact prohibits 2FXB from coordinating a second cluster. The superposition associated with this alignment covers 43 amino acids with an rms of 1.25 Å.

This experiment gives evidence for our hypothesis that combining evolutionary sequence information with information on structural preferences can detect structurally meaningful alignments even where methods purely based on sequence information or purely based on structure information are being misled. Encouraged by the results presented in this paper, we are equiping the RDP method with additional variants of sequence and structure information. In the near future, we will run extensive experiments on sequence and structure data and rate the RDP method in comparison to other existing methods for aligning sequences and/or structures.

References

Bowie, J.; Lüthy, R.; and Eisenberg, D. 1991. A method to identify protein sequences that fold into a known three-dimensional structure. *Science* 253:164–170.

Bryant, S., and Lawrence, C. 1993. An empirical energy function for threading protein sequence through the folding motif. *PROTEINS: Structure, Function and Genetics* 16:92–112.

Dayhoff, M. O. 1978. A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure, Supplement 3* 5:345–352.

Godzik, A.; Kolinski, A.; and Skolnick, J. 1992. Topology fingerprint approach to the inverse protein folding problem. *JMB* 227(1):227–238.

Hubbard, T. 1994. Use of β-strand interaction pseudopotentials in protein structure prediction and modelling. In Lathrop, R., ed., *Proceedings of the Biotechnology Computing Track*. IEEE Computer Society Press.

Jones, D.; Taylor, W.; and Thornton, J. 1992. A new approach to protein fold recognition. *Nature* 358:86–89.

Lathrop, R., and Smith, T. 1994. A branch-and-bound algorithm for optimal protein threading with pairwise (contact potential) amino acid interactions. In *Proceedings 27th Hawaii International Conference on System Sciences*, volume V. Los Alamitos, CA: IEEE.

Lathrop, R. 1994. The protein threading problem with sequence amino acid interaction preferences is NP-complete. *Protein Engineering* 7(9):1059–1068.

Lessel, U., and Schomburg, D. 1994. Similarities between protein 3-d structures. *Protein Engineering* 7(10):1175–1187.

Maiorov, V., and Crippen, G. 1992. Contact potential that recognizes the correct folding of globular proteins. *Journal of Molecular Biology* 227:876–888.

McLachlan, A. 1982. Rapid comparison of protein structures. *Acta Crystallographica* A 38:871–873.

Mevissen, H., and Vingron, M. 1995. Quantifying the local reliability of a sequence alignment. submitted.

Mevissen, H.; Thiele, R.; Zimmer, R.; and Lengauer, T. 1994. Analysis of protein alignments – the software environment toplign. GMD Technical Report (in preparation).

Miyazawa, S., and Jernigan, R. 1985. Estimation of effective interresidue contact energies from protein crystal structure: Quasi-chemical approximation. *Macromolecules* 18:534–552.

Orengo, C. 1994. Classification of protein folds. *Current Opinion in Structural Biology* 4(3):429–440.

Russell, R. B., and Barton, G. 1994. Structural features can be unconserved in proteins with similar folds. *Journal of Molecular Biology* 244(3):332–350.

Sander, C., and Schneider, R. 1991. Database of homology-derived protein structures and the structural meaning of sequence alignment. *PROTEINS: Structure, Function and Genetics* 9:56–68.

Sippl, M. 1990. Calculation of conformational ensembles from potentials of mean force: An approach to the knowledge-based prediction of local structures in globular proteins. *Journal of Molecular Biology* 213:859–883.

Taylor, W., and Orengo, C. A. 1989. Protein structure alignment. *JMB* 208:1–22.