Search Strategies in Structural Bioinformatics

Mark T. Oakley[1], Daniel Barthel[2], Yuri Bykov[2], Jonathan M. Garibaldi[2], Edmund K. Burke[2], Natalio Krasnogor[2*] and Jonathan D. Hirst[1*]

[1]School of Chemistry, University of Nottingham, University Park, Nottingham NG7 2RD, United Kingdom

[2]School of Computer Science and Information Technology, University of Nottingham, Jubilee Campus, Nottingham, United Kingdom

[*]corresponding authors

Abstract

Optimisation problems pervade structural bioinformatics. In this review, we describe recent work addressing a selection of bioinformatics challenges. We begin with a discussion of research into protein structure comparison, and highlight the utility of Kolmogorov complexity as a measure of structural similarity. We then turn to research into de novo protein structure prediction, in which structures are generated from first principles. In this endeavour, there is a compromise between the detail of the model and the extent to which the conformational space of the protein can be sampled. We discuss some developments in this area, including off-lattice structure prediction using the great deluge algorithm. One strategy to reduce the size of the search space is to restrict the protein chain to sites on a regular lattice. In this context, we highlight the use of memetic algorithms, which combine genetic algorithms with local optimisation, to the study of simple protein models on the two-dimensional square lattice and the face-centred cubic lattice.

1. Introduction

In this review article we will cover several aspects of computational studies of protein structure, focusing on structure comparison, structure prediction and the aggregation of structures. These topics are, on the one hand, quite diverse and yet, on the other, certainly in no way span the entire breadth of activity in the field of structural bioinformatics. The selection of topics reflects our particular research interests, but in describing some of own work, we will also set it in the broader context of activity within the discipline. Much of this research is driven by new advances and ideas in computer science and we will highlight novel applications of search algorithms in structural bioinformatics.

Structural analysis is at the core of homology modelling and comparative modelling [1] and has provided invaluable insights into the biological function of proteins and the evolutionary relationships among proteins [2]. Structure comparison is also an essential part of the assessment of predicted protein structures, and plays a key role in the biennial Critical Assessment of Techniques for Protein Structure Prediction (CASP) exercises [3]. Similarity beyond trivial near-identity is a somewhat subjective concept and so there are many approaches to structure comparison, which emphasize different aspects. One method that we have been actively developing and applying to the comparison of protein structures [4] and small molecules [5] is the so-called Universal Similarity Metric (USM), which, as we discuss later, expresses all other similarity metrics [6]. We survey current algorithms for protein structural comparison, distinguishing between methods based on the comparison of three-dimensional coordinates and those based on the comparison of two-dimensional distance matrices or contact maps, and we summarize current web servers and meta-servers for protein structure comparison.

The number of groups participating in CASP has grown substantially over the past decade, reflecting, amongst other factors, an increasing effort in predicting protein structures *de novo*. The

field is broad and has been the subject of numerous reviews, including summaries of the CASP experiments themselves [7, 8]. Three key issues in prediction are: how a structure is represented, how candidate structures are generated and how these candidate structures are assessed, *i.e.*, the nature of the energy function or scoring function. A richer level of structural representation will, by necessity, mean that a smaller number of structures can be assessed. We focus on two strategies for structure prediction in this review: the use of lattice models to represent proteins in a coarse-grain manner and the use of heuristic search techniques for generating candidate structures. The techniques of classical molecular dynamics simulation, Monte Carlo, simulated annealing and genetic algorithms are well established. We have been investigating newer heuristic approaches. We review activity in this area and only touch on the more established techniques in passing, for example, we do not discuss all-atom molecular dynamics simulations in detail.

The mechanisms of protein mis-folding and aggregation are perhaps even more challenging than the problem of predicting protein folding, but are, naturally, of great interest, due to the association of protein aggregates with diseases, such as Alzheimer's, amongst others. The relevance of aggregation has only been recognised in the last few years, but due to its importance it has already been the subject of several computational studies. Many of the technical problems related to protein folding also apply to protein aggregation, but there are additional difficulties, not least the obvious issue in terms of sampling when there are several protein chains. We discuss lattice studies of aggregation.

2. Protein Structure Comparison

The large scale analysis of protein sequence, structure and function is a fundamental part of current research in the biosciences. Effective algorithms, which can handle the massive influx of new data generated by current high throughput technologies, have made contributions to many advances in structural biology, especially in protein structural biology. For example, structural alignments can

reveal the evolutionary history of proteins and allow inferences of their function, as proteins with high structural similarity usually show related biological functions. Structural comparisons also play a key role in the modelling of new proteins and, more generally, in shaping our understanding of the organisation of the known protein universe [9, 10]. In the context of *de novo* and homology modelling, one may want to identify common structural building blocks that can be collected in fragment libraries, to allow better predictions of native protein structures [11]. Also, protein structure comparison is necessary when a representative or consensus structure must be obtained from a large number of structural variants. Regardless of whether one is interested in comparative modelling, fold recognition or new fold prediction, fast, robust and accurate methods for comparing protein structures are needed. Prominent examples for this demand are CASP [12] and the Evaluation of Automatic Protein Structure Prediction (EVA) [13].

Protein structure comparison methods use a variety of similarity concepts and there is no consensus on which similarity measure (or metric) is the best. Indeed, different biological problems may require different similarity measures and/or metrics. In evolutionary terms, structure is more conserved than sequence, because structures have more constraints imposed upon them by chemical and physical factors. Thus, the number of viable protein folds is limited [10, 14, 15]. This implies that there are remote homologous proteins with highly conserved structures (and hence possibly function), but without recognizable sequence similarity [16]. In some cases, one may want to focus on local structural similarities; in other cases, global similarities may be of interest. For instance, when considering distant homologues, a global similarity assessment may miss common features if one of the proteins contains shifts in the orientation of equivalent secondary structure elements, or if it has extensive deletions or insertions of residues [17]. On the other hand, considering only local sub-structures could lead to situations where the global alignment is missed [10]. When aligning a new protein structure against a database of known structures, rapid secondary structure-based approaches could be used to indicate a protein's class, architecture, topology and homologous

superfamily [18], and subsequently more accurate, residue-based methods can narrow down the number of putative relatives. The process of comparison can actually use different "features" on which to base the similarity assessment, such as a subset of the atomic three-dimensional coordinates (*e.g.*, the backbone C_α atoms, the side chain C_β atoms, or the residues' centre of mass), secondary structure elements (*e.g.* α-helices, β-sheets, loops), environmental profiles, internal mappings (*e.g.*, distance matrices or contact maps), or a combination of these [15, 19]. Furthermore, these comparisons could combine different strategies. Sequence-dependent methods would use features taken from the proteins' sequence only when aligning "equivalent" residues, sequence-independent methods neglect sequence data in favour of higher structural features, while hybrid methods would combine the two [17].

2.1 Structure Comparison: Methods and Algorithms

Methods for protein structure comparison usually apply one or more of the previously mentioned concepts [10]. For example, some algorithms use fragment matching [20, 21], while others rely on geometric hashing [3], the comparison of distance matrices [22], contact map overlaps [23-27], maximum sub-graph detection [28], local geometry matching [29], consensus structures [30], or other criteria as the source of similarity. Moreover, as many of these involve a combinatorial or continuous optimisation problem, a search and optimisation technique must be used to compute the similarity involved. Optimisation strategies such as incremental combinatorial extension of the optimal path [31], Monte Carlo algorithms and simulated annealing [22], dynamic programming [32-34], genetic algorithms [35] and memetic algorithms [25, 27] are regularly employed. We focus next on just a few illustrative examples, covering the range of available techniques, to give a flavour of the rich literature and software that is available.

The root mean square deviation (RMSD) is a global, sequence-dependent measure of 3D similarity often used to find an optimal rigid body superposition of two structures [36]. The superposition

depends on one structure being fixed, while the second is translated and rotated, to minimise the Euclidean distance measure between pairs of amino acids. The minimisation process depends on having (a) the Cartesian coordinates for each amino acid in each of the two proteins and (b) an alignment at the sequence level between the two proteins. Usually, amino acids are represented by their C_α or C_β atoms, although residues' centres of mass are sometimes also used. The optimal translation will make the centres of mass of the two structures coincide, while for the optimal rotation, a correlation matrix between both sets of Cartesian coordinates must be considered. As RMSD is a global similarity measure, it is more sensitive to small regions with large differences than to large but fairly similar regions. Therefore, one needs to take into account the actual number of aligned residues [17]. For non-rigid body comparisons that take account for flexible structures containing loops and hinges, a Gaussian-weighted RMSD value has been introduced recently [37].

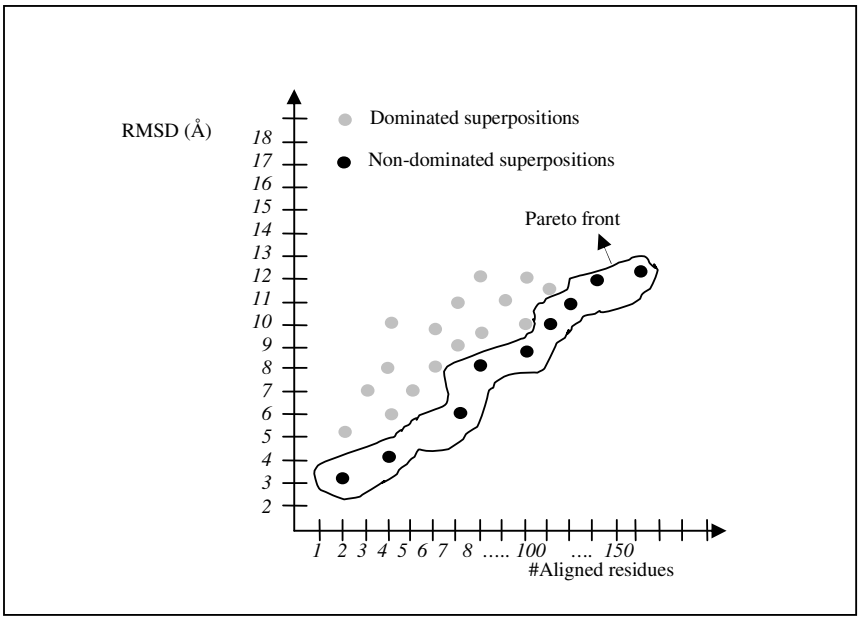

Figure 1. Schematic representation of alternative solutions to a bi-objective protein structure superpositioning problem. In structure alignment, minimizing the RMSD and maximizing the number the number of aligned residues may be competing objectives. The Pareto front gives a set of possible solutions.

Despite these caveats, RMSD is commonly used in conjunction with other similarity indicators, such as the number of equivalent (structurally aligned) residues. It is not trivial to optimise both RMSD and, say, the number of aligned residues, as they represent conflicting objectives [39]. Usually, these two indicators are, either implicitly or explicitly, weighted by the similarity algorithms, effectively rendering an intrinsically multi-objective problem into a single objective one. It would be more flexible to have an algorithm where the user, or a decision support system, could generate a pareto front for the bi-objective problem of superposing two protein structures (see Figure 1). Having the set of non-dominated solutions would inform a decision on which point or points, *i.e.*, structural superpositions, of the pareto front are more appropriate for the problem at hand. More specifically, if the proteins are not closely homologous, a solution with low RMSD but a smaller number of aligned residues might be better. Conversely, if the proteins are close homologues then a high number of aligned residues should be expected.

One algorithm and web server providing both sequence-dependent and sequence-independent methods for protein structure comparison, is LGA (local-global alignment) [38], which calculates a similarity profile that takes into account regions with both local and global structure commonalities. It achieves this by using two different scoring functions, the global distance test (GDT) and the longest continuous segments (LCS) [38]. These two similarity measures are combined into a single objective by a mechanism that balances the relative contribution of LCS and GDT. GDT identifies residues that meet a given RMSD threshold, located anywhere in the structure (sequence-independent calculation). Residues that fit the global RMSD profile are considered to be equivalent. LCS, on the other hand, finds all the longest continuous fragments of similar residues that deviate by no more than a specified distance cut-off (sequence-dependent calculation). LCS can be efficiently computed by a dynamic programming algorithm, but GDT, being a global combinatorial problem, requires a heuristic method for its calculation. Furthermore, as the similarity of two

structures cannot be captured using just one RMSD threshold and one distance cut-off, LGA generates many local superpositions, using a set of increasing RMSD thresholds and distance cut-offs. All the results are combined into a single similarity measure (LGA_S). This method is routinely used by CASP assessors to measure the quality of predicted structures against their respective targets [3, 39].

Another approach is based on the comparison of paths and graphs. Employing an incremental combinatorial extension (CE) of the optimal path [31], one can build up an optimal match between two protein structures using aligned fragment pairs. These are fragments of each protein of variable size with similar structural features, and are derived from the local geometry, rather than the overall topology or the orientation of secondary structure elements. From the set of all possible combinations of aligned fragment pairs that adhere to a given similarity criterion, those that produce the longest continuous alignment path are incrementally extended. The heuristic target function describing structural similarity is based only on inter-residue distances, and the solution is evaluated for statistical significance using *Z*-scores. For alignments above a certain *Z*-score threshold, a final optimisation step is added considering not only RMSD values, but shifts of gaps within a certain window, too.

The secondary structure matching (SSM) algorithm [40] defines a two step procedure to compare protein structures. In the first step, a fast graph-matching algorithm aligns a pre-defined set of the secondary structure elements present in the proteins. The second (iterative) step attempts optimal superposition of the protein backbones. A 3D graph is fully defined using the secondary structure elements as labelled vertices, with the length between any two edges defined as the number of residues involved. This graph-based representation of the secondary structures allows a coarse estimation of the structural similarity between the two graphs. A subsequent step refines the initial alignment by finding equivalent C_α atoms among the two protein structures. A quality measure is

computed, which balances the RMSD value between aligned atoms, the length of the alignment and the overall lengths of the two structures. The algorithm attempts to maximise this similarity measure by tweaking the alignments; it can remove less similar pairs or short fragments in order to avoid locking the structures in a particular orientation. The algorithm reports not only the alignment and its quality, but also its significance. A *p*-value is provided that approximates the probability that the final alignment is obtained simply by chance when comparing two random structures; a *Z*-score is also provided. This method works well for closely related structures, but may result in imperfect alignments for structures with low common similarity.

The FAST Alignment and Search Tool [16] tries to find the maximal clique in a pair graph built from the two structures that are being compared. The graph's vertices are the set of (possible) matching C_α atoms. In turn, the edge set is composed of those pairs of C_α atoms with intra-molecular distances below a given threshold. As the maximum clique problem is NP-hard, a heuristic method is applied to eliminate incompatible residue-residue pairs in order to reduce the size and density of the graph for which the maximum clique must be found. After a first step that builds up and then prunes the graph (in favour of consecutive high-scoring segments), edges are chosen by comparing the local geometric properties of the two structures. Side chain orientations are also considered, albeit implicitly, when looking for the maximum clique. An initial alignment is found using dynamic programming and fine-tuned by finding additional equivalent residue pairs and by eliminating residues that are unlikely to appear in the optimal alignment. The final raw score is normalised following an extreme value distribution in order to calculate the statistical significance of the alignment.

We turn now to methods and algorithms based on the comparison of distance matrices, contact maps, contact vectors and Voronoi contacts. The distance matrix of a protein structure describes the pairwise Euclidian distances between all (or a defined subset of) its atoms. It can also be interpreted

as the incidence matrix of a weighted complete graph between all the atoms that make up a protein, with the weight of an edge representing the distance that separates the two end points.

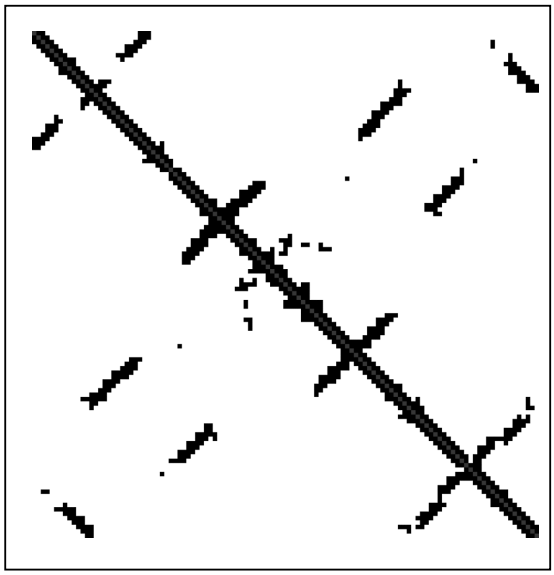

Figure 2. Contact map for protein 1QFO from the Protein Data Bank. Each axis comprises the sequential order of amino acids. Black pixels represent a C_α - C_α contact between two residues, based on a 7.5Å threshold.

A (crisp) contact map (Figure 2) is a binary filtered version of a distance matrix, *i.e.*, an incidence matrix, that defines a contact, C_{ij}, for any pair of elements i and j whose separation, R_{ij}, is below a specified threshold t [41].

$$CM = (C_{ij}) : C_{ij} = \begin{cases} 1 & if\ R_{ij} \le t \\ 0 & otherwise \end{cases} \qquad \text{(Eqn. 1)}$$

As a crisp contact map may leave out important features or protein topological fingerprints, fuzzy contact maps using multiple thresholds (for short and long distance patterns) and fuzzy functions (rather than crisp boundaries as in Eqn. 1) have been introduced as intermediate between full distance matrices and binary contact maps. Fuzzy contact maps can represent structural motifs of

various length scales and also allow one to integrate a degree of uncertainty in the definition of these motifs [23]. Contact maps are, thus, 2D representations of protein structures that contain sufficient information to recover the original 3D structure, except for the overall chirality [42]. If two structures are similar, their distance matrices or contact maps are also expected to be similar, and vice versa [17]. A further simplification of (crisp) 2D contact maps leads to 1D contact vectors (CV), which count the number of contacts, N_i, for each element i in the contact map [43, 44]:

$$CV = (N_i) : N_i = \sum_{i,j} C_{ij} \qquad \text{(Eqn. 2)}$$

Recently, contact vectors have been employed to compare two protein structures, using a histogram representation of the structure's contact lengths [45]. In contrast to distance-based contacts, which are derived from a contact map using a given threshold, Voronoi contacts are defined as the nearest-neighbour contacts of a residue that span a convex polyhedron sharing a common face with its direct neighbours [46]. Using double dynamic programming, Voronoi contacts have been employed for similarity comparison and alignment [46].

In the following, we describe in more detail the algorithms behind DALI/DaliLite, MaxCMO (maximum contact map overlap) and USM (Universal Similarity Metric). The former uses distance matrices, while the latter two are based on contact maps. The DALI/DaliLite algorithm compares two structures by dividing each into hexapeptide fragments and comparing the corresponding 6 by 6 contact maps, in order to simplify the alignment task [22]. It finds common local patterns within these fragments (contact patterns), which are then merged into larger consistent alignments and optimised further by a Monte Carlo algorithm. As this is not guaranteed to converge to the global optimal solution, multiple alignments are optimised in parallel [19]. As its key similarity measure, DALI provides the statistical significance of the final alignment (Z score), its length and RMSD value.

The *maximum contact map overlap* (MaxCMO) problem is an alternative formulation for finding similarities based on contact maps, in which comparing two protein structures is equivalent to finding the maximum overlap of their contact maps. This involves finding the largest sequence-independent, but non-crossing (*i.e.*, ordered), alignment of equal contacts (overlap) between two protein structures [24, 26]. In this case, each contact map is represented as a graph, where each atom/residue is a vertex and each contact between two atoms/residues is encoded by an edge that joins the corresponding vertices. The MaxCMO problem captures a different meaning of similarity, because rather than only aligning equivalent residues, it also takes into account local topological similarity (see Figure 3). Although the MaxCMO problem is NP hard [47], approximation algorithms exist using Dynamic Programming [48], Integer Linear Programming using Lagrangian relaxation [26] and branch and cut approaches [24]. An approach based on maximum cliques has recently been introduced [49]. For large and challenging protein data sets, meta-heuristic

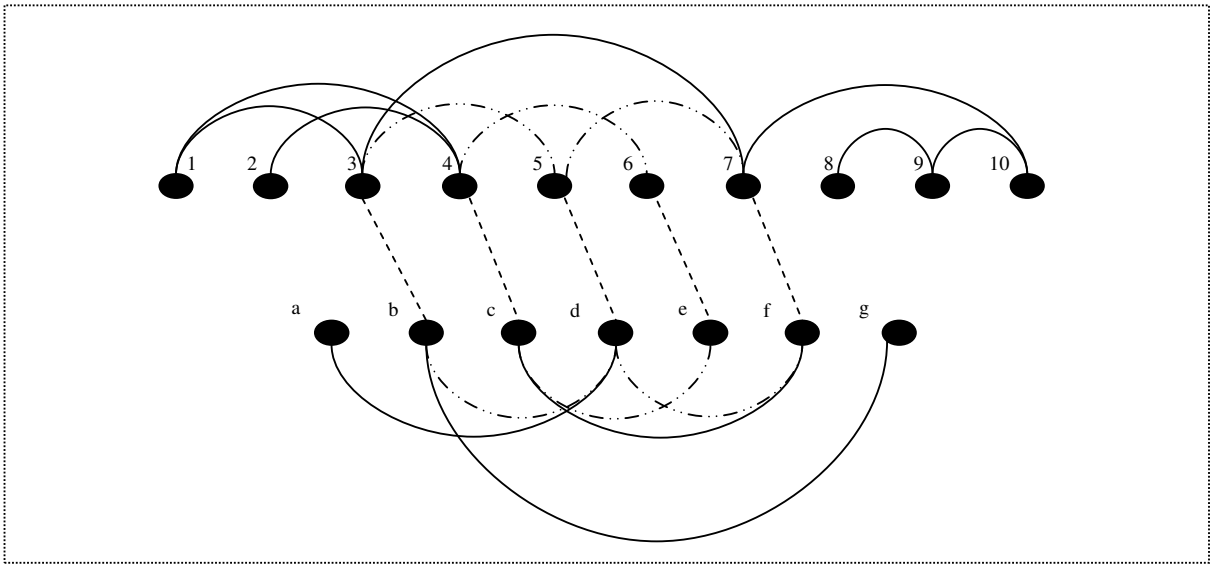

Figure 3. A contact map overlap. Residues 3, 4, 5, 6 and 7 are aligned (dashed lines) with residues b, c, d, e and f. Their local environments are similar, e.g., residues b and d share a contact (dot-dashed line) and are aligned with residues 3 and 5 (above), which also share a contact (dot-dashed line).

approaches for MaxCMO have been used [50]. Contact maps are also used for the calculation of the Universal Similarity Metric (USM), which is used to compare pairs of protein structures without performing any alignment, rotation or translation of the proteins [4, 6]. The USM is founded on the concept of Kolmogorov complexity and compares the information content of two contact maps. Their similarity is expressed as the normalised excess of information between each other, calculating their normalised compression distance (NCD) using a (string) compressor program [51]. The USM is an effective, concept-independent and domain-independent measure of protein similarity, which works particularly well with distantly related structures [4] and sequences [52]. However, as it subsumes every possible similarity concept [6], it is only useful as a first approximation to similarity assessments and, as proposed in [4], it should be used in conjunction with other, more problem-specific methods. The USM of large protein data sets can be calculated through a publicly available server at http://www.procksi.net.

To conclude this section of the review of protein structure comparison, we summarize some of the many web servers and databases, which provide public access to the algorithms and methods described above. For a full overview of on-line databases, the reader might want to consult [53] and the articles in the same database special issue of *Nucleic Acids Research* and related web server issues; such issues have appeared annually for a number of years. Some of the most mature and popular databases are SCOP [54, 55], CATH [56], HOMSTRAD [57], CAMPASS [58], and FSSP [9], amongst others. Table 1 provides a list of URLs for a selection of web servers and meta-servers for protein structure comparison. For a more comprehensive overview, the reader is referred to the Bioinformatics Links Directory [59], a curated catalogue of recommended, well-tested resources, tools and databases for protein 3D structure and sequence comparison.

Fragment Finder [60] is a web-based search-engine and database that allows finding similar 3D

structural fragments against a user-defined protein structure fragment by comparing the conformational dihedral angles of the main chain. The resulting fragments can be superimposed using STAMP [61] or ProFit [62], and are visualised with RASMOL [63].

Server	Web Address
CATH-GRATH	http://www.cathdb.info/cgi-bin/cath/Grath.pl
CATH-SSAP	http://www.cathdb.info/cgi-bin/cath/SsapServer.pl
CE	http://cl.sdsc.edu/ce.html
ContactMetric	http://mammoth.bcm.tmc.edu/cm
DALI	http://www.ebi.ac.uk/dali
DaliLite	http://www.ebi.ac.uk/DaliLite
FAST	http://biowulf.bu.edu/FAST
FATCAT	http://fatcat.burnham.org
FlexProt	http://bioinfo3d.cs.tau.ac.il/FlexProt
FragmentFinder	http://cluster.physics.iisc.ernet.in/ff
LGA	http://predictioncenter.gc.ucdavis.edu/local/lga/lga.html
MAMMOTH	http://ub.cbm.uam.es/mammoth/
Matras	http://biunit.naist.jp/matras/index.html
Pandora	http://www.pandora.cs.huji.ac.il
PRIDE2	http://hydra.icgeb.trieste.it/pride/
POSA	http://fatcat.burnham.org/POSA
ProSa	http://www.came.sbg.ac.at/typo3/index.php?id=prosa
ProCKSI	http://www.procksi.net
ProtoNet	http://www.protonet.cs.huji.ac.il
Everest	http://www.everest.cs.huji.ac.il
Protarget	http://www.protarget.cs.huji.ac.il
SSM	http://www.ebi.ac.uk/msd-srv/ssm
Vorolign	http://www.bio.ifi.lmu.de/Vorolign

Table 1. Protein structure comparison web servers and their URLs.

The FATCAT web server [64] implements a structure comparison algorithm based on *Flexible*

structure Alignment by Chaining Aligned Fragment Pairs allowing Twists [65]. The algorithm takes into account that proteins are flexible and automatically detects hinges and internal rearrangement. First, the algorithm identifies a list of continuously aligned fragment pairs; these are then chained together through the possible introduction of twists (rotation/translation), gaps and simple extensions to improve the superposition and refine the alignment. The similarity measure significance is assessed through a *p*-value that obeys an empirically fitted extreme value distribution. The superposed structures can be visualised using the Chime plugin or RASMOL. A different method for multiple flexible structure comparison using a Partial Order Structure Alignment (POSA) was developed by the same authors [66].

ProCKSI is a workbench and decision support system for *Protein (Structure) Comparison, Knowledge, Similarity and Information*. It implements a protocol for protein structure comparison using the USM comparing contact maps and contact vectors, and a fast meta-heuristic that computes the MaxCMO. ProCKSI also harvests results from other established protein comparison and alignment methods, such as DaliLite, FAST, Vorolign, amongst others [50]. Additionally, it provides further information for each protein, *e.g.*, its classifications from CATH and SCOP, and related scientific literature from iHOP [67]. The collected measures can be analysed with a variety of standard clustering methods that in turn are visualised using a linear, circular or hyperbolic representation of the hierarchical protein structure tree. In addition to the similarity measures provided by ProCKSI, the user may input his own similarity matrix. All available information can be integrated into a unique distance matrix representing a consensus similarity. The current version of ProCKSI is geared towards the comparison of large data sets. This is in contrast to, for example, CATH, SCOP or DALI where the user usually inputs one structure that is then compared to a pre-defined database.

From the above discussion, it is clear that the comparison of protein structures is not a trivial

challenge. This is not simply a reflection that different conceptions of similarity are important in various problems, but also arises from inherent combinatorial aspects. Several particular algorithmic approaches, including stochastic sampling methods and heuristic techniques, have been discussed. We have also touched upon more general strategies, such as the use of lower-detail models (e.g., using just C_α atoms) and divide-and-conquer approaches (like the local hexapeptide alignments employed in DALI). Structure comparison is a core component in the evaluation of structure prediction. The latter is clearly a larger and even more diverse field. We turn to it now and we will again review some specific studies, but also highlight some of the more generic strategies used to make energy evaluations cheaper and to reduce or focus search spaces. We sub-divide the following discussion based on the convenient distinction between lattice and off-lattice models. We describe the use of these models to study and predict protein folding and misfolding.

3. De Novo Protein Structure Prediction using Off-Lattice Models

The protein folding problem (the prediction of a protein's native structure from its sequence) is one of the greatest challenges in structural bioinformatics. The thermodynamic hypothesis states that a protein adopts the structure with the lowest free energy as its native state. So a widely-used approach to structure prediction is to search through the possible structures to find the most stable one. There are two major obstacles to this *de novo* prediction. The first of these is accurately reproducing the relative energies of protein conformations, without requiring excessive computer time. Secondly, the conformational space that a protein can access is so large that an exhaustive search is impossible. We focus mainly on the latter of these problems and describe some search algorithms that have been used to find low energy structures of proteins.

3.1 Search Algorithms

Several search algorithms have been used in studies of protein folding and aggregation. Conventionally, the search process starts from a random conformation and new conformations are

iteratively generated by making random changes and then applying some criteria for acceptance or rejection of each new conformation. Usually all conformations having lower energy are accepted. However, the rejection of all trial conformations with higher energy would lead to a local minimum. So some higher energy conformations should be accepted and the criterion for acceptance influences the overall performance of the algorithm.

A widely used algorithm is the Metropolis Monte Carlo method [68-71]. This is an effective search algorithm and it can also be used to generate the thermodynamic properties of the system being studied. After calculating the energy of the new structure, it is accepted as the starting point for the next move with a probability, p, given by,

$$p = e^{-\Delta E / RT} \qquad \text{(Eqn. 3)}$$

where ΔE is the difference in energy between the initial and new structures, R is the universal gas constant and T is the temperature. In lattice simulations, ΔE, R and T are considered in reduced units and T is an adjustable parameter that can make the search favour local optimisation of a structure or wide coverage of conformational space. Simulated annealing [72-75], where a search starts at a high temperature and is then slowly cooled, is a popular variant. One of the more successful Monte Carlo techniques is replica exchange, in which several Monte Carlo searches are run in parallel at different temperatures [76, 77]. At regular intervals, the structures in two replicas are given the opportunity to exchange, with a probability, p, given by:

$$p = e^{-\left(\frac{1}{RT_j} - \frac{1}{RT_i}\right)\left(E_i - E_j\right)} \qquad \text{(Eqn. 4)}$$

Other modified Monte Carlo algorithms have been used, including ensemble growth Monte Carlo

[78, 79], electrostatically driven Monte Carlo method [80], simulated annealing with genetic crossover [81], the multicanonical algorithm [82] and many others. Genetic algorithms are another well-established search method. These searches start with a population of randomly-generated parent structures, which are combined (crossover) and/or subjected to random changes (mutation) to produce a generation of structures. The most stable child structures become the parents for a new generation of structures. This process is iterated until no better structures are found.

3.2. Force Fields

Protein models in which conformational space is not restricted to a lattice are often termed off-lattice models. To compute the energy of a protein conformation, these models rely on empirical force fields, which take geometric parameters, such as bond lengths and bond angles from crystallographic data [83, 84] and quantum chemical calculations. Such models often consider all atomic degrees of freedom. Thus, off-lattice models represent native structures of proteins more precisely than the lattice models. However, this precision is computationally expensive, which makes it more difficult to locate the global energy minimum [85]. Protein folding has been studied using many off-lattice models of different complexity. Generally, off-lattice models may be divided into residue-level and atomic-level ones. The former are relatively simple, in which each amino acid residue is modelled as a single point, for example, [86, 87]; the latter consider every atom. Atomic-level models can be simplified by the use of so-called extended atoms, whereby aliphatic and aromatic hydrogen atoms are subsumed within their corresponding carbon atoms [69, 73, 88]. The intermediate representation of Fujitsuka *et al*. [74] considers all atoms in the backbone only, while each side chain is represented as a sphere located at the corresponding centre of mass of the real side chain. The model proposed by Irback represents side chains as "large C_β atoms" [75].

There have been few studies of protein aggregation using off-lattice models and search algorithms. Instead, most involve molecular dynamics (MD) algorithms [89], which are beyond the scope of

this review. Mousseau and Derreumaux have modelled the aggregation of several peptides using a united atom representation [90] and Monte Carlo searches. They generate the candidate structures for the Monte Carlo tests using the activation-relaxation technique, which generates off-lattice structures that are local energy minima. These simulations give β-sheets as the most stable structures of small aggregates. In larger aggregates comprising 6-8 chains, other structures, such as β-barrels and micelles, are seen as well.

The force fields of residue-level models are usually similar to those of lattice models: they operate with an empirical potential, derived by statistical analysis of protein databases. Atomic-level models utilize more accurate force fields, based on physical principles and often enhanced by quantum mechanics calculations [73, 84, 91]. Typically, these force fields comprise several terms, including: (a) non-bonded pairwise interactions, usually Lennard-Jones (between all atoms) and electrostatic interactions (between partially charged atoms), (b) orientation-dependent hydrogen-bond interactions and (c) an interaction of the macromolecule with the surrounding solvent. The last one is particularly challenging, as explicit modelling of all solvent atoms is computationally expensive. Therefore, many approaches calculate the solvent effects implicitly [92]. A key aspect is the hydrophobic effect, giving rise to a protein globule containing hydrophobic residues in the core surrounded by polar residues. This is often accounted for by an energy term proportional to the area of the solvent-accessible surface [93]. Although an implicit representation is less computationally expensive than an explicit one, it still requires calculation of the surface area of the protein. Besides these general terms, some additional potentials are sometimes used. For example, Momany *et al.* [84] proposed an empirical "torsion" potential, which recognizes preferences for some particular torsion angles in side chains.

In our own work, we have developed an off-lattice model [94], which represents all heavy atoms, and non-polar hydrogen atoms. Although this is computationally expensive, we have focused some

effort on efficient energy evaluation. For example, energy evaluations are made only for portions of the structure which have changed. The force field of our model is based on some of the work of Scheraga [84, 93], while the solvent effect is enhanced by penalizing charged atoms buried inside protein without establishing hydrogen bonds. This forces them to appear on the surface, in order to have a contact with the solvent. This is a crude treatment of the bulk effect of solvent, which neglects microscopic detail and does not account for specific effects like counter-ions. The speed of the energy calculation is further enhanced by discretisation of the energy values and the use of pre-calculated tables.

The kinematics of all-atom models includes rotations around all bonds, except peptide bonds, which are kept planar. More detailed models also consider rotation around peptide bonds. Additionally, one can allow variation of bond angles and bond lengths [91]. Rotation around dihedral angles is a useful strategy for searching conformational space. Several approaches change a number of dihedral angles together [86, 94]. Avbelj and Moult [69] suggested choosing a single residue and changing all its associated dihedral angles. They also collected a library of most likely angles for each amino acid, which were adopted by preference when generating a new conformation. This suggestion was followed in some other studies [95]. One algorithm [96] changes seven consecutive torsion angles: here, one angle is chosen arbitrarily and others are identified so that the new conformation produced by the move is geometrically closest to the current one. This is believed to help avoid local minima [97].

In our model we did not use a library of most likely angles, as this might unduly restrict the search process by precluding important intermediate structures. We employed random rotations around dihedral angles (including peptide bonds). The single rotations are combined with seven-fold moves to give a local deformation of the current state. In addition to this, it was helpful to employ co-axial rotations of peptide groups, in which two neighbouring amino acids are chosen and the angles φ_i

and ψ_{i+1} are changed by the same value in different directions. The axes of these rotations are almost parallel, so the overall transformation of polypeptide chain is close to parallel shift. This provides relatively small overall transformation, while the calculation of parameters of such a move is much easier than for the seven-fold move.

3.3 Application of the Great Deluge Algorithm to an Off-Lattice Model

Wenzel and co-workers [73, 88, 95, 98] have applied several specially designed techniques, such as the basin hopping method (with artificial energy minimization of intermediate conformations), the stochastic tunnelling method (where the probability of an uphill move is calculated on the basis of the best achieved conformation) and parallel tempering. Genetic algorithms have been applied to the folding of short proteins [86]. However, these methods are quite sensitive to parametrisation and require a preliminary setting up of the necessary parameters. In an attempt to avoid this, we have applied [94] the Great Deluge optimisation method [99]. This algorithm does not require preliminary parametrisation and allows one to specify the total search time in advance. The Great Deluge local search is an iterative procedure, where at each step a new conformation is randomly selected from a set of candidates generated from the current conformation (its neighbourhood). The chosen candidate is accepted as the new current conformation if it fits into an artificial feasible space, which is gradually reduced during the search. This mechanism, unlike the Monte Carlo method, makes the local search process highly controllable by the user. In particular, it allows improvements to the accuracy of prediction by regulating the processing time and exploring different areas of a multiobjective search space. To make the algorithm more effective, different neighbourhood structures are explored with different priority, for example, the rotation of the backbone compared to side chains.

Most successful all-atom folding has been reported for short chains forming α-helices (or a bundle of a few α-helices) or β-hairpins, e.g., [69, 75, 86, 94]. Another popular benchmark is the 20-residue

Trp-cage protein [88, 100], whose native state consists of one α-helix and a polyproline II conformation. Other examples are the 56-residue fragment of the B-domain of staphylococcal protein A [80] and the 60-residue bacterial ribosomal protein [95]. We applied the Great Deluge algorithm to some these literature systems, with the main purpose of investigating the detailed processes followed by our search procedure on different proteins and to identify the major obstacles that should be overcome. One challenge for simulation and search based protein structure prediction is achieving reliable and consistent folding to the native conformation over many runs. However, our experiments showed quite high variation across different runs. Although in the best cases the lowest energy conformations were relatively close to native state (Table 2), these were achieved in only approximately 10-20% of cases.

This situation is often indicative of premature convergence, where the search has frozen before reaching the global optimum. This tendency is associated with high energy barriers between geometrically close conformations. In many runs, our algorithm was able to approach the native state in the middle of the search, but subsequently moved away from it. An example of the search profile for the Trp-Cage protein, where the initial structure was a single α-helix, is given in Figure 4. The algorithm relatively quickly finds a conformation around 3 Å RMSD to the native state, but

Protein	PDB Code	Length	RMSD to native state (Å)
Hydrophilic amphipathic helical basic peptide	1DJF	14	0.65
Beta-hairpin peptide	1J4M	14	2.37
TRP-Cage miniprotein	1L2Y	20	2.07
Beta-beta-alpha peptide	1FME	28	4.34
Thermostable subdomain from chicken villin headpiece	1VII	36	5.28
Peripheral subunit-binding domain of dihydrolipoamide acetyltransferase	2PDD	43	4.92
C-terminal domain of DNA fragmentation factor alpha subunit	1KOY	62	5.36

Table 2. Lowest RMSD results of the Great Deluge algorithm on short proteins.

then almost immediately moves away to less native-like structures. In the middle of the search,conformations close to the native state (RMSD ≈ 2Å) are found, but again the search moves away.

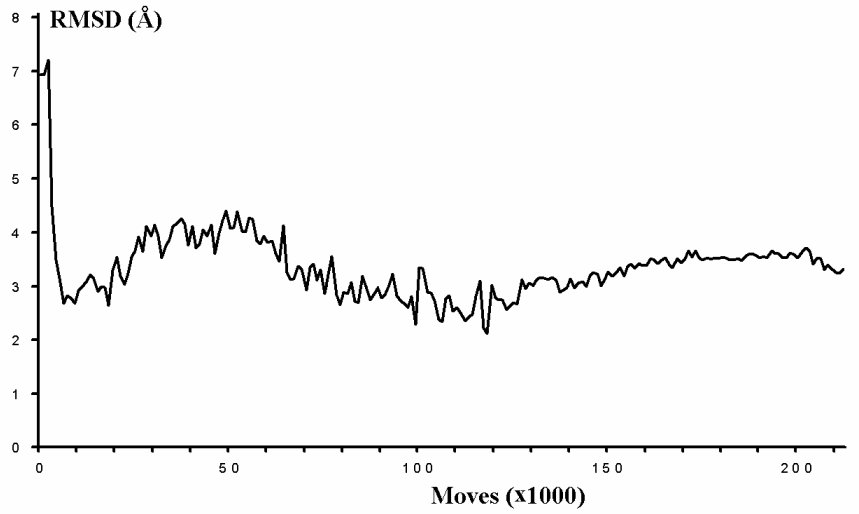

Figure 4. The search profile for 20-residue Trp-Cage protein

This behaviour may be caused by biasing of the search procedure into different regions of the search space. One promising direction for improving the performance of optimisation methods may be the investigation of the role of different moves in the folding process, for example, exploration of adaptive methods with different acceptance condition for different moves. Increasing or reducing the proportion of rotation of backbone to side chains can force a polypeptide chain to fold into α-helix or β-sheet respectively. This illustrates the potentially dramatic effect of biasing. Several other aspects are also a subject of further investigations, including the role of initial conformation (random compared to a regular structure, such as a fully extended state or a helical structure) as well as processing time requirements for different phases of the search process.

4. Lattice Models

In a simplified model, the protein conformation may be restricted, such that each residue occupies a different vertex on a lattice. The protein chain is self-avoiding and consecutive residues in the sequence occupy adjacent positions on the lattice. This effectively sets the inter-residue distance (3.9 Å on average for real proteins) to the length of the lattice spacing. The two-dimensional triangular and square lattices are frequently used. Clearly, these are not physically realistic, but they can be used to model some of the principles underpinning protein folding and to develop search algorithms. In three dimensions, frequently used lattices are diamond, cubic and face-centred cubic, with each residue surrounded by four, six or twelve others.

Simplified models have played a key role in shaping our understanding of proteins and in shedding light upon the type of algorithms that may work well for protein structure prediction. Minimalist models have been used, amongst other things, to study the nature of the energy landscape [101], the uniqueness of the native state [102], the origin of two-state thermodynamic characteristics of protein folding [103], and structure-function influences on evolution [104]. Simplified models have also been used in real-world structure prediction, by combining experimental information about secondary and tertiary structure with optimised conformation from lattice simulations [105, 106]. Thus, simplified protein models have contributed to our understanding of the fundamental physics of proteins, whilst paving the way for the development of algorithms for the prediction of native conformations.

An important part of any lattice model is the potential used to compute the energies of structures. Probably the best known is the HP model [107], which abstracts the hydrophobic interaction central in the folding process, by reducing the 20 naturally occurring amino acids to a binary alphabet {H, P}, of hydrophobic and polar residues. In the HP model, two hydrophobic residues have an interaction energy of $-|\varepsilon|$, if they are in contact (and are not adjacent in sequence). All other

interaction energies are zero. One interesting variant of the HP model involves shifting the average interaction energy between residues, by introducing repulsive forces, so that two hydrophobic residues in contact have an interaction energy of -2ε and all other residue-residue contacts have an interaction energy of ε, where ε is positive (and may be set to unity). The shifted-HP model has been used to define native conformations with non-maximally compact structures, which often have a binding pocket, *i.e.*, an empty lattice surrounded by residues [104]. Such model proteins, thus, have a minimalist function in addition the usual features of the standard HP model, and these functional model proteins have been used to investigate the distinct influences of structure and function on evolution [108-110]. Figure 5 shows functional model proteins embedded in a square lattice and a diamond lattice. To be deemed a viable protein, a functional model protein must fold into a unique native state and the native structure is required to have a binding pocket. Moreover, there must be an energy gap between the minimum energy conformation and the next excited state. These constraints present additional challenges to search algorithms applied to the folding of functional model proteins: getting trapped in local minima is more likely (due to the energy gap), algorithms cannot exploit assumptions about the compactness of native structures and, unlike many standard HP model proteins, there are no degenerate global minima. Many examples of functional model proteins can be downloaded from: http://www.cs.nott.ac.uk/~nxk/hppdb.html.

The HP model captures many of the properties of proteins, but models including more types of residue can provide a more realistic representation. The HP model has been extended by inclusion of positively- and negatively-charged residues to make the HCPC potential [111, 112]. A more detailed model is the Miyazawa-Jernigan (MJ) potential [113], which includes pairwise interaction energies between all 20 naturally occurring amino acids. These terms are derived from the contact frequencies in a large set of protein structures. There are many extensions to the MJ potential that include additional energy terms, to model properties like solvation [114] or high packing density

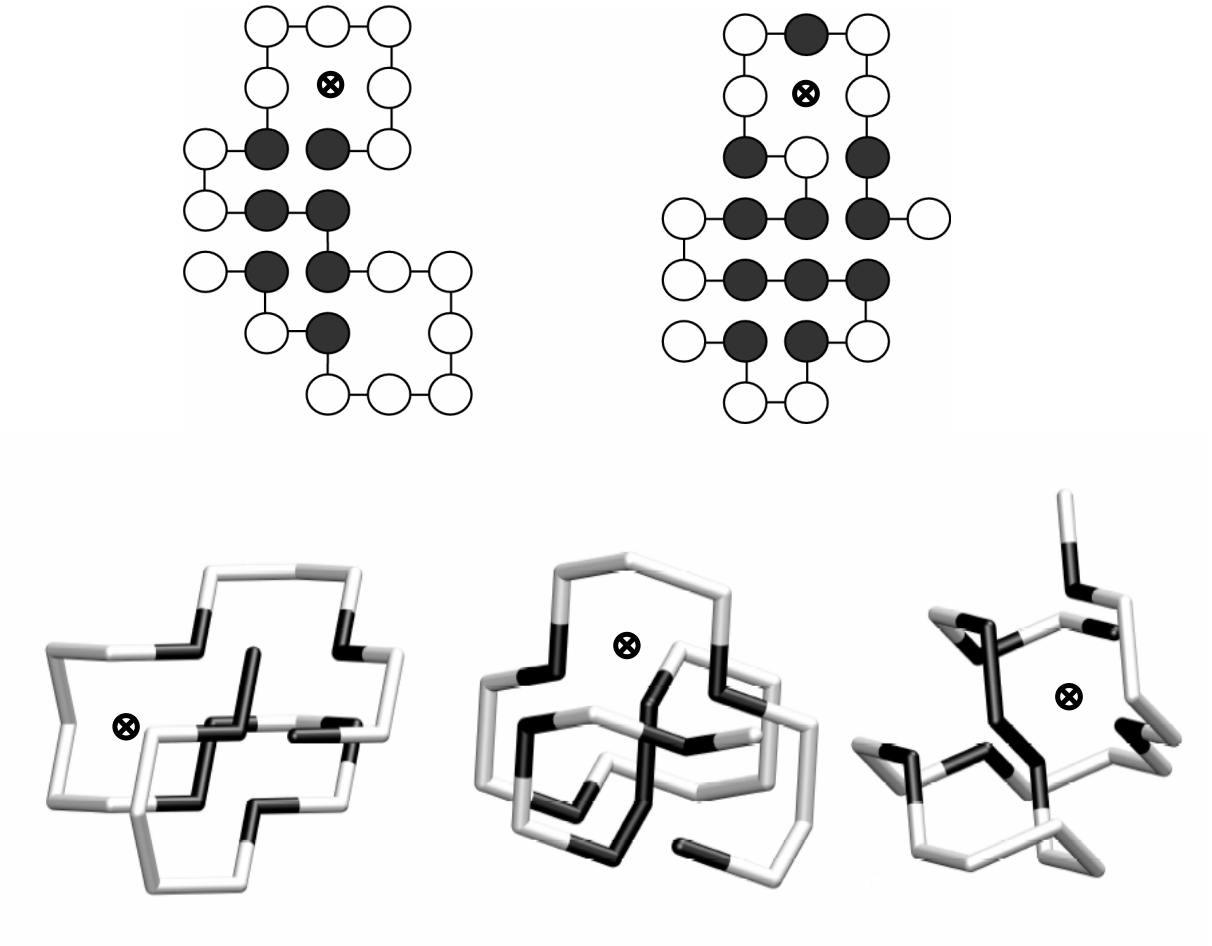

Figure 5. Functional model proteins on the square (upper panel) and diamond lattice (lower panel). Hydrophobic residues are black, polar residues white. Binding pockets (⊗) are evident in all cases.

[115]. Another simple model is the Gō model [116], in which all contacts that are present in the native state of the protein contribute an equal stabilising energy and any other contacts make no contribution to the energy.

Minimalist models greatly simplify the underlying complexity of structure prediction. However, in the worst case, they remain computationally intractable. Although several approximation algorithms with guaranteed performance exist [117], in practical benchmarking these perform much worse than current metaheuristics. Metaheuristics can be defined as "a master strategy that guides and modifies other heuristics to produce solutions beyond those that are normally generated in a quest for local optimality" [118]. Some metaheuristics that have been used to study simplified models include genetic algorithms [119, 120], evolutionary Monte Carlo hybrids [121], memetic algorithms [122],

ant algorithms [123-125] and genetic tabu seach hybrids [126].

4.1 Memetic Algorithms Applied to Lattice Models

Memetic algorithms are evolutionary algorithms that include, as part of the evolutionary cycle of crossover-mutation-selection, a local search stage. In our implementation [122], in the local search phase, the algorithm has access to several different local search move operators from which it can select, according to how the search progresses. As multiple local search strategies are utilised, this approach is termed a multimeme algorithm. A population of "individuals" (rather than solutions) is kept. An individual comprises its genetic material (which represents a candidate protein structure) and its memetic material (which defines the local search move operator to use). The mechanisms of genetic exchange and variation are the usual crossover and mutation operators, albeit tailored to the specific problem at hand. During the crossover stage, new individuals are created. The offspring inherit both genetic and memetic material; the local search move operator of the parent with the lowest energy conformation is inherited. In order to avoid revisiting previous regions of the search space, a (contact map) memory is introduced in the mating stage.

The local search strategies available to the multimeme algorithm are (Figure 6). Pivot moves, stretch or unfolding of a substructure, random macro-mutation of a substructure, reflection of a substructure, non-local k-opt (see below) and local k-opt. Substructures of lengths 4, 8 and 16 were considered. Stretch operators facilitate disentangling structures early in the search and also provide a means of unfolding from local minima. Macro-mutation involves randomly shuffling substructures (of lengths 4, 8 or 16). For the k-opt move operators, k is the number of residues considered for re-positioning (k = 2, 3 or 4). Local k-opt moves involve residues which are consecutive in sequence; in the non-local version, the residues are not neighbours in the sequence.

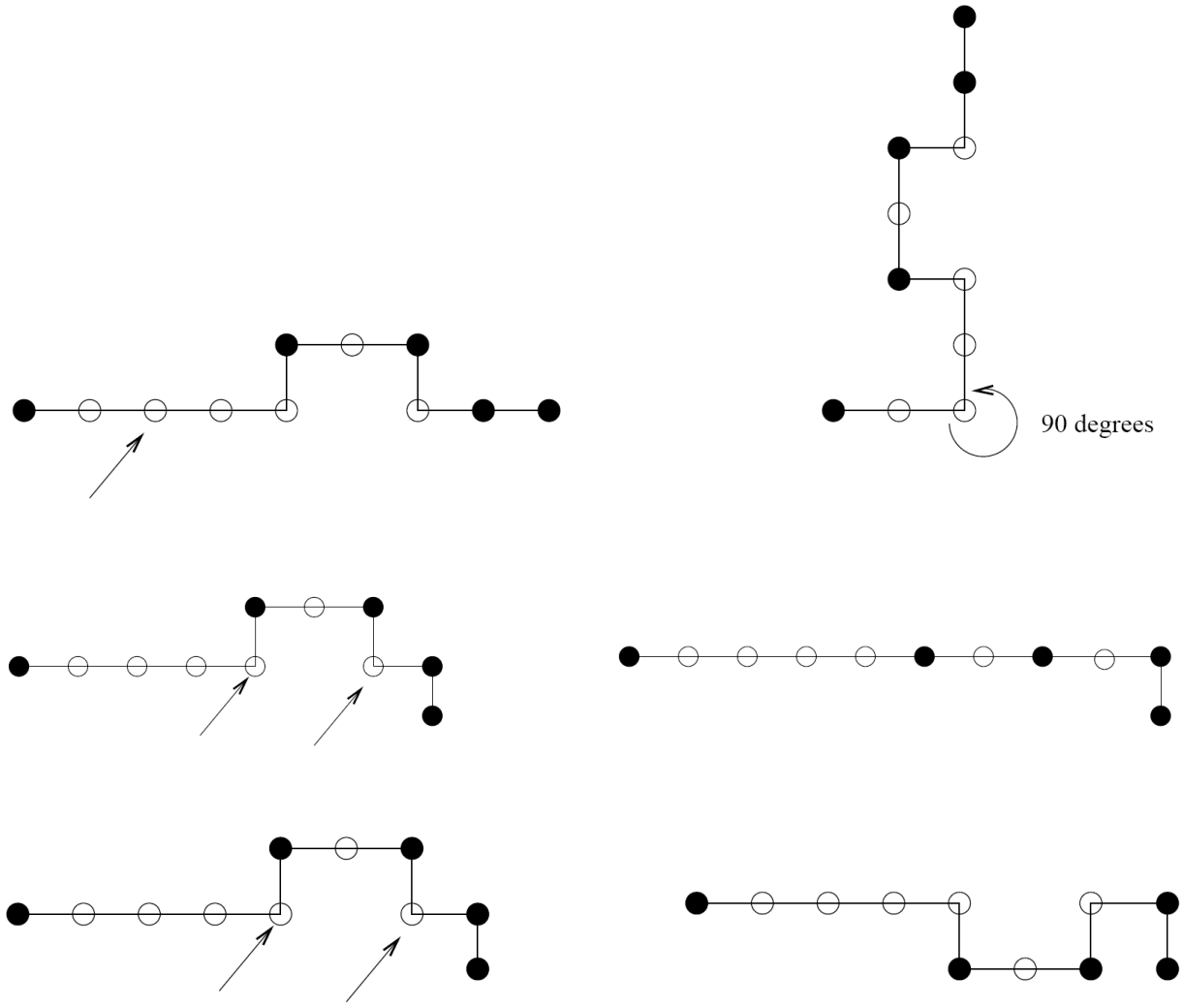

Figure 6. Local search move operators: left – initial structure; right – resulting structure; upper panel – pivot move; centre panel – stretch; lower panel – reflection.

In addition, a contact map memory was included into the multimeme algorithm, using a hash table, with entries of the form (*key*, *occupancy*). The *key* represents the contact of two residues. A contact was taken to be compatible if no more than 66% of the individuals already in the population shared that contact. During the reproduction phase, each generated offspring was deemed compatible with the memory if at least 25% of the contacts in the structure were compatible. These fractions were determined empirically. As a contact map can be realized by several different structures,

compatibility with the contact map memory drives the search towards a more exploratory regime, thereby increasing diversity in the population. This strategy improved on results previously obtained with multimeme algorithms on the standard HP model [47] and also solved to optimality instances of functional model proteins that our previous algorithms were not capable of solving.

Each generation of the multimeme algorithm consisted of a mating stage (two-point crossover with tournament selection), mutation (one and two-point mutation), local search and replacement. The basic evolutionary parameters and settings for the multimeme algorithm were tournament sizes of two and four for the mating stage. A crossover probability of 0.8 and a mutation probability of 0.3 were used. The runs were executed based on a (μ, λ) replacement strategy, where at a given generation of population of μ individuals produces λ offspring; (μ, λ) of (50, 200), (100, 400) and (500, 1000) were examined. Every individual in the population underwent three iterations of optimisation (stopping short of convergence) with the local search move operator specified in the memetic material. Other parameters were set according to the criteria described previously [47].

Five independent runs were executed for each instance. Some standard HP model sequences on the square lattice taken from the literature [119, 121] and the optima found by the multimeme algorithm are shown in Table 3. Several examples were considered and Table 3 compares the number of energy evaluations made by the multimeme algorithm and by two other established methods: the genetic algorithm and the Monte Carlo method reported Unger and Moult [119]. The evolutionary Monte Carlo method [121] solves to optimality all the instances in Table 4 (and longer instances), but direct comparison of the number of energy evaluations is not readily made, as only the number of energy evaluations on feasible conformations (excluding the infeasible conformations) was reported.

#	Sequence	Length	E_{opt}	E_{MMA}
1	HPHPPHHPHPPHPHHPPHPH	20	-9	-9
2	$P^3HHPPHHP^5H^7PPHHP^4HHPPHPP$	36	-14	-14
3	$HH(PH)^4H^3PHP^3HP^3HP^4HP^3HP^3HPH^4(PH)^4H$	50	-21	-21
4	$H^{12}(PH)^2(P^2H^2)^2(PPH)^2(HP^2H)^2(P^2H^2)^2P^2(HP)^2H^{12}$	64	-42	-39[a]
5	HPHPPHHPHPPHPHHPPHPH	20	-9	-9
6	$PPHPPHHP^4HHP^4HHP^4HH$	25	-8	-8
7	$(P^2H)^2HPPHHP^5H^{10}P^6(H^2P^2)^2HPPH^5$	48	-22	-23[b]
8	$PHPPHPH^3PHHPH^5$	18	-9	-9
9	$HPHPH^3P^3H^4PPHH$	18	-8	-8
10	$HHP^5HHP^3H^4PPHH$	18	-4	-4
11	$H^3PPHPHPHPPHPHPHPPH$	20	-10	-10
12	$PPH^3PH^8P^3H^{10}PHP^3H^{12}P^4H^6PHHPHP$	60	-34	-35[b]

Table 3. Instances of the standard HP model on a square lattice, with literature optimum energies, E_{opt}, and the lowest energies found by the multimeme algorithm, E_{MMA}. [a]Multimeme algorithm did not find optimum energy. [b]New optimum discovered by the multimeme algorithm.

The pruned-enriched Rosenbluth method (PERM), which uses Monte Carlo based chain growth to generate partial conformations and prunes unfavourable conformations, performs very well on standard HP model sequences on the square lattice [127], although, interestingly, it also fails to find the global minimum of instance 4 in Table 3. PERM utilizes sequence-specific information and has been tailored to find highly compact structures. Consequently, it is superior to the multimeme algorithm on the standard HP model. However, on functional model proteins, which are not

Sequence # (from Table 2)	Number of energy evaluations		
	Genetic algorithm	Monte Carlo	Multimeme algorithm
1	30492	292443	14621
2	301339	6557189[a]	208233
3	592887	15151203	336763
6	20400	2694572	18736
7	126547	9201755[b]	1155656

Table 4. For HP model sequences on the square lattice, comparison of the number of energy evaluations used by the memetic algorithm with literature methods [119]. [a]Monte Carlo method finds only a local optimum (of energy -13). [b]Monte Carlo method finds only a local optimum (of energy -20).

maximally compact (so that they can accommodate a binding pocket), we anticipate that PERM would be inferior to the multimeme algorithm. For longer instances of this model, PERM does not always finish within an allocated running time of two days, in contrast to the multimemetic algorithm. Recently, methods based on artificial immune systems [128, 129], ant colony optimisation [123-125, 130, 131] and estimation of distribution algorithms [132] have also been applied to the structure prediction of minimalist models.

The construction of effective algorithms for structure prediction on simplified models, like the HP model and functional model proteins, is a stepping-stone towards structure prediction of real proteins that are not amenable to homology or threading methods. Simplified models can be used to seed searches employing more detailed models. Therefore, improvement in optimisation techniques for lattice models is to be welcomed. Strategies that can be useful include the combination of global

and local search methods, mechanisms (such as the contact map memory) for preserving diversity in a population of solutions, and mechanisms for escaping poor local optima and traversing extended neutral plateaus. The multimeme algorithm discussed above embodies these strategies, and proves to be robust across many lattice types (triangular, square, diamond) and models (standard HP and functional model proteins).

4.2 Lattice Studies of Aggregation

Many of the challenges of modelling aggregates are the same as those with single proteins, but with multiple chains the computational demands are much greater. As always, there is a compromise between detail and speed, and models ranging from simplified lattice models to all-atom off-lattice models have been employed to study the aggregation of proteins. Some short peptide sequences with four or more residues are known to form amyloid fibrils with structures very close to those formed by amyloidogenic proteins. Many computational studies focus on these short peptides to reduce the size of the optimisation problem.

When studying the folding of single proteins, the effects of the simulation's boundary can be ignored. However, for aggregates the boundary conditions are important, because concentration has a large influence on aggregation. This is a particularly serious problem for lattice models, where there are no long-range forces, so a protein chain that dissociates from an aggregate would be free to drift away without a boundary. The problem can be solved by confining the aggregate to a fixed volume by adding either a solid boundary or a periodic boundary.

The square lattice combined with the HP potential is simple enough that, for short chains, all of the structures of all possible sequences of a given length can be enumerated. The dimerisation of short HP peptides with up to 16 residues has been studied by this method [133]. Giugliarelli *et al.* used this approach to look at 16 and 25 residue lattice proteins [134], but reduced the size of the problem

by only considering maximally compact structures, *i.e.*, those that fill a 4×4 or 5×5 square.

For larger systems, search algorithms must be used. The earliest example of protein aggregation on a lattice [135] involved Monte Carlo simulations on a model 20-residue HP peptide with a unique native state. Increasing the strength of the HH interactions led to the peptide becoming trapped in mis-folded conformations. When systems comprising up to 40 chains were simulated, some of these conformations were prone to aggregation. Recently, Nakanishi and Kikuchi [136] performed a rigorous thermodynamic analysis on the aggregation of two HP proteins. To overcome problems with the excluded volume, they used multi-self-overlap-ensemble Monte Carlo, which allows chains to move through each other. They studied the effect of concentration on aggregation and found that a dimer is thermodynamically stable only when the system is confined to a small volume. Two groups have performed Monte Carlo searches on square lattices with the HCPC model [137, 138]. As in the HP studies, both systematic enumeration and Monte Carlo searches were used. They both found sequences that folded into a single native state when only one chain was considered. However, when multiple chains were considered, aggregates could form from several self-propagating conformations.

The cubic lattice is a simple three-dimensional model. Bratko and Blanch have performed Monte Carlo and replica exchange searches on this lattice using a version of the Gō potential that had been extended to include contacts between chains [139, 140]. Due to the high specificity of the Gō model, these proteins are relatively stable with respect to aggregation. Most of the work on the cubic lattice has used the MJ potential [114, 141]. Broglia *et al*. found that having two sites on different chains that interact strongly leads to irreversible aggregation [141]. To avoid this, Leonhard *et al*. introduced a solvation energy term to the MJ potential to reduce the strength of these interactions [114]. Subsequent studies [142-144] have used this modified MJ potential for thermodynamic analysis of aggregation.

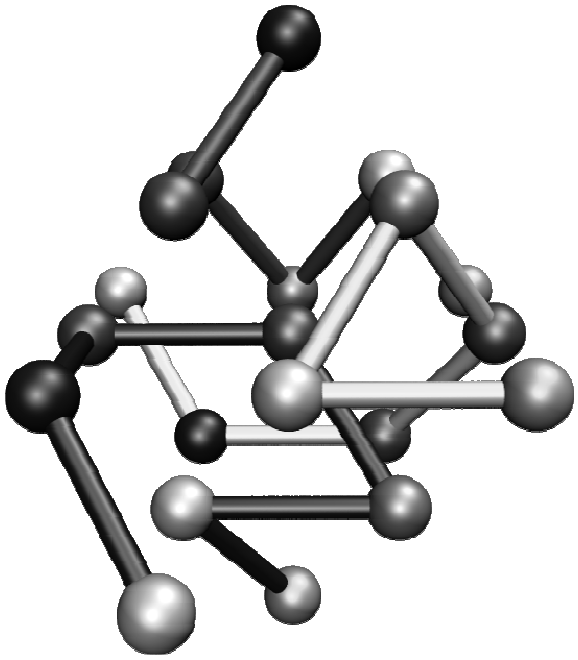

Figure 7. A model aggregate on the face centred cubic lattice.

We have developed a model of protein aggregation based on the face-centered cubic (FCC) lattice [156]. This has a finer resolution than the cubic lattice, with each point surrounded by 12 others. An example conformation is shown in Figure 7. The FCC lattice matches the structures of proteins more closely than the cubic lattice, but comes at an increased computational cost. With a finer lattice, it is worthwhile using a more detailed potential to calculate the energy of structures. We use the MJ potential including a later modification that introduces a repulsive term for any residues with a high packing density [115], which becomes more important as the size of an aggregate increases. This is the most detailed model that has been used to study aggregation on a lattice. However, this model produces with little recognisable secondary structural elements like α-helices or β-sheets and it needs to be improved by including interactions such as hydrogen bonding. We have studied the aggregation of three short peptides that are known to undergo aggregation. Several Monte Carlo based algorithms were used, with the replica exchange Monte Carlo method consistently finding more stable structures than any other algorithm. We also used the tabu search algorithm [118],

which is designed to force searches into new areas of conformational space by maintaining a list of previously visited structures that cannot be revisited. However, it generally performed worse than the Monte Carlo based algorithms.

5. Conclusion

In this review, we have focused on a selection of applications within structural bioinformatics, to highlight the application of novel search strategies to optimisation problems. We have emphasized the use of reduced models, for example, contact maps in protein structure comparison and lattice models in protein structure prediction and studies of aggregation. In the context of the structure prediction of HP lattice models, we have described the use of multi-memetic algorithms, a hybrid metaheuristic, which combines genetic algorithms with local search. In the domain of off-lattice protein folding, we have discussed the Great Deluge algorithm. Clearly, these structural bioinformatics problems are challenging, and will require continued innovation not only in search strategies and optimisation techniques, but also in the representation of protein structures and the force fields modelling the energetic interactions that govern protein folding and aggregation.

Acknowledgments

We are grateful for access to the University of Nottingham's high performance computer and for financial support from EPSRC (GR/T07534/01), BBSRC (grant BB/C511764/1) and the EU (NoE BIOPATTERN: contract no. FP6-508803).

References

[1] Hirst, J.D. (**2002**) in *Modern Protein Chemistry*, (Howard, G. C. and Brown, W. E., Eds.). pp. 123-144. CRC Press LLC, Boca Raton, Florida, USA.

[2] Kolodny, R., Petrey, D. and Honig B. (**2006**) *Curr. Opin. Str. Biol.*, *16*, 393-398.

[3] Siew, N., Elofsson, A., Rychlewski, L. and Fischer, D. (**2000**) *Bioinformatics*, *6*, 776-785.

[4] Krasnogor, N. and Pelta, D.A. (**2004**) *Bioinformatics*, *20*, 1015-1021.

[5] Melville, J.L., Riley, J. F. and Hirst, J.D. (**2007**) *J. Chem. Inf. Model.*, *47*, 25-33.

[6] Li, M., Chen, X., Li, X.; Ma, B. and Vitanyi, P.M.B. (**2004**) *IEEE Trans. Inf. Theory*, *50*, 3250-3264.

[7] Moult, J. (**2006**) *Phil. Trans. R. Soc. B*, *361*, 453–458.

[8] Moult, J., Fidelis, K., Rost, B., Hubbard, T. and Tramontano, A. (**2005**) *Proteins: Struct., Funct., Bioinf.*, *61, Suppl. 7*, 3-7.

[9] Holm, L. and Sander, C. (**1996**) *Science*, *273*, 595-602.

[10] Koehl, P. (**2001**) *Curr. Opin. Struct. Biol.*, *11*, 348-353.

[11] Kolodny, R., Koehl, P., Guibas, L. and Levitt, M. (**2002**) *J. Mol. Biol.*, *323*, 297-307.

[12] Tress, M., Ezkurdia, I., Graña, O., López, G. and Valencia, A. (**2005**) *Proteins: Struct., Funct., Bioinf.*, *61, Suppl. 7*, 27-45.

[13] Koh, I.Y.Y., Eyrich, V.A., Marti-Renom, M.A., Przybylski, D., Madhusudhan, M.S., Eswar, N., Graña, O., Pazos, F., Valencia, A., Sali, A. and Rost, B. (**2003**) *Nucleic Acids Res.*, *31*, 3311-3315.

[14] Chothia, C. and Lesk, A.M. (**1986**) *EMBO J.*, *5*, 823-826.

[15] Sierk, M.L. and Pearson, W.R. (**2004**) *Protein Sci.*, *13*, 773-785.

[16] Zhu, J. and Weng, Z. (**2005**) *Proteins: Struct., Funct., Bioinf.*, *58*, 618-627.

[17] Lancia, G. and Istrail, S. (**2003**) *Protein Structure Comparison: Algorithms and Applications*. Springer-Verlag, Heidelberg, Germany.

[18] Orengo, C.A., Pearl, F.M.G., Bray, J.E., Todd, A.E., Martin, A.C., Lo Conte, L. and Thornton,

J.M. (**1999**) *Nucleic Acids Res*., *27*, 275-279.

[19] Novotny, M., Madsen, D. and Kleywegt, G.J. (**2004**) *Proteins: Struct., Funct., Bioinf.*, *54*, 260-270.

[20] Alexandrov, N.N., Takahashi, K. and Gō, N. (**1992**) *J. Mol. Biol*., *225*, 5-9.

[21] Vriend, G. and Sander, C. (**1991**) *Proteins: Struct., Funct., Gen.*, *11*, 52-58.

[22] Holm, L. and Sander, C. (**1993**) *J. Mol. Biol*., *233*, 123-138.

[23] Pelta, D.A., Krasnogor, N., Bousono-Calzon, C., Verdagay, J.L., Hirst, J.D. and Burke, E. (**2005**) *Fuzzy Sets and Systems*, *152*, 103-123.

[24] Caprara, A., Carr, R., Istrail, S., Lancia, G. and Walenz, B. (**2004**) *J. Comput. Biol*., *11*, 27-52.

[25] Krasnogor, N. (**2004**) *Genetic Programming and Evolvable Machines*, *5*, 181-201.

[26] Caprara, A. and Lancia, G. (**2002**) In *Proceedings of the Research in Computational Molecular Biology Conference (RECOMB)* ACM Press, New York, NY, USA, 100-108.

[27] Carr, B., Hart, W., Krasnogor, N., Burke, E.K., Hirst, J.D. and Smith, J. (**2002**) In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1027-1034.

[28] Artymiuk, P.J., Poirrett, A.R., Rice, D.W. and Willet, P. (**1995**) *Top. Curr. Chem.*, *174*, 73-103.

[29] Wu, T.D., Schmidler, S.C., Hastie, T. and Brutlag, D.L. (**1998**) *J. Comput. Biol*., *5*, 585-595.

[30] Leluk, J., Konieczny, L. and Roterman, I. (**2003**) *Bioinformatics*, *19*, 117-124.

[31] Shindyalov, I.N. and Bourne, P.E. (**1998**) *Protein Engng*., *11*, 739-747.

[32] Yang, A.S. and Honig, B. (**2000**) *J. Mol. Biol*., *301*, 665-678.

[33] Taylor, W.R. (**1999**) *Protein Sci*., *8*, 654-665.

[34] Gerstein, M. and Levitt, M. (**1998**) *Protein Sci., 7*, 445-456.

[35] Szustakowski J.D. and Weng Z.P. (**2000**) *Proteins: Struct., Funct., Bioinf.*, *38*, 428-440.

[36] Kabsch, W. (**1978**) *Acta Crystallogr., Sect. A: Cryst. Phys., Diffr., Theor. Gen. Cryst.*, *34*, 827-828.

[37] Damm, K. L. and Carlson, H. A. (**2006**) *Biophys. J.*, *90*, 4558-4573.

[38] Zemla, A. (**2003**) *Nucleic Acids Res.*, *31*, 3370-3374.

[39] Vincent, J.J., Tai, C.H., Sathyanarayana, B.K. and Lee, B. (**2005**) *Proteins: Struct., Funct., Bioinf.*, *61*, *Suppl. 7*, 67-83.

[40] Krissinel, E. and Henrick, K. (**2004**) *Acta Crystallogr., Sect. D: Biol. Crystallogr.*, *60*, 2256-2268.

[41] Havel, T.F., Kuntz, I.D. and Crippen G.M. (**1983**) *Bull. Math. Biol.*, *45*, 665-720.

[42] Vendruscolo, M., Kussell, E. and Domany, E. (**1997**) *Folding Des.*, *2*, 295-306.

[43] Pollastri, G., Baldi, P., Fariselli, P. and Casadio, R. (**2002**) *Proteins: Struct., Funct., Bioinf.*, 47, 142-153.

[44] Bacardit, J., Stout, S., Hirst, J.D., Blazewicz, J. and Krasnogor, N. (**2006**) In *Proceedings of the 8th Genetic and Evolutionary Computation Conference (GECCO)* ACM Press, 247-254.

[45] Lisewski, A. M. and Lichtarge, O. (**2006**) *Nucleic Acids Res.*, *34*, e152.

[46] Birzele, F., Gewehr, J. E., Csaba, G. and Zimmer, R. (**2007**) *Bioinformatics*, *23*, e205-e211.

[47] Krasnogor, N. (**2002**) PhD thesis, University of West England, Bristol, UK; http://www.cs.nott.ac.uk/~nxk/PAPERS/thesis.pdf

[48] Gramm, J. (**2004**) *IEEE/ACM T. Comput. Biol. Bioinformat.*, *1*, 171-180.

[49] Strickland, D.M., Barnes, E. and Sokol, J.S. (**2005**) *Operations Res.*, *53*, 389-402.

[50] Barthel, D., Hirst, J.D., Blacewicz, J., Burke, E.K. and Krasnogor, N. (**2007**) *BMC Bioinf.*, *8*, 416.

[51] Cilibrasi, R. and Vitanyi, M.B. (**2005**) *IEEE Trans. Inf. Theory*, *51*, 1523-1545.

[52] Kocsor, A., Kertesz-Farkas, A., Kajan, L. and Pongor, S. (**2006**) *Bioinformatics*, *22*, 407-412.

[53] Galperin, M.Y. (**2006**) *Nucleic Acids Res.*, *34*, D3-D5.

[54] Murzin, A.G., Brenner, S.E., Hubbard, T. and Chothia, C. (**1995**) *J. Mol. Biol.*, *247*, 536-540.

[55] Andreeva, A., Howorth, D., Brenner, S.E., Hubbard, T.J., Chothia, C., Murzin, A.G. (**2004**) *Nucleic Acids Res.*, *32*, D226-D229.

[56] Pearl, F., Todd, A., Sillitoe, I., Dibley, M., Redfern, O., Lewis, T., Bennett, C., Marsden, R., Grant, A., Lee, D., Akpor, A., Maibaum, M., Harrison, A., Dallman, T., Reeves, G., Diboun, I., Addou, S., Lise, S., Johnston, C., Sillero, A., Thornton, J. and Orengo, C. (**2005**) *Nucleic Acids Res.*, *33*, D247-D251.

[57] Mizuguchi, K, Deane, C.M., Blundell, T.L. and Overington, J.P. (**1998**) *Protein Sci.*, *7*, 2469-2471.

[58] Sowdhamini, R., Burke, D.F., Huang, J.F., Mizuguchi, K., Nagarajaram, H.A., Srinivasan, N., Steward, R.E. and Blundell, T.L. (**1998**) *Structure*, *6*, 1087-1094.

[59] http://bioinformatics.ca/links_directory/?subcategory_id=136

[60] Ananthalakshmi, P., Kumar, C.K., Jeyasimhan, M., Sumathi, K. and Sekar, K. (**2005**) *Nucleic Acids Res.*, *33*, W85-W88.

[61] Russell, R.B. and Barton, G.J. (**1992**) *Proteins: Struct., Func., Gen.*, *14*, 309-323.

[62] Martin, A.C.R. (**2005**) ProFit, http://www.bioinf.org.uk/software/profit.

[63] Sayle, R.A. and Milner-White, E.J. (**1995**) *Trends Biochem. Sci.*, *20*, 374-376.

[64] Ye, Y. and Godzik, A. (**2004**) *Nucleic Acids Res.*, *32*, W582-W585.

[65] Ye, Y. and Godzik, A. (**2003**) *Bioinformatics*, *19*, ii246-ii255.

[66] Ye, Y. and Godzik, A. (**2005**) *Bioinformatics*, *21*, 2362-2369.

[67] Hoffmann, R. and Valencia, A. (**2005**) *Bioinformatics*, *21 Suppl 2*, ii252-ii258.

[68] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. and Teller, E. (**1953**) *J. Chem. Phys.*, *21*, 1087-1092.

[69] Avbelj, F. and Moult, J. (**1995**) *Proteins: Struct., Funct., Gen.*, *23*, 129-141.

[70] Kussell, E., Shimada, J. and Shakhnovich, E.I. (**2002**) *Proc. Natl. Acad. Sci. USA*, *99*, 5343-5348.

[71] Zagrovic, B., Snow, C.D., Shirts, M.R. and Pande, V.S. (**2002**) *J. Mol. Biol.*, *323*, 927-937.

[72] Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P. (**1983**) *Science*, *220*, 671-680.

[73] Herges, T. and Wenzel, W. (**2004**) *Biophys. J.*, *87*, 3100-3109.

[74] Fujitsuka, Y., Chikenji, G. and Takada, S. (**2006**) *Proteins: Struct., Funct., Bioinf.*, *62*, 381-398.

[75] Irback, A. (**2003**) *J. Phys.: Cond. Mat*., *15*, S1797-S1807.

[76] Swendsen, R.H. and Wang, J.S. (**1986**) *Phys. Rev. Lett*., *57*, 2607-2609.

[77] Hansmann, U.H.E. (**1997**) *Chem. Phys. Lett*., *281*, 140-150.

[78] Bratko, D., Chakraborty, A.K. and Shakhnovich, E.I. (**1997**) *J. Chem. Phys*., *106*, 1264-1279.

[79] Bratko, D., Chakraborty, A.K. and Shakhnovich, E.I. (**1996**) *Phys. Rev. Lett*., *76*, 1844-1847.

[80] Vila, J.A., Ripoll, D.R. and Scheraga, H.A. (**2003**) *Proc. Natl. Acad. Sci. USA*, *100* 14812-14816.

[81] Yoshida, T., Hiroyasu, T., Miki, M., Ogura, M. and Okamoto, Y. (**2002**) *Proceedings of 2002 Genetic and Evolutionary Computation Conference (GECCO 2002)* 49-51.

[82] Okamoto, Y. (**1998**) *Recent Research Developments in Pure & Applied Chemistry*, *2*, 1-22.

[83] Engh, R.A.; Huber, R. (**1991**) *Acta Cryst. A47*, 392-400.

[84] Momany, F.A., McGuire, R.F., Burgess, A.W. and Scheraga, H.A. (**1975**) *J. Phys. Chem*., *79*, 2361-2381.

[85] de Bakker, P. I., Furnham, N., Blundell, T. L. and DePristo, M.A. (**2006**) *Curr. Opin. Str. Biol*., *16*, 160-165.

[86] Koskowski, F. and Hartke, B. (**2005**) *J. Comput. Chem*., *26*, 1169-1179.

[87] Thirumalai, D. and Klimov, D.K. (**1999**) *Curr. Opin. Struct. Biol*., *9*, 197-207.

[88] Verma, A., Schug, A., Lee, K.H. and Wenzel, W. (**2006**) *J. Chem. Phys.*, *124*, 044515.

[89] Hall, C.K. and Wagoner, V.A. (**2006**) *Meth. Enzymol*., *412*, 338-365.

[90] Melquiond, A., Gelly, J.C., Mousseau, N. and Derreumaux P. (**2007**) *J. Chem. Phys*., *126*, 065101.

[91] Katagiri, D., Ode, H., Ishikawa, H., Hattori, T., Syoji, Y. and Hoshino, T. (**2004**) *Proceedings of the Sixth Meeting on the Critical Assessment of Techniques for Protein Structure Prediction (CASP6)*, 27-28.

[92] Chen, J.H., Im, W.P. and Brooks, C.L. (**2006**) *J. Am. Chem. Soc*., *128*, 3728-3736.

[93] Ooi, T., Oobatake, M., Nemethy, G. and Scheraga, H.A. (**1987**) *Proc. Natl. Acad. Sci. USA*, *84*, 3086-3090.

[94] Bykov, Y., Oakley, M.T., Burke, E.K. and Hirst, J.D. (**2004**) *Proceedings of the Sixth Meeting on the Critical Assessment of Techniques for Protein Structure Prediction (CASP6)*, 70.

[95] Schug, A. and Wenzel, W. (**2006**) *Biophys. J*., *90*, 4273-4280.

[96] Gō, N. and Scheraga H.A. (**1970**) *Macromolecules*, *3*, 178-187.

[97] Cahill, S., Cahill, M. and Cahill, K. (**2003**) *J. Comput. Chem.*, *24*, 1364-1370.

[98] Gopal, S. M. and Wenzel W. (**2006**) *Angew. Chem. Intl. Ed*., *45*, 7726-7728.

[99] Dueck, G. (**1993**) *J. Comput. Phys.*, *104*, 86-92.

[100] Simmerling, C.; Strockbine, B.; Roitberg, A.E. (**2002**) *J. Am. Chem. Soc.*, *124*, 11258-11259.

[101] Nakamura, H.K., Sasaki, T.N. and Sasai, M. (**2001**) *Chem. Phys. Letts.*, *347*, 247-254.

[102] Sun, S.J., Brem, R., Chan, H.S. and Dill, K.A. (**1995**) *Protein Engng.*, *8*, 1205-1213.

[103] Yue, K. and Dill, K.A. (**1995**) *Proc. Natl. Acad. Sci. USA*, *92*, 146-150.

[104] Hirst, J.D. (**1999**) *Protein Engng.*, *12*, 721-726.

[105] Xia, Y., Huang, E.S., Levitt, M. and Samudrala, R. (**2000**) *J. Mol. Biol.*, *300*, 171-185.

[106] Kihara, D., Lu, H., Kolinski, A. and Skolnick, J. (**2001**) *Proc. Natl. Acad. Sci. USA*, *98*, 10125-10130.

[107] Dill, K.A. (**1985**) *Biochemistry*, *24*, 1501-1509.

[108] Blackburne, B.P. and Hirst, J.D. (**2001**) *J. Chem. Phys.*, *115*, 1935-1942.

[109] Blackburne, B.P. and Hirst, J.D. (**2003**) *J. Chem. Phys.*, *119*, 3453-3460.

[110] Blackburne, B.P. and Hirst, J.D. (**2005**) *J. Chem. Phys.*, *123*, 154907.

[111] Dima, R.I. and Thirumalai, D. (**2002**) *Protein Sci*., *11*, 1036-1049.

[112] Harrison, P.M., Chan, H.S., Prusiner, S.B. and Cohen, F.E. (**2001**) *Protein Sci*., *10*, 819-835.

[113] Miyazawa, S. and Jernigan, R.L. (**1985**) *Macromolecules*, *18*, 534-552.

[114] Leonhard, K., Prausnitz, J.M. and Radke, C.J. (**2003**) *Phys. Chem. Chem. Phys*., *5*, 5291-

5299.

[115] Miyazawa, S. and Jernigan, R.L. (**1996**) *J. Mol. Biol*., *256*, 623-644.

[116] Gō, N. (**1983**) *Ann. Rev. Biophys. Bioeng*., *12*, 183-210.

[117] Chandru, V., Dattasharma, A. and Kumar, V.S.A. (**2003**) *Discrete Applied Mathematics*, *127*, 145-161.

[118] Glover, F. (**1986**) *Computers and Operations Res.*, *13*, 533-549.

[119] Unger, R. and Moult, J. (**1993**) *J. Mol. Biol.*, *231*, 75-81.

[120] Cox, G.A., Mortimer-Jones, T.V., Taylor, R.P. and Johnston, R.L. (**2004**) *Theor. Chem. Acc.*, *112*, 163-178.

[121] Liang, F.M. and Wong, W.H. (**2001**) *J. Chem. Phys.*, *115*, 3374-3380.

[122] Krasnogor, N., Blackburne, B.P., Burke, E.K. and Hirst, J.D. (**2002**) *Lecture Notes in Computer Science*, *2439*, 769-778.

[123] Shmygelska, A. and Hoos, H.H. (**2005**) *BMC Bioinformatics*, *6*, 30.

[124] Shmygelska, A., Hernandez, R. and Hoos, H.H. (**2002**) *Lect. Notes Comp. Sci*., *2463*, 40-52.

[125] Shmygelska, A. and Hoos, H.H. (**2003**) *Lect. Notes Comp. Sci.*, *2671*, 400-417.

[126] Jiang, T.Z., Cui, Q.H., Shi, G.H. and Ma, S.D. (**2003**) *J. Chem. Phys.*, *119*, 4592-4596.

[127] Bastolla, U., Frauenkron, H., Gerstner, E., Grassberger, P. and Nadler, W. (**1998**) *Proteins: Struct., Funct., Gen.*, *32*, 52-66.

[128] Cutello, V. Nicosia, G. and Pavone, M. (**2004**) *Proceedings of the 2004 Evolutionary Computation Congress*, *1*, 1074-1080.

[129] Cutello, V., Nicosia, G., Pavone, M. and Timmis, J. (**2007**) *IEEE Trans. Evol. Comput*., *11*, 101-117.

[130] Chu, D., Till, M. and Zomaya, A. (**2005**) *Parallel and Distributed Processing Symposium, Proceedings. 19th IEEE International*, 193b - 193b.

[131] Song, J., Cheng, J. and Zheng, T. (**2006**) *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06)*, 410-415.

[132] Santana, R., Larranaga, P. and Lozano, J.A. (**2004**) *Lect. Notes Comp. Sci*., *3337*, 388-398.

[133] Harrison, P.M., Chan, H.S., Prusiner, S.B. and Cohen, F.E. (**1999**) *J. Mol. Biol*., *286*, 593-606.

[134] Giugliarelli, G., Micheletti, C., Banavar, J.R. and Maritan, A. (**2000**) *J. Chem. Phys*., *113*, 5072-5077.

[135] Gupta, P., Hall, C.K. and Voegler, A.C. (**1998**) *Protein Sci*., *7*, 2642-2652.

[136] Nakanishi, K. and Kikuchi, M. (**2006**) *Journal of the Physical Society of Japan*, *75*, 064803.

[137] Tycko, R. (**2003**) *Biochemistry*, *42*, 3151-3159.

[138] Jaroniec, C.P., MacPhee, C.E., Bajaj, V.S., McMahon, M.T., Dobson C.M. and Griffin R.G. (**2004**) *Proc. Natl. Acad. Sci. USA*, *101*, 711-716.

[139] Bratko, D. and Blanch, H.W. (**2003**) *J. Chem. Phys*., *118*, 5185-5194.

[140] Bratko, D. and Blanch, H.W. (**2001**) *J. Chem. Phys.*, *114*, 561-569.

[141] Broglia, R.A., Tiana, G., Pasquali, S., Roman, H.E. and Vigezzi, E. (**1998**) *Proc. Natl. Acad. Sci. USA*, *95*, 12930-12933.

[142] Cellmer, T., Bratko, D., Prausnitz, J.M. and Blanch, H.W. (**2005**) *Proc. Natl. Acad. Sci. USA*, *102*, 11692-11697.

[143] Bratko, D., Cellmer, T., Prausnitz, J.M. and Blanch, H.W. (**2006**) *J. Am. Chem. Soc*., *128*, 1683-1691.

[144] Cellmer, T., Bratko, D., Prausnitz, J.M. and Blanch, H.W. (**2005**) *J. Chem. Phys*., *122*, 174908.

[145] Oakley, M.T., Garibaldi, J.M. and Hirst, J.D. (**2005**) *J. Comput. Chem*., *26*, 1638-1646.