

The performance of current methods in ligand–protein docking

Brendan J. McConkey*, Vladimir Sobolev and Marvin Edelman

Department of Plant Sciences, Weizmann Institute of Science, Rehovot, Israel 76100

Computer-based methods for predicting the structure of ligand–protein complexes or docking algorithms have application in both drug design and the elucidation of biochemical pathways. The number of solved structures of ligand–protein complexes now permits the testing and validation of docking algorithms, by comparison of predicted complexes with structures extracted from protein databases. This paper outlines the methodologies and compares their performance in predicting the structure of ligand–protein complexes.

COMPUTER-aided methods for the identification and characterization of ligand–protein interactions have undergone considerable advances in the past decade. Ligand docking and screening algorithms are now frequently used in the drug-design process, and have additional application in the elucidation of fundamental biochemical processes. There are several well-established docking algorithms which have been previously reviewed^{1,2}, as well as many more recently introduced methods.

The purpose of docking algorithms is now expanding beyond the original goal of fitting a given ligand into a specific protein structure. Newer applications include database screening, lead generation and *de novo* drug design. Each of these applications can be used for the identification of novel protein inhibitors. Lead generation and database screening search a large database of known chemicals for compounds having a moderate to strong affinity for the target protein. The identified set of compounds can be used for the construction of novel inhibitors with high affinity and specificity for the target protein. *De novo* drug design is a computer-based method of designing potent inhibitors^{3–6}, where possible inhibitors are assembled piecewise within a given binding pocket, from a set of chemical fragments within a molecular fragment library. While the methodology underlying single ligand–protein docking and database screening is quite similar, the criterion by which they are judged differs. A database search algorithm is required to be able to screen many thousands of possible ligands in a reasonable time period, typically no more than several days. This entails an upper limit on each docking of no more

than a few minutes. However, if a specific ligand–protein interaction is to be rigorously modelled, accuracy is the primary concern, with the algorithm execution time a secondary factor.

Protein–protein docking methodology has considerable overlap with that of ligand–protein docking; however, differences between the two docking tasks have resulted in most algorithms being intended for either one purpose or the other. The methods discussed in detail here are generally intended for ligand–protein docking. The issues faced in designing docking algorithms have been reviewed^{2,7–9}. This review will focus on recent papers testing the performance of the various docking algorithms currently available.

A docking procedure consists of three interrelated components: identification of the binding site, a search algorithm to effectively sample the search space (the set of possible ligand positions and conformations on the protein surface) and a scoring function. Most docking algorithms rely on the binding site being predefined, so that the search space is limited to a comparatively small region of the protein. This is often a reasonable assumption, as the location of the binding region of the protein can frequently be inferred from comparison with other known protein structures or biochemical constraints. The binding site may be identified by comparison with the protein co-crystallized with a different ligand, or comparison with proteins of similar function. In the absence of a priori information, putative docking sites may be identified by cavity detection programs^{10,11}.

The search algorithm must effectively sample the search space of the ligand–protein complex, i.e. the translation, rotation, and conformation space of the ligand relative to the protein. However, for even moderate sampling increments, a search of all possible combinations within the search space results in the number of combinations, being frequently in the billions. For example, given a ligand with ten rotating bonds, sampled in 60° increments, there are over 10⁷ possible conformations in the rotation space alone. Thus a search algorithm must be able to effectively sample regions of the search space in the vicinity of the correct solution, without an exhaustive search of all conformations and orientations.

The scoring function used within docking algorithms has two roles – as a target function for the search algorithm (a quantity to be optimized), and to give a ranking

*For correspondence. (e-mail: brendan.mcconkey@weizmann.ac.il)

to the set of final solutions generated by the search. Ideally, the combination of the search algorithm and the scoring function should result in a single solution close to the actual ligand position. In practice, docking algorithms are tested for their ability to reproduce known ligand conformations in an X-ray crystal structure within a given margin, and to recognize one of the conformations closest to the experimental structure as the best solution.

Several of the most common types of search algorithms and scoring functions are briefly described here. The different methods mentioned are not strict divisions, as many docking programs combine aspects from two or more search algorithms.

Search algorithms

Random search

In a random search, a large set of random ligand positions and conformations are generated, and those ligand poses are ranked with a suitable scoring function. A true random search of the translational, rotational, and conformational space of a ligand–protein pair is usually effective only if both protein and ligand can be treated as rigid bodies. This method, if used alone for ligand–protein docking, is less successful than the more knowledge-based algorithms¹². It is however frequently used as a starting point for other algorithms, where a set of randomly positioned and oriented ligands are then refined or optimized using other search and minimization techniques.

Rigid body docking

The first search algorithms for docking used the rigid body assumption, where ligand and protein flexibility was not explicitly considered. This limits the search space of the algorithm to three translational and three rotational degrees of freedom, greatly reducing the number of possible complexes to test. Within a rigid body docking algorithm, ligand flexibility could be addressed by using a set of possible ligand conformations, or allowing for a degree of atom–atom overlap between the protein and ligand. If used alone, the utility of rigid body docking is limited to ligands without numerous rotating bonds, as the number of possible ligand conformations to be tested increases exponentially with each additional rotating bond. DOCK¹³ was one of the first docking algorithms to be widely used, and the initial version was a rigid body procedure. An algorithm based on rigid body methodology, but accounting for limited protein and ligand flexibility is LIGIN¹⁴. Some recent shape-based procedures for protein–protein docking, such as FTDOCK¹⁵ and QSDOCK¹⁶, are essentially rigid body procedures. The rigid body assumption is comparatively well suited to protein–protein docking, as the number of possible

conformations and orientations of protein side chains makes the explicit consideration of protein flexibility generally infeasible. While most current docking algorithms have some method of addressing ligand flexibility, many use a rigid body approximation for the protein receptor.

Simulated annealing

Simulated annealing is based on a model of cooling processes. In the model, the initial state of the system has random thermal motion within a specified potential force field. The effective temperature of the system (the degree of random motion) is decreased over time, until a final stable docked position is obtained. The random motion of the ligand allows for exploration of the local search space, and the decreasing temperature of the system acts to drive it to a minimum energy. One of the most widely used simulated annealing procedures is the Metropolis Monte Carlo simulated annealing algorithm¹⁷. In this algorithm, a given set of parameters representing the current location, orientation and conformation of the ligand, is modified by random perturbation of the parameters. The new set is scored, and if it is better than the previous score it is kept; if not, it has a probability of being kept, and the probability is a function of the change in score and the effective temperature of the system. The process is repeated, and the temperature is systematically lowered to simulate an annealing process. Simulated annealing which has been applied to numerous docking procedures^{18–24}, was used by an earlier version of AutoDock²⁵, and in a comparative study²⁶ of simulated annealing versus two other global optimization algorithms.

Genetic algorithms and evolutionary programming

Genetic algorithms are loosely modelled on concepts borrowed from Darwinian evolution – survival of the fittest and decent with modification. In a genetic algorithm, there is a population of solutions that undergo unary (mutation) and higher order (crossover) transformations. The newly generated solutions undergo selection, biased towards the most fit among them²⁷. The algorithm maintains a selective pressure towards an optimal solution, with a randomized information exchange permitting exploration of the search space²⁸. Examples of application of genetic algorithms to docking procedures include GOLD^{29,30}, AutoDock³¹, DARWIN³², in Judson *et al.*³³, and in combination with the DOCK suite of programs³⁴. The application of genetic algorithms to drug design has been reviewed^{35,36}.

Evolutionary programming algorithms are related to genetic algorithms in that there is a selection pressure and mutation operators. However, there are no crossover operators for exchange between intermediate solutions.

Instead, each individual in the population has an associated set of parameters for guiding the mutations, which can vary over time. Execution stops after a given number of generations. Evolutionary programming has been discussed^{12,37,38} and applied to ligand docking^{39,40}.

Tabu search

Tabu search is a more recently developed search algorithm, which combines a minimization procedure with restrictions on the search path, such that the solution is forced into previously unexplored regions of the search space^{41,42}. It proceeds stepwise from an initial solution, while maintaining a list of previous solutions. The list of previous solutions provides both a ranking of solutions and a partial record of explored regions of the search space. The Tabu algorithm generates a set of N new solutions from the previous solution, and one of the N solutions is kept. A solution is added to the list if it is the best solution so far, or the solution explores a new region of the search space. The efficiency of the Tabu search has been compared to that of genetic algorithms, evolutionary programming, and simulated annealing¹². The Tabu search has been used in the PRO_LEADS^{43,44} docking algorithm, and in conjunction with genetic algorithms⁴⁵.

Incremental construction

Incremental construction algorithms use a piecewise assembly of the ligand within a defined binding pocket in the receptor. Rigid fragments are generated from the ligand by breaking it at rotating bonds, to create a set of fragments to be used by the docking algorithm. One or more of these fragments, usually the largest, is selected as the starting fragment and is docked to the receptor. A set of possible docked orientations for this fragment is kept, and other fragments are added in various orientations and scored. This process is repeated until the entire ligand is assembled. Docking applications using incremental construction include FlexX⁴⁶, Hammerhead⁴⁷, HOOK⁴⁸, and as a component of DOCK4.0 (ref. 49). *De novo* ligand design algorithms using incremental construction include LUDI⁵⁰ and GROWMOL^{3,51}.

Matching algorithms

Several docking procedures have been classed here as matching algorithms, as they share the common method of aligning structural features of the protein and ligand. The constraints on the matching algorithm can be based on shape and/or chemical information. Frequently, complementary atom types such as hydrogen bond donors and acceptors are paired between the protein and ligand. Clique detection, a pattern-matching technique from graph theory⁵², can be used to match structural features of ligands and receptor pockets. For example, a set of

atoms with given chemical properties can be matched to a set of complementary atom positions within the binding pocket, and the inter-atomic distances between ligand atoms provide constraints on solution set. Generally these algorithms are applied to a rigid body or sets of rigid bodies representing possible ligand conformations. Another approach uses geometric hashing, a procedure used in object recognition within computer vision algorithms^{53–55}. Pattern-matching algorithms for ligand-docking include DOCK¹³ and FLOG⁵⁶; pattern matching has also been used for protein–protein docking^{55,57,58}.

Scoring functions

Given a molecular docking algorithm, a means is required to identify the best of a set of possible solutions. This usually takes the form of a scoring function, in which the structural parameters of the ligand and protein are used to estimate the strength of the interaction. The scoring function can be used to rank the possible solutions, and if the scoring function is expressed in terms of free energy, to provide an estimate of the binding affinity as well. The components and accuracy of scoring functions have been reviewed^{59,60}.

Free energy perturbation

Free energy perturbation methods^{61,62} are the most rigorous for calculating binding affinities. These methods are quite time-consuming, as they require molecular dynamics simulations and the intervention of a knowledgeable user⁶⁰. Execution time per ligand–protein conformation varies from minutes to hours of computation time. The accuracy of these methods is still dependent upon a reliable parameterization of energetic terms, such as dielectric constants and van der Waals weighting factors⁶³. The current application of free energy perturbation methods to database searching or ligand–protein docking is limited, as an automated search procedure for ligand docking should not require user intervention or extensive molecular simulations. They may however be used to refine affinity estimates of predicted complexes obtained from docking algorithms.

Partition of energy terms

The most common means of estimating a binding affinity to be used as a scoring function is by partitioning of the free energy into recognizable components. The number and type of terms vary between scoring functions, but in general there are terms for hydrogen bonding, van der Waals, electrostatic and hydrophobic interactions, and entropy penalties. Methods of this type comprise the bulk of the scoring functions in use. The distance dependence of each of the terms is estimated based on theoretical considerations, and frequently is similar in form. For

example, the Lennard-Jones 6–12 potential is usually used for estimation of van der Waals interactions, whereas a coulombic term with a distance-dependent dielectric is used for modelling electrostatic interactions. The weighting of each of the interaction terms is estimated by fitting a regression model to a test set of ligand–protein complexes with known binding affinities. Whether the dependence and magnitude of these terms are valid for ligand–protein complexes outside of the original training set remains a matter of debate. A good data set of X-ray crystal structures and corresponding experimental binding affinities is required for estimation of parameters. Additionally, the database would be large, with several hundred entries of diverse type, and binding affinities ranging over several orders of magnitude.

One of the first and most widely used partitioned energy functions for docking was LUDI^{64,65}. Other scoring functions are based on CHARMM⁶⁶, AMBER⁶⁷, and the Merck Molecular Mechanics Force Field^{68–70}. The performance of empirical potentials such as these has been compared⁷¹, and some of the theoretical considerations underlying scoring functions discussed⁶⁰.

Knowledge-based scoring functions

These functions are based on statistical analysis of pairing frequencies of atoms within known ligand–protein complexes, with the underlying assumption that the more favourable an interaction is, the greater the frequency of occurrence. This can be translated into a statistically based scoring function without the need to fit regression parameters. These methods have been successfully used in protein-folding studies^{72–74}. There remains some confusion between the terms ‘knowledge-based potentials’ and ‘potential of mean force’. A true potential of mean force is based on the theory of liquids and is derived from statistical mechanics^{75,76}. The underlying assumption of knowledge-based potentials is that the relative positions of the atoms approximately follow the Boltzmann distribution with respect to energy, and that the total interaction can be approximated by the sum of pairwise potentials⁷⁷. This is not strictly applicable to protein-folding and docking, as the positions of atoms within a protein and ligand are correlated through secondary and tertiary structure. However, using methods analogous to that for calculation of a potential of mean force in liquids, effective scoring functions for ligand–protein interactions may be derived. The relationship between a potential of mean force as a defined quantity in physical chemistry and knowledge-based scoring functions has been critically discussed^{78,79}.

Complementarity functions

Several types of scoring functions do not explicitly estimate a binding energy, but instead calculate a score as

a non-energetic measure of the geometric, chemical or electrostatic fit of the ligand and receptor. Methods based solely on geometric complementarity or shape-fitting alone have been shown to be successful, particularly for protein–protein docking^{59,74,80,81}. Recently, shape-based methods have been used to dock rigid ligands by means of quadratic shape descriptors¹⁶.

Other complementarity measures can be used in addition to shape, such as electrostatics and matching of atomic surfaces by chemical type. Calculation of the electrostatic complementarity of biomolecules has been made possible by the development of fast numerical and computational methods to solve the Poisson–Boltzmann equation⁸², allowing estimates of protein electrostatic field estimates to be used in docking procedures⁸³. The electrostatic and shape profile of a protein has been effectively represented by Fourier transforms, and has been applied to protein–protein docking^{15,84}. Surface complementarity has been further subdivided according to the atom types in ligand docking^{14,85}. This method has also been used for identifying and categorizing ligand–protein and protein–protein interactions (LPC and CSU software)⁸⁶. Another algorithm using a combination of shape and chemical type is HOOK⁴⁸. Shape complementarity has also been used as part of a two-stage protocol, where the ligand is first docked based on shape, and then an energetic potential is used to refine the structure⁴⁵.

Validation of docking procedures

An objective comparison of docking methods is not completely feasible, as results are highly dependent upon the ligand–protein complexes used as a test set, the parameter settings used for the different algorithms, and similarity of the test complexes to the training set used to develop the scoring function. Still, it is informative to look at the successes and drawbacks of the various methods.

There are relatively few instances in the literature of completely independent tests of docking algorithms. One such test was the docking component of Critical Assessment of Techniques for Protein Structure Prediction (CASP2)^{87,88}. The CASP2 docking section presented a set of target ligand–protein complexes for solved but not publicly released structures. The basis for comparison of the docked results was root mean square deviations (RMSDs) from the experimental structure. It was noted that there is a trade-off between the accuracy and speed of docking algorithms⁸⁸, and thus RMSDs, while providing a quantitative measure of accuracy, are not the only means by which docking algorithms should be judged. For the docking section, nine research groups submitted predictions for seven ligand–protein complexes.

The use of known structural or sequence information can help determine the binding site for the targets, and was applied to varying degrees in the submissions to CASP2. A more rigorous test of automated docking

algorithms would involve a search of the entire protein⁸⁹, or incorporating binding site identification algorithms to the docking procedures. In the CASP2 experiment, no single docking method seemed to consistently perform best⁸⁸. Using the rather strict criterion of a weighted mean RMSD less than 2 Å, none of the groups made more than one correct prediction out of seven for the small molecule targets. About half of the submitted solutions met the criterion of an RMSD less than 4 Å. It was noted that the predicted ligand positions closest to experimental results were not always ranked the highest. If the results are compared to more recent versions of docking algorithms, the benchmark set by the CASP2 competition has been exceeded, as docking procedures continue to be refined.

Rigid body/minimization algorithms

The core algorithm of earlier versions of DOCK used a graph-based pattern-matching algorithm for the position and orientation of a ligand in a binding pocket, and treated the ligand and receptor as rigid bodies¹³. The algorithm was updated with the addition of chemical matching parameters to the previously used shape-based matching⁹⁰, improving of the ranking of known inhibitors. In the development of the most recent version of DOCK, several different pattern-matching algorithms were compared⁴⁹. The best performing one, an exhaustive single graph matching algorithm, was incorporated into DOCK4.0. Ligand flexibility is accommodated within DOCK4.0 by either docking multiple rigid conformers or an incremental construction algorithm. Conformers can be minimized using the AMBER force field^{91,92}, which has energetic terms for bond angle, bond length, torsion angle, plus van der Waals and electrostatic interactions. The test set used in the paper by Ewing and Kuntz⁴⁹ docked a selected set of predominantly rigid ligands to a set of five proteins of known structure. The test was intended to compare the efficiency of the various search algorithms used, and thus compared the speed and accuracy of convergence of the methods. The results were not quantitatively compared to known ligand–protein complexes. As a rapid search procedure, DOCK4.0 is well suited to database screening for lead compounds.

The rigid docking algorithm of DOCK 3.5 (ref. 13) was used in conjunction with multiple low-energy conformers, torsional searching, energy minimization and simulated annealing⁹³. Here, a grid-based method was used to sample the conformation space of the ligand and select low-energy conformers for rigid docking. The set of low-energy conformers was docked using DOCK3.5, with some initial modifications to the scoring function. The usual van der Waals potential was replaced with a ‘soft’ van der Waals term, scaled down by a factor of 1000 to allow some overlap of ligand and protein atoms.

The initial docked solution was minimized using the AMBER95 force field⁹² with a distance-dependent dielectric. The minimization removed poor van der Waals contacts from the ‘soft’ docking. However, as the minimization procedure could not surmount local energy barriers, torsional scanning and a short, simulated annealing procedure was used to explore the search space in the vicinity of the initial docked position. The stepwise search procedure⁹³ was tested on a set of twelve ligand–protein complexes, that had previously proved troublesome in earlier studies of docking algorithms^{94,95}. Each of the twelve ligands was docked with a range of RMSDs of 0.64 to 2.01 Å, for the top-ranked solutions. Execution time was 20 to 40 min on an SGI/R10000 195 MHz workstation, for moderately sized ligand containing eight to ten rotating bonds. Although the solutions ranked first were close to the experimental ligand position, the closest structure to experimental was not the top-ranked solution in five of twelve cases.

Another rigid body pattern-matching algorithm is FLOG (Flexible Ligands on a Grid)⁵⁶. A clique detection algorithm is used to match ligand atoms to grid points, generating a trial position and orientation of the ligand in the binding pocket. Flexibility is handled by selecting up to 25 conformations of each ligand. The initial solutions obtained are minimized using a simplex-based procedure. Using dihydrofolate reductase as a sample case, FLOG was able to correctly identify known inhibitors from within a large database.

The docking algorithm LIGIN¹⁴ uses a scoring function based on contact surface areas⁸⁵. Two of the most important energetic interactions occurring between a protein and ligand in docking are desolvation and hydrophobic interactions, both of which are highly correlated with surface areas^{96,97}. Furthermore, surface contact areas of polar atoms can approximate electrostatic interactions⁹⁸. The scoring function used in LIGIN is based on complementary atom types, such as hydrophobic–hydrophobic interactions and hydrogen bond donor–acceptor pairs. The initial docking procedure is similar to DOCK, in that it is a rigid body placement algorithm, and may also be used to dock ligand fragments. The search method proceeds from a large set of random starting positions, then uses a flexible polyhedron search algorithm⁹⁹ for optimization of the area-based complementarity function. Limited protein side-chain flexibility is accounted for by allowing a degree of steric overlap with receptor residues. In the paper introducing LIGIN¹⁴, a test set of 14 ligand–protein complexes was used to validate the search procedure. For the top-ranked solutions, 13 of 14 ligands were docked within 2.0 Å RMSD. The study also involved the docking of ligands into aporeceptors (proteins crystallized without a ligand). If apoprotein dockings are included, the total success rate is 15/18. This compares well with those generated by other docking algorithms.

As part of the CASP2 experiment, LIGIN was used as an *ab initio* docking algorithm⁸⁹, in which no prior information about the binding site or structural similarities to known ligand–protein complexes was used. This strategy differs from the approach used by other docking algorithms, as most require user intervention to define the binding pocket. This can be a reasonable assumption in many cases, but in some cases the binding site is not known, and a search of the entire protein surface is required. The performance of LIGIN was comparable with the other submissions to the CASP2 docking section. In six of seven cases, LIGIN correctly determined the binding pocket on the protein surface, and in two of those six cases, the ligand was docked to within 4.0 Å of the crystal structure. It was found that LIGIN performed best when there was a high degree of shape complementarity, and the ligand and protein had a high degree of surface contact. The typical run time of the LIGIN algorithm was 20 min on a SGI Indigo 150 MHz system. Implementation of a genetic algorithm and a refined scoring function based on LIGIN methodology is currently in progress (Najmanovich and McConkey, unpublished).

Incremental construction algorithms

The FlexX program⁹⁵ uses a rapid incremental construction algorithm to assemble the ligand in the binding pocket. The scoring function is based on that of Bohm⁶⁵, and estimates the interaction energy as a sum of terms for electrostatic interactions, directional hydrogen bonds, rotational entropy, and aromatic and lipophilic interactions⁹⁵. The receptor is treated as a rigid body, and the binding site is user-defined. In the CASP2 evaluation of docking algorithms⁸⁸, it was the fastest algorithm, making it well suited to database screening. The placement of the initial ligand fragment is done using a hierarchical pattern-matching algorithm, which matches hydrogen bond donor/acceptor pairs, metal ions/metal acceptors, and aromatic ring interactions between the ligand and protein. The initial validation set contained 19 complexes with 0 to 17 rotating bonds. Each of the docking runs contained solutions within 1.20 Å RMSD in the ranked list, though the solution closest to the experimental result was not always ranked first. Fourteen of 19 complexes (74%) had the top-ranked solution within 2.0 Å RMSD of the experimental structure⁹⁵.

The FlexX algorithm was later modified to include hydrophobic interaction types in the base placement algorithm¹⁰⁰, resulting in a slight improvement of the algorithm performance. For the given test set, the percentage of first rank structures docked within 2.0 Å increased from 64 to 70%. FlexX was recently validated with a much larger test set consisting of 200 diverse ligand–protein complexes⁴⁶. The test set consisted of ligands with 0 to 36 rotating bonds and 5 to 65 non-

hydrogen atoms. Computation time was 0.1 s to 8.2 min per ligand–protein combination on a 296 MHz workstation, with an average run time of 93 s. Forty-seven per cent of the top-ranked solutions had an RMSD of less than 2.0 Å. The success rate increases to approximately 80% if only ligands with less than 15 rotating bonds are considered. If the criterion is expanded to include any generated solution and not just the first-ranked, FlexX had a 91% accuracy for less than 5 rotating bonds, 77% for 5–15 rotating bonds, and approximately 25% accuracy for more than 15 rotating bonds. The overall accuracy for solutions of any rank was 70%. The scoring function was successful at identifying the ligand pose closest to the experimental result in many but not all cases, suggesting that there is potential for improvement of the scoring function.

FlexX has been used within a two-stage protocol¹⁰¹, where the initial set of FlexX solutions are minimized then re-ranked using a more precise force field. The force field used for re-ranking was composed of a pairwise potential modelled using the CHARMM22 force field⁶⁶, an electrostatic term from the solution to the Poisson equation^{82,102}, and a solvation term proportional to the total non-polar, solvent-accessible surface area. The test set included ligands previously found to be problematic to FlexX. In the results for eight complexes, the average RMSD using the re-ranking procedure decreased from 4.65 to 1.66 Å. For three of eight cases tested, it was necessary to introduce explicit crystallographic waters to the docking site to obtain good results. Without the explicit waters, the initial and final mean RMSDs of the eight complexes are 2.53 and 5.12 Å respectively. To address the problem of explicit waters, FlexX has been extended with an algorithm for integration and placement of water molecules during docking¹⁰³. The incorporation of the water placement algorithm did not result in a significant improvement in performance, but was able to predict the location of waters involved in docking in several instances.

Knegtel *et al.*¹⁰⁴ compared the incremental construction algorithms of DOCK4.0 and FlexX, using 32 thrombin inhibitors. For this receptor, neither algorithm performed particularly well, with an accuracy of only 10 to 35% for top-ranked solutions within 2.0 Å RMSD of the experimental structure. The chemical scoring within DOCK outperformed both DOCK energy scoring and the FlexX potential. In a related study, DOCK was used to search a database for ligands which bind to thrombin or the progesterone receptor¹⁰⁵. It was found that the energy scoring was more effective at identifying thrombin inhibitors, and chemical scoring was more effective for docking to the progesterone receptor. This demonstrates that the choice of test set and receptor can have a significant effect on the performance of the scoring functions.

Another incremental construction algorithm introduced within the last five years is Hammerhead⁴⁷. The Hammer-

head program has three parts: a protein 'pocket finder', a scoring function, and a fragment based alignment and conformational search procedure. The scoring function is based on that developed by Jain¹⁰⁶. The incremental construction algorithm initially divides the ligand into large pieces until the fragment is less than 20 atoms or has less than three rotating bonds. The protein is treated as a rigid body, and the degree of overlap of ligand and protein atoms is used to screen predicted complexes. The screening allows for a threshold amount of ligand and receptor inter-penetration, mimicking a degree of protein flexibility. As a test of the algorithm a database of 80,000 compounds was docked to the receptor streptavidin, and biotin, the ligand with the highest known affinity, was ranked first. Four moderate to strong binding ligand-protein complexes, including biotin/streptavidin, were used to test the docking algorithm and scoring function within Hammerhead. The binding affinities of the four ligands were predicted within 1.3 log K_i units, and all first ranked solutions were within 1.8 Å of the crystal structure. Typical run times are 0.3 to 6.7 min on an SGI R4400 150 MHz processor.

Genetic algorithms

Morries *et al.*³¹ have compared two types of genetic algorithms with the simulated annealing algorithm used in previous versions of AutoDock (refs 25 and 94). The first genetic algorithm procedure tested was a conventional genetic algorithm, with a set of variables specifying the ligand position, orientation and conformation, acted upon by crossover, mutation and selection operators. The second algorithm was a 'Lamarckian' genetic algorithm, which incorporates a local minimization procedure for a given fraction of the population. The three algorithms were tested with a set of seven ligands containing zero to eight non-hydroxyl rotating bonds and nine to 46 non-hydrogen atoms. As AutoDock is a stochastic procedure, each docking was repeated ten times, with the lowest energy solution reported. The simulated annealing algorithm placed five of seven ligands within 2.0 Å RMSD, while both genetic algorithms placed all seven ligands within 2.0 Å RMSD. The range of RMSDs for seven test cases with the genetic algorithm was 0.32–1.11 Å, and with the Lamarckian genetic algorithm was 0.45–1.14 Å. Considering the average over all ten runs, the simulated annealing algorithm, genetic algorithm and Lamarckian genetic algorithm had RMSDs of 3.63, 3.06, and 0.88 Å respectively. Although both genetic algorithms had comparable best results over ten runs, the Lamarckian genetic algorithm was more consistent in identifying a low RMSD solution in each run.

The scoring function of AutoDock3.0 was modelled after the AMBER force field⁶⁷, and uses a pairwise sum of energetic terms with parameters for van der Waals,

hydrogen bonding and electrostatic interactions, as well as conformational entropy and solvation terms. The solvation term used a pairwise volume-based method¹⁰⁷, which was found to work well with the grid-based algorithm used in AutoDock. A training set of 30 complexes was used to fit five parameters to the model. Initially the hydrogen-bonding term was calculated with explicit water molecules, but it was found that the regression term tended to zero if waters were included. A correction factor of approximately 0.2 kcal/mol was added to penalize ligand and protein atoms that did not form hydrogen bonds. The final model had residual standard error of 9.11 kJ/mol (2.117 kcal/mol). The run time of AutoDock3.0 on a 200 MHz SG MIPS was 4.5 to 41.3 min.

Jones *et al.*³⁰ introduced the docking program GOLD (Genetic Optimization for Ligand Docking) based on a genetic algorithm²⁹ that incorporates desolvation and limited protein flexibility. A test set of 100 complexes was used to validate the docking procedure. Entries in the test set had 0 to 30 rotating bonds with 6 to 55 non-hydrogen atoms. Twenty runs were done for each complex and the highest score from all runs was kept. As GOLD uses identification of hydrogen bonds to position the ligand in the binding pocket, at least one hydrogen bond is required by the algorithm. Out of the test set of 100 complexes, the first-ranked solution was within 2.0 Å RMSD of the experimentally determined structure in 41 cases, and within 4 Å RMSD in 71 cases. With five docking repetitions, 63% of the top-ranked solutions were within 4 Å, and with no repetitions 49% of cases were within 4 Å RMSD. The genetic algorithms were determined to be more likely to fail if the ligand is large or highly flexible, or has insufficient hydrogen bonds. Interestingly, it was noted that the docking algorithm performs better using higher resolution X-ray structures. If the crystallographic data had a resolution of 2.5 Å or better, GOLD docked 77% of top-ranked solutions within 4 Å RMSD, whereas PDB files with a resolution greater than 2.5 Å were docked within 4 Å in only 52% of the cases.

The scoring function in GOLD is an energy partition function, with terms for hydrogen bonding, ligand-protein interactions and the internal energy of the ligand. Multiple atom types were used to calculate hydrogen bond energies between donor and acceptor pairs. The interaction energies for atom pairs were pre-calculated using model fragments and water displacement terms²⁹, estimated using a combination of methods^{108–110}. In some cases, the hydrogen bond donor-acceptor pairs had a positive energy, showing a preference for solvation instead of the given interaction. The distance and angle dependence of hydrogen bond energies was accounted for by a distance and angle-dependent weighting function. In this scoring function, the usual 6–12 Lennard-Jones potential for van der Waals interactions was replaced by a more forgiving 4–8 potential with a distance cut-off, allowing the ligand to form close contacts with the

protein more readily. The internal energy of the ligand was calculated using a 6–12 van der Waals potential and the Tripos force field¹¹⁰. The scoring function accounts for the desolvation of polar residues, but does not address desolvation or lipophilic interactions explicitly. It was effective in identifying a top-ranked solution within 2 Å RMSD in approximately half the cases. After publication of the validation paper, recognition of hydrophobic–hydrophobic interactions was added to the algorithm and tested on an additional 34 ligands, and the data were published on the GOLD website¹¹¹. For the test set, 12 complexes were docked within 2 Å RMSD (35%), and 25 were within 4 Å RMSD (74%) for the top-ranked solutions.

Tabu search algorithms

The docking algorithm PRO_LEADS^{43,44} uses a Tabu search and a rigid body approximation for the receptor. The best solution obtained by the Tabu search is minimized using a simplex algorithm¹¹² to give a final result. The scoring function is modelled after the empirical pairwise potential developed by Eldridge *et al.*¹¹³, which has an estimated error of 8.68 kJ/mol. The procedure was validated using a set of 70 test complexes, containing 0 to 18 rotating bonds and 5 to 46 non-hydrogen atoms. There was some overlap of the complexes used in the validation of the algorithm with those used to fit the parameters of the scoring function. There were 46 complexes that were common to the 70 complexes used in the test set and the 82 complexes used in developing the scoring function. Several variations of the Tabu search were used in docking tests, to investigate the trade-off between speed and accuracy of the algorithm. The most rigorous search used 100 initial ligand positions, and the most rapid used five. The percentage of top-ranked solutions within 2.0 Å RMSD was 84 and 60, with average run times of 55.2 and 0.94 min, and an estimated error in binding affinity of 9.4 and 13.8 kJ/mol respectively. Due to the overlap of the training and data sets, the binding affinity was not an unbiased test of the scoring function. The scoring function was, however, able to consistently recognize the best solution within the data sets.

Quantitative structure activity relationship of scoring parameters

VALIDATE⁹⁸ is a hybrid approach to a scoring function, developed using a QSAR approach. The enthalpy of binding is calculated by molecular mechanics, while properties such as complementary hydrophobic surface area are used to estimate the entropy of binding through heuristics. Fifty-one complexes were used as the training set for developing the model. For each complex, the physical and chemical properties of the ligand–protein interaction were calculated, and models were fitted using partial least squares statistics and neural network analysis. The

major properties investigated were electrostatic energy, steric interaction energy, steric fit, H log P partition coefficient^{114,115}, the number of rotatable bonds, ligand strain energy, lipophilic contact surface area, polar/non-polar contact surface area and hydrophilic contact surface area. The standard error of the final model was 1.006 log K_i units. Interestingly, there was not an explicit parameter for hydrogen bonding in the final model. Attempts were made by the authors to include a hydrogen bonding term, but each reduced the accuracy of the resulting model. It was suggested that hydrogen bonding was accurately represented by other parameters in the model, including electrostatic interaction and complementary polar surface areas. The results indicate that the contribution of electrostatics was much smaller than expected, and explained only 2.9% of the variation in the model. It was concluded that either the representation of charges is inadequate, or much of the relevant information from electrostatics is found in other parameters, such as complementary polar surface areas.

Knowledge-based scoring

Muegge and Martin⁶³ developed a knowledge-based scoring function to estimate ligand–protein binding affinity, based on the method of Sippl *et al.*^{74,116}. X-ray crystal structures from the Protein Data Bank¹¹⁷ were used to create a radial distance-dependent function (termed PMF) that approximates the Helmholtz-free interaction energies of ligand–protein atom pairs. A volume-based correction term was added to account for solvation effects of the ligand on binding. It was shown that there was significant correlation between the sum of atom pair potentials and total binding free energy, and that the sum is therefore a good measure of binding affinities⁶³. The standard deviation of the potential function from the experimentally determined binding energies was 1.8 log K_i units, comparable to force field-based scoring functions fitted to experimental data. An attempt was made to incorporate explicit terms for solvation and loss of rotational entropy on ligand binding. The added terms did not improve the correlation between calculated and observed free energies, and so were omitted in the final model. A similar result was found using another knowledge-based potential, BLEEP^{118,119}. The potential function was generated with and without explicit water atoms being present. Water atoms were added using the program Aquarius^{120,121}. The procedure for incorporating solvent was not found to improve correlation with experimental values. Both potentials were tested on a set of 90 ligand–protein complexes, and found to have a correlation coefficient of $R^2 = 0.74$ over a range of 49 kJ/mol experimental binding energy. When the solvent term was included, the correlation was reduced to $R^2 = 0.63$.

The knowledge-based potential of Muegge and Martin (PMF)⁶³ was tested by comparison with 77 ligand–protein complexes. The potential function was further compared

with two popular force field-based methods, LUDI⁶⁵ and SMOG¹²², with eight different sets of related protein structures. In the majority of cases, the PMF scoring was found to perform better than the empirical functions. To translate the scoring function to a binding energy, a different linear scaling factor was used for each of the eight data sets. Most of the scaling factors were in the same range, but in three of the eight test sets, the scaling factors had a two to three-fold difference. For this reason, the authors state that the generality of the approach may be limited. The scoring function may be more reliable for screening different ligands against the same target, or comparing different ligand–protein configurations within a docking study. Thus the procedure is effective for ranking ligands, but still may not provide a globally accurate estimate of binding affinities. The PMF score was also used with DOCK4.0 in a database screening application¹²³ in which 3247 small molecules were docked to FK506-binding protein. Molecules known to weakly bind to FK506 had higher docking scores, and were enriched in the upper ranks of the computational scoring lists. The enrichment obtained using the PMF score was greater than that obtained by the standard DOCK4.0 force field.

In another study¹²⁴, the PMF score was compared to the force field-based scoring functions in DOCK4.0 and FlexX, using the matrix metalloproteinase stromelysin (MMP-3) with a set of 61 biphenyl-type inhibitors. The experimentally determined binding affinities were compared with those estimated using DOCK/force field, DOCK/PMF and the FlexX potential. As crystal structures were available for only six out of the 61 inhibitors, the conserved position of the biphenyl moiety was used as the basis of comparison. Out of the 61 inhibitors, DOCK/PMF docked 93%, DOCK/force field docked 85%, and FlexX docked 80% within 2.0 Å RMSD. DOCK/PMF was the only function among the three that found a significant correlation between binding affinity and predicted score of docked inhibitors. It was suggested that FlexX may be better at ‘fine-tuning’ the ligand position, as the ligands correctly docked by FlexX had a smaller average RMSD (0.9 Å) than either Dock/PMF (1.3 Å) or Dock/FF (1.8 Å)¹²⁴. It is possible that the good performance of both programs is partially a result of the data set used – incremental construction algorithms, available in both DOCK and FlexX, are likely to choose the biphenyl portion of the ligands as the initial placed fragment. As the RMSDs are calculated only from the biphenyl portion, these values may not be an unbiased test of performance. Estimation of the binding affinity, however, is calculated over the entire molecule; so the good correlation of measured and predicted binding affinities is an indication of both an effective search algorithm and scoring function. Each of the four incorrectly docked structures using DOCK/PMF had scored higher than the anticipated correct binding

mode, suggesting that the poor docking was a result of insufficient sampling rather than a scoring error. It was suggested by the authors that the PMF scoring function also had a lower rate of false negatives (complexes with high docking scores but low binding affinity) making it useful in screening applications.

The knowledge-based scoring function DrugScore⁷⁶ incorporated solvent effects into the potential using solvent-accessible surfaces. The solvent exposure probabilities of ligand and protein atoms were converted to potentials, analogous to the methodology used to develop the pairwise knowledge-based potential. The weighting of the atom pair frequencies and the solvation terms was optimized empirically to be 0.5 each. The scoring function was used to analyse docked solutions obtained from FlexX and DOCK4.0. Ninety-one PDB structures were selected from the 200 used in FlexX validation⁹⁵, and contained 0 to 27 rotating bonds. With this set, FlexX finds top-ranked solutions with an RMSD of less than 2.0 Å in 54% of the cases. Using the DrugScore potential, the percentage of top-ranked docked solutions increases to 73. A second set of 68 complexes, not used in the development of the scoring function, was used to cross validate the results. In the set of 68 complexes, 41% of the solution sets generated by FlexX contained a solution within 2.0 Å from the crystal ligand position. In the 29 sets containing a solution within 2.0 Å RMSD, DrugScore and FlexX obtain nearly identical results and identified the top-ranked solution within 2.0 Å in 92 and 93% of the cases respectively. The DOCK4.0 algorithm was also tested, using 100 protein–ligand complexes pooled from both of the FlexX test sets. The energy scoring in DOCK resulted in 61 complexes having a solution within 2.0 Å; of these, 54% were the top-ranked solution. Re-scoring the conformations with DrugScore resulted in 51% of the top-ranked solutions being within 2.0 Å RMSD. Using DOCK chemical scoring, 43 complexes had a solution within 2.0 Å, and 46% of these had the top-ranked solution within 2.0 Å RMSD. Thus, the DOCK energy scoring was the most effective in this case.

Open issues and algorithm development

The various search algorithms present in the literature and discussed here are each theoretically capable of effectively exploring the search space of a ligand–protein interaction, given enough time, computer resources, and effective parameterization of the method. However, there is still room for improvement in the efficiency and accuracy of the commonly used implementations of these algorithms. The performance of stochastic docking algorithms was increased somewhat by multiple runs, suggesting that although the search space is frequently sampled sufficiently, multiple docking runs can increase the accuracy of the docking. The more successful docking studies frequently used a combination of methods,

often with a minimization component. The use of local minimization in conjunction with a global search method seems particularly effective^{12,31}.

There was a decrease in performance of nearly all methods with an increasing number of rotating bonds. It is likely that this is a reflection of the increased size of the search space, but it is also a possibility that it is due to limitations inherent in the implementations of the docking algorithms. It would be expected that the effectiveness of an incremental construction could be reduced when docking large, flexible ligands, due to the smaller relative size of the initial fragment compared to the entire ligand. Similarly, within a genetic algorithm, optimization of the ligand position could require a concerted change of several variables coded into the algorithm, and the probability of this occurring diminishes with increasing number of parameters. The use of a local minimization procedure in conjunction with either of these methods might reduce this effect considerably.

There are several issues in generating empirical scoring functions that remain difficult to surmount. In a thermodynamically-based energetic model, the terms are usually divided into enthalpy and entropy contributions. These terms are large and opposing, and can introduce a large error into the sum, even if the individual contributions are known with reasonable accuracy. Additionally, interactions between protein and ligand atoms do not always occur in the intuitively expected form. For example, it would be expected that a pair of oppositely charged atoms would tend to have a favourable interaction. Yet in some cases, there is an increase in free energy on binding of oppositely charged residues, due to the displacement of favourable interactions with the solvent^{29,82}.

The inclusion, or omission, of water molecules in molecular docking has not yet been effectively automated. Existing programs can identify likely positions for water molecules, and use explicitly modelled water molecules in docking algorithms. However, this has not increased the efficacy of docking yet.

Receptor flexibility is neglected in most docking algorithms due to the large increase in complexity of the problem when it is explicitly considered. The rigid body approximation for the receptor is not always justifiable, however, particularly when docking a novel ligand to an apoprotein. Whenever a ligand binds to a receptor, the receptor is likely to undergo some structural modification, such as reduced side-chain motion in the vicinity of the docked ligand, or possibly larger structural rearrangements such as hinge-bending. Frequently, these changes are minimal, but in some cases, changes in structure upon ligand binding are integral to the function of the protein, such as allosteric regulation and signal transduction⁸⁸. Currently, most docking algorithms incorporate only minimal, if any, protein flexibility. The rigid body assumption may prove to be adequate in many circumstances, but is still an approximation.

In most tests of scoring functions and docking algorithms, X-ray crystal structures of ligand–protein complexes are used in the test sets. The ligand is removed from the structure model, then re-docked back into the receptor pocket. This likely provides the best test of the scoring function, as it is required to score the final docked structure the highest, in order to be effective. However, this is not an accurate model of the docking process. The protein structure and ligand pose prior to binding are almost certainly different from the final docked conformation. Hence, recognition of the ligand and protein is an adaptive process, though in many cases the degree of protein motion is small.

The binding affinity of ligands is also affected by protein mobility. If a docked ligand is in a deep pocket of a flexible protein, the protein itself may have to undergo a conformational change in order to release the bound ligand, and so the ligand is effectively ‘trapped’ by the protein. Thus, the binding affinity may be a function of the rate of motion of the protein, as well as the sum of interactions between the protein and the ligand. The ligand–protein complex association/disassociation rate may also be affected by the presence or absence of another ligand or protein. The large measured binding affinity of biotin to streptavidin is a good example of this process. As the ligand–protein complex with the highest known affinity, it is included frequently in the training set for many scoring functions. However, its binding affinity is highly susceptible to changes in the protein structure. Biotin/streptavidin increases its binding affinity by eight orders of magnitude for the tetramer compared to the monomer structure present in the PDB. It is unlikely that this change in affinity can be accounted for by any of the scoring functions in use. It is possible that the association and dissociation of the complexes such as this may be estimated by the repeated molecular modelling of ligand–protein complex over time.

A limit of the accuracy of both empirical and knowledge-based scoring functions is the resolution of PDB structures used to generate the scoring functions. The position of each atom will be less precise in a structure with poorer resolution, affecting both energetic and knowledge-based scoring functions. PDB structures with poor resolution in the vicinity of the binding site could be a result of motion of the ligand and/or side receptor pocket around an average structure. Here, the entropy penalty of the docked structure is less than that predicted, and an error is introduced into the calculation of enthalpy terms as well. The experimentally determined binding affinities of ligands are also subject to variation and error. Binding affinities are temperature-dependent, and data have not been collected at one fixed temperature. There are several methods for the determination of binding efficiencies, and errors in measurement can be considerable. Error estimates of less than 10% are an exception¹²⁵.

There are inherent limits to some scoring approaches when used independently of other methods. The Poisson–Boltzmann equation, for example, can provide an accurate description of the electrostatic field around a protein, but if used alone does not account for desolvation or other interactions. Conversely, most pairwise potentials do not effectively predict the global electrostatic interactions, which can be an important factor in docking. Even if all known effects are accounted for within an energetic model, the magnitude and even the sign of each term is not precisely known. It is still a subject of debate whether or not van der Waals forces are greater between protein and ligand, or protein–solvent and ligand–solvent interactions are greater⁶⁰.

Most scoring functions have some level of approximation for identifying atom types, and the interaction between a given pair of atom types is usually treated as having equivalent strengths, given equivalent separation distances. This may be true on average, but there are exceptions. Subtle changes in electrostatic properties can result in large changes in binding affinity within a given type of interaction^{60,126}. A scoring function that is a sum of pairwise additive terms also neglects non-additive behaviour between and among groups. Hydrophobic interactions are a good example of this.

In summary, the most likely area for improvement of docking and scoring algorithms is within the scoring functions. A reasonable level of prediction has been achieved recently, but the ligand pose closest to the experimentally determined structure is still not always identified as the best solution. Regarding the limitations discussed here, some may be surmounted through further study of protein–ligand interactions. At some point there will likely be a maximum possible accuracy attainable through the use of scoring functions, particularly in estimation of binding affinities.

1. Cherfils, J. and Janin, J., *Curr. Opin. Struct. Biol.*, 1993, **3**, 265–269.
2. Kuntz, I. D., Meng, E. C. and Shoichet, B. K., *Acc. Chem. Res.*, 1994, **27**, 117–123.
3. Bohacek, R. S. and McMartin, C., *J. Am. Chem. Soc.*, 1994, **116**, 5560–5571.
4. Bohm, H. J., *Perspect. Drug Disc. Design*, 1995, **3**, 21–33.
5. DeWitte, R. S., Ishchenko, A. V. and Shakhnovich, E. I., *J. Am. Chem. Soc.*, 1997, **119**, 4608–4617.
6. Klebe, G., *J. Mol. Med.*, 2000, **78**, 269–281.
7. Jones, G. and Willett, P., *Curr. Opin. Biotechnol.*, 1995, **6**, 652–656.
8. Lengauer, T. and Rarey, M., *Curr. Opin. Struct. Biol.*, 1996, **6**, 402–406.
9. Verkhivker, G. M. et al., *J. Comput.-Aided Mol. Des.*, 2000, **14**, 731–751.
10. Peters, K. P., Fauck, J. and Frommel, C., *J. Mol. Biol.*, 1996, **256**, 201–213.
11. Brady, G. P. Jr. and Stouten, P. F., *J. Comput.-Aided Mol. Des.*, 2000, **14**, 383–401.
12. Westhead, D. R., Clark, D. E. and Murray, C. W., *ibid.*, 1997, **11**, 209–228.
13. Kuntz, I. D., Blaney, J. M., Oatley, S. J., Langridge, R. L. and Ferrin, T. E., *J. Mol. Biol.*, 1982, **161**, 269–288.
14. Sobolev, V., Wade, R. C., Vriend, G. and Edelman, M., *Proteins*, 1996, **25**, 120–129.
15. Gabb, H. A., Jackson, R. M. and Sternberg, M. J., *J. Mol. Biol.*, 1997, **272**, 106–120.
16. Goldman, B. B. and Wipke, W. T., *Proteins*, 2000, **38**, 79–94.
17. Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P., *Science*, 1983, **220**, 621–680.
18. Apostolakis, J., Pluckthun, A. and Caflisch, A., *J. Comput. Chem.*, 1998, **19**, 21–37.
19. Caflisch, A., Fischer, S. and Karplus, M., *ibid.*, 1997, **18**, 723–743.
20. Cummings, M. D., Hart, T. N. and Read, R. J., *Protein Sci.*, 1995, **4**, 885–899.
21. Liu, M. and Wang, S. M., *J. Comput.-Aided Mol. Des.*, 1999, **13**, 435–451.
22. Trosset, J. Y. and Scheraga, H. A., *Proc. Natl. Acad. Sci. USA*, 1998, **95**, 8011–8015.
23. Trosset, J. Y. and Scheraga, H. A., *J. Comput. Chem.*, 1999, **20**, 244–252.
24. Totrov, M. and Abagyan, R., *Proteins*, 1997, **S1**, 215–220.
25. Goodsell, D. S. and Olson, A. J., *ibid.*, 1990, **8**, 195–202.
26. Diller, D. J. and Verlinde, C., *J. Comput. Chem.*, 1999, **20**, 1740–1751.
27. Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, Berlin, 1994, 2nd edn.
28. Goldberg, D. E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, New York, 1989.
29. Jones, G., Willett, P. and Glen, R. C., *J. Mol. Biol.*, 1995, **245**, 43–53.
30. Jones, G., Willett, P., Glen, R. C., Leach, A. R. and Taylor, R., *ibid.*, 1997, **267**, 727–748.
31. Morris, G. M., Goodsell, D. S., Halliday, R. S., Huey, R., Hart, W. E., Belew, R. K. and Olson, A. J., *J. Comput. Chem.*, 1998, **19**, 1639–1662.
32. Taylor, J. S. and Burnett, R. M., *Proteins*, 2000, **41**, 173–191.
33. Judson, R. S., Jaeger, E. P. and Treasurywala, A. M., *Theochem.-J. Mol. Struct.*, 1994, **114**, 191–206.
34. Oshiro, C. M., Kuntz, I. D. and Dixon, J. S., *J. Comput.-Aided Mol. Des.*, 1995, **9**, 113–130.
35. Parrill, A. L., *Drug Disc. Today*, 1996, **1**, 514–521.
36. Willett, P., *Trends Biotechnol.*, 1995, **13**, 516–521.
37. Clark, D. E. and Westhead, D. R., *J. Comput.-Aided Mol. Des.*, 1996, **10**, 337–358.
38. Yao, X. and Liu, Y., Proceedings of the Fifth Annual Conference on Evolutionary Programming (eds Fogel, L. J., Angeline, P. J., Back, T.), MIT Press, 1996.
39. Gehlhaar, D. K., Verkhivker, G. M., Rejto, P. A., Sherman, C. J., Fogel, D. B., Fogel, L. J. and Freer, S. T., *Chem. Biol.*, 1995, **2**, 317–324.
40. Yang, J. M. and Kao, C. Y., *J. Comput. Chem.*, 2000, **21**, 988–998.
41. Glover, F. and Laguna, M., in *Modern Heuristic Techniques for Combinatorial Problems* (ed. Reeves, C. R.), Blackwell, Oxford, 1993, pp. 70–150.
42. Cvijovic, D. and Klinowski, J., *Science*, 1995, **267**, 664–666.
43. Baxter, C. A. et al., *J. Chem. Inf. Comput. Sci.*, 2000, **40**, 254–262.
44. Baxter, C. A., Murray, C. W., Clark, D. E., Westhead, D. R. and Eldridge, M. D., *Proteins*, 1998, **33**, 367–382.
45. Hou, T., Wang, J., Chen, L. and Xu, X., *Protein Eng.*, 1999, **12**, 639–648.
46. Kramer, B., Rarey, M. and Lengauer, T., *Proteins*, 1999, **37**, 228–241.
47. Welch, W., Ruppert, J. and Jain, A. N., *Chem. Biol.*, 1996, **3**, 449–462.
48. Eisen, M. B., Wiley, D. C., Karplus, M. and Hubbard, R. E., *Proteins*, 1994, **19**, 199–221.
49. Ewing, T. J. A. and Kuntz, I. D., *J. Comput. Chem.*, 1997, **18**, 1175–1189.
50. Bohm, H. J., *J. Comput.-Aided Mol. Des.*, 1992, **6**, 61–78.
51. Bohacek, R. S. and McMartin, C., *Curr. Opin. Chem. Biol.*, 1997, **1**, 157–161.

52. Brint, A. T. and Willett, P., *J. Chem. Inf. Comput. Sci.*, 1987, **27**, 152–158.
53. Fischer, D., Norel, R., Wolfson, H. and Nussinov, R., *Proteins*, 1993, **16**, 278–292.
54. Norel, R., Fischer, D., Wolfson, H. J. and Nussinov, R., *Protein Eng.*, 1994, **7**, 39–46.
55. Nussinov, R. and Wolfson, H. J., *Comb. Chem. High Throughput Screen*, 1999, **2**, 249–259.
56. Miller, M. D., Kearsley, S. K., Underwood, D. J. and Sheridan, R. P., *J. Comput.-Aided Mol. Des.*, 1994, **8**, 153–174.
57. Gardiner, E. J., Willett, P. and Artymiuk, P. J., *J. Chem. Inf. Comput. Sci.*, 2000, **40**, 273–279.
58. Sandak, B., Nussinov, R. and Wolfson, H. J., *Comput. Appl. Biosci.*, 1995, **11**, 87–99.
59. Shoichet, B. K. and Kuntz, I. D., *J. Mol. Biol.*, 1991, **221**, 327–346.
60. Tame, J. R. H., *J. Comput.-Aided Mol. Des.*, 1999, **13**, 99–108.
61. Kollman, P., *Chem. Rev.*, 1993, **93**, 2395–2417.
62. Briggs, J. M., Marrone, T. J. and McCammon, J. A., *Trends Cardiovasc. Med.*, 1996, **6**, 198–204.
63. Muegge, I. and Martin, Y. C., *J. Med. Chem.*, 1999, **42**, 791–804.
64. Bohm, H. J., *J. Comput.-Aided Mol. Des.*, 1992, **6**, 593–606.
65. Bohm, H. J., *ibid.*, 1994, **8**, 243–256.
66. Brooks, B. R., Bruccoleri, R. E., Olafson, B. D., States, D. J., Swaminathan, S. and Karplus, M., *J. Comput. Chem.*, 1983, **4**, 187–217.
67. Pearlman, D. A. *et al.*, *Comput. Phys. Commun.*, 1995, **91**, 1–41.
68. Halgren, T. A., *J. Comput. Chem.*, 1996, **17**, 490–519.
69. Halgren, T. A., *ibid.*, 1996, **17**, 520–552.
70. Halgren, T. A., *ibid.*, 1996, **17**, 553–586.
71. Hobza, P., Kabelac, M., Sponer, J., Mejzlik, P. and Vondrasek, J., *J. Comput. Chem.*, 1997, **18**, 1136–1150.
72. Jones, D. T., Taylor, W. R. and Thornton, J. M., *Nature*, 1992, **358**, 86–89.
73. Domingues, F. S. *et al.*, *Proteins*, 1999, **S3**, 112–120.
74. Sippl, M. J., *Curr. Opin. Struct. Biol.*, 1995, **5**, 229–235.
75. Israelachvili, J., *Intermolecular and Surface Forces*, Academic Press, London, 1992, 2nd edn.
76. Gohlke, H., Hendlich, M. and Klebe, G., *J. Mol. Biol.*, 2000, **295**, 337–356.
77. Muegge, I., *Abstr. Pap. Am. Chem. Soc.*, 1999, **218**, 51-COMP.
78. BenNaim, A., *J. Chem. Phys.*, 1997, **107**, 3698–3706.
79. Vajda, S., Sippl, M. and Novotny, J., *Curr. Opin. Struct. Biol.*, 1997, **7**, 222–228.
80. Cherfils, J., Duquerroy, S. and Janin, J., *Proteins*, 1991, **11**, 271–280.
81. Helmer-Citterich, M. and Tramontano, A., *J. Mol. Biol.*, 1994, **235**, 1021–1031.
82. Honig, B. and Nicholls, A., *Science*, 1995, **268**, 1144–1149.
83. Ausiello, G., Cesareni, G. and HelmerCitterich, M., *Proteins*, 1997, **28**, 556–567.
84. Ritchie, D. W. and Kemp, G. J., *ibid.*, 2000, **39**, 178–194.
85. Sobolev, V. and Edelman, M., *ibid.*, 1995, **21**, 214–225.
86. Sobolev, V., Sorokine, A., Prilusky, J., Abola, E. E. and Edelman, M., *Bioinformatics*, 1999, **15**, 327–332.
87. Dunbrack, R. L., Gerloff, D. L., Bower, M., Chen, X. W., Lichtarge, O. and Cohen, F. E., *Fold. Des.*, 1997, **2**, R27–R42.
88. Dixon, J. S., *Proteins*, 1997, **S1**, 198–204.
89. Sobolev, V., Moallem, T. M., Wade, R. C., Vriend, G. and Edelman, M., *ibid.*, 1997, 210–214.
90. Shoichet, B. K. and Kuntz, I. D., *Protein Eng.*, 1993, **6**, 723–732.
91. Weiner, S. J., Kollman, P. A., Nguyen, D. T. and Case, D. A., *J. Comp. Chem.*, 1986, **7**, 230–252.
92. Cornell, W. D. *et al.*, *J. Am. Chem. Soc.*, 1995, **117**, 5179–5197.
93. Wang, J., Kollman, P. A. and Kuntz, I. D., *Proteins*, 1999, **36**, 1–19.
94. Morris, G. M., Goodsell, D. S., Huey, R. and Olson, A. J., *J. Comput.-Aided Mol. Des.*, 1996, **10**, 293–304.
95. Rarey, M., Kramer, B., Lengauer, T. and Klebe, G., *J. Mol. Biol.*, 1996, **261**, 470–489.
96. Eisenberg, D., Wilcox, W. and McLachlan, A. D., *J. Cell. Biochem.*, 1986, **31**, 11–17.
97. Eisenberg, D. and McLachlan, A. D., *Nature*, 1986, **319**, 199–203.
98. Head, R. D., Smythe, M. L., Oprea, T. I., Waller, C. L., Green, S. M. and Marshall, G. R., *J. Am. Chem. Soc.*, 1996, **118**, 3959–3969.
99. Himmelblau, D. M., *Applied Non-Linear Programming*, McGraw-Hill, New York, 1972.
100. Rarey, M., Kramer, B. and Lengauer, T., *Bioinformatics*, 1999, **15**, 243–250.
101. Hoffmann, D., Kramer, B., Washio, T., Steinmetzer, T., Rarey, M. and Lengauer, T., *J. Med. Chem.*, 1999, **42**, 4422–4433.
102. Warwicker, J. and Watson, H. C., *J. Mol. Biol.*, 1982, **157**, 671–679.
103. Rarey, M., Kramer, B. and Lengauer, T., *Proteins*, 1999, **34**, 17–28.
104. Knegtel, R. M., Bayada, D. M., Engh, R. A., von der Saal, W., van Geerestein, V. J. and Grootenhuys, P. D., *J. Comput.-Aided Mol. Des.*, 1999, **13**, 167–183.
105. Knegtel, R. M. and Wagener, M., *Proteins*, 1999, **37**, 334–345.
106. Jain, A. N., *J. Comput.-Aided Mol. Des.*, 1996, **10**, 427–440.
107. Stouten, P. F. W., Frommel, C., Nakamura, H. and Sander, C., *Mol. Simul.*, 1993, **10**, 97–120.
108. Cramer, C. J., Lynch, G. C., Hawkins, G. D. and Truhlar, D. G., *AMSOL 3.5c User Manual*, Quantum Chemical Program Exchange, Department of Chemistry, University of Indiana, Bloomington, IN, 1993.
109. Rinaldi, D., Rivail, J. L. and Rguini, N., *J. Comput. Chem.*, 1992, **13**, 675–680.
110. Clark, M., Cramer, R. D. and Vanopdenbosch, N., *ibid.*, 1989, **10**, 982–1012.
111. <http://cisrg.shef.ac.uk/projects/gold/examples.html>
112. Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P., in *Numerical Recipes in Fortran: The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1992, 2nd edn. p. 402.
113. Eldridge, M. D., Murray, C. W., Auton, T. R., Paolini, G. V. and Mee, R. P., *J. Comput.-Aided Mol. Des.*, 1997, **11**, 425–445.
114. Wireko, F. C., Kellogg, G. E. and Abraham, D. J., *J. Med. Chem.*, 1991, **34**, 758–767.
115. Kellogg, G. E., Semus, S. F. and Abraham, D. J., *J. Comput.-Aided Mol. Des.*, 1991, **5**, 545–552.
116. Sippl, M. J., Ortner, M., Jaritz, M., Lackner, P. and Flockner, H., *Fold. Des.*, 1996, **1**, 289–298.
117. Bernstein, F. C. *et al.*, *J. Mol. Biol.*, 1977, **112**, 535–542.
118. Mitchell, J. B. O., Laskowski, R. A., Alex, A. and Thornton, J. M., *J. Comput. Chem.*, 1999, **20**, 1165–1176.
119. Mitchell, J. B. O., Laskowski, R. A., Alex, A., Forster, M. J. and Thornton, J. M., *ibid.*, 1999, **20**, 1177–1185.
120. Pitt, W. R., Murrayrust, J. and Goodfellow, J. M., *ibid.*, 1993, **14**, 1007–1018.
121. Goodfellow, J. M., Pitt, W. R., Smart, O. S. and Williams, M. A., *Comput. Phys. Commun.*, 1995, **91**, 321–329.
122. DeWitte, R. S. and Shakhnovich, E. I., *J. Am. Chem. Soc.*, 1996, **118**, 11733–11744.
123. Muegge, I., Martin, Y. C., Hajduk, P. J. and Fesik, S. W., *J. Med. Chem.*, 1999, **42**, 2498–2503.
124. Ha, S., Andreani, R., Robbins, A. and Muegge, I., *J. Comput.-Aided Mol. Des.*, 2000, **14**, 435–448.
125. Weber, G., *Protein Interactions*, Chapman and Hall, New York, 1992.
126. Zimmerman, S. C. and Murray, T. J., *Philos. Trans. R. Soc. London, Ser. A*, 1993, **345**, 49–56.