Journal of Computer-Aided Molecular Design, 9 (1995) 213–225
ESCOM

J-CAMD 292

PRO_LIGAND: An approach to de novo molecular design. 4. Application to the design of peptides

David Frenkel, David E. Clark, Jin Li*, Christopher W. Murray, Barry Robson, Bohdan Waszkowycz and David R. Westhead

Proteus Molecular Design Ltd., Proteus House, Lyme Green Business Park, Macclesfield, Cheshire SK11 0JL, U.K.

Received 9 December 1994
Accepted 1 February 1995

Keywords: Computer-aided molecular design; De novo peptide design; HIV-1 protease inhibitors; Lysozyme epitopes; Synthetic vaccine design

Summary

In some instances, peptides can play an important role in the discovery of lead compounds. This paper describes the peptide design facility of the de novo drug design package, PRO_LIGAND. The package provides a unified framework for the design of peptides that are similar or complementary to a specified target. The approach uses single amino acid residues, selected from preconstructed libraries of different residues and conformations, and places them on top of predefined target interaction sites. This approach is a well-tested methodology for the design of organics but has not been used for peptides before. Peptides represent a difficulty because of their great conformational flexibility and a study of the advantages and disadvantages of this simple approach is an important step in the development of design tools. After a description of our general approach, a more detailed discussion of its adaptation to peptides is given. The method is then applied to the design of peptide-based inhibitors to HIV-1 protease and the design of structural mimics of the surface region of lysozyme. The results are encouraging and point the way towards further development of interaction site-based approaches for peptide design.

Introduction

An ever-increasing number of protein structures is being determined by X-ray crystallography or NMR studies [1–4]. The success of some pioneering projects based upon such structures has led to the growing acceptance of Structure-Based Drug Design (SBDD) as a useful tool in the discovery of lead compounds [5–9]. In cases where the 3D structure of the receptor is not available or derivable, so-called 'indirect' design approaches [10–13] can be used to extract and utilise information about known ligands to drive further design optimisation.

One computational method that is generating intense interest at the moment is the field of de novo design. The aim is to use structural information to derive constraints on the shape and chemistry of potential ligands and then to build acceptable molecules that are consistent with these constraints. Many programs have now been described for the de novo design of ligands [14–41] and the field has recently been reviewed [42]. The majority of these programs build structures from basic chemical building blocks and can be classified according to the kind of building block they employ. Some, for instance, build structures in an atom-by-atom manner [18,19,26,30], whilst others employ libraries of small organic molecular fragments [15,20–22,27–30].

In both cases, this means that a great diversity of organic molecules are produced as solutions, but there is little or no control over the synthetic accessibility or toxicity of solutions and it is often a considerable task to sort out the unattractive designs from the more interesting ones. One way around this problem is to use amino acid residues as the basic building blocks [14] and thus ensure that only peptides are produced.

Oligopeptides may be attractive candidates during the first stage of drug design, since they can be easily synthesised for testing in various assays. This enables the therapeutic relevance of a particular molecular target to be

*To whom correspondence should be addressed.

rapidly confirmed or rejected before the possibly time-consuming process of organic synthesis is undertaken. In addition, peptide designs may also serve as starting points for a combinatorial library-based approach – it is apparent that experimental methods based on combinatorial libraries are much more efficient when constrained through external knowledge [43,44]. Furthermore, peptides can be credible fragments with which to close discontinuous epitope loops which may be known to be associated with a particular biological activity.

Some attempts at peptide de novo design have appeared in the literature. The program GROW [14] uses preformed libraries of conformations of peptide fragments. As the structure is formed, a new residue is selected by testing all conformations of the next residue and using a molecular mechanics scoring function to choose the highest scoring peptides. However, despite the early success of the GROW method of Howe and Moon, very little further work has been published in this area. The program GEMINI [45] uses information on the packing of amino acid side chains stored in a database to suggest conformations of peptide ligands when bound to their receptors. It does not build structures in the receptor cavity like most de novo design programs, but it has illustrated the utility of a rule-based method in predicting the conformations of active peptides.

PRO_LIGAND [39–41] is our in-house facility for automated de novo design and it is an integral part of our system for molecular design and simulation, PROMETHEUS. PRO_LIGAND uses a rule-based method to derive from the target an approximation to an ideal drug template in terms of interaction sites. These interaction sites represent the 3D configuration of various physicochemical properties of a putative ligand. The program then builds designs consistent with the template from libraries of fragments. The approach is most similar to that used in the program LUDI [20]. Until now, the program has been used for the design of organics, but given the desirability of a peptide growing facility, PRO_LIGAND has been extended to build peptides. The purpose of this work was to test our approach in applications to peptides which represent special problems, due to their great conformational flexibility. Our design approach, based on interaction sites derived from sets of rules, is very different from any method that has been published so far in the area of peptide design.

This paper contains a description of our method, followed by its application to two design problems typical of the type that one would wish to address with a de novo design program. The first example applies the method to the design of inhibitors of HIV-1 protease, given the structure of the active site as a starting point. The second example looks at the design of a peptide similar to an epitope of lysozyme. The results and the method used to produce them are discussed.

Methods

Overview of PRO_LIGAND

PRO_LIGAND is composed of four main modules. The first module to operate in the design process is **Design-base Generation**. Its purpose is to take the input information (i.e., molecular structure and command files) and to generate the features that are important for the design process, e.g., the atoms comprising the active site of a receptor or a pharmacophore derived from a set of active structures. The output from this module is termed the *design base*.

The second module, **Design-model Generation**, employs a rule-based algorithm to construct a *design model* whose physico-chemical characteristics are either *similar* or *complementary* to those of the design base, according to the user's choice. The design model is a template that describes the idealised steric and hydrogen-bonding features of the chemical structures to be designed. These features are represented as *interaction sites* [20,21,46]. The rules used to derive the positions and types of interaction sites are derived from statistical analyses of nonbonded contact geometries found within the Cambridge Structural Database [47]. Hydrogen-bond acceptors and donors are represented by **A-Y** and **D-X** vectors, respectively, while lipophilic regions are represented by L or R sites, according to whether the region around the site is aliphatic or aromatic in nature.

Once the design model has been produced, the **Structure Generation** module is invoked and it produces structures by assembling 3D fragments from preconstructed libraries. These library fragments are labelled to indicate the type of interaction site they may match and a rapid graph-theoretical algorithm is used to seek fits of the fragments to the design model. The output from Structure Generation is a set of structures that are consistent with the constraints of the design model. This module is most relevant to peptide growth and it will be discussed in more detail below.

A further module, called **Structure Refinement**, allows for the mixing together of the various structures produced by the Structure Generation module [41]. It employs a genetic algorithm to mix and mutate the structures in order to increase the quality of the solutions, but this has only been applied to organic molecules so far.

Building peptides in PRO_LIGAND

Figure 1 shows the structure of the peptide libraries. The first level of demarcation concerns the overall extent to which the fragments are labelled. Fragments are labelled:

(1) sparsely – only a few side-chain features are labelled as interaction sites;

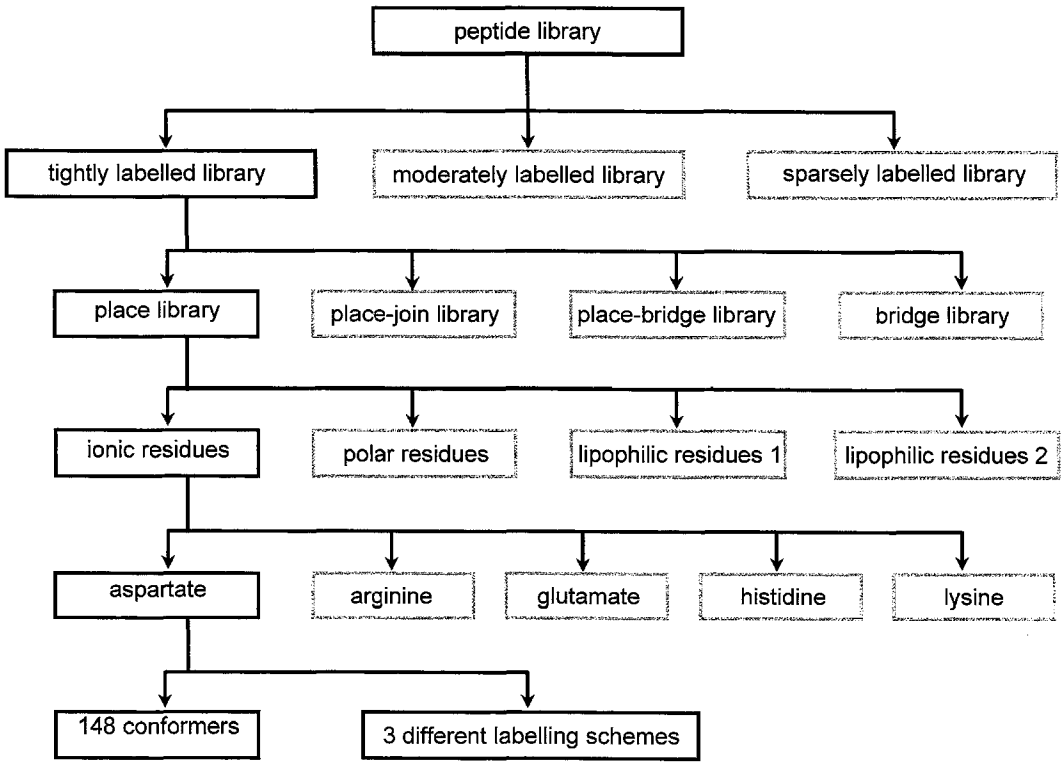

Fig. 1. Overview of the structure of the peptide library.

(2) moderately – only one of the backbone features (the amide carbonyl as an acceptor group or the amide N-H donor group) is labelled in addition to side-chain features; or

(3) tightly – both backbone features are labelled as interaction sites, as well as side-chain features.

Which of these gross labelling schemes is used will depend a lot upon the application. Fragments that are labelled more tightly are appropriate for dense and well-specified design models (such as the HIV-1 protease example), whereas more sparsely labelled residues are more appropriate where a less specific backbone configuration is required (such as in the lysozyme example). Each of these libraries is subdivided into place, place-join, place-bridge and bridge libraries; this subdivision reflects the different modes of building that are possible in structure generation, which will be expanded upon in the next section. The next subdivision is into various classes of amino acid residues such as hydrophobic, ionic and polar. The library thus has a directory structure with the library files themselves residing in the bottom directory, corresponding to the classification of the residues. Obviously, our choice of how to classify the residues has been somewhat subjective and will not be ideal for any particular application. This is compensated for by the fact that the user can rearrange and reclassify the library files relatively easily. The organisation of this bottom level of directories is important, because the user can rank the various sets of fragments in these directories to ensure, for instance, that the ionic and polar residues are always accessed before the hydrophobic ones.

Each library file contains a set of different labelling schemes, which will be described below. There is also a pointer in the library files to a number of preconstructed conformations for that residue. A description of the construction of the conformations will be given below.

Labelling of residues

The library files contain the labelling of the interaction sites for each residue, i.e., residue atoms are designated as aromatic (**R**) or aliphatic (**L**) lipophilic sites, or pairs of atoms are designated as hydrogen-bond acceptors (**A-Y**) or donors (**D-X**). Even within a particular gross class of labelling, often different choices of labelling are required. For instance, in aspartate either or both of the oxygens of the carboxylate group are suitable to be defined as acceptor interaction sites. In our 'tight' labelling scheme, aspartate has three different groups of interaction sites to describe this case (Fig. 2), with the group having both carboxylate oxygens labelled given a higher rank than the two groups where only one oxygen is labelled. This ensures that hits with the tighter labelling scheme (which indicates the full hydrogen-bonding capability of the aspartate side chain) are always located in preference to the labellings involving only one of the acceptor groups. It is important to note that the interaction-site labels

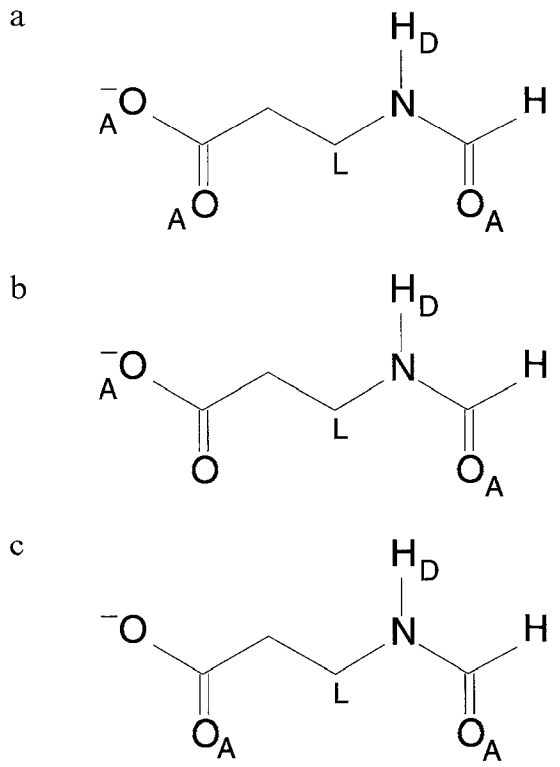

Fig. 2. Diagram showing a 'tight' labelling scheme for an aspartate fragment in the place library. The fragment with labelling scheme (a) will be ranked higher than the fragments labelled (b) or (c). The labels A, D and L denote acceptor, donor and lipophilic aliphatic interaction sites, respectively. X and Y sites are added automatically to fragments by the program.

reflect the properties of individual atoms, not the residue as a whole. Thus, for instance, the C^{β} atom of arginine is labelled as lipophilic (**L**), even though the residue is generally considered 'ionic'. The presence of such lipophilic sites is important to aid the program in growing structures across a design model.

As well as interaction sites, the library fragments can also have sites for joining marked on them. A typical fragment used for bridging between already placed fragments is shown in Fig. 3. In this figure, one hydrogen atom is suffixed with a 'J' to indicate that it is one type of join site, whilst another has been marked with a 'P' as another type of join site. Obviously, a polypeptide will result by repeatedly laying 'P' sites on top of 'J' sites and deleting the two labelled hydrogens. This is essentially what happens in the building process. Note that we have chosen to retain the amide character in our library fragments. This is because there is restricted conformational freedom about the relevant C-N amide bond and this reduces the number of conformations that need to be stored. The fragment shown is the only reasonable choice that preserves the amide bond, because otherwise proline has to be treated as a special case. A further reduction in the number of rotatable bonds could have been obtained by splitting residue fragments into a backbone part and a side-chain part, but we have preferred to keep intact residues, since this is conceptually simpler and more natural.

Construction of conformers

The library files themselves contain a pointer to all the conformations that have been prestored for that fragment. The strategy of prestoring peptide conformations has been used by other workers in this field [14], but is open to debate. A more complicated and theoretically satisfying approach would be to store one conformation and use a different fitting procedure that takes account of the residue's rotatable bonds in a manner analogous to 3D database searching methods [48,49]. This method is attractive in some respects, but it is unclear how efficient it would actually be; we therefore decided to adopt the more straightforward approach of prestoring conformers, since this has allowed us to quickly test the potential of our rule-based interaction-site approach.

Several factors are important in the construction of a library of appropriate conformations. Firstly, it is important that the conformers are energetically reasonable, since no attempt is made to adapt the conformation after the fragment is fitted. Secondly, for similar reasons it is important that an 'adequate' coverage of conformational space be attained. The concept of 'adequate' is rather woolly, but our definition has centred on the ability of the program to approximately reproduce segments of peptides extracted from the crystal structure of a large protein. These segments were about 8–10 residues long and ranged from loop regions to tightly ordered helices. This type of test can be done by producing a design model that is similar to the extracted segments and performing a Structure Generation calculation using this design model and the peptide library. It is also vital that the conformational libraries be as small as possible, since the presence of too many conformations will significantly slow down the performance of the program. This means that no more than an 'adequate' coverage of space is represented and also that the various regions of conformational space are sampled evenly. It is very difficult to satisfy these criteria in an objective manner, but it should be borne in mind that our purpose here is to test our whole theoretical approach to de novo peptide design and to prove that it works rather than produce the best and

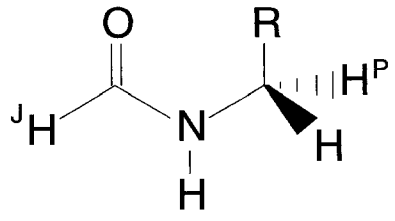

Fig. 3. Generic fragment definition for amino acid residues.

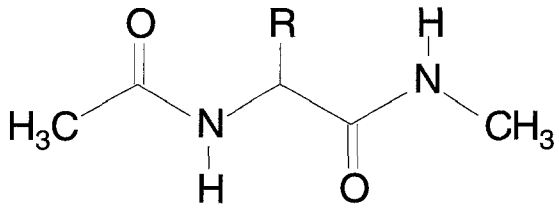

Fig. 4. Generic dipeptide used in the preparation of the amino acid conformations in the library.

most objective sampling scheme for the conformers. We tested several different approaches to conformer generation, but finally settled on the following approach, which is similar to that used by Moon and Howe [14]:

(1) An appropriate number of dipeptide conformers of the type shown in Fig. 4 were created using a random search of torsional space. Obviously, more conformers are required for residues which are more conformationally flexible.

(2) These structures were then partially minimised using a simplex method and an energy for each conformer was calculated.

(3) For each amino acid residue, this complete set of conformers was clustered according to the similarity of interatomic distances within each conformer. This clustering exercise enabled some measure of the 'evenness' of spread of the fragments to be ascertained.

(4) Structures were then removed from the set on the basis of unreasonably high energy or close similarity to other conformations.

(5) Finally, the relevant portion of the dipeptide was extracted and put in binary form for storage in the residue conformation file.

Most of this procedure was automated using the expert system facilities present in our in-house molecular simulation and design package, PROMETHEUS [50,51]. The POLY force field of Robson and Platt was used for energy evaluations [52]. An alternative approach to the construction of a conformer library might be to select preferred backbone and side-chain conformations of each peptide residue, based on theoretical and/or experimental data [53,54]. However, it may well be unwise to assume that the conformations preferred in either an unbound peptide or a protein crystal structure are necessarily those that would be preferred by an oligopeptide binding in an active site. Thus, we feel that our approach, like that of Moon and Howe, is a valid one, particularly in the early stages of program development.

All 20 naturally occurring amino acids were obtained using this method, except for glycine and proline which were treated slightly differently. The final number of conformations for each residue produced by the approach described above is shown in Table 1. It is noticeable that we are using about an order of magnitude fewer conformations than Howe and Moon used in GROW. One reason for this is that because of the definition of our fragment (see Fig. 3), we have one less rotatable bond; another reason could be that because of the great differences between the two approaches, we require a less complete coverage of conformational space.

Structure generation

Four different phases of fragment fitting are available in the Structure Generation module:

(1) *Placing* – the residues are placed onto the interaction sites specified in 3D coordinate space in the design model.

(2) *Place-bridging* – the residues are placed onto interaction sites, but are also constrained to form two or more bonds with fragments that have already been fitted.

(3) *Place-joining* – the residues are placed onto interaction sites, but are also constrained to form one bond with fragments that have already been fitted.

(4) *Bridging* – the residues simply bridge (form two or more bonds) between the previously placed fragments.

Which phase of fitting is used at a particular point in the building of a structure is to a large degree controlled by the user and a wide variety of building strategies are available. In peptide building we have generally used a 'growing' strategy, where one fragment is placed and subsequent fragments are all place-joined; this prevents the production of disjoint structures.

Both the design model and a given library fragment can be considered as *graphs*, where the interaction sites correspond to the graph's nodes and the inter-site distances to the graph's edges. This being so, it is possible to employ a rapid subgraph isomorphism algorithm [55] to detect whether a particular fragment can fit upon a subset of the design model sites to within a distance tolerance specified by the user. Specifically, a fragment is considered to match a subset of design model sites if a one-to-one correspondence exists between the types of the fragment interaction sites and those of the design model sites, and if the corresponding inter-site distances are all within

TABLE 1
NUMBER OF CONFORMATIONS USED FOR EACH AMINO ACID

Residue	No. of conformations	Residue	No. of conformations
Ala	72	Leu	117
Arg	315	Lys	297
Asn	203	Met	167
Asp	148	Phe	152
Cys	128	Pro	19
Gln	305	Ser	92
Glu	275	Thr	213
Gly	19	Trp	173
His	139	Tyr	207
Ile	123	Val	112

Amino acids are denoted by the standard three-character code.

the specified distance tolerance. This kind of approach has been widely used in the field of searching 2D and 3D chemical structure databases [56]. Once a fit has been found for a fragment, the fragment is placed onto the appropriate design model interaction sites or link sites using an rms superimposition algorithm. The fit is rejected if the new residue clashes with any already placed residues or with receptor atoms. The geometry about any new bonds is corrected, so as to take account of any bad bond lengths or bond angles produced by the fitting procedure. It is important to realise that any fragment that fits to within the user-defined tolerances and passes these few simple clash-checking tests is accepted; no attempt is made to differentiate between good and bad fits. The sequential building of peptides from a choice of residues and conformations rapidly leads to a combinatorial explosion in the size of the solution space as more residues are added. Our method corresponds to a depth-first solution to the problem and the program rapidly locates solutions of varying quality and great diversity. Howe and Moon employ a method that is more akin to a breadth-first search and which will be much slower at locating solutions, although the average quality of the solutions should be higher. However, there is a danger with this strategy that high-quality solutions will be missed when a particular conformation which scores badly is rejected, even if it would have allowed later residues to make especially good contacts. In order to operate the depth-first strategy, it is important that conformations are accessed in random order, and that interaction site groupings for a particular fragment are accessed randomly. The exceptions to this are when the user has specifically ranked particular fragments or particular interaction site labelling schemes higher than others.

After the structure has been built, it will be automatically scored. In the case of peptides, the score simply reflects how well the structure fits the design model and is a weighted sum of the number of interaction sites of each type of hit. The magnitude of the weights can be specified by the user, so that structures with the desired characteristics are given the best scores. Full details of this scoring function are given in Ref. 39. The use of an empirical scoring function is in keeping with the rule-based interaction site strategy; other, more complicated empirical scoring functions are available [23], but we prefer to retain this conceptually simpler and easily amendable function. If nothing else, it prevents too great a reliance be placed on empirical scoring functions, which may be misleading if trusted implicitly. Obviously, any design that we wish to take further will be subjected to energy-based calculations. Howe and Moon use a molecular mechanics-based scoring function, which is in keeping with the fact that it is more important in a breadth-first strategy that the criteria for judging the quality of emerging solutions is as accurate as possible.

Several options are available in structure generation that are relevant to peptide growth. Firstly, it is possible to define a *seed*, i.e., a fragment to be included in the built structure and to be used as a starting point for structure generation. Also, (parts of) previously generated structures can be used as seeds and this can be thought of as trapping information about attractive combinations of residues and directing the depth-first strategy into favourable areas of the solution space. A second option that is useful in some peptide runs is prescreening. This involves carrying out an initial placement of all library conformers upon the design model, using the subgraph isomorphism algorithm mentioned above with the fitting tolerances specified by the user. Any conformers found not to fit will not be accessed further during the subsequent building process. In the protease example given later, this produces a speed-up of many times, since the majority of the conformations of the ionic and polar residues will not fit on the design model.

Results

As a test of the capabilities of the peptide building functionality in PRO_LIGAND, we have applied it to two examples. The first is the design of an inhibitor of HIV-1 protease using information derived from the crystal structure of the enzyme. This is typical of the type of problem which de novo design programs have been tested on and for which they are useful. The second example is more unusual and is aimed at trying to design a peptide that is similar to an epitope region on the protein lysozyme. The idea was to extract only the key features in part of a lysozyme epitope and to try to grow novel peptide solutions which hit some of these key features. Clearly, this leads to a sparse design model and is a much more testing example for PRO_LIGAND. The ability to design similar to epitopes would be helpful in the design of synthetic peptide vaccines.

Complementary design to HIV-1 protease

In the search for therapeutics and vaccines to treat and prevent acquired immunodeficiency syndrome (AIDS), much attention has focussed upon its causative agent, human immunodeficiency virus (HIV). In the life cycle of HIV, the processing of *gag* and *gag-pol* polyproteins by the enzyme HIV-1 protease has been shown to be essential for viral replication. Thus, it is generally believed that if the activity of the protease can be inhibited, the spread of viral infection can be attenuated [57,58]. The protease has thus become a popular target for rational drug design efforts and a number of novel inhibitors have been designed using SBDD methods [59–61].

PRO_LIGAND was used to design peptides that are complementary to the active site of the protease. The crystal structure used in this example was that of HIV-1

Fig. 5. A peptide design from PRO_LIGAND (thick-stick representation), superimposed on the design model for inhibitors to HIV-1 protease. For clarity, the active-site water molecule is also shown in a thick-stick representation.

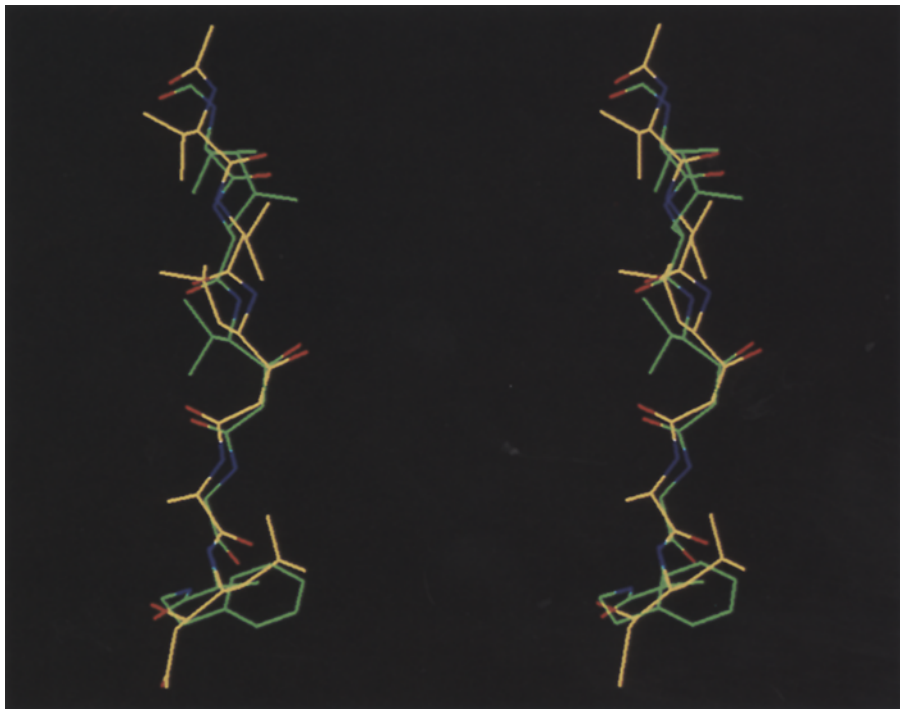

Fig. 6. Acetyl pepstatin (with the carbon atoms shown in yellow) superimposed onto the PRO_LIGAND design for an inhibitor to HIV-1 protease.

protease complexed with the inhibitor acetyl pepstatin (PDB entry 5HVP) [62]. The design base and design model for the protease were obtained in the manner described by Clark et al. [39]. This led to a design base of 491 protease atoms, including a water molecule necessary for mediation in the contact between Val^3 and Sta^4 in the inhibitor and Ile^{50} and Ile^{250} in the protease [62]. The design model contained 855 interaction sites. The large number of interaction sites results from the fact that in 'complementary design' mode, the Design-model Generation module generates a cluster of interaction sites complementary to each feature in the receptor active site that is capable of hydrogen bonding. These clusters represent the various possible hydrogen-bond geometries between the putative ligand and the receptor. Thus, for instance, a C=O group in the design base will yield a cluster of **D-X** donor sites in the design model. These sites will be placed at an appropriate distance from the C=O group and will form an appropriate range of angles with it.

To build structures to fit this model, a 'growing' strategy for structure generation was employed (where the first fragment was placed and all subsequent fragments were place-joined). We have often found that this kind of strategy is preferable to the 'outside-in' kind of strategy (where fragments are exhaustively placed and then bridges between them are sought as a separate step), simply because it guarantees that fully connected structures will be generated. With such a large design model, it is highly unlikely that an 'outside-in' strategy would succeed in joining all the fragments placed in the placing phase of building, because of the limited number and conformational rigidity of the bridging fragments. Owing to the fairly high density of design model interaction sites in this example, the tightly labelled and moderately labelled residue libraries were selected. A screening option was used to screen out all conformations which could never fit onto the design model. This caused most polar and ionic residue conformations to be screened out, in agreement with the expectation that the protease favours cleavage around hydrophobic residues.

Peptides with lengths similar to that of the natural inhibitor were grown to fit into the active site. However, such solutions are not attractive designs for inhibitors since they represent substrates for the enzyme. To overcome this problem, a seed fragment was used which had the desired property of occupying the active site region without being cleaved by the protease. A suitable seed fragment should be a transition-state isostere of the protease and many choices are available from known inhibitors (e.g., statines, reduced amides, dihydroxyethylenes, etc.) [63]. The seed we have used is an ethanol fragment derived from acetyl pepstatin in its bound conformation in the crystal structure. Moon and Howe used a similar strategy with a larger seed in their investigations on HIV-1 protease using the GROW program [14,64]. One of the best structures that PRO_LIGAND constructed is shown superimposed on the design model in Fig. 5. The blue design model sites are donor sites where, for instance, N-H bonds from amides can be placed; the dark blue and lighter blue portions denote the hydrogen atom position and the attached heteroatom position, respectively. The red/orange sites are acceptor positions where, for instance, C=O bonds from amides can be placed; the red portion indicates the position of the acceptor atom and the orange portion the position of any attached atom. Finally, the white dots are lipophilic features which map the shape of the active site. As can be seen, the chemistry of the designed molecule matches the design model well. For instance, the important water molecule in the centre of Fig. 5 is hydrogen bonded to the carbonyl groups of the valine and glycine residues adjacent to the ethanol seed fragment. All placed residues have hit at least one hydrogen-bond site and should therefore form one hydrogen bond with the protease. An exception is the ethanol seed fragment itself, where the O-H bond does not line up properly with a donor site vector; this indicates a problem with the crystal structure of the bound inhibitor, where there is a clash between the oxygen from the relevant O-H in acetyl pepstatin and an aspartate oxygen in the enzyme [62].

The sequence of the designed structure is Val-Leu-Val-ethanol-Gly-Ala-Phe and this compares with the sequence of acetyl pepstatin, which can be written as acetyl-Val-Val-Sta-Ala-Sta, where Sta indicates a statine residue. Obviously, the end points of the designed peptide cannot be similar to those of pepstatin, since PRO_LIGAND has been constrained to construct from the 20 amino acid fragments in our libraries. As expected, the designed ligand contains mainly hydrophobic residues, in agreement with the observation that the protease preferentially cleaves around such residues. Figure 6 shows a stereo image of acetyl pepstatin in its crystallographic conformation, with the designed molecule in the conformation produced by PRO_LIGAND. The rms deviation between the backbone positions is 0.58 Å. This is a good result considering that, apart from the starting coordinates of the two backbone seed atoms, the only information used to design the ligand was obtained from the enzyme structure.

Similar design to epitope regions of lysozyme

The identification of epitopes is a key area in synthetic vaccine design. However, there is no guarantee that an isolated epitopic sequence will elicit an immune response when synthesised, because the immune system recognises the 3D arrangement of key features in the epitope rather than the sequence itself. This is exemplified by the existence of conformational or discontinuous epitopes. It is important, therefore, to develop methods for designing peptides similar to the conformation of known or pre-

dicted epitopes. For this reason, we have looked at an epitopic region in lysozyme and extracted some of the key features that are responsible for recognition by antibodies, with the intention of using PRO_LIGAND to grow peptides similar to this region. This example critically tests the program's ability to generate peptides in regions where only a few interaction sites are defined.

We have used the crystal structure of lysozyme complexed with the antibody D1.3 (PDB entry 1FDL) [65]. The epitope regions on lysozyme have been given as residues 18–27 and 116–129 [66]. For the purposes of this example, we have chosen to simplify the design problem by focussing on the residues between positions 117 and 125. Not all residues in this sequence are in contact with the antibody. We recognised the following hydrogen-bond features on lysosyme as being in contact with the antibody: the carbonyl backbone group on Gly^{117}; the carboxylate group on Asp^{119}; the N-H backbone group on Val^{120}; the amide group on the side chain of Gln^{121}; and the N-H bonds on the side chain of Arg^{125}. Precisely these features will form the design base. From this design base, a design model template is generated onto which peptides of appropriate chemistry will be fitted and grown. Figure 7 shows the design model used for this problem, superimposed onto the parts of the lysozyme sequence from which it is derived. The design model features are coloured in the same manner as for the HIV-1 protease example. Note that, by contrast to that example, in this case we wish to design structures which are similar to the design base. In similar design, only one design model interaction site results from each hydrogen-bonding feature in the design base. The fragments of the lysozyme sequence (the design base) are discontinuous, since the side chain of the arginine residue is not attached to the other portions of the peptide. Many of the atoms in the design base that are not in contact with the lysozyme were specified as lipophilic sites (white dots) to encourage growth between the other design model sites. Our goal is to design a continuous peptide which fits as many of the hydrogen-bond interaction sites as possible.

The medium and loosely labelled libraries were used for the structure generation runs and the program was run in a 'growing' mode. In this sparse design problem, a very large number of solutions of varying quality are available to the program and a large number of structures need to be generated to obtain reasonable solutions. However, the program automatically ensures that only peptides beyond a specified length are printed out and these can be sorted easily. For this example we routinely generated 2500 structures overnight on an HP 735 workstation. Of these, about 10 would be four or more residues long. The solutions given here were obtained from several runs of this type. Figure 8 shows three of the best solutions superimposed on the design model; hydrogens have been omitted from the figure and the peptides have been coloured differently for clarity. The sequences as well as information on the features hit for these peptides is given in Table 2.

TABLE 2
DESIGNED PEPTIDE SEQUENCES AND INTERACTION SITES HIT FOR SIMILAR DESIGNS TO THE LYSOZYME EPITOPE REGION

Peptide	Sequence	No. of acceptor sites hit	No. of donor sites hit
A	Glu-Lys-His-Asn-Arg	3	3
B	His-Asp-Gln-Arg	4	2
C	His-Asp-Asn-Ser	3	3

As can be seen, the program successfully grows peptides that span the design model and hit an appreciable number of the features specified. This occurs despite the fact that the design model is very sparse and constructed from a discontinuous design base. The potential of our approach in the area of similar design is thus demonstrated, although in a real design problem more care would be needed in the construction of the design model. We are currently looking into methods of improving the design model generation code for similar design. Another concern is that the generated peptides may not be in a low-energy conformation since, at present, the program does not permit relaxation of the growing peptide. The inclusion of force field calculations in structure generation is one element of our ongoing research in this area. It should be noted that the designed peptides do not adopt a similar conformation to the fragments of the lysozyme sequence used as a design base. This is to be expected, because the original sequence is discontinuous so there is probably not a continuous solution that follows this backbone conformation. A further point of interest is that two of the designs shown in Fig. 8 contain the consensus sequence His-Asp, which could serve as a starting point for the design of further variants using a combinatorial library procedure [67].

Discussion

The HIV results are encouraging. Using the simple rule-based approach and the concept of interaction sites, we have produced a solution which essentially reproduces the backbone configuration of a known inhibitor, with side chains which seem reasonable. In this sense, the rule-based approach can be considered to be viable for these flexible fragments. The lysozyme example was less encouraging, although it is worth mentioning that only an interaction site-based approach could deal with this type of problem which does not lend itself to the scoring of ligands through an energy function. What made this problem more difficult than the protease example was the sparse design model, comprising few interaction sites. This meant that an enormous number of peptides could

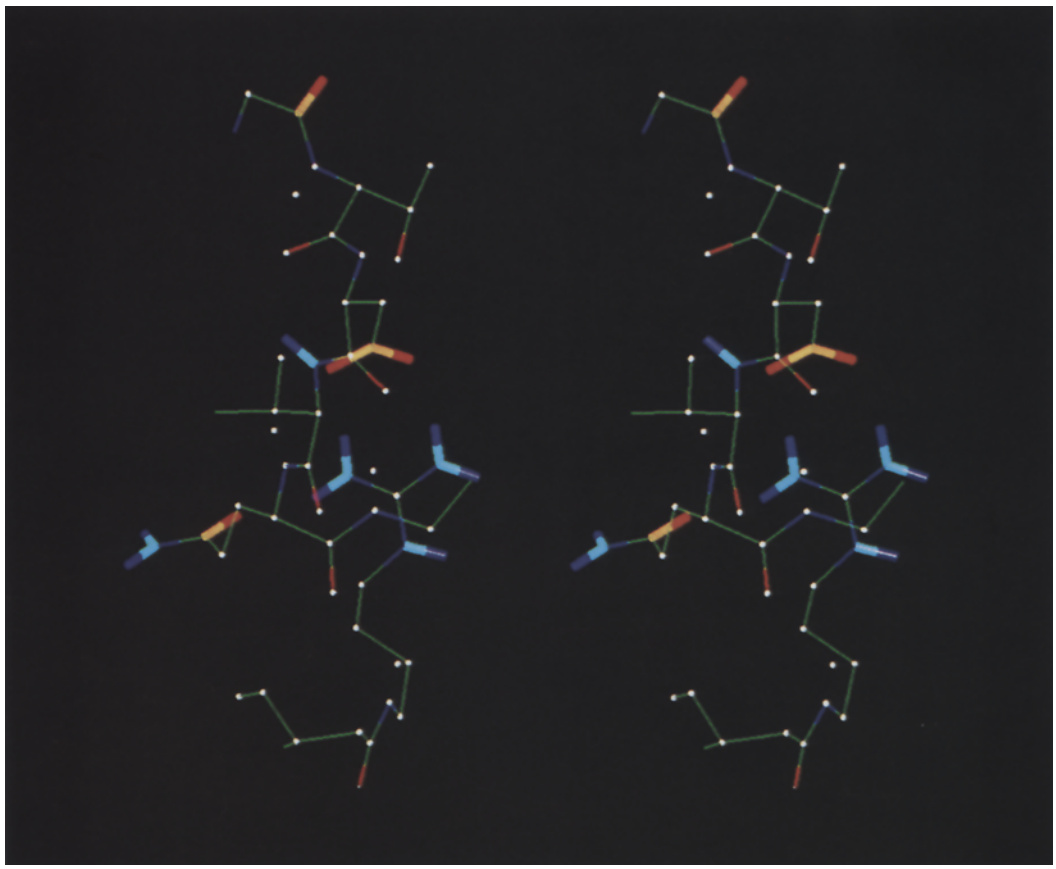

Fig. 7. Part of the epitope region of lysozyme, superimposed on the design model derived from it. White dots denote lipophilic sites, blue and red/orange thick-stick vectors denote hydrogen-bond donor and acceptor sites, respectively.

fit onto the design model and there was a very large solution space to be searched through. Additionally, it is not clear that there was a good solution to this very difficult problem since there were so few sites in the design model.

The success of the protease example is comparable to that obtained by Howe and Moon with the GROW program, albeit with an entirely different methodology. They went on to synthesise their peptides and prove that they had activity. At this stage, we have not attempted to synthesise any of our designs since our aim was more to test our rule-based approach to this problem and to point the way to improvements. It is perhaps appropriate to consider an explicit comparison between PRO_LIGAND and GROW – both aimed at the de novo design of peptides. The similarities are:

(1) The use of prestored peptide conformations which are aimed at giving an even sampling of low-energy conformational space.

(2) The use of seeding and growing strategies to produce good solutions.

The differences are:

(1) PRO_LIGAND uses interaction sites to decide on the placement of fragments; GROW uses energy-based methods.

(2) PRO_LIGAND takes the conformations directly to form the final molecule; GROW allows some force field relaxation of the peptide as it is being built.

(3) PRO_LIGAND uses a depth-first strategy; GROW's strategy for accepting hits is more like a breadth-first search.

Certainly an unattractive feature of both methods is their reliance on preformed libraries of conformations; one can never be sure if enough or too many conformations are in the library without extensive and painstaking testing. An alternative which we are actively pursuing is to use the directed tweak method [48], which would allow one to adjust the conformation of the fragment to fit design model sites.

The second similarity is the use of seeding and growing strategies to produce solutions. In fact, PRO_LIGAND has greater flexibility than GROW with regard to the available building strategies, but in our tests on peptides we have found that the 'grow' strategy (i.e., sequential joining of the fragments) is the most effective. The use of seeding is an important way of optimising the performance of PRO_LIGAND; however, the program can operate entirely de novo. This illustrates an important difference between GROW and PRO_LIGAND, since GROW requires that the position of the first fragment be spec-

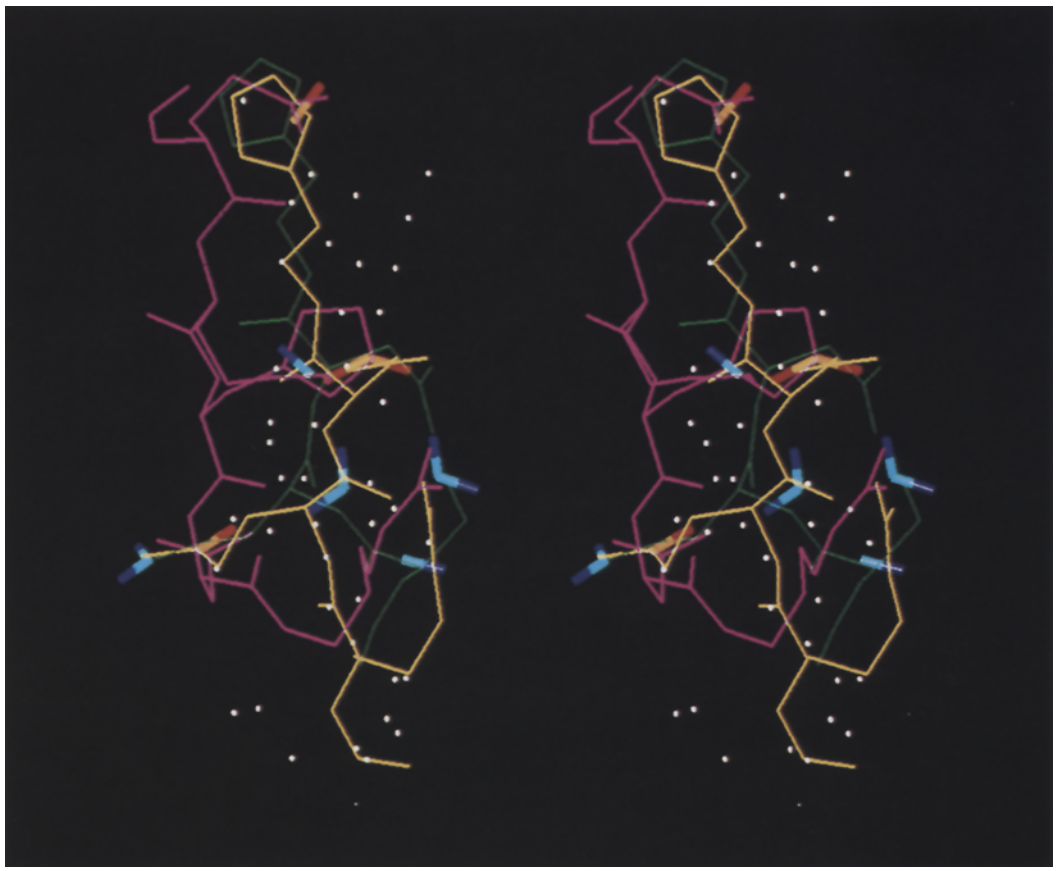

Fig. 8. PRO_LIGAND designs for structural analogues of the epitope region of lysozyme, superimposed on the design model. The sequences of peptides A (magenta), B (yellow) and C (green) are given in Table 2.

ified. Additionally, GROW appears to operate best when the sequence (and even the conformation) of the growing peptide is restricted. This is because it is difficult to compare and make judgements on the scores of fragments when they are very different (i.e., a good score for valine cannot be directly compared with a good score for aspartate). This is less of a problem with interaction site approaches, partially because they allow one to use a depth-first strategy.

The first difference mentioned above is really the fundamental difference between the two algorithms, and the purpose of the current work is to test how viable an interaction site methodology is for flexible fragments. Such a method has been proven to be of merit for organic design, but was hitherto untested for peptides. We believe that the speed, interpretability and simplicity of rule-based methods make them an attractive alternative to molecular mechanics-based approaches. It should be emphasised, however, that designs produced by PRO_LIGAND should be subjected to subsequent and rigorous computational evaluation using energy-based methods to probe the strength and specificity of the ligand binding. Another advantage of rule-based approaches is that they can be readily applied to design problems where energy methods are not appropriate, such as similar design to pharmacophores.

In the interaction site method, explicit calculation of the interaction between the receptor and the growing peptide is avoided. As a result, no kind of force field calculation is performed on the design during its construction, and no relaxation of the peptide conformation occurs. This is a second difference between PRO_LIGAND and GROW, since GROW allows some relaxation of the fragments after they have been placed. We believe that this is a deficiency of our program – especially for these flexible molecules. Because the ideal conformation is never in the library, the geometry of a placed fragment tends to be suboptimal, and as the peptide grows these errors accumulate. The problems are apparent with similar design experiments, where one tries to reproduce a segment of a peptide extracted at random from some crystal structure. Once the segment grows beyond a length of about eight residues, it becomes very difficult to build structures that are similar to the original sequence. Part of the answer may be the introduction of the directed tweak method, which will make all possible conformations accessible to the program. Another approach that we hope to implement is to introduce a force field and allow the constrained relaxation of the peptides onto the design model interaction sites that they hit. This work will be reported in a future paper.

The final difference highlighted between the two approaches is the use of different strategies for searching through the solution space. We believe that our depth-first strategy is the more natural one to use with rule-based methods, because it is very difficult to get an accurate estimate of the score of placed fragments (although Böhm has recently made progress in this area [23]). In fact, even in situations where an accurate assessment of a fragment's score is possible, it is still difficult to prune the solution space 'tree' with any certainty. The reason for this is that, when assessing whether a fragment with a particular conformation should be placed, it is impossible to know if this conformation will allow the subsequent placement of especially favourable residues later. The depth-first strategy also has the advantage of allowing a relatively fast production of a wide variety of solutions. This enables one to identify promising areas of the solution space which can then be explored further by ranking the residues appropriately or using a seed. It should be noted that a depth-first strategy is more difficult to operate with energy-based methods, where it is hard to establish that a fragment should be placed without first testing the energies of all other conformations and residues.

Conclusions

We have extended our de novo design package, PRO_LIGAND, to incorporate a specific strategy for designing peptides. It is based on our previous methodology, which uses rules to define interaction sites that are derived from the design target. This methodology has been shown to be effective in the design of organics, but has never been applied to peptides, which offer special problems due to their great flexibility. Our results with HIV-1 protease show that the approach is promising for this type of problem, which is typical of the de novo drug design problems that are currently being attempted by other workers in the field. The more difficult example of growing a peptide similar to an epitope region in lysozyme did produce peptide solutions that span the space of the region studied. This demonstrated the utility of the program, but the generated solutions did not look particularly promising, reflecting the difficulty of the design problem rather than deficiencies in the methodology. All in all, we are encouraged by the results so far, and are currently investigating more sophisticated approaches aimed at avoiding the need for a library of conformations for each of the residues.

Acknowledgements

We thank Stephen A. Levy for the development of the binary representation of the conformations of amino acids.

References

1 Walkinshaw, M.D., Med. Res. Rev., 12 (1992) 317.
2 Ealick, S.E. and Armstrong, S.R., Curr. Opin. Struct. Biol., 3 (1993) 861.
3 Fesik, S.W., J. Biomol. NMR, 3 (1993) 261.
4 Zuiderweg, E.R.P., Van Doren, S.R., Kurochkin, A.V., Neubig, R.R. and Majumdar, A., Perspect. Drug Discov. Design, 1 (1993) 391.
5 Reich, S.H. and Webber, S.E., Perspect. Drug Discov. Design, 1 (1993) 371.
6 Greer, J., Erickson, J.W., Baldwin, J.J. and Varney, M.D., J. Med. Chem., 37 (1994) 1035.
7 Whittle, P.J. and Blundell, T.L., Annu. Rev. Biophys. Biomol. Struct., 23 (1994) 349.
8 Verlinde, C.L.M.J. and Hol, W.G.J., Structure, 2 (1994) 577.
9 Colman, P.M., Curr. Opin. Struct. Biol., 4 (1994) 868.
10 Mayer, D., Naylor, C.B., Motoc, I. and Marshall, G.R., J. Comput.-Aided Mol. Design, 1 (1987) 3.
11 Sheridan, R.P., Nilakantan, R., Dixon, J.S. and Venkataraghavan, R., J. Med. Chem., 29 (1986) 899.
12 Cramer, R.D., Patterson, D.E. and Bunce, J.D., J. Am. Chem. Soc., 110 (1988) 5959.
13 Martin, Y.C., Bures, M.G., Danaher, E.A., DeLazzar, J., Lico, I. and Pavlik, P.A., J. Comput.-Aided Mol. Design, 7 (1993) 83.
14 Moon, J.B. and Howe, W.J., Protein Struct. Funct. Genet., 11 (1991) 314.
15 Moon, J.B. and Howe, W.J., In Wermuth, C.G. (Ed.) Trends in QSAR and Molecular Modelling 92 (Proceedings of the 9th European Symposium on Structure–Activity Relationships: QSAR and Molecular Modelling), ESCOM, Leiden, 1993, pp. 11–19.
16 Miranker, A. and Karplus, M., Protein Struct. Funct. Genet., 11 (1991) 29.
17 Caflisch, A., Miranker, A. and Karplus, M., J. Med. Chem., 36 (1993) 2142.
18 Nishibata, Y. and Itai, A., Tetrahedron, 47 (1991) 8985.
19 Nishibata, Y. and Itai, A., J. Med. Chem., 36 (1993) 2921.
20 Böhm, H.-J., J. Comput.-Aided Mol. Design, 6 (1992) 61.
21 Böhm, H.-J., J. Comput.-Aided Mol. Design, 6 (1992) 593.
22 Böhm, H.-J., In Kubinyi, H. (Ed.) 3D QSAR in Drug Design: Theory, Methods and Applications, ESCOM, Leiden, 1993, pp. 386–405.
23 Böhm, H.-J., J. Comput.-Aided Mol. Design, 8 (1994) 243.
24 Böhm, H.-J., J. Comput.-Aided Mol. Design, 8 (1994) 623.
25 Lewis, R.A., Roe, D.C., Huang, C., Ferrin, T.E., Langridge, R. and Kuntz, I.D., J. Mol. Graph., 10 (1992) 66.
26 Rotstein, S.H. and Murcko, M.A., J. Comput.-Aided Mol. Design, 7 (1993) 23.
27 Rotstein, S.H. and Murcko, M.A., J. Med. Chem., 36 (1993) 1700.
28 Gillet, V.J., Johnson, A.P., Mata, P., Sike, S. and Williams, P., J. Comput.-Aided Mol. Design, 7 (1993) 127.
29 Gillet, V.J., Newell, W., Mata, P., Myatt, G., Sike, S., Zsoldos, Z. and Johnson, A.P., J. Chem. Inf. Comput. Sci., 34 (1994) 207.
30 Pearlman, D.A. and Murcko, M.A., J. Comput. Chem., 14 (1993) 1184.
31 Tschinke, V. and Cohen, N.C., J. Med. Chem., 36 (1993) 3863.
32 Ho, C.W.M. and Marshall, G.R., J. Comput.-Aided Mol. Design, 7 (1993) 623.
33 Leach, A.R. and Lewis, R.A., J. Comput. Chem., 15 (1994) 233.
34 Leach, A.R. and Kilvington, S.R., J. Comput.-Aided Mol. Design, 8 (1994) 283.

35 Eisen, M.B., Wiley, D.C., Karplus, M. and Hubbard, R.E., Protein Struct. Funct. Genet., 19 (1994) 199.
36 Bohacek, R.S. and McMartin, C., J. Am. Chem. Soc., 116 (1994) 5560.
37 Cohen, A.A. and Shatzmiller, S.E., J. Mol. Graph., 11 (1993) 166.
38 Cohen, A.A. and Shatzmiller, S.E., J. Comput. Chem., 15 (1994) 1393.
39 Clark, D.E., Frenkel, D., Levy, S.A., Li, J., Murray, C.W., Robson, B., Waszkowycz, B. and Westhead, D.R., J. Comput.-Aided Mol. Design, 9 (1995) 13.
40 Waszkowycz, B., Clark, D.E., Frenkel, D., Li, J., Murray, C.W., Robson, B. and Westhead, D.R., J. Med. Chem., 37 (1994) 3994.
41 Westhead, D.R., Clark, D.E., Frenkel, D., Li, J., Murray, C.W., Robson, B. and Waszkowycz, B., J. Comput.-Aided Mol. Design, 9 (1995) 139.
42 Lewis, R.A. and Leach, A.R., J. Comput.-Aided Mol. Design, 8 (1994) 467.
43 Lam, K.S., Salmon, S.E., Hersh, E.M., Hruby, V.J., Kazmierski, W.M. and Knap, R.J., Nature, 394 (1991) 82.
44 Houghten, R.A., Pinilla, C., Blondelle, S.E., Appel, J.R., Dooley, C.T. and Cuervo, J.H., Nature, 394 (1991) 84.
45 Singh, J., Saldanha, J. and Thornton, J.M., Protein Eng., 4 (1991) 251.
46 Klebe, G., J. Mol. Biol., 237 (1994) 212.
47 Allen, F.H., Bellard, S., Brice, M.D., Cartwright, B.A., Doubleday, A., Higgs, H., Hummelink, T., Hummelink-Peters, B.G., Kennard, O., Motherwell, W.D.W., Rodgers, J.R. and Watson, D.G., Acta Crystallogr., B35 (1979) 2331.
48 Hurst, T., J. Chem. Inf. Comput. Sci., 34 (1994) 190.
49 Clark, D.E., Jones, G., Willett, P., Kenny, P.W. and Glen, R.C., J. Chem. Inf. Comput. Sci., 34 (1994) 197.
50 Ball, J., Fishleigh, R.V., Greaney, P., Li, J., Marsden, A., Platt, E., Pool, J.L. and Robson, B., In Bawden, D. and Mitchell, E.M. (Eds.) Chemical Structure Information Systems: Beyond the Structure Diagram, Ellis Horwood, Chichester, 1990, pp. 107–123.
51 Robson, B., Ball, J., Fishleigh, R.V., Greaney, P., Li, J., Marsden, A., Platt, E. and Pool, J.L., Biochem. Soc. Symp., 57 (1991) 91.
52 Robson, B. and Platt, E., J. Mol. Biol., 188 (1986) 258.
53 Ramachandran, G.N., Ramakrishnan, C. and Sasiekharan, V., J. Mol. Biol., 7 (1963) 95.
54 Ponder, J.W. and Richards, F.M., J. Mol. Biol., 193 (1987) 775.
55 Ullmann, J.R., Biochem. Biophys. Res. Commun., 23 (1976) 31.
56 Brint, A.T. and Willett, P., J. Mol. Graph., 5 (1987) 49.
57 Kohl, N.E., Emini, E.A., Schlief, W.A., David, L.J., Heimbach, J.C., Dixon, R.A.F., Scolnick, E.M. and Sigal, I.S., Proc. Natl. Acad. Sci. USA, 85 (1988) 4686.
58 McQuade, T.J., Tomaselli, A.G., Liu, L., Karacostas, V., Moss, B., Sawyer, T.K., Heinrikson, R.L. and Tarpley, W.G., Science, 247 (1990) 454.
59 Appelt, K., Perspect. Drug Discov. Design, 1 (1993) 23.
60 Fitzgerald, P.M.D., Curr. Opin. Struct. Biol., 3 (1993) 868.
61 Redshaw, S., Exp. Opin. Invest. Drugs, 3 (1994) 273.
62 Fitzgerald, P.M.D., McKeever, B.M., VanMiddlesworth, J.F., Springer, J.P., Heimbach, J.C., Leu, C.-T., Herber, W.K., Dixon, R.A.F. and Darke, P.L., J. Biol. Chem., 265 (1990) 14209.
63 Waller, C.L., Oprea, T.I., Alessandro, G. and Marshall, G.R., J. Med. Chem., 36 (1994) 4152.
64 Sawyer, T.K., Staples, D.J., Liu, L., Tomasselli, A.G., Hui, J.O., O'Connel, K., Schostarez, H., Hester, J.B., Moon, J., Howe, W.J., Smith, C.W., Decamp, D.L., Craik, C.S., Dunn, B.M., Lowther, W.T., Harris, J., Poorman, R.A., Wlodawer, A., Jaskolski, M. and Heinrikson, R.L., Int. J. Pept. Protein Res., 40 (1992) 274.
65 Fischmann, T.O., Bentley, G.A., Bhat, T.N., Boulot, G., Mariuzza, R.A., Phillips, S.E.V., Tello, D. and Poljak, R.J., J. Biol. Chem., 266 (1991) 12915.
66 Davies, D.R. and Padlan, E.A., Annu. Rev. Biochem., 59 (1990) 439.
67 Gallop, M.A., Barrett, R.W., Dower, W.J., Fodor, S.P.A. and Gordon, E.M., J. Med. Chem., 37 (1994) 1233.