Modelling protein side-chain conformations using constraint logic programming

Martin T. Swain and Graham J.L. Kemp

Department of Computing Science, University of Aberdeen, King's College, Aberdeen, Scotland, UK, AB24 3UE

Abstract

Side-chain placement is an important sub-task in protein modelling. Selecting conformations for side-chains is a difficult problem because of the large search space to be explored. This problem can be addressed using constraint logic programming (CLP), which is an artificial intelligence technique developed to solve large combinatorial search problems. The side-chain placement problem can be expressed as a CLP program in which rotamer conformations are used as values for *finite domain variables*, and bad steric contacts involving rotamers are represented as *constraints*. This paper introduces the concept of null rotamers, and shows how these can be used in implementing a novel iterative approach. We present results that compare the accuracy of models constructed using different rotamer libraries and different domain variable enumeration heuristics. The results obtained using this CLP-based approach compare favourably with those obtained by other methods.

Key words:
constraint logic, protein modelling, protein structure, rotamers, side-chain

1 Introduction

Protein structures determined using X-ray crystallography or nuclear magnetic resonance provide essential information for understanding protein function, and it is hoped that this information will lead to novel, protein-based therapies. Although the determination of protein sequences may be fast, determining the detailed 3-dimensional structure of a protein is far more time

Email address: {mswain,gjlk}@csd.abdn.ac.uk (Martin T. Swain and Graham J.L. Kemp).

URL: http://www.csd.abdn.ac.uk/bioinf/ (Martin T. Swain and Graham J.L. Kemp).

consuming. As an alternative to experimental structure determination, homology modelling (Chinea et al., 1995; Holm and Sander, 1992; Laughton, 1994; Guex and Peitsch, 1997) is often used to construct model structures from amino acid sequences. The modelling procedure normally includes three main sub-tasks: aligning the sequence of the protein with unknown structure against the sequence of a homologous protein with a solved structure, creating the model backbone by adapting that of the known structure, and then placing amino acid side-chains onto the modelled backbone. In this paper we present a new method of performing this last task, modelling side-chains, using *constraint logic programming* (CLP).

The principal problem encountered when modelling side-chains is the extremely large number of possible combinations of side-chain conformations — infinite if we consider side-chain bonds to be continuously variable. For practical purposes the search space can be discretised by considering a finite set of possible torsion angles for each side-chain. However, this can still result in an enormous search space: if we were to consider a rotational step size of 10^o, then a protein with 100 amino acid residues, with 2 rotatable bonds per residue would yield a search space of $(36 \times 36)^{100}$ possible side-chain combinations (Lee and Subbiah, 1991).

The discovery that the distribution of side-chain conformations fell into statistically significant clusters (Ponder and Richards, 1987), known as *rotamers* (Schrauber, 1993; Dunbrack and Cohen, 1997; Heringa and Argos, 1999), have brought notable advances in side-chain modelling. Rather than considering regular torsion angle increments for each rotatable bond, one can instead choose from a much smaller set of torsion angles representing the most common side-chain conformations, or rotamers, observed in experimentally determined protein structures. Thus the vast combinatorial search space can be greatly reduced.

Rotamer libraries (Ponder and Richards, 1987; Tuffery et al., 1997; Dunbrack and Karplus, 1993; Dunbrack and Cohen, 1997; Lovell et al., 2000) typically contain up to 30-50 rotamers for side-chains with several rotatable bonds like lysine and arginine, and about three rotamers for side-chains with only one rotatable bond like serine, cysteine and threonine. An exception to this is the very extensive rotamer library developed by Dunbrack et al. (1993,1997) which has separate rotamer conformations listed for every 10^o of backbone ϕ and ψ torsion space. Like their backbone independent (BBIND) rotamer library (Dunbrack and Karplus, 1993), the backbone dependent (BBDEP) library typically contains three angular conformations per rotatable side-chain bond, resulting in 81 rotamers for lysine and arginine.

Lovell et al. (2000) state that within all published rotamer libraries there exist some rotamers that, if built in ideal geometry with all hydrogen atoms,

would contain impossible internal atomic overlaps. They built their library from high resolution structures, considering the uncertainties, or B-factors, in the position of the deposited structure co-ordinates. Having built their library they observed some significant differences to the rotamer distributions found in other published libraries.

Typical side-chain modelling methods consist of a potential energy function to describe the interactions of the side-chains represented by rotamers. This energy function plays an important role in discriminating between possible side-chain combinations since it is generally accepted that as the potential energy is lowered the protein model becomes more accurate. In a review of side-chain modelling, Vasquez (1996) states that most side-chain modelling algorithms use comparatively simple energy functions. These functions, based mainly upon van der Waals interactions, can give excellent results for residues in hydrophobic cores, however the results for surface and buried polar residues can be poor, and very little difference is made by adding terms for electrostatics and hydrogen bonding.

Various algorithms are used to search through the energy landscape generated by the potential energy function's description of the different combinations of rotamers. Typical optimisation methods include Monte Carlo algorithms (Holm and Sander, 1991, 1992), genetic algorithms (Tuffery et al., 1991) and dead end elimination (DEE) (Desmet et al., 1992; Taylor, 1992; Lasters et al., 1995). The DEE theorem is based upon a mathematical expression, the DEE criterion, that eliminates rotamers that cannot be part of the global minimum energy conformation. Thus, if the DEE method finds a solution, it will be guaranteed to be the global minimum. However, the potential energy function can only contain terms describing interactions between pairs of rotamers and, if the protein is very large, the DEE method may fail to converge on a solution.

Other methods (Wilson et al., 1993; Holm and Sander, 1991; Kono and Doi, 1996; Gordon and Mayo, 1999) using rotamers and energy functions include the self-consistent mean field method (Koehl and Delarue, 1994). A mean field description of the energy landscape is used to allow each residue to feel the average of all possible environments, smoothing out the energy landscape and avoiding local minima. Two very rapid methods depend on steric overlap to distinguish between rotamers: Bower *et. al* (1997) use a small back-bone dependent rotamer library, while Shenkin *et. al* (1996) incorporate the differences in rotamer probability into their optimisation procedures.

Not all algorithms with potential energy functions use rotamer libraries (Levitt, 1992; Eisenmenger et al., 1993). Such methods include simulated annealing (Lee and Subbiah, 1991) to search through every 10 degrees of side-chain dihedral angles, neural nets (Hwang and Liao, 1995) to generate distributions of side-chain dihedral angles used for Monte Carlo optimisation and a thermody-

namically based method to calculate the energy of a flexible rotamer (Mendes et al., 1999). Here a flexible rotamer is a continuous ensemble of conformations that cluster around the standard rigid rotamer. Another method compares the local environment of each unknown side-chain to a database of fragments with a similar backbone conformations. The optimal best matches are then found using a Monte-Carlo procedure (Laughton, 1994).

Constraint logic programming

Constraint logic programming (CLP) (Fruhwirth et al., 1993; Wallace, 1995; Carlsson and Ottosson, 1997), is a paradigm that has been developed by the artificial intelligence community for hard search problems. It uses constraints with a simple built in search method: the constraints eliminate impossible alternatives and so restrict and guide the search. CLP is usually applied to problems with such a significant combinatorial component that heuristics must be used. Typical application areas include planning, scheduling, packing, and resource allocation.

A CLP program consists of a set of variables that are each confined to their own discrete set of values. This set of values is known as a variable's *finite domain*. Constraints are the relationships between finite domains expressed using arithmetic, algebraic and boolean statements. A solution is a set of values for domain variables that satisfies the constraints.

This paper shows that the side-chain placement problem can be expressed in terms of constraints on rotamers, and that CLP can solve these constraints to predict the conformations of side-chains. In modelling, the most important constraint on side-chain atom packing is the avoidance of steric overlap (Vasquez, 1996; Shenkin et al., 1996; Bower et al., 1997). We compare the different CLP search methods that are part of the SICStus Prolog finite domain constraint solver (Carlsson and Ottosson, 1997), and the results of using the these CLP methods with different rotamer libraries (Dunbrack and Karplus, 1993; Dunbrack and Cohen, 1997; Lovell et al., 2000; Tuffery et al., 1997). Finally, we show that the CLP method can perform predictions with similar accuracy to other published methods (Tuffery et al., 1991; Levitt, 1992; Laughton, 1994; Lee and Subbiah, 1991; Holm and Sander, 1992; Hwang and Liao, 1995).

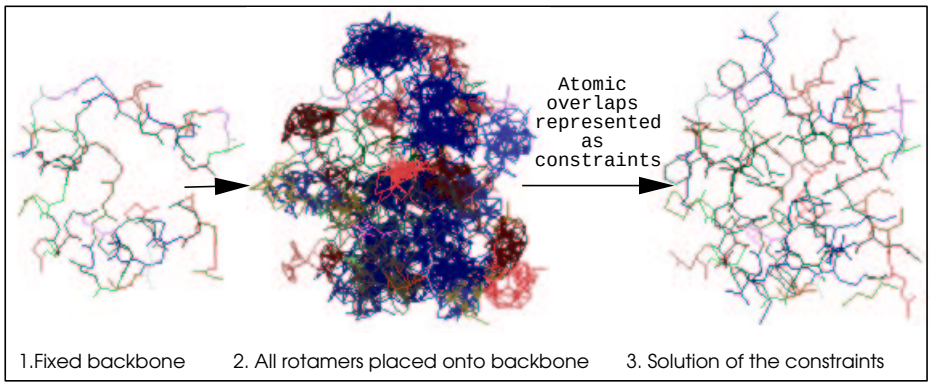

Fig. 1. The CLP method begins with a given backbone. All rotamers in the library for every backbone reside are placed onto the backbone. Steric overlaps are represented as constraints on the rotamers, and are solved by CLP to give a single rotamer for each residue.

2 Materials and methods

2.1 CLP method for side-chain placement

We are using CLP together with rotamer libraries to predict side-chain conformations. We assume that we already have a modelled backbone, and in testing our method we consider the extreme situation in which all side-chains have to be placed.

In outline, our method starts by placing the complete set of rotamers from a rotamer library onto the protein backbone. Steric overlaps between rotamers and the backbone, and between rotamers of different residues, are calculated using C code, and formulated as constraints. Our method then selects compatible side-chain conformations, as illustrated in Figure 1. In the experiments presented in this paper we do not fix any residue conformations.

The main features of a CLP program and its solution are *domain variables*, *constraints* and the *search strategy* used by the CLP solver.

Domain variables

CLP domain variables are used to represent residue positions along the protein chain, and a variable's finite domain is the set of rotamers corresponding to possible conformations of that residue's side-chain. In the rotamer libraries used in this paper, each rotamer has a stored probability value indicating how common the conformation is for that type of residue. When CLP methods try to find a value for a domain variable, they either start with the smallest value in the finite set and then consider others in ascending order, or else they start

with the largest in the set and consider others in descending order. Thus, the values assigned to rotamers are important in determining the order in which these are tried by the solver, and can affect the choice of conformations in the final solution. Therefore, we choose the value 1 to identify the most common side-chain conformation for each residue type, and higher values for successively less common rotamers.

Constraints

There are two types of constraints placed on the rotamers in the CLP method:

(1) rotamers cannot be involved in any steric overlaps with the fixed backbone;
(2) rotamers cannot be involved in any steric overlaps with rotamers from other residues.

Rotamers that overlap with the fixed backbone will be eliminated from their residue's finite domain. Two rotamers that overlap with each other cannot be part of the solution, and so one of the pair must be eliminated from its residue's finite domain. For example, the domain variables of the 6 residues in the peptide VKYQGS would be represented as VAL1, LYS2, TYR3, GLN4, GLY5, SER6. The CLP solver will try to find a rotamer, for each residue, that satisfies the constraints placed on the values that are in each variable's finite domain. The SICStus Prolog syntax (Carlsson and Ottosson, 1997) for this example is shown in Figure 2: CLP variables, finite domains and constraints are declared, and a predicate for solving them is defined.

Search strategy used by the CLP solver

The `labeling` predicate shown in Figure 2 has search options to control the order in which variables are selected and assigned a rotamer. These heuristics for enumerating domain variables alter the way in which the constraint solver carries out its search through the different rotamer combinations.

leftmost: the residues are selected in order of position from the N terminus.
min: the residue closest to N terminus with the smallest lower bound. i.e. with a constraint on its most probable rotamer.
max: the residue closest to N terminus with the greatest upper bound. i.e. with a constraint on its most improbable rotamer.
ff: the residue closest to N terminus with the least number of rotamers.
ffc: the residue closest to N terminus with the least number rotamers and with the most constraints suspended on it.

```
%
% Predicate to solve the constraints using
% the most constrained [ffc] heuristic.
%
solve_constraints :-
        constraints(ResiduePositions),
        labeling([ffc], ResiduePositions).


%
% SICStus Prolog CLP syntax for
% constraints on side-chain placement
%
constraints([VAL1, LYS2, TYR3, GLN4, GLY5, SER6]) :-

        % Finite domains for variables, e.g.
        % VAL1 has 3 rotamers, LYS2 has 81 rotamers
        %
        VAL1 in 1..3,
        LYS2 in 1..81,
        TYR3 in 1..4,
        GLN4 in 1..10,
        GLY5 in 1..1,
        SER6 in 1..3,

        % Clashes with the backbone, e.g.
        % VAL1 cannot be rotamer 2
        %
        VAL1 #\= 2,
        TYR3 #\= 3,
        TYR3 #\= 4,
        GLN4 #\= 1,
        GLN4 #\= 5,

        % Clashes between rotamers, e.g.
        % if VAL1 is rotamer 1 then TYR3 cannot be rotamer 1
        %
        VAL1 #= 1 #=> TYR3 #\= 1,
        VAL1 #= 1 #=> LYS2 #\= 1,
        VAL1 #= 1 #=> LYS2 #\= 2,
        SER6 #= 1 #=> LYS2 #\= 2,
        SER6 #= 1 #=> GLN4 #\= 2,
        SER6 #= 1 #=> GLN4 #\= 8,
        true.
```

Fig. 2. SICStus Prolog syntax for declaring CLP variables, finite domains and constraints.

Since in each finite domain the first value represents the most probable rotamer, the CLP solver has been set up to initially try satisfying the constraints with the most common rotamers. If the most common rotamer fails, the constraint solver will backtrack and try the second most common, then the third, and so on until one succeeds. If a solution is possible it usually takes CLP just a few seconds to find it.

2.2 Problems with a simple CLP approach

An unfortunate characteristic of this CLP method is that it either works, or it doesn't — a common reason for the constraint solver to fail is too many constraints placed on one or more side-chains. This means that the system is over-constrained and no side-chain models can be produced. Failure to find even a poor solution for an over-constrained system is a disadvantage of the CLP method for side-chain placement, since a poor solution with known weaknesses can still provide scientists with useful information about a protein's structure, and can serve as a starting point for further structural refinement.

To ensure that a solution to the constraints is possible, a simple heuristic was used to determine if two atoms were involved in steric overlaps: two atoms were said to be overlapping if the interatomic distance, i.e. the distance between the centres of the two atoms, was less than *ConDist*. ConDist is the minimum allowed interatomic distance constraint. It was assumed that higher values of ConDist would lead to tighter atomic packing constraints on the side-chains, with the expectation that the modelling predictions would be more accurate. The largest value of ConDist that produces a solution can vary greatly between proteins. Typically, small proteins, with less than 100 residues, can be modelled with a ConDist of about 2.4 Å. Larger proteins, with over 200 residues, can be modelled with a ConDist value of only 1.6 Å — a value that represents some very severe steric overlaps.

The severe steric overlaps present in the models created by the CLP method highlight some of the problems experienced by approximating continuous side-chain conformations by fixed rotamers. In side-chain modelling methods that use explicit energy functions such close contacts lead to very high van der Waals terms that approach infinity as the distance shrinks to zero. This has led some researchers to fix the van der Waals term to a certain value for small interatomic distances (Lee and Subbiah, 1991; Holm and Sander, 1992; Koehl and Delarue, 1994; Hwang and Liao, 1995). One method that does not suffer from this problem is that of Mendes et al. (1999). They use flexible rotamers to create thermodynamic expressions that are analogous to the terms in potential energy functions. These thermodynamic expressions consider a ensemble of conformations for each rotamer and thus avoid severe steric overlaps.

Even when this CLP method does work many side-chains are modelled without having had any constraints placed upon them. This is because when the ConDist is lowered to a value small enough to produce a solution, the constraints on many of the other residues will be so weak that they will have been poorly modelled. A more flexible and accurate method would allow tighter constraints to be placed on less densely packed residues while looser constraints would be placed on those residues liable to become over-constrained and cause the constraint solver to fail. Such a method is described in the following section.

2.3 An iterative implementation of the CLP method

One method of identifying variables that are likely to be over-constrained is to use *null values* (or *null rotamers*). In doing this, we add an extra value to the end of each variable's finite domain, after the least common rotamer, that corresponds to "no (real) value found". When this value is part of the solution it means that no rotamer can be placed for the corresponding residue. As the null rotamer has no physical representation there can be no constraints placed upon it. No matter how tight the constraints on a variable may be, there will always be a solution that contains the null rotamer. Thus, under very tightly constrained conditions, the residues in the core of the protein may be over-constrained and allocated null rotamers, whereas those under-constrained residues found towards the surface of the protein will be allocated real rotamers.

The CLP method of side-chain placement may be modified to make use of null rotamers. The basic idea is that ConDist is increased iteratively from zero to around 3.2 Å in steps of 0.2 Å. As ConDist is increased, residues become over-constrained and are allocated null rotamers. When a residue is allocated a null rotamer, the rotamer that was part of the solution solved under the previous iteration is chosen as the side-chain conformation for that residue. This side-chain is considered to be fixed and any steric overlaps it makes with the other rotamers are considered in the same way as overlaps to the backbone. As the constraints are tightened more and more side-chains become fixed until the iterations are stopped. An outline of the improved CLP side-chain placement algorithm is shown in Figure 3.

2.4 Rotamer libraries

Several rotamer libraries (Dunbrack and Karplus, 1993; Dunbrack and Cohen, 1997; Lovell et al., 2000; Tuffery et al., 1997) have been used, with some modifications having been made to the BBDEP library of Dunbrack and Cohen

```
find close inter-atomic distances between rotamers
ConDist = 0

while ConDist < 3.2 Angstroms

      turn off null rotamers
      automatically write CLP program for current value of ConDist
      try to solve constraints with CLP

      if CLP fails to find a solution

            turn on null rotamers
            automatically write CLP program for current value of ConDist
            solve constraints with CLP (success guaranteed)
            relace any null rotamers with rotamer found for that residue
                in the previous iteration

      else CLP found a solution

            store rotamers
            evaluate model
            ConDist := ConDist + 0.2

      end else

end while
```

Fig. 3. Pseudo-code description of the iterative CLP side-chain placement algorithm.

(1997). Lovell et al. (2000) observe that many of the very uncommon rotamers in the BBDEP library contain large internal clashes, and are unlikely to be seen in high resolution X-ray structures. We have culled the BBDEP library to remove the rotamers with extremely low probabilities: the rotamers of all side-chains with 3 or 4 rotatable bonds that have probabilities of less than 0.02 or 0.01, respectively, have been removed. Rotamers of Asp, Asn and Trp with probabilities of less than 0.05 have also been removed, and for the other side-chains a minimum probability of 0.10 was allowed.

Reducing the size of the BBDEP library has two advantages: first the size of the computational problem is reduced, and second, it is not uncommon for the accuracy of the predictions to increase as the constraint solver cannot backtrack through less common rotamers. After reducing the library to its most common rotamers, extra rotamers (with torsion angles differing by $\pm 10^o$) were added in an attempt to avoid some of the bad steric overlaps.

2.5 Evaluating side-chain predictions

The comparison of side-chain modelling methods is complicated by the different criteria used by authors to assess the accuracy of their predictions. Predicted side-chain conformations are commonly compared to the X-ray structures obtained from the Protein Data Bank (Bernstein et al., 1977) by calculating the RMSD of the side-chain atoms, or by comparing side-chain dihedral (χ) angles (as defined in IUPAC-IUB Commission on Biochemical Nomenclature (1970)).

One difference between authors using RMSD is whether or not Cβ atoms are included in the calculations (Shenkin et al., 1996); this is sometimes excluded since the position of the Cβ atom is determined by backbone geometry. Some authors include all side-chains, whereas others only include only side-chains placed by their modelling method. Also, results are sometimes presented separately for buried side-chains.

When comparing side-chain dihedral angles, $\chi 1$ is typically regarded as correct if it is within 40^o of the same angle in the X-ray structure (Dunbrack and Karplus, 1993; Hwang and Liao, 1995; Koehl and Delarue, 1994; Shenkin et al., 1996). Authors choose to evaluate the $\chi 2$ angle predictions in a variety of ways: the number of $\chi 2$ angles within 40^o independently of $\chi 1$, or the number of $\chi 2$ angles within 40^o given that the side-chain's $\chi 1$ is correct, or the number of side-chains with both $\chi 1$ and $\chi 2$ within 40^o.

In this paper, unless stated otherwise, we assess the accuracy of the modelling predictions using RMSD calculations over all heavy side-chain atoms, including Cβ.

3 Results and discussion

In this section we compare results obtained using different CLP enumeration heuristics. Next, we compare the accuracy of our CLP method with other side-chain placement algorithms. Finally, we compare results obtained using the CLP method to model a set of proteins using different rotamer libraries.

The value chosen for the upper bound of the ConDist parameter will affect the accuracy of the model. To help determine a suitable maximum value for ConDist to be used when modelling unknown structures, we modelled several known protein structures with ConDist iterating from 0.2 Å to 3.2 Å in steps of 0.2 Å. Figure 4 shows that, in each case, the most accurate model is obtained when ConDist has a value between 2 Å and 3 Å.

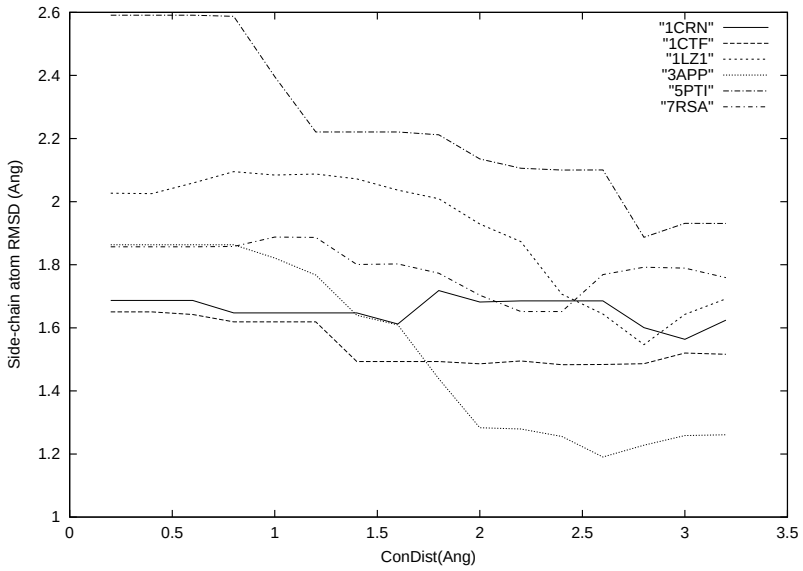

Fig. 4. Side-chain RMSD of models of six proteins listed in Table 1 built using the CLP method with the `ffc` enumeration heuristic, and ConDist parameter increasing from 0.2 Åto 3.2 Å. The rotamer library used was the modified BBDEP library described in Section 2.4.

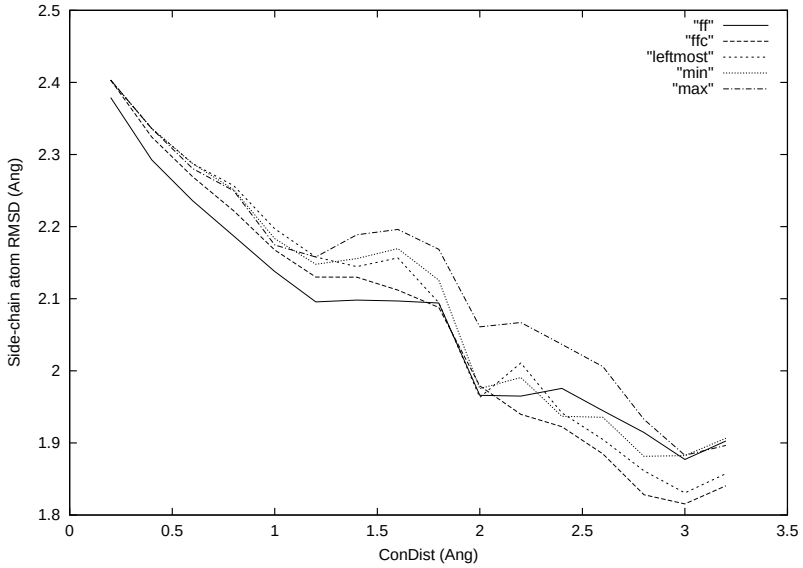

Fig. 5. Average side-chain RMSD of models of six proteins listed in Table 1 built using the CLP method with five different enumeration heuristics, and ConDist parameter increasing from 0.2 Å to 3.2 Å. The rotamer library used was that of Tuffery et al. (1997).

Figure 5 shows the accuracy of the modelling method when using different CLP variable enumeration heuristics. The `ffc` search option is the most successful, with `max`, `min` and `ff` all minimising the RMSD at a very similar values. The results presented in Figure 5 were obtained using the rotamer library of Tuffery et al. (1997). Tests with other rotamer libraries give similar results.

Table 1
Comparison of the CLP modelling method (using the `ffc` enumeration heuristic and the modified BBDEP library described in Section 2.4) at ConDist=2.80 Å with other published methods. For each method the scores listed are the RMSD (Å) for all heavy side-chain atoms in each protein, including Cβ.

Protein	CLP method	Laughton (1994)	Lee & Subbiah (1991)	Hwang & Liao (1995)	Levitt (1992)	Hom & Sander (1992)	Tuffery et al. (1991)
1CRN	1.60	2.00	1.65	1.34	1.57	-	1.48
5PTI	1.89	2.31	2.61	1.80	2.43	1.9	-
1CTF	1.49	1.95	1.86	1.40	1.37	1.7	1.44
7RSA	1.79	-	2.01	1.73	2.00	1.8	-
1LZ1	1.55	2.04	1.62	1.61	1.59	1.6	1.82
3APP	1.23	1.64	1.12	1.24	1.42	-	-
Average	1.59	1.99	1.81	1.52	1.73	1.75	1.58

It has been observed that the accuracy of models can vary greatly from protein to protein (Lee and Subbiah, 1991; Tuffery et al., 1991; Dunbrack and Karplus, 1993). In Table 1 we compare six proteins that have been modelled by several authors. For these proteins, the CLP method yields RMSD predictions that compare favourably with the other methods. The method of Hwang and Liao (1995), using neural networks with simulated annealing, gives the best average predictions, however it should be noted that the method was tested on a subset of proteins taken from the set used to train the neural networks.

Table 2 compares different rotamer libraries used with the CLP side-chain modelling method and the `ffc` enumeration heuristic. Results are presented for models of forty-three proteins. These proteins have been collated from those modelled by Koehl and Delarue (1994), Shenkin et al. (1996) and Holm and Sander (1992). Each protein has a resolution less than or equal to 2.0 Å, no side-chain atoms are missing from any residues, and, with the exception of 7RSA and 5PTI, only one conformation is given for each side-chain. In the case of 7RSA and 5PTI the models were only compared to the first out of the alternative conformations. Figure 6 shows the average RMSD of all heavy side-chain atoms between the model structures and the X-ray structures, and Figure 7 shows the percentages of $\chi 1$ angles that are modelled correctly.

Although the modified BBDEP library gives the best results overall, it gives the worst RMSD prediction for the protein 1UBQ. In this case, increasing the tightness of the constraints by increasing ConDist results in the RMSD remaining generally constant, while the percentage of correct $\chi 1$ angles increases by almost 9% to a maximum of 72% with a ConDist value of 3.0 Å(not tabulated)

Table 2
Comparison of different rotamer libraries used with the CLP side-chain modelling method and the `ffc` enumeration heuristic. For each protein we show the RMSD values (Å) and the percentage of correct $\chi 1$ angles. The value of ConDist giving the lowest average RMSD for all 43 proteins is shown for each rotamer library.

PBD Code	Resolution	No. of Residues	BBDEP Modified	BBDEP	BBIND	Lovell Library	Tuffery Library
ConDist			2.8	2.4	2.8	2.8	3.0
1BP2	1.7	123	1.97 68	1.98 67	2.02 66	2.28 57	2.03 64
1CA2	2.0	256	2.01 79	1.71 80	1.89 67	2.00 62	1.74 70
1CCR	1.5	111	1.82 75	2.01 75	2.07 68	1.83 63	1.71 68
1CRN	1.5	46	1.60 81	1.54 86	1.66 73	1.93 65	1.68 73
1CTF	1.7	68	1.49 77	1.50 72	1.78 66	1.99 53	1.56 74
1HOE	2.0	74	1.65 73	1.63 73	2.29 63	2.60 50	2.18 62
1LZ1	1.5	130	1.55 83	1.78 81	2.01 71	2.12 65	2.05 70
1MBA	1.6	146	1.91 78	1.98 75	1.69 81	2.10 64	2.10 75
1PAZ	1.55	120	1.64 81	1.57 82	1.77 74	1.98 64	1.70 78
1PPD	2.0	212	1.80 84	2.05 79	2.08 73	2.13 68	2.10 72
1PPT	1.37	36	1.88 88	1.88 88	1.45 82	1.88 64	2.08 82
1R69	2.0	63	2.18 74	2.28 69	2.59 70	2.57 54	2.00 74
1RDG	1.4	52	1.58 77	1.46 84	1.73 67	1.75 63	1.53 70
1UBQ	1.8	76	2.11 68	2.09 65	2.03 60	2.03 40	1.86 57
256B	1.4	106	1.82 81	2.04 77	2.06 77	2.03 69	2.07 77
2CAB	2.0	256	1.76 81	1.80 82	1.86 70	2.08 60	1.92 68
2CDV	1.8	107	1.57 82	1.79 82	1.74 73	2.02 62	2.17 66
2CGA	1.8	245	1.76 79	1.68 77	2.33 63	2.18 58	2.00 69
2CI2	2.0	65	1.94 73	2.11 68	2.10 64	2.27 44	1.83 66
2CTS	2.0	437	2.09 70	2.08 71	2.09 68	2.21 60	2.14 69
2I1B	2.0	153	1.78 72	1.86 67	1.77 66	1.81 66	1.79 69
2LYZ	2.0	129	1.61 73	1.77 70	1.88 68	1.93 57	1.80 60
2LZT	1.97	129	1.69 82	2.00 79	1.70 76	1.82 67	1.78 73
2MLT	2.0	26	2.20 62	2.20 62	2.35 57	1.49 62	1.91 52
2OVO	1.5	56	1.94 77	1.88 79	2.19 65	2.08 63	1.68 71
2RHE	1.6	114	1.90 76	1.79 77	2.06 70	2.22 45	2.02 68
2UTG	1.64	70	1.94 66	2.13 65	1.93 63	2.21 55	1.93 66
3APP	1.8	323	1.23 85	1.25 85	1.59 71	1.75 61	1.62 70
3GRS	1.54	461	1.84 78	1.80 77	1.99 73	1.96 62	2.02 69
3LZM	1.7	164	1.92 82	1.88 83	2.22 74	2.43 65	2.03 77
4HHB	1.74	141	1.91 70	1.73 77	1.96 68	1.98 58	2.07 70
4LYZ	2.0	129	1.74 73	1.82 69	1.94 65	1.97 60	1.99 61
4PEP	1.8	326	1.35 75	1.44 73	1.83 63	1.86 59	1.90 63
4PTI	1.50	58	1.91 89	2.18 85	1.96 83	2.24 72	2.12 83
4TNC	2.0	160	2.08 63	2.07 63	2.01 60	2.08 55	2.06 57
5CYT	1.5	103	1.96 77	1.80 76	2.16 67	1.98 65	2.00 66
5PCY	1.80	99	1.41 76	1.61 73	1.54 68	1.83 61	1.61 67
5PTI	1.0	58	1.89 76	2.10 78	2.22 67	2.40 59	2.04 78
5RXN	1.20	54	1.60 79	1.49 81	1.66 69	1.75 67	1.77 67
6LDH	2.0	329	1.82 70	1.79 70	1.92 65	1.92 61	2.02 62
6LYZ	2.0	129	1.90 72	1.86 70	1.89 68	1.96 56	1.92 64
7RSA	1.26	124	1.79 71	1.65 74	2.19 57	2.08 57	1.95 66
8DFR	1.7	186	1.84 77	1.72 79	1.91 70	2.02 57	1.98 67
Average			1.80 76	1.83 75	1.95 69	2.04 60	1.92 69

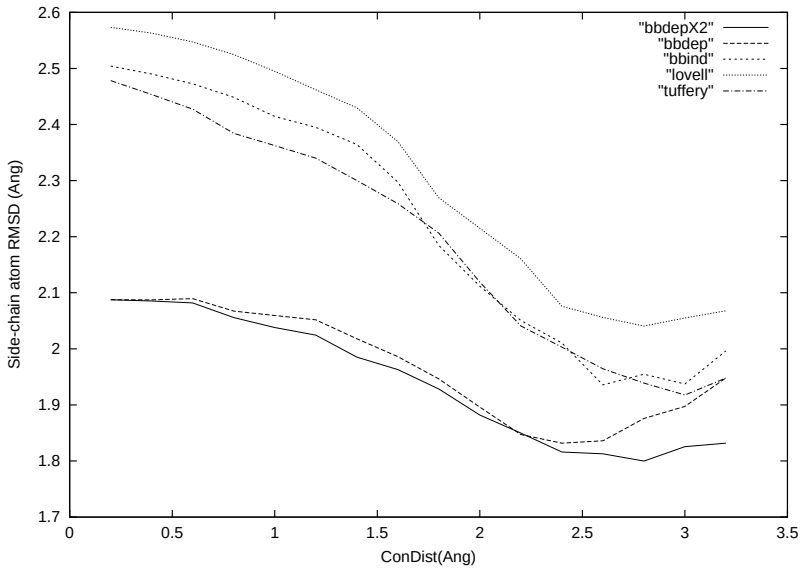

Fig. 6. The average RMSD of all heavy side-chain atoms between the forty-three X-ray structures listed in Table 2 and models built using the CLP method with the `ffc` enumeration heuristic and different rotamer libraries (Dunbrack and Karplus, 1993; Dunbrack and Cohen, 1997; Tuffery et al., 1997; Lovell et al., 2000). The modified BBDEP library, described in Section 2.4, is labelled "bbdepX2".

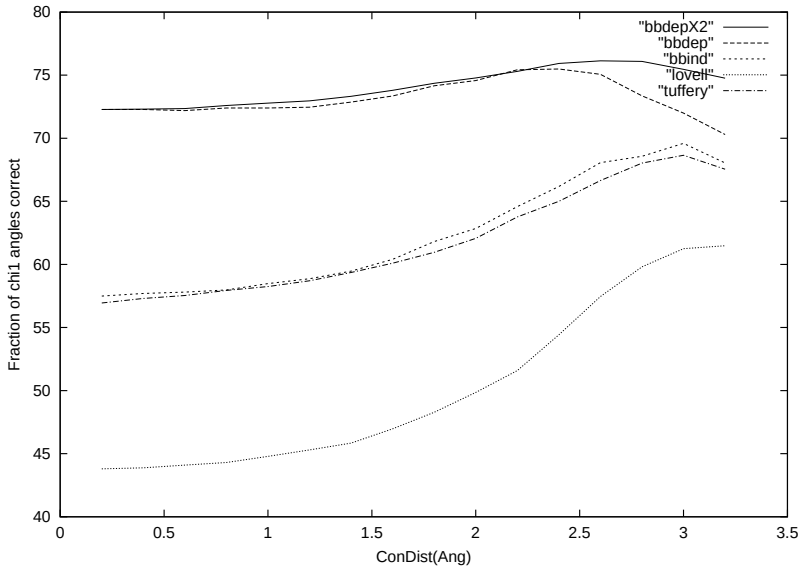

Fig. 7. The average percentage of modelled side-chains with $\chi 1$ angles within 40^o of those in the forty-three X-ray structures listed in Table 2. The models were built using the CLP method with the `ffc` enumeration heuristic and different rotamer libraries.

— the highest of all the libraries used.

Figure 7 and Table 2 show that the CLP method used with the modified BBDEP library predicts 76% of $\chi 1$ angles correctly at ConDist=2.8Å. Koehl

Table 3
The RMSD values (Å) for all heavy side-chain atoms, and the percentage of $\chi 1$ angles correct, are shown for each residue when modelled by the CLP side-chain modelling method using the `ffc` enumeration heuristic and different rotamer libraries.

Residue	Number	BBDEP Modified	BBDEP	BBIND	Tuffery Library	Lovell Library
ARG	226	2.58 65	2.67 65	2.79 62	2.75 60	2.75 63
ASN	333	1.44 76	1.50 73	1.36 70	1.48 69	1.45 66
ASP	379	1.50 74	1.41 78	1.64 61	1.67 59	2.17 30
CYS	170	0.83 64	0.89 60	0.98 54	0.98 54	0.98 54
GLN	228	1.70 68	1.73 67	1.91 67	1.79 64	1.74 65
GLU	306	1.78 62	1.78 64	1.92 57	1.86 61	1.89 58
HIS	127	1.61 81	1.68 79	1.72 79	1.58 80	1.58 82
ILE	303	0.58 88	0.67 85	0.87 76	0.77 80	0.85 78
LEU	490	0.85 79	0.91 76	0.88 79	0.96 74	1.12 66
LYS	432	1.85 67	1.86 67	1.96 63	1.92 64	1.96 64
MET	126	1.49 74	1.59 64	1.58 70	1.45 70	1.60 69
PHE	222	1.45 89	1.44 89	1.59 86	1.61 83	1.70 84
PRO	275	0.40 87	0.33 89	0.41 76	0.44 71	0.48 70
SER	483	0.76 58	0.69 61	0.87 51	0.89 49	1.12 34
THR	362	0.43 85	0.48 81	0.90 59	0.84 61	1.22 41
TRP	103	1.58 87	1.46 89	1.51 88	1.51 88	1.89 83
TYR	219	1.38 94	1.42 93	1.59 89	1.67 87	1.58 89
VAL	437	0.37 87	0.40 86	0.59 75	0.54 79	0.93 58

and Delarue (1994) reported 72%, Shenkin et al. (1996) 74%, and Bower et al. (1997) 77%. The flexible rotamer method of Mendes et al. (1999), including an energy refinement stage, gives average predictions of 85.8% correct $\chi 1$ angle with a smaller set of twenty high quality structures. They claim that to be 2.4% better than the method of Bower et al. (1997) and 7.3% better than the method of Koehl and Delarue (1994) with that same test set.

In Table 3 we show the modelling predictions for all residues in the proteins listed in Table 2. The range of results obtained for threonine (from 85% to 41% of $\chi 1$ angles correct) indicates that the different probability values given for the rotamers in the libraries used can have a very significant effect on the predictions. Large side-chains with rings are often heavily constrained, and

are modelled well using each of the rotamer libraries. In our present implementation we do not use constraints to drive Cys residues towards having favoured disulphide bond geometry. We believe that adding such constraints should improve the accuracy of modelled Cys residues, and we are currently investigating this.

To conclude, we have presented a new side-chain placement method based upon the artificial intelligence technique of constraint logic programming. Heuristics describing atomic packing interactions are used with a backbone dependent rotamer library to model side-chains by iteratively packing them into tighter and tighter spaces. The method may be easily modified to be used in homology modelling, or to model only a few side-chains at at time.

Acknowledgements

M.T.S. is supported by a BBSRC CASE award with Biovation Ltd.

References

Bernstein, F. C., Koetzle, T. F., Williams, G. J. B., Mayer, E. F., Bruce, M. D., Rodgers, J. R., Kennard, O., Shimanouchi, T., Tasumi, M., 1977. The Protein Data Bank: a Computer-Based Archival File for Macromolecular Structures. J. Mol. Biol. 112, 535–542.

Bower, M. J., Cohen, F. E., Dunbrack, R. L., 1997. Prediction of protein side-chain rotamers from a backbone-dependent rotamer library: A new homology modeling tool. J. Mol. Biol. 267, 1268–1282.

Carlsson, M., Ottosson, G.and Carlson, B., 1997. An open-ended finite domain constraint solver. Proc. Programming Languages: Implementations, Logics, and Programs .

Chinea, G., Padron, G., Hooft, R. W. W., Sander, C., Vriend, G., 1995. The use of position specific rotamers in model building by homology. Prot. Struct. Funct. Genet. 23 (415–421).

Desmet, J., De Maeyer, M., Hazes, B., Lasters, I., 1992. The dead-end elimination theorem and its use in protein side-chain positioning. Nature 356, 539–542.

Dunbrack, R. L., Cohen, F. E., 1997. Bayesian statistical analysis of side-chain rotamer preferences. Protein Science 6, 1661–1681.

Dunbrack, R. L., Karplus, M., 1993. Backbone-dependent rotamer library for proteins: application to side-chain prediction. J. Mol. Biol. 230 (543–574).

Eisenmenger, F., Argos, P., Abagyan, R., 1993. A method to configure protein

side-chains form the main-chain trace in homology modelling. J. Mol. Biol. 231, 849–860.

Fruhwirth, T., Herold, A., Kuchenhoff, V., Le Provost, T., Pierre, L., Monfroy, E., Wallace, M., 1993. Constraint logic programming: An informal introduction. Tech. rep., European Computer-Industry Research Centre.

Gordon, D. B., Mayo, S. L., 1999. Branch-and-terminate: a combinatorial optimization algorithm for protein design. Structure 7 (1089–1098).

Guex, N., Peitsch, M. C., 1997. SWISS-MODEL and the Swiss-PdbViewer: An environment for comparative protein modeling. Electrophoresis 18, 2714–2723.

Heringa, J., Argos, P., 1999. Strain in protein structures as viewed through nonrotameric side-chain: I. their positions and interaction. Prot. Struct. Funct. Genet. 37, 30–43.

Holm, L., Sander, C., 1991. Database Algorithm for Generating Protein backbone and Side-chain Co-ordinates from a Cα Trace. J. Mol. Biol. 218, 183–194.

Holm, L., Sander, C., 1992. Fast and Simple Monte Carlo Algorithm for Side Chain Optimization in Proteins: Application to Model building by Homology. Prot. Struct. Funct. Genet. 14, 213–233.

Hwang, J. K., Liao, W. F., 1995. Side-chain prediction by neural networks and simulated annealing optimization. Prot. Eng. 8, 363–370.

IUPAC-IUB Commission on Biochemical Nomenclature, 1970. Abbreviations and Symbols for the Description of the Conformation of Polypeptide Chains. Eur. J. Biochem. 17, 193–201.

Koehl, P., Delarue, M., 1994. Application of a self-consistent mean field theory to predict protein side-chains conformation and estimate their conformational entropy. J. Mol. Biol. 239, 249–275.

Kono, H., Doi, J., 1996. A new method for side-chain conformation prediction using a hopfield network and reproduced rotamers. J. Comp. Chem. 17, 1667–1683.

Lasters, I., De Maeyer, M., Desmet, J., 1995. Enhanced dead-end elimination in the search for the global minimum energy conformation of a collection of protein side chains. Prot. Eng. 8, 815–822.

Laughton, C. A., 1994. Prediction of Protein Side-chain Conformations from Local Three-dimensional Homology Relationships. J. Mol. Biol. 235, 1088–1097.

Lee, C., Subbiah, S., 1991. Prediction of protein side-chain conformation by packing optimization. J. Mol. Biol. 217, 373–388.

Levitt, M., 1992. Accurate modeling of protein conformation by automatic segment matching. J. Mol. Biol. 226, 507–533.

Lovell, S. C., Word, M., Richardson, J. S., Richardson, D. C., 2000. The Penultimate Rotamer Library. Prot. Struct. Funct. Genet. 40, 389–408.

Mendes, J., Baptista, A., Carrondo, M., Soares, C., 1999. Improved modeling of side-chains in proteins with rotamer-based methods: A flexible rotamer model. Prot. Struct. Funct. Genet. 37, 530–543.

Ponder, J. W., Richards, F. M., 1987. Tertiary templates for proteins. J. Mol. Biol. 193 (775–791).

Schrauber, H., 1993. Rotamers: to be or not to be? J. Mol. Biol. 230, 592–612.

Shenkin, P., Farid, H., Fetrow, J., 1996. Prediction and evaluation of side-chain conformations for protein backbone structures. Prot. Struct. Funct. Genet. 26, 323–352.

Taylor, W., 1992. New paths from dead ends. Nature 356, 748–479.

Tuffery, P., Etchebest, C., Hazout, S., 1997. Prediction of protein side chain conformations: a study on the influence of backbone accuracy on conformation stability in the rotamer space. Prot. Eng. 10, 361–372.

Tuffery, P., Etchebest, C., Hazout, S., Lavery, R., 1991. A new approach to the rapid determination of protein side-chain conformations. J. Biomol. Struct. Dynam. 8, 1267–1289.

Vasquez, M., 1996. Modeling side-chain conformations. Curr. Opin. Struct. Biol. 6, 217–221.

Wallace, M., 1995. Constraint programming. Tech. rep., Imperial College.

Wilson, C., Gregoret, L. M., Agard, D. A., 1993. Modeling side-chain conformation for homologous proteins using an energy-based rotamer search. J. Mol. Biol. 229, 996–1006.