Acta Crystallographica Section D
Biological Crystallography
ISSN 0907-4449

ARP/*wARP*'s model-building algorithms. I. The main chain

Richard J. Morris,[a]† Anastassis Perrakis[b] and Victor S. Lamzin[a]*

[a]EMBL Hamburg c/o DESY, Notkestrasse 85, D-22603 Hamburg, Germany, and [b]NKI, Department of Molecular Carcinogenesis, Plesmanlaan 121 1066 CX, Amsterdam, The Netherlands

† Present address: Global Phasing Ltd, Sheraton House, Castle Park, Cambridge CB3 0AX, England.

Correspondence e-mail: victor@embl-hamburg.de

Received 14 December 2001
Accepted 25 March 2002

Algorithms underlying the automatic model-building functionality of the *ARP*/*wARP* software suite are presented. Finding the most likely set of Cα atoms from a given set of atoms is formulated as a constrained integer programming problem. The objective function is a density-weighted score for the match between observed and expected chain conformation. Graph-search algorithms are presented that find solutions to this problem in an efficient manner.

1. Introduction

Once the phase problem has been solved and an electron-density distribution has been computed from the set of phase estimates and structure-factor amplitudes, the macromolecular crystallographer is faced with the problem of interpreting a map that is not self-explanatory. To reduce this three-dimensional continuous electron-density distribution to the desired set of atoms with type assignments and bonds that constitute a model is a cumbersome task requiring much expertise and time.

The *ARP*/*wARP* software suite (Lamzin *et al.*, 2001) is a package of utilities aimed at delivering an essentially complete macromolecular atomic model from a given electron-density map. Given data extending to at least 2.3 Å and *reasonable* initial phase estimates, *ARP*/*wARP* is capable of building a complete and refined protein model within hours.

1.1. Main-chain tracing

Protein models can be represented by a set of long single non-branching polypeptide chains consisting of a highly flexible arrangement of repetitive identical units (the planar peptide unit of the main chain or backbone) and a number of short well defined units attached to it (the side chains). The full main chain itself can be determined to a high degree of accuracy by the positions of the Cα atoms alone (Esnouf, 1997). As the side-chain placement problem is well determined given a correct main chain and the main chain well defined by the Cα atoms, protein model building is often seen as the problem of locating the Cα positions. So, traditionally one of the first steps in the interpretation of a protein electron-density map is the identification of the protein's main chain.

The first papers on automatic map interpretation and pattern recognition in a crystallographic context appeared in the early 1970s, most notably by Greer (1974) and Koch (1974). Greer's skeletonization approach was highly successful and is still the most common aid for tracing the main chain in a given electron-density map.

Recent developments in the area of map interpretation have been mainly related to the core-tracing algorithm developed by Swanson (1994) and the extensive use of databases (Jones & Thirup, 1986; Kleywegt & Jones, 1997*a*; Finzel, 1997) also in combination with topological approaches and artificial intelligence techniques (Leherte *et al.*, 1994; Fortier *et al.*, 1997). Bricogne (1997*a*,*b*) has formulated a general framework for knowledge-based structure solution and map interpretation based on a Bayesian molecular-replacement approach. A similar scheme has been implemented by Kleywegt & Jones (1997*b*) in their template-convolution routine *ESSENS*, by Cowtan (1998) in *FFFEAR* using Fast-Fourier methods and by Terwilliger (2001). Progress has been made recently with the development of the database-assisted template-matching program *DADI* (Diller, Pohl *et al.*, 1999; Diller, Redinbo *et al.*, 1999), the neural network approach to pattern recognition in *TEXAL* (Holton *et al.*, 2000), automated fragment searching in *MAID* (Levitt, 2001) and conformation matching in *CONFMATCH* (Wang, 2000). Many of these latter programs use a wide variety and combination of optimization, recognition and searching techniques that cannot be given full justice in this brief summary.

Although very sophisticated implementations exist for assisting main-chain tracing and building, for instance the popular packages *QUANTA* (Oldfield, 1996), *O* (Jones & Kjeldgaard, 1996), *XtalView* (McRee, 1992), *TURBO-FRODO* (Roussel & Cambillau, 1991) and others, the current state of automation is still rather modest, relying on the user to actually make all the relevant decisions (such as the density level at which connectivity is sought, to choose the best skeleton *etc.*). For X-ray data extending to a resolution of about 2.3 Å and higher, the *warpNtrace* module of *ARP/wARP* (Perrakis *et al.*, 1999) offers capabilities that come close to full automation that allow a refined model, complete to within a few residues, to be constructed. Here, we present in detail the management system behind the main-chain tracing automation. The full *ARP/wARP* model-building flowchart is depicted in Fig. 1.

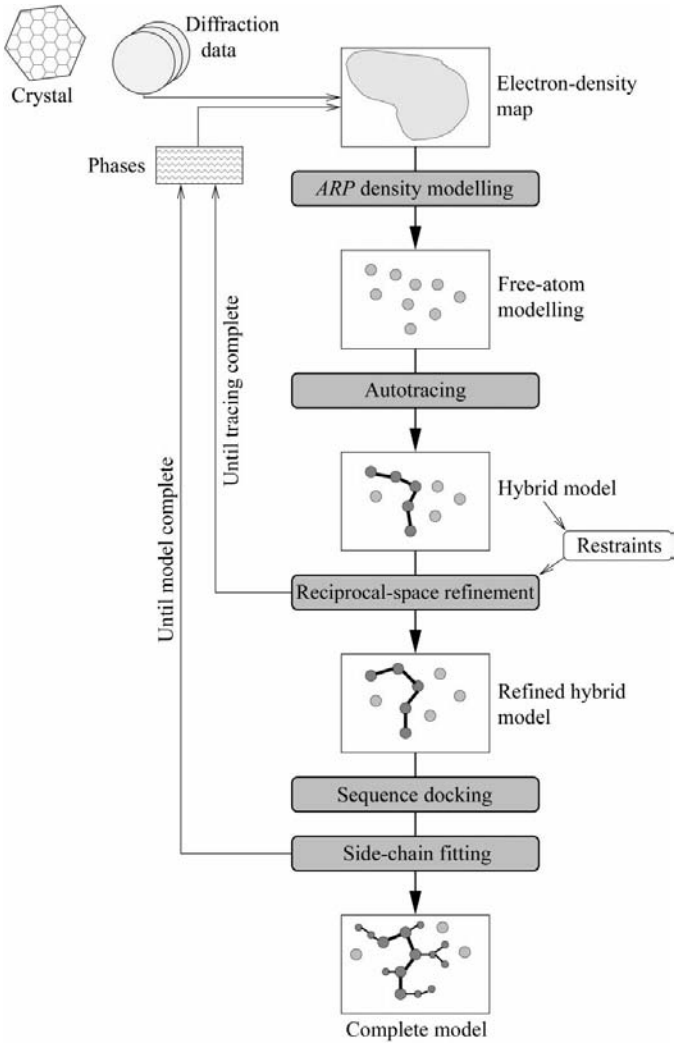

Figure 1
The *ARP/wARP* model-building (*warpNtrace*) flowchart.

1.2. Refinement, overfitting and restraints

The observation-to-parameter ratio is a key factor in any optimization of model parameters against experimental data. For the optimization of a crystallographic model, diffraction data extending to at least atomic resolution (about 1.2 Å) are necessary to provide a sufficient number of reflections and thus an adequate observation-to-parameter ratio to fully justify an atomic model. Also insisting on an atomic model for lower resolution data implies a significant drop in the observation-to-parameter ratio and an increase in noise fitting arising from over-parametrization. The degree of overfitting can be monitored by the use of cross-validation techniques such as the free *R* factor (Brünger, 1992). For refinement of a full atomic protein model against non-atomic diffraction data to proceed smoothly without significant overfitting, restraints must be added to correlate the model parameters (or to artificially add observations). The introduction of additional observations in the form of stereochemical restraints (Konnert & Hendrickson, 1980) as a means of decreasing the chances of overfitting has been demonstrated on a theoretical basis by Bacchi *et al.* (1996) and Tickle *et al.* (1998), with the use of $R_{\rm free}$ factor to *R* factor ratio expectation values. However, to keep model parameters within sensible limits of prior stereochemical knowledge, a model with the correct atom-type assignments and their connectivity must already exist. It should be noted, however, that recent developments (Scheres & Gros, 2001) challenge this view by the proposed 'conditional optimization' that should allow the application of restraints to a collection of unlabelled atoms.

1.3. *ARP* and free atoms

The automated refinement procedure *ARP* (Lamzin & Wilson, 1993) may be viewed as a density-modelling program. *ARP* attempts to reproduce the distribution of a given electron-density distribution by the placement of atoms. Since the atom type cannot be assigned based solely on a density distribution calculated from data extending to less than atomic

resolution, *ARP* treats each newly placed atom as a dummy O atom (a *water*). The atomic parameters are then adjusted to optimize the match with the given density by refinement against the set of structure-factor amplitudes. Given an electron-density map, *ARP* will flood it with water. The model so created by *ARP* is not a conventional chemical model with atom-type assignments and a list of connections, but just a set of free atoms that reproduce the density well. Given no further information, this set of free atoms represents one of many valid atomic interpretations of the given density. The requirement that the model should also make chemical sense in terms of prior knowledge about atoms, bonds, bond distances, bond angles, amino-acid properties, protein secondary structure, protein folds *etc.* allows the criteria for judging these free-atoms models to be made considerably more stringent than those used by *ARP* to actually build this atomic representation of density features. As the true atom type is unknown, the generation of stereochemical restraints is not possible and the refinement is carried out in unrestrained mode (free from restrictions): atoms placed by *ARP* are therefore often referred to as *free atoms*. Computationally they are not free but relatively cheap: *ARP* density modelling with free atoms (density analysis, atom placement and positional refinement) may require workstation CPU times of the order of several seconds to minutes. Unrestrained refinement of the atoms placed by *ARP* will, however, suffer increasingly from the danger of overfitting with decreasing resolution.

2. From free atoms to a protein model

The ability of *ARP* to accurately place atoms has been demonstrated in many applications. Atoms placed by *ARP* are typically within 0.5 Å of the corresponding position in the final structure (see especially Figs. 3 and 15 of Lamzin & Wilson, 1997), but obviously depending on phase estimates, resolution limits and overall quality of the data. It therefore seems reasonable to assume that in a free-atoms representation of a given density map many of the atoms will be close to corresponding positions in the final structure. The model-building algorithms have the task of identifying these atoms, assigning the atom type and establishing their connectivity. Once atom type and bonds have been determined, stereochemical restraints can be derived and applied in restrained refinement of the current model. This increases the observation-to-parameters ratio, reduces overfitting, enhances refinement and provides better phase estimates. The set of built protein fragments with the remaining free atoms we refer to as a *hybrid model*. Iterating this map interpretation, model building and refinement cycle will ultimately result in the best possible phase estimates and a complete protein model. The power of *ARP*/*wARP* lies in this unification of model building and refinement into one big phase-refinement cycle and the iteration over such cycles (see Fig. 1).

In the following, the algorithms for picking the most probable protein main-chain atoms from a given set of candidate atom positions will be described. This is a key step in the assignment of atom types.

2.1. Formulation of the problem

Given a set S of N free atoms with coordinates $x_i = (x_i, y_i, z_i)$ and $S = \{x_i | i = 1, \ldots, N\}$, the task is to find a subset of positions, denoted by $M \subseteq S$, such that the stereochemical parameters calculated from the interatomic vectors of this set agrees best with the prior knowledge of protein stereochemistry

$$M = \text{argmax}_{s \in S} P[G(s)]. \quad (1)$$

$G(s)$ is the list of derived geometrical parameters of the chosen set, s, and P is the probability of this set of parameters being observed. This probability may be approximated by the frequencies of various geometrical parameters derived from analyses of known protein structures and the expected number of residues. The similarity of this approach to maximum-likelihood refinement should be noted and indeed model building and refinement are treated as one unified phase-refinement procedure within the framework of *ARP*/*wARP*. For the model-building formalism the argument in the above equation is the set of positions, *i.e.* which ones to choose, whereas for refinement this choice has already been made and the arguments are the atomic parameters of this set.

2.2. α-Carbon (Cα) parametrization

We reformulate the task of finding the best subset of atoms that looks like a protein as trying to identify which atoms will be used as Cα atoms and their (directed) connections to other chosen Cα atoms. Every Cα atom should have at least one other candidate Cα atom in its neighbourhood. This other Cα atom should be approximately 3.8 Å away and can be connected either in the form $-C(=O)-N-C\alpha$ or $-N-C(=O)-C\alpha$ to the original Cα atom; this directionality is denoted as forward (outgoing) and backward (incoming), respectively. The methods used by *ARP*/*wARP* to search for possible single-peptide units are explained in detail in Lamzin & Wilson (1997). In the final structure, every Cα should have at most one incoming and one outgoing connection. Introducing a connection variable, c_{ij}, to represent a forward connection between free-atom number i and free-atom number j ($c_{ij} = 1$ if the connection is present; $c_{ij} = 0$ otherwise) allows the task of finding the most protein-like set of atoms to be formulated as a constrained integer programming problem,

$$\text{maximize } P[G(s)] = P[G(\{c_{ij} | i, j \in S\})], \quad (2)$$

subject to $c_{ij} \in \{0, 1\}$ for all free atoms i and j and $\sum_{ij} c_{ij}$ should be as large as possible but less than the total number of residues. The list of possible connections is generated by distance checks between all atoms followed by peptide-plane density analysis (Lamzin & Wilson, 1997; Perrakis *et al.*, 1999). In brief, if two free atoms are approximately 3.8 Å apart and the density between them can provide values at the atomic centres of the remaining peptide plane atoms above a chosen threshold, then these two free atoms are marked as being a putative pair of Cα atoms. Two pairs of Cα atom units that have one position in common are flagged as being potentially connected (see Fig. 2). A list of connections (a so-called

adjacency list) can thus be obtained. This is the set of variables c_{ij} that must undergo optimization in the above sense.

2.3. The need for approximations

The only strategy that is guaranteed to always give the optimal solution to a general integer programming problem is the brute-force approach of trying every possible solution, ranking them and selecting the best. If the initial electron-density map were of such high quality that every placed free atom that might be a Cα atom has only one incoming and one outgoing connection, this brute-force approach would be fast, effective and reliable. However, even for a refined model, multiple potential connections may occur if only the positional information is used. Searching for distances of 3.8 ± 1.0 Å in a refined model would give of the order of a few hundred (depending on protein shape and solvent content) of the number of correct directed C$\alpha-$Cα connections. For electron-density maps calculated from initial phase estimates, the strict density criteria used by *ARP* must be relaxed, causing many atoms to be placed with less accuracy and also a number of false placements (with respect to the final model). The result is that many candidate Cα atoms have many more than one possible incoming and outgoing connection. We assume that the number of outgoing (and incoming) connections, n, is distributed according to a probability mass function $P(n)$. The distribution $P(n)$ depends on phase quality, resolution and the acceptance criteria within *ARP*/*wARP* for peptide planes. In general, the probability for higher n will increase with decreasing map quality. The probability of finding a chain of length L is given by the product of $(L-1)$ probabilities of n being greater than zero multiplied by the probability of n being equal to zero (the chain stops after L connections if there are no further possible connections),

$$P(L) = P(n \neq 0)^{L-1} P(n = 0). \qquad (3)$$

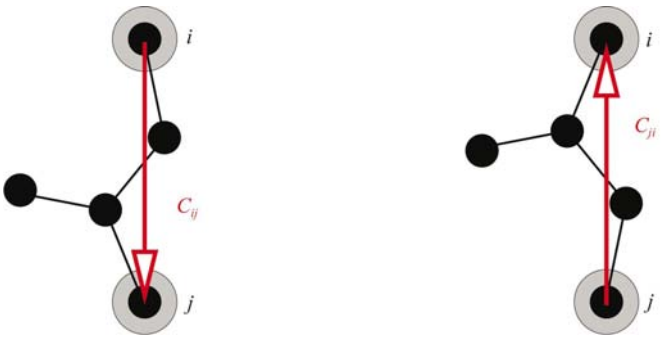

Figure 2
Determination of possible Cα atom connections. If two atoms, *i* and *j*, are approximately 3.8 Å apart, then a peptide plane is unit placed between these atoms and rotated about the *i*–*j* axis. If the interpolated density values at the atomic positions for the best orientation are above a given density threshold, the atoms *i* and *j* are flagged as being possible Cα atoms with a directed connection from *j* to *i*. The same procedure is repeated for the other peptide-plane direction. Both directions can be accepted at this stage.

The expected number of chains that will need checking in order to find the best one, can be expressed as

$$E(L) = \sum_L P(L) \prod_{k \leq L} \sum_n nP(n) = \sum_L P(L) \langle n \rangle^L, \qquad (4)$$

where $\langle n \rangle$ is the average number of branch points per node. The summation over L only need be taken to the maximum number of residues. For this sum not to diverge, the probabilities of chain length L must approach zero as fast as $\langle n \rangle_L$ tends towards infinity. In practical applications, of the order of 10% of all found Cα atoms have only incoming connections [$P(n \neq 0) \simeq 0.1$, $P(n = 0) \simeq 0.9$] and the average number of branch points typically exceeds two. This soon leads to what is known as a combinatorial explosion and enumerating all possible chains becomes intractable.

2.4. Divide and conquer

The optimization problem described above has the disadvantage that each putative chain has to be determined in full. The list of all geometrical parameters in each chain must be compared with some general high-dimensional distribution of expected parameters marginalized to those parameters found in each chain. For the development of efficient solution strategies, it is necessary to be able to take decisions based on evaluations at a more local level.

Each putative chain is divided into smaller structural units and the total chain score built from these unit evaluations,

$$\begin{aligned} \text{maximize } P[G(s)] &= P[G(\{c_{ij} | i, j \in S\})] \\ &\simeq \textstyle\prod_u \varphi_u(\{c_{ij} | i, j \in u \subset S\}), \end{aligned} \qquad (5)$$

subject to the same constraints as above. The index u extends over all structural units along a chain and over all possible chains and φ_u is a unit-based score. Recalling that one is trying to maximize a probability score P for each chain, the above approximation is implicitly making the assumption that the probabilities of the structural units are independent – a rather drastic assumption that is obviously violated even in favourable cases such as the structural units being complete helices or whole domains. This formulation of the problem, however, has a rich underlying structure that can be used efficiently in the development of solution strategies. Taking the logarithm of the above equation allows the product to be replaced by a summation over log probabilities and transforms the task of tracing the main chain to an integer linear-programming problem,

$$\text{maximize } \sum_u \varphi_u(\{c_{ij} | i, j \in u \subset S\}). \qquad (6)$$

The structural units mentioned above should be large enough to contain enough stereochemical information to facilitate a good discrimination of non-main-chain atoms and small enough to allow decisions to be made at a local level to build up the whole solution from smaller pieces. For free atoms, we have found that a set of four Cα atoms is a good compromise.

2.5. Great expectations

We have carried out a number of Cα-geometry analyses based on recently deposited high-resolution protein structures in the PDB (Bernstein *et al.*, 1977; Berman *et al.*, 2000). Frequency distributions for all Cα(*n*)—Cα(*n* + 1)—Cα(*n* + 2)—Cα(*n* + 3)— for all interatomic distances, pseudo-valence angles and dihedral angles have been computed. Two-dimensional distributions consisting of the first pseudo-valence angle Cα(*n*)—Cα(*n* + 1)—Cα(*n* + 2) and the dihedral angle Cα(*n*)—Cα(*n* + 1)—Cα(*n* + 2)—Cα(*n* + 3) have been studied by Oldfield & Hubbard (1994) and Kleywegt (1997). Oldfield uses these distributions to assist in the skeletonization-to-model step in *QUANTA* and Kleywegt uses them for structural-validation purposes, although he also mentions explicitly the possibility of already incorporating this validation tool at the model-building stage.

For estimating a continuous distribution from a set of data points we used the Parzen window technique (Fukunaga, 1990). A so-called kernel function is attached to each data point and the spread of the function is made dependent on the number of data points such that it vanishes for an infinite number of points. The choice of kernel functions, however, makes no assumptions about the form of the distribution. We chose multivariate Gaussians as kernel functions.

The thus obtained multidimensional distance and angle distributions derived directly from database analyses are well suited to the recognition of Cα atoms in sets of accurately positioned candidate atoms: all-*trans* peptide units of the test-set structures could be correctly identified as such. The presence of *cis*-peptides adds an extra complication that we have chosen to ignore: as a consequence, *ARP*/*wARP* will introduce chain breaks at these positions. The accuracy of the free-atom positions is, however, dependent on the current

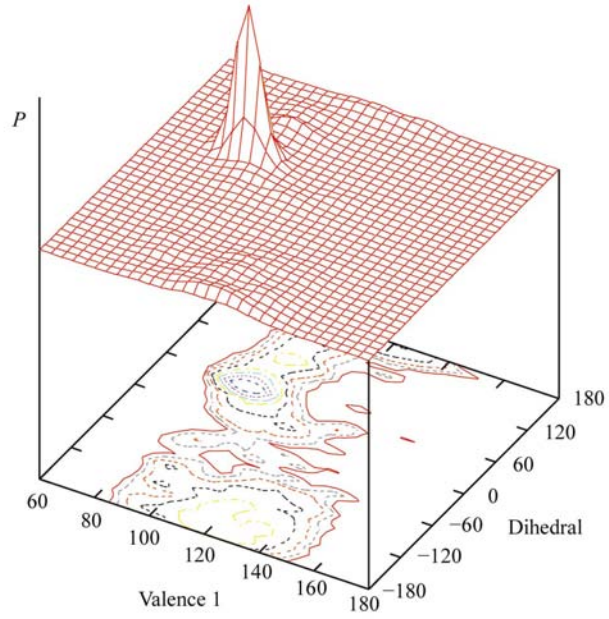

Figure 3
The pseudo-valence and dihedral distribution within a four-Cα-atom fragment from deposited high-resolution structures.

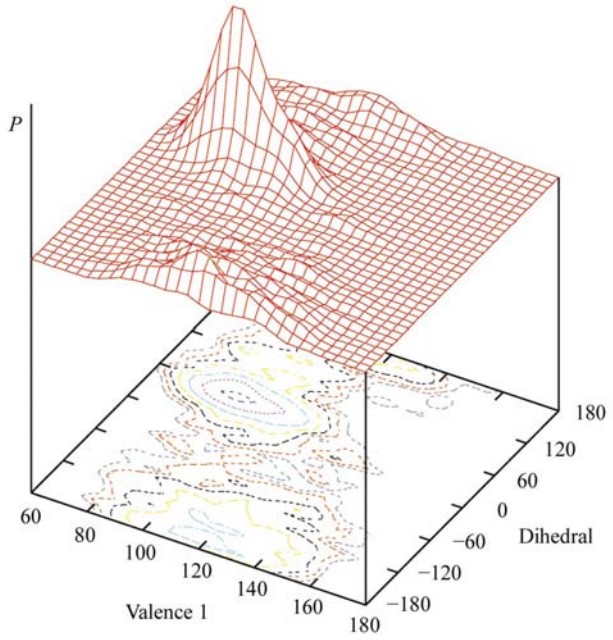

Figure 4
The pseudo-valence and dihedral distribution within four-Cα-atom fragments from deposited high-resolution structures that have had a random coordinate error of ± 0.6 Å added to their *xyz* positions.

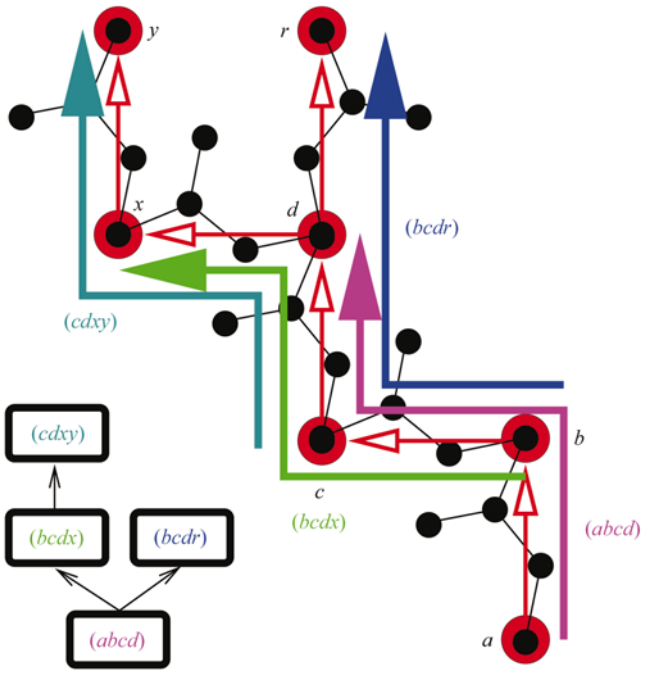

Figure 5
Fragments. The above drawing shows as red balls (*a*, *b*, *c*, *d*, *r*, *x*, *y*) a number of candidate Cα atoms with directed peptide-plane units fitted between them. All groups of four such candidate atoms that have connections in the same direction are given scores based on their pseudo-valence and dihedral angles. If these angles match expectation values, the fragment is stored as a node. Connections between nodes are established in the same manner and the chains are built by overlapping these fragments. A graph representation is shown in the lower drawing.

phase quality and resolution of the data. The patterns in a free-atoms model differ to a varying degree from those of well refined structures. Therefore, frequency distributions have been computed that correspond to structures with random coordinate errors. Figs. 3 and 4 show the projection of the multivariate distributions on the two pseudo-valence angles and the dihedral angle between four Cα atoms in sequence for the original data (Fig. 3) and the structures perturbed by a random coordinate error of 0.6 Å (Fig. 4). The distribution decreases rapidly in classification power beyond this limit.

2.6. Fragment overlapping

Distance and angle distributions of the above form, compiled over four-Cα-atom units, allow (minimal) three-dimensional information about protein structure to be used at the local decision-making stage. Furthermore, the units are small enough for all possible such fragments to be scanned and evaluated quickly. Given a list of free atoms, all possible Cα fragments of length four can be identified and stored in a database of building blocks for the problem at hand (see Fig. 5). The main chain can then be built by overlapping the last three atoms of each fragment with the first three of the following fragment. Each fragment is given a score proportional to the log odds ratio of the probability of the fragment parameters being produced by four Cα atoms and the probability of the fragment parameters being produced by any four atoms. As the [C$\alpha(n)-$C$\alpha(n+1)$] distances have already been used at the peptide-identification stage, their inclusion here contributes no additional information. The evaluation distributions contain only the two pseudo-valence angles and the dihedral angle within a four-Cα-atom unit. Distance and signed distances (the sign is that of the dihedral angle) corresponding to these angles could equally well be used (Morris, 2000). In the objective function (6) we use fragment scores weighted by the average peptide density. The chain scores are then built up by summation of these fragment scores.

2.7. Graph searching

With the above approximations, the optimization problem may now reformulated as finding a set of chains in a weighted graph that give the highest total score. The nodes in this graph represent four-Cα-atom fragments and the arcs are overlapping connections to other four-Cα-atom fragments (Fig. 6). The weights are computed from the geometrical scores of the fragments and their average density over the atomic positions. Note that the arcs (and also the weights) are directed to reflect the directionality of the peptide bond (C$-$N, N$-$C).

A standard algorithm for probing as deep as possible into a graph structure is the depth-first search algorithm (DFS) of graph theory (Papadimitriou & Steiglitz, 1998). For a given starting node, a list is maintained of all possible further nodes to visit in the form of a stack. The most recently found nodes are always the first to be taken off the stack and further processed. This achieves a systematic deep probing of the graph: only when the list of nodes to visit next for a specific node has been exhausted does the algorithm jump back to the previous (upstream) node. The search is continued for the remaining nodes of this upstream node and so on until the stack is empty. The time requirements of the standard DFS algorithm in an adjacency list implementation are proportional to the number of nodes plus arcs in the graph (Sedgewick, 1992) or, in terms of chain length and branching average, proportional to the branching average to the power of the chain length (Russel & Norvig, 1995).

One problem of the DFS algorithm is that is does not jump back more than one node and and does not consider different paths. Instead, it will always head down deep into the graph if that possibility exists. This means that early mistakes cannot be corrected later and the optimal solution cannot be guaranteed. We have modified an important modification by the introduction of many additional bookkeeping facilities and unlimited back stepping. The algorithm retrieves the score of each next fragment and keeps track of the total chain score of all nodes used so far and of all the Cα atoms of every used node and takes care to avoid loops in the chain and geometrical clashes. From every node in the graph, all chains up to a maximum length representing the total number of residues are sought. This results in a list of best chains for every node that are ranked by their scores. The best set of chains is determined by first starting with the highest scoring chain, accepting it, deleting all the nodes from the remaining set and then accepting the next highest scoring chain that can be built from the remaining nodes. Choosing the best set in this manner takes care of all possible chain breaks and rearrangements among them. The algorithm bears close resemblance to the depth-first search algorithm in its depth-limited form (Russel & Norvig, 1995) – the difference being our introduction of additional searching *via* back stepping with the required bookkeeping to establish the best set of solutions for each node. This systematic graph search is guaranteed to find the optimal solution to the problem. Exhaustive graph searches of this kind can, however, be intractable if the number of possible connections at each node is too high. In fact, we face a similar combinatorial explosion as above. The complexity has been reduced so far only by lowering the number of choices to make at each at each Cα atom by elimination of highly improbable fragments.

2.8. Further approximations

As outlined above, each accepted four-Cα fragment is assigned a quality score based on the probability of observing its geometry in known structures. The search algorithm can be set up to require a minimum quality of the fragments whilst scanning for chains. For large problems (above 10 000 candidate positions with an average number of forward connections greater than two), the algorithm first attempts to build chain stretches of high quality before reducing the fragment threshold level. The high-score fragments are most commonly helices or β-strands. The stepwise threshold lowering resembles a selective stepping through secondary-structure elements by their observed frequency in previously solved structures.

Furthermore, the search depth can be limited in an iterative manner. The search space is thus significantly reduced. This strategy is closely related to the iterative deepening search (Russel & Norvig, 1995).

An initial implementation of resolving the branch points was based on local (and isolated) decision-making. If branch points were encountered during the tracing, the algorithm would give preference to the highest density option. This model resembles a Markov decision process (Ross, 1992): the local density (the present) determines where the chain goes next (the future) independently of what has been already modelled (the past). The geometry of the chain is checked in a similar restrictive manner: only the distance and angle to the current peptide unit is used to indicate whether the next atom is approximately in an allowed position.

3. Discussion

The superiority of a decision-making strategy that can consult the future to evaluate the consequence of each decision before actually making it is obvious and represents an optimal form of management of systems with uncertainties. Our casting of main-chain tracing into the framework of an optimization problem and the efficient graph-search algorithms allow for local model-building decisions to be dealt with in a near-optimal manner by consulting the result at a more global level. This is a necessary step to automate the procedure further. Research is ongoing to improve the recognition of protein fragments in an electron-density map, but without significant breakthroughs in this area that are capable of dealing with maps of strongly varying quality the actual management part of the process will continue to play an important role.

In the current implementation only Cα distributions are used. The parametrization of the problem in terms of Cα geometry represents only an approximation to the problem, but the idea may be extended to incorporate other parameters of importance. The introduction of extra features and the combination of features may enhance the power of recognition and classification. A common approach to determine good parameters for pattern recognition is to simply start off with all conceivable features. Principal components analysis (PCA) may then be applied to provide better features for recognition *via* linear combination of the initially chosen parameterization and also to reduce the dimensionality of the problem (Fukunaga, 1990; Duda & Hart, 1973). In initial test cases we found the features suggested by PCA to be superior, although the mutual information of the individual features before and after PCA changed only marginally, indicating a high dependence of the new features despite the lack of correlation.

The success of the procedure outlined in this paper depends rather critically on the accuracy of the Cα positions. Up to a placement error of about 0.5 Å this approach works well; above this value the power of the distributions for classification breaks down. At a Cα coordinate error of 0.7 Å and above, the usefulness declines steeply. This is the error range that one must expect for free-atoms modelling of electron-density maps calculated from data with resolution lower than about 2.3 Å, owing to the breakdown of atomicity.

Although the search techniques described here for finding the best chain would in principle allow one to simply test an enormous number of trial positions and thus avoid the problem of requiring such a precise identification of Cα atoms, our current algorithms are not yet powerful enough to deal with such exhaustive searches with reasonable time and disk-space usage. The execution time for finding the best set of main-chain fragments from a set of free atoms that have undergone the peptide-plane recognition procedure is typically of the order of seconds. However, this time increases approximately with the average number of branch points to

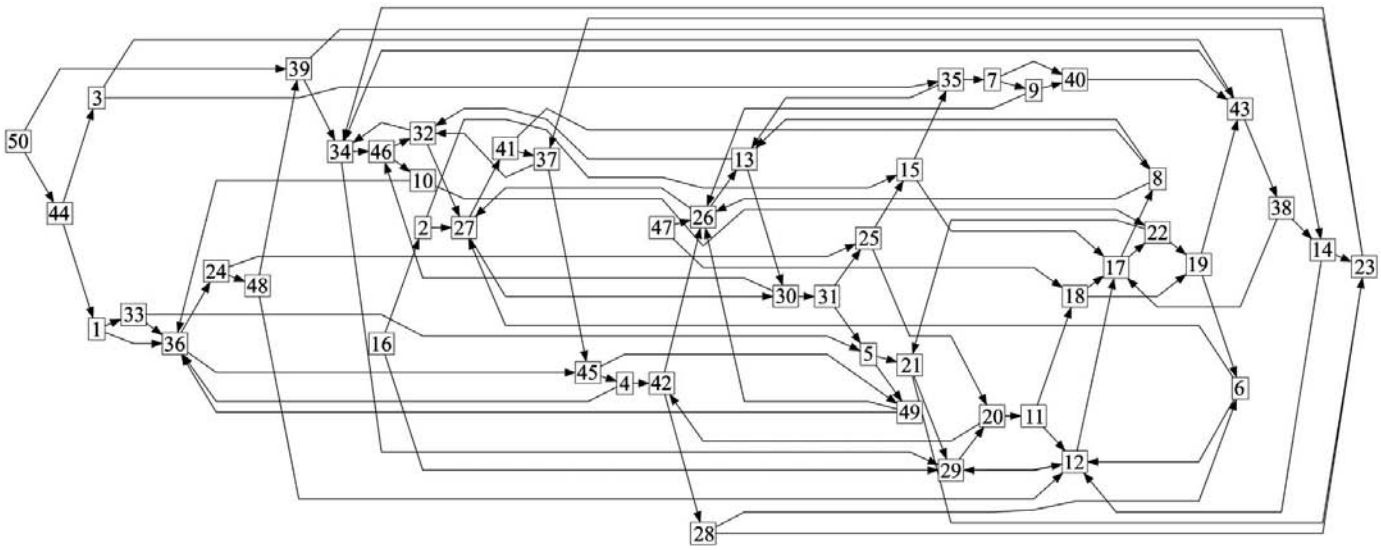

Figure 6
Simplified representation of the main-chain tracing problem. 54 Cα-atom units are shown with an average of two outgoing connections. The graph layout has been optimized with *daVinci* V2.1 (http://www.informatik.uni-bremen.de/daVinci/) for clarity: the connection lengths are in no relation to the weights.

the power of five in the current implementation and the above approximations. This sets practical limits on the maximum branching average of about four. The average number of branch points from free-atoms modelling combined with peptide-plane recognition is commonly of the order of two to four; systematic grid-point searches often accumulate well over ten. Especially for cases in which *ARP* cannot identify atoms in the density (poor initial phases, resolution less than 2.3 Å) a grid search seems an attractive way to circumvent this problem, given sufficient algorithmic advances. We believe that improved methods for Cα location and/or better filtering methods combined with a similar management-system algorithm as described here will allow complete automation of the main-chain model-building process.

The iterative nature of *ARP*/*wARP* model building and its coupling with refinement remains the key to success in automation. *ARP*/*wARP forgets* its previous model-building decisions at each new cycle and bases its model always only on the current electron density. This approach combined with the employment of maximum-likelihood refinement (Murshudov *et al.*, 1997) allows bias from density misinterpretations (wrong model-building decisions) to largely be corrected for and avoided in the final model.

RJM is grateful for financial support from EU Grant BIO2-CT920524/BIO4-CT96-0189, for the facilities provided by the EMBL Outstation Hamburg during long-term leave for PhD research from the Karl-Franzens Universität Graz, and to Professor Christoph Kratky for kind support and assistance throughout. All authors would like to thank R. Meijers and P. Zwart for useful comments and fruitful discussions and G. Bricogne for critical reading of the manuscript.

References

Bacchi, A., Lamzin, V. S. & Wilson, K. S. (1996). *Acta Cryst.* D**52**, 641–646.
Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. & Bourne, P. E. (2000). *Nucleic Acids Res.* **28**, 235–242.
Bernstein, F. C., Koetzle, T. F., Williams, G. J. B., Meyer, E. F. Jr, Brice, M. D., Rodgers, J. R., Kennard, O., Shimanouchi, T. & Tasumi, M. (1977). *J. Mol. Biol.* **112**, 535–542.
Bricogne, G. (1997*a*). *Methods Enzymol.* **276**, 361–423.
Bricogne, G. (1997*b*). *Methods Enzymol.* **277**, 14–18.
Brünger, A. T. (1992). *Nature (London)*, **355**, 472–475.
Cowtan, K. (1998). *Acta Cryst.* D**54**, 750–756.
Diller, D. J., Pohl, E., Redinbo, M. R., Hovey, T. & Hol, W. G. J. (1999). *Proteins Struct. Funct. Genet.* **36**, 512–525.
Diller, D. J., Redinbo, M. R., Pohl, E. & Hol, W. G. J. (1999). *Proteins Struct. Funct. Genet.* **36**, 526–541.
Duda, R. O. & Hart, P. E. (1973). *Pattern Classification and Scene Analysis.* New York: Wiley-Interscience.
Esnouf, R. M. (1997). *Acta Cryst.* D**53**, 665–672.
Finzel, B. C. (1997). *Methods Enzymol.* **277**, 230–242.
Fortier, S., Chiverton, A., Glasgow, J. & Leherte, L. (1997). *Methods Enzymol.* **277**, 131–157.
Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*, 2nd ed. New York: Academic Press.
Greer, J. (1974). *J. Mol. Biol.* **82**, 279–301.
Holton, T., Ioerger, T. R., Christopher, J. A. & Sacchettini, J. C. (2000). *Acta Cryst.* D**56**, 722–734.
Jones, T. A. & Kjeldgaard, M. (1996). *Methods Enzymol.* **277**, 173–198.
Jones, T. A. & Thirup, S. (1986). *EMBO J.* **5**, 819–822.
Kleywegt, G. J. (1997). *J. Mol. Biol.* **273**, 371–376.
Kleywegt, G. J. & Jones, T. A. (1997*a*). *Methods Enzymol.* **277**, 208–230.
Kleywegt, G. J. & Jones, T. A. (1997*b*). *Acta Cryst.* D**53**, 179–185.
Koch, M. H. J. (1974). *Acta Cryst.* A**30**, 67–70.
Konnert, J. H. & Hendrickson, W. A. (1980). *Acta Cryst.* A**36**, 344–350.
Lamzin, V. S., Perrakis, A. & Wilson, K. S. (2001). *International Tables for Crystallography*, Vol. F, edited by M. Rossmann & E. Arnold, pp. 720–722. Dordrecht: Kluwer Academic Publishers.
Lamzin, V. S. & Wilson, K. S. (1993). *Acta Cryst.* D**49**, 129–149.
Lamzin, V. S. & Wilson, K. S. (1997). *Methods Enzymol.* **277**, 269–305.
Leherte, L., Fortier, S., Glasgow, J. & Allen, F. H. (1994). *Acta Cryst.* D**50**, 155–166.
Levitt, D. G. (2001). *Acta Cryst.* D**57**, 1013–1019.
McRee, D. E. (1992). *J. Mol. Graph.* **10**, 44–47.
Morris, R. J. (2000). PhD thesis. Karl-Franzens Universität Graz, Austria.
Murshudov, G. N., Vagin, A. A. & Dodson, E. J. (1997). *Acta Cryst.* D**53**, 240–255.
Oldfield, T. J. (1996). In *Crystallographic Computing 7. Proceedings of the IUCr Macromolecular Computing School*, edited by K. D. Watenpaugh & P. E. Bourne. Oxford University Press. In the press.
Oldfield, T. J. & Hubbard, R. E. (1994). *Proteins Struct. Funct. Genet.* **18**, 324–337.
Papadimitriou, C. H. & Steiglitz, K. (1998). *Combinatorial Optimization*, pp. 194–200. New York: Dover.
Perrakis, A., Morris, R. J. & Lamzin, V. S. (1999). *Nature Struct. Biol.* **6**, 458–463.
Ross, S. M. (1992). *Applied Probability Models with Optimization Applications*, pp. 119–153. New York: Dover.
Roussel, A. & Cambillau, C. (1991). *Silicon Graphics Geometry Partners Directory*, p. 81. Mountain View, CA, USA: Silicon Graphics.
Russel, S. & Norvig, P. (1995). *Artificial Intelligence*, pp. 77–80. New Jersey: Prentice Hall.
Scheres, S. H. W. & Gros, P. (2001). *Acta Cryst.* D**57**, 1820–1828.
Sedgewick, R. (1992). *Algorithms in C++*, pp. 415–435. Reading, MA, USA: Addison Wesley.
Swanson, S. M. (1994). *Acta Cryst.* D**50**, 695–708.
Terwilliger, T. (2001). *Acta Cryst.* D**57**, 1755–1762.
Tickle, I., Laskowski, R. & Moss, D. S. (1998). *Acta Cryst.* D**54**, 547–557.
Wang, C. E. (2000). *Acta Cryst.* D**56**, 1591–1611.