Acta Crystallographica Section D
**Biological Crystallography**
ISSN 0907-4449

# A stochastic approach to molecular replacement

**Nicholas M. Glykos[a]* and Michael Kokkinidis[a,b]**

[a]IMBB, FORTH, PO Box 1527, 71110 Heraklion, Crete, Greece, and [b]Department of Biology, University of Crete, PO Box 2208, 71409 Heraklion, Crete, Greece

Correspondence e-mail: glykos@crystal2.imbb.forth.gr

Received 6 July 1999
Accepted 25 November 1999

The classical approach to the problem of placing $n$ copies of a search model in the asymmetric unit of a target crystal structure is to divide this $6n$-dimensional optimization problem into a succession of three-dimensional searches (rotation-function followed by translation-function searches for each of the models). Here, it is shown that a structure-determination method based on a reverse Monte Carlo minimization of a suitably chosen statistic in the $6n$-dimensional space defined by the rotational and translational parameters of the $n$ molecules is both feasible and practical, at least for small $n$. Because all parameters of all molecules are determined simultaneously, this algorithm is expected to improve the signal-to-noise ratio in difficult cases involving high crystallographic/non-crystallographic symmetry in tightly packed crystal forms. Preliminary results from the application of this method (obtained with a space-group general computer program which has been developed for this purpose) are presented.

## 1. Introduction

As the number of macromolecules with known three-dimensional structures continues to increase, so does the importance of molecular replacement as a tool for the determination of new crystal structures. Although the number of contributions and enhancements to the method since its original derivation by Rossmann & Blow (1962) is very large (see, for example, Rossmann, 1972, 1990; Machin, 1985; Dodson *et al.*, 1992; Carter & Sweet, 1997), it can be argued that in current practice there are two major approaches for solving molecular-replacement problems. The first is based on the calculation (and subsequent examination) of a large number of both rotation-function and translation-function solutions in a fast and more or less automatic manner [typified by the program *AMoRe* (Navaza & Saludjian, 1997) from the *CCP*4 suite of programs (Collaborative Computational Project, Number 4, 1994)]. The second approach uses a Patterson correlation refinement procedure to simultaneously identify putatively correct solutions from the rotation function and to improve the search model before continuing with a translation-function calculation (Brünger, 1992, 1997). In both cases, however, the rotational and translational parameters of the search model(s) are treated separately and an independent determination of their values is attempted. This may result in a reduction of the signal-to-noise ratio in problems involving a large number of molecules per unit cell and/or crystal forms with low solvent content.

A third more recently developed class of algorithms attempts to improve the sensitivity and accuracy of the method by increasing the dimensionality of the parameter space explored. This is achieved by performing successive six-dimensional searches for each of the molecules present in the crystallographic asymmetric unit. Published examples of such methods include a genetic algorithm approach (Chang & Lewis, 1997), an evolutionary search methodology (Kissinger *et al.*, 1999) and a systematic six-dimensional search using a fast translation function (Sheriff *et al.*, 1999).

Here, we describe an alternative $6n$-dimensional molecular-replacement procedure which is based on the simultaneous determination of the rotational and translational parameters of all molecules present in the crystallographic asymmetric unit of a target structure.

## 2. Algorithms and implementation

### 2.1. Algorithms

The volume of the configurational space defined by the rotational and translational parameters of the molecules present in the asymmetric unit of a target crystal structure is so large that a systematic examination of all possible combinations of their positions and orientations is beyond present-day computing capabilities. On the other hand, simulated-annealing methods have repeatedly been shown to be able to deal with multidimensional combinatorial optimization problems in near-optimal ways and in a fraction of the time required for a systematic search (Kirkpatrick *et al.*, 1983; Press *et al.*, 1992). The two most popular simulated-annealing techniques are molecular dynamics and Monte Carlo simulations. In the case of macromolecular refinement (where the atoms are covalently linked), molecular dynamics is the method of choice (Brünger & Rice, 1997, and references therein). For the problem under examination, in which the search model is fixed and no interatomic potential is used,[1] a Monte Carlo minimization method appears to be a natural solution.

We have chosen to use a modification of the reverse Monte Carlo technique (McGreevy & Pusztai, 1988; Keen & McGreevy, 1990) where, instead of minimizing the quantity $\chi^2 = \sum_{hkl}[(F_o - F_c)/\sigma(F_o)]^2$, one minimizes any of the following (user-defined) target functions: (i) the conventional crystallographic $R$ factor, $R = \sum_{hkl}|F_o - F_c|/F_o$, (ii) the quantity $1.0 - \mathrm{Corr}(F_o, F_c)$ and (iii) the quantity $1.0 - \mathrm{Corr}(F_o^2, F_c^2)$, where Corr is the linear correlation coefficient function, $F_o$ and $F_c$ are the observed and calculated structure-factor amplitudes, respectively, of the $hkl$ reflection and $\sigma(F_o)$ is the standard uncertainty of the corresponding measurement. To avoid unnecessary repetition and to simplify the discussion that follows, we will hereafter refer only to the $R$-factor statistic, on the understanding that any of the correlation-based targets can be substituted for it.

The minimization procedure follows closely the original Metropolis algorithm (Metropolis *et al.*, 1953) and its basic steps are outlined below. Random initial orientations and positions are assigned to all molecules present in the crystallographic asymmetric unit of the target structure and the $R$ factor ($R_{\rm old}$) between the observed and calculated structure-factor amplitudes is noted. In the first step of the basic iteration, a molecule is chosen randomly and its orientational and translational parameters are randomly altered. The $R$ factor ($R_{\rm new}$) corresponding to this new arrangement is calculated and compared with $R_{\rm old}$: if $R_{\rm new} \leq R_{\rm old}$, then the new configuration is accepted and the procedure is iterated with a new (randomly chosen) molecule. If $R_{\rm new} > R_{\rm old}$ (that is, if the new configuration results in a worse $R$ factor), the new configuration is accepted with a probability $\exp[(R_{\rm old} - R_{\rm new})/T]$, where $T$ is a control parameter which plays the role of temperature in statistical mechanical simulations. This probabilistic treatment again relies on the random number generator: if $\exp[(R_{\rm old} - R_{\rm new})/T] > \xi$, where $\xi$ is a random number between 0.0 and 1.0, the new configuration is accepted and the procedure iterated. If $\exp[(R_{\rm old} - R_{\rm new})/T] \leq \xi$, we return to the previous configuration (the one that resulted in an $R$ factor equal to $R_{\rm old}$) and reiterate.

It would appear at first sight that this is not a practical algorithm: for every iteration, the atomic coordinates of the search model must be rotated and translated, a new electron-density map calculated and, more importantly, a fast Fourier transform (FFT) step must be performed. Assuming that the limiting step is the FFT of $N$ grid points, then this procedure would require ($N\log_2 N$) operations per cycle (Press *et al.*, 1992) and would not be practical for simulations longer than a few tens of thousands of cycles. These appearances are deceiving. By trading computer memory for speed of execution, this algorithm can be converted to a linear $O(K)$ procedure, where $K$ is the number of reflections (of the target structure) expanded to the space group $P1$. This is achieved by calculating (and storing in memory) the molecular transform of the search model before the actual minimization is started. For the rest of the simulation, in order to calculate a structure-factor amplitude $F_c(hkl)$ we only have to add the (complex) values of the molecular transform at the coordinates that the $hkl$ reflection would take if rotated accordingly to the orientation of each molecule in the unit cell[2] (a detailed account on the usage of the molecular transform to accelerate the structure-factor calculation for this type of problem can be found in §2.1 of Chang & Lewis, 1997). A reader may object that if at every step we calculate the contribution of each molecule to

[1] The inclusion of a van der Waals repulsion term would convert the simulated-annealing algorithm to an inefficient packing-function optimizer: once an arrangement is found that allows the efficient packing of the search models and their symmetry equivalent in the target unit cell, no further major rearrangements of the molecular configuration will be possible (especially in tightly packed crystal forms) and the minimization would come to a halt. A more thoughtful procedure based on a time-dependent weighting scheme that would slowly increase the contribution from the interatomic potential during the length of the minimization would greatly increase the computational requirements of the algorithm.

[2] In other words, instead of rotating the search models, we keep the model (actually, its molecular transform) fixed and rotate the reciprocal lattice of the target structure. The translational parameters of the search models (and crystallographically related molecules) enter the calculation as phase shifts applied to the complex transform values.

every reflection, then the computer time per step of the minimization would depend not only on the number of unique reflections and crystallographic symmetry operators, but also on the number of the search models in the asymmetric unit. The dependence on the number of molecules in the asymmetric unit of the target structure can be removed if we recall that at each step of the minimization we only modify the parameters of one of the search models. If the contribution of each molecule to every reflection is stored in memory, then at each step we only have to recalculate the contribution from the molecule that is being tested.

### 2.2. Implementation

A space-group general computer program has been developed which implements the algorithms described in the previous section. As is always the case with the Monte Carlo method, the efficiency of the minimization depends greatly on the optimal (or otherwise) choice of (i) an annealing schedule which specifies how the temperature of the system will vary with time and (ii) of a set of moves that determine how the next configuration (the one that will be tested) can be obtained from the current configuration (the one that has already been tested).

The current implementation of the program supports three annealing modes. In the first mode, the temperature is kept constant throughout the minimization. The second is a slow-cooling mode, with the temperature linearly dependent on the simulation time. In the third mode, the temperature of the system is automatically adjusted in such a way as to keep the fraction of moves made against the gradient of the $R$ factor constant and equal to a user-defined value.[3] It is possible to obtain reasonable estimates of the temperature range required for a slow-cooling run automatically: this is achieved by monitoring a quantity analogous to the specific heat (from statistical mechanics) during a short slow-cooling simulation which is started from a sufficiently remote (high) temperature (Kirkpatrick *et al.*, 1983).

The selection of an optimal set of possible moves and the control of their magnitudes depends on the nature of the individual problems, making it difficult to find a satisfactory solution without losing generality. Instead of artificially making the optimization problem discontinuous (by restricting the configurational parameters to take values from a pre-defined fixed grid), we have chosen to work with the continuous case (in which any parameter can take any value from within its defining limits). The program stores the orientational parameters of the search models using the polar angle $(\omega, \varphi, \kappa)$ convention, with $\omega$ defining the latitude and $\varphi$ the longitude of a rotation axis about which the molecule is rotated by $\kappa$ degrees. The translational parameters are stored in terms of the fractional coordinates of the geometrical centres of the molecules in the crystallographic frame of the target structure. The choice of polar angles simplifies the task of updating and controlling the orientational parameters: for the whole length of the minimization, an orientation for the rotation axis is chosen randomly and uniformly from the full-half sphere (that is, $0 \leq \omega \leq \pi/2$ and $0 \leq \varphi < 2\pi$), leaving only the rotational offset $\Delta\kappa$ and the translational offsets $\Delta x$, $\Delta y$, $\Delta z$ to be specified before a new configuration can be obtained from the current one. The program supports two modes of move-size control. In the first, the maximum possible values that the random offsets $\Delta\kappa$, $\Delta x$, $\Delta y$ and $\Delta z$ can take are kept constant throughout the simulation with $\max(\Delta\kappa) = d_{\min}$ (in degrees) and $\max(\Delta x, \Delta y, \Delta z) = d_{\min}/\max(a, b, c)$, where $d_{\min}$ is the minimum Bragg spacing of the input data and $a$, $b$, $c$ are the unit-cell translations of the target structure (in Å). In the second mode, the maximum move sizes (as defined above) are linearly dependent on both time and the current $R$ factor, with $\max(\Delta\kappa) = \pi R t/t_{\text{total}}$ and $\max(\Delta x, \Delta y, \Delta z) = 0.5 R t/t_{\text{total}}$, where $R$ is the current $R$ factor, $t$ is the current time step and $t_{\text{total}}$ is the total number of time steps for the minimization. The dependence on the $R$ factor is justified on the grounds that as we approach a minimum of the target function, we should be sampling the configurational space on a finer grid.[4] The time dependence follows from a similar argument.

## 3. Results

Although the final proof of the utility of any new structure-determination method is its ability to determine previously unknown structures, we think that it is useful to illustrate the applicability of the algorithms presented above with model calculations based on known structures deposited in the Protein Data Bank.

The first example has been chosen to show that for simple problems this stochastic approach can be approximately as fast as the traditional methods. The example was constructed as follows: structure-factor amplitudes were calculated from PDB entry 2ihl containing the atomic coordinates of a monoclinic form (space group $C2$) of the Japanese quail lysozyme with unit-cell parameters $a$ = 103.90, $b$ = 38.70, $c$ = 34.00 Å, $\beta$ = 100.60°. The resulting amplitudes were modified by adding an offset ranging randomly and uniformly from −20 to +20% of their modulus. This 'noisy' data set was treated as the observed data set of the target structure. The search model for the calculation was turkey egg-white lysozyme (PDB entry 2lz2), which has an r.m.s. deviation from the Japanese quail lysozyme of 1.42 Å. Fig. 1 shows the evolution of the average $R$ factors from five independent minimizations using the 281 strongest reflections to 4 Å resolution (about 24% of all data to this resolution). All five simulations converged to the correct solution (giving $R$ and free $R$ values of about 0.27). The important thing to note, however, is that

[3] The program counts the number of times that a new configuration is accepted even though it results in a higher $R$-factor value. After a predefined number of iterations, the fraction of moves that have been made against the $R$-factor gradient is calculated: if it is less than a target value (defined by the user) the temperature is increased; otherwise it is decreased.

[4] The word 'grid' is used here metaphorically. For all practical purposes, the values of $\Delta\kappa$, $\Delta x$, $\Delta y$ and $\Delta z$ returned by the random number generator are continuous (if, for example, the generator returns values in the range 0 to $2^{31} - 1$ and $\max(\Delta\kappa) = \pi$, then the 'grid size' on $\Delta\kappa$ is less than $9 \times 10^{-8}$ degrees).

on a modern workstation each minimization took less than 2 min of Central Processing Unit (CPU) time.[5] Given that in all cases the solution was found in less than 40 000 steps, this is equivalent to about 40 s of CPU time per solution (and because the parameter space is continuous, this includes the equivalent of a rigid-body refinement step).

The second example shows results from a six-dimensional search using real data obtained from the PDB entry for the orthorhombic form of chicken egg-white lysozyme (PDB entry 1aki, data set code r1akisf.ent). The space group of the target structure is $P2_12_12_1$, with unit-cell parameters $a$ = 59.062, $b$ = 68.451, $c$ = 30.517 Å and one molecule per asymmetric unit. The search model for this calculation was Japanese quail lysozyme (PDB entry 2ihl), which has an r.m.s. deviation from the target structure of 1.20 Å and a maximum displacement of 8.72 Å. Fig. 2 shows the evolution of the average $1.0 - \mathrm{Corr}(F_o, F_c)$ values from five independent minimizations using the 558 strongest reflections in the resolution range 15–4.0 Å (about 50% of all data to this resolution). As it is obvious from this figure, four out of five minimizations converged to the correct solution [with $(1 - C)$ values of about 0.39]. The total CPU time for each minimization was 118 min, with the four solutions of the successful runs appearing after 30, 36, 59 and 64 min, respectively. Clearly, longer simulation times would improve the success rate of the algorithm even further.[6]

The third example is based on a target structure containing two molecules in the crystallographic asymmetric unit and has an added difficulty arising from the presence of a pseudo-$B$-centred lattice. Data for this example were calculated from PDB entry 1lys containing the atomic coordinates of a monoclinic form (space group $P2_1$) of hen egg-white lysozyme with unit-cell parameters $a$ = 27.23, $b$ = 63.66, $c$ = 59.12 Å, $\beta$ = 92.9°. The two molecules in the asymmetric unit have approximately the same orientation and are related by a translation vector (in fractional coordinates) of (0.51, 0.02, 0.53). The closeness of the translation vector to $(\frac{1}{2}, 0, \frac{1}{2})$, generates a super-lattice corresponding to a pseudo-$P2_1$ cell with unit-cell parameters $a$ = 31.9, $b$ = 63.6, $c$ = 33.2 Å, $\beta$ = 130.6° and only one molecule in the asymmetric unit. To make the example more realistic, noise with a maximum amplitude of ±20% was added to the calculated $F$s and we used a search model (Japanese quail lysozyme, PDB entry 2ihl) which has r.m.s. deviations from the two target molecules of 1.52 and 1.56 Å, respectively. Fig. 3 shows the evolution of the average $R$ factors from three independent minimizations using the 1066 strongest reflections to 4 Å resolution (about 60% of all data to this resolution). All three simulations converged to the correct solution giving $R$ and free $R$ values of about 0.34, with each minimization taking approximately 5.5 h of CPU time on the reference workstation. As can be seen from the diagrams,

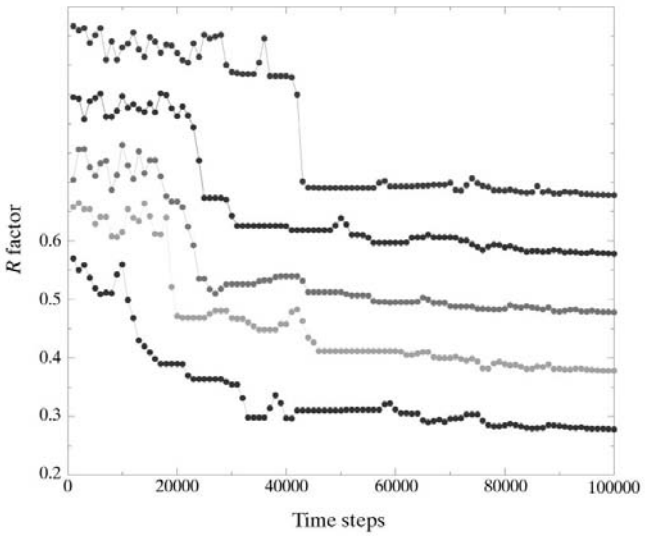


**Figure 1**
Evolution of the average $R$ factors for five minimizations from a five-dimensional problem. See text for details. The $R$-factor values refer only to the lower curve, with the other four curves translated by +0.1, 0.2, 0.3 and 0.4 $R$-factor units in order to improve clarity. All graphs were prepared using the program *Xmgr* (http://plasma-gate.weizmann.ac.il/Xmgr/).

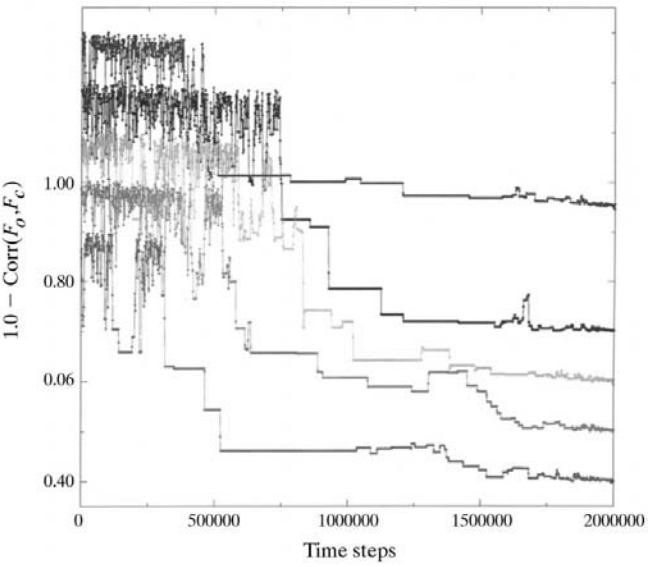


**Figure 2**
Evolution of the average $1.0 - \mathrm{Corr}(F_o, F_c)$ values for five minimizations from a six-dimensional problem using real data. The $(1 - C)$ values refer only to the lower curve, with the other four curves translated by +0.1, 0.2, 0.3 and 0.4 units in order to improve clarity. See text for details.

[5] All references to physical time measurements of the program's speed of execution refer to a UNIX workstation which in single-user mode gave the following SPEC95 benchmark results: SPECint95 = 16.6, SPECint_rate95 = 149, SPECfp95 = 21.9 and SPECfp_rate95 = 197 (Standard Performance Evaluation Corporation, 10754 Ambassador Drive, Suite 201, Manassas, VA 21109, USA; http://www.specbench.org/). UNIX is a registered trademark of UNIX System Laboratories, Inc.

[6] The reduced success rate for this example (when compared with the previous one, shown in Fig. 1) can be attributed to a combination of two factors. The first is the higher dimensionality of the problem (combined with the relatively short simulation time). The second is the absence of a bulk-solvent correction from the current implementation of the program. This introduces a systematic error for the low-resolution data and makes a low-resolution cut-off necessary. This, in turn, generates series-termination errors which complicate the target-function landscape, making the identification of the global minimum more difficult.

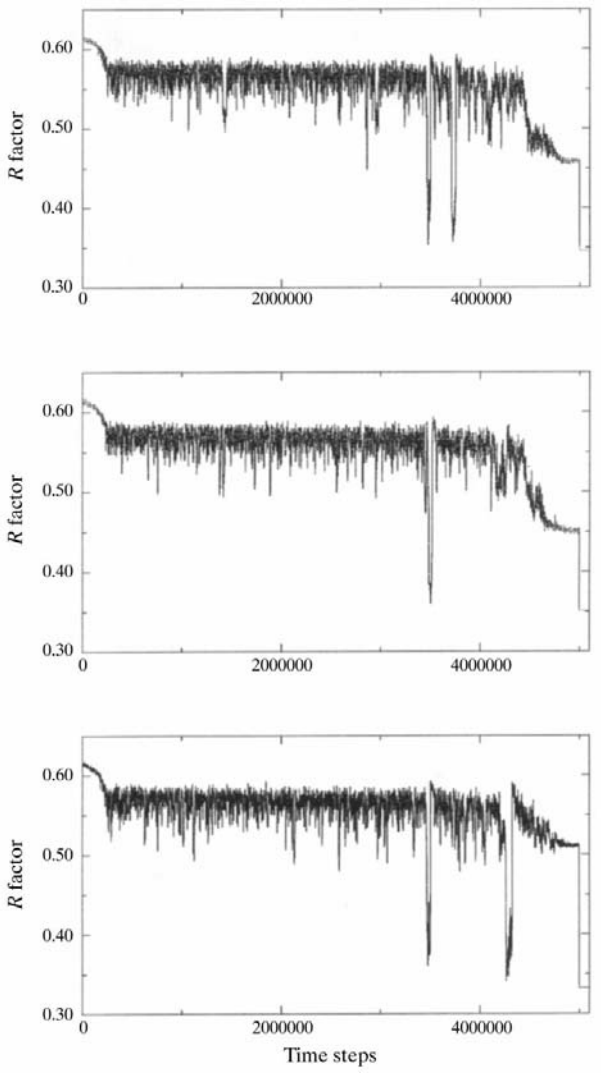


**Figure 3**
Evolution of the average $R$ factors for three minimizations from an 11-dimensional problem. See text for details.

although all three simulations found the (correct) global minimum after about 3.5 million moves (corresponding to about 3.8 h of CPU time), none of them remained there until the end of the minimization. The reason is that for this example we used an annealing protocol which automatically increases the temperature when the system is trapped inside a minimum.[7]

It is instructive to note here that when an attempt was made to reduce the CPU time requirements by performing the simulation using only the 279 strongest reflections to 5 Å, the program converged to two distinct types of solutions: the first was the correct solution as described above. The second was a solution corresponding to the structure of the pseudo-$P2_1$ cell, with the two search models having approximately the same orientational and translational parameters, resulting in a complete overlap of the two molecules. The reason for this behaviour is that the information about the real cell is mostly contained in the weaker reflections. If these are systematically excluded from the calculation, the program will converge with similar frequencies to the two solutions that are consistent with the given (strong) reflections.

## 4. Discussion

We have shown that a stochastic molecular-replacement method which is able to determine the rotational and translational parameters of all search models simultaneously is not only feasible but is also practical for the majority of everyday crystallographic problems.

The model calculations presented above showed that for relatively simple problems this method can perform as efficiently as the traditional algorithms. For more complex problems, the cost of the significantly higher CPU time requirements may be balanced by the improved signal-to-noise ratio offered by this approach.

This is not to imply that the method as it stands does not suffer from serious limitations. The first and most important is that in its current implementation it is assumed that the target crystal structure consists exclusively of only one molecular species. Although there is no *a priori* reason for limiting the algorithm to only one type of search model, the amount of physical memory required for storing two (or more) molecular transforms simultaneously would make the implementation impractical. The second (not unrelated) limitation is that the molecular structure of the search model is kept fixed throughout the calculation. Again, there is no practical way for modifying the search model during the calculation without losing the advantages offered by a pre-calculated molecular transform.[8] One final problem concerns the incorporation of known non-crystallographic symmetry elements (determined, for example, from the self-rotation or Patterson functions) in the calculations described above. In the case of exclusively translational non-crystallographic symmetry (as was the case in the third example of §3), this prior knowledge can be directly incorporated in the current implementation of the program (in the form of additional fixed symmetry elements). Incorporation of general non-crystallographic symmetry elements restraints is not possible with the current implementation of the program, as this would entail independent refinement of the positions of all non-crystallographic symmetry axes with a known orientation. If, however, both the orientation and position of the non-crystallographic symmetry axes is known, then this prior knowledge can be directly used with the current version of the program.

[7] This annealing protocol (which can escape even from the global minimum) is functional only because the program saves the configuration that gave the lowest $R$ factor during the minimization. When the simulation is finished, the configuration that resulted to the lowest $R$ factor is restored and a few extra cycles of refinement at a low temperature are performed before writing out the final coordinates of all search models.

[8] An obvious solution would be to treat the individual domains (or other substructure) as independent search models, but this would not only be impractical owing to physical memory limitations but would also unnecessarily increase the number of free parameters (and the dimensionality of the problem).

## 5. Program availability

The program (*Queen of Spades*), together with its documentation and some example scripts, is distributed free of charge to both academic and non-academic users. It is available for download from the WWW at http://origin.imbb.forth.gr:8888/~glykos/.[9] The distribution contains executable images suitable for the majority of the most commonly used workstation architectures.

We should like to thank the referees for their useful comments and suggestions.

## References

Brünger, A. T. (1992). *X-PLOR Version* 3.1. *A System for X-ray Crystallography and NMR*. New Haven, Connecticut, USA: Yale University Press.
Brünger, A. T. (1997). *Methods Enzymol.* **276**, 558–580.
Brünger, A. T. & Rice, L. M. (1997). *Methods Enzymol.* **277**, 243–269.
Carter, C. W. Jr & Sweet, R. M. (1997). *Methods Enzymol.* **276**, 558–618.

[9] The version of this program described in this paper is also available from the IUCr electronic archive (Reference: ad0083). Services for accessing this material are described at the back of this issue.

Chang, G. & Lewis, M. (1997). *Acta Cryst.* D**53**, 279–289.
Collaborative Computational Project, Number 4 (1994). *Acta Cryst.* D**50**, 760–763.
Dodson, E. J., Glover, S. & Wolf, W. (1992). Editors. *Proceedings of the CCP4 Study Weekend. Molecular Replacement.* Warrington: Daresbury Laboratory.
Keen, D. A. & McGreevy, R. L. (1990). *Nature (London)*, **344**, 423–425.
Kirkpatrick, S., Gelatt, C. D. Jr & Vecchi, M. P. (1983). *Science*, **220**, 671–680.
Kissinger, C. R., Gehlhaar, D. K. & Fogel, D. B. (1999). *Acta Cryst.* D**55**, 484–491.
McGreevy, R. L. & Pusztai, L. (1988). *Mol. Simul.* **1**, 359–367.
Machin, P. A. (1985). Editor. *Proceedings of the CCP4 Study Weekend. Molecular Replacement.* Warrington: Daresbury Laboratory.
Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. (1953). *J. Chem. Phys.* **21**, 1087–1092.
Navaza, J. & Saludjian, P. (1997). *Methods Enzymol.* **276**, 581–593.
Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1992). *Numerical Recipes in C. The Art of Scientific Computing*, 2nd ed. Cambridge University Press.
Rossmann, M. G. (1972). *The Molecular Replacement Method.* London: Gordon & Breach.
Rossmann, M. G. (1990). *Acta Cryst.* A**46**, 73–82.
Rossmann, M. G. & Blow, D. M. (1962). *Acta Cryst.* **15**, 24–31.
Sheriff, S., Klei, H. E. & Davis, M. E. (1999). *J. Appl. Cryst.* **32**, 98–101.