Molecular replacement with multiple different models
Nicholas M. Glykos and Michael Kokkinidis

Copyright © International Union of Crystallography

Author(s) of this paper may load this reprint on their own web site provided that this cover page is retained. Republication of this article or its storage in electronic databases or the like is not permitted without prior permission in writing from the IUCr.
Molecular replacement with multiple different models

Nicholas M. Glykos*‡ and Michael Kokkinidis*‡

*IMBB, FORTH, PO Box 1527, 71110 Heraklion, Crete, Greece, and ‡Department of Biology, University of Crete, PO Box 2208, 71409 Heraklion, Crete, Greece. Correspondence e-mail: glykos@imbb.forth.gr

Classical molecular replacement methods and the newer six-dimensional searches treat molecular replacement as a succession of sub-problems of reduced dimensionality. Due to their ‘divide-and-conquer’ approach, these methods necessarily ignore (at least during their early stages) the very knowledge that a target crystal structure may comprise, for example, more than one copy of a search model, or several models of different types. An algorithm for a stochastic multi-dimensional molecular replacement search has been described previously and shown to locate solutions successfully, even in cases as complex as a 23-dimensional 4-body search. The original description of the method only dealt with a special case of molecular replacement, namely with the problem of placing \( n \) copies of only one search model in the asymmetric unit of a target crystal structure. Here a natural generalization of this algorithm is presented to deal with the full molecular replacement problem, that is, with the problem of determining the orientations and positions of a total of \( n \) copies of \( m \) different models (with \( n \geq m \)) which are assumed to be present in the asymmetric unit of a target crystal structure. The generality of this approach is illustrated through its successful application to a 17-dimensional 3-model problem involving one DNA and two protein molecules.

1. Introduction

Traditional molecular replacement methods (Rossmann, 1990; Navaza, 1994; Navaza & Saludjian, 1997) and the newer six-dimensional searches (Chang & Lewis, 1997; Kissinger et al., 1999, 2001; Jamrog et al., 2003) use a ‘divide-and-conquer’ approach to tackle molecular replacement problems. Nevertheless, and as the success of these methods convincingly demonstrates, the heuristic lying behind those approaches (namely, the properties of the Patterson function and its equivalence to the measured crystallographic data) is so powerful that treating the molecular replacement problem in its full dimensionality is more often than not a waste of computational resources. It is only when the assumptions behind those methods break, that a full multi-dimensional molecular replacement search could provide a useful alternative. We have previously described an algorithm (and its implementation in the form of a computer program) for a stochastic multi-dimensional molecular replacement search (Glykos & Kokkinidis, 2000; Glykos & Kokkinidis, 2001; hereinafter referred to as GK0 and GK1, respectively) and have recently applied it to the determination of a previously unknown structure via a 23-dimensional 4-body search (Glykos & Kokkinidis, 2003). Here we report an extension of this algorithm (and of its corresponding computer program) to tackle the problem of simultaneously and independently placing an arbitrary number of different search models in the asymmetric unit of a target crystal structure.

2. Formulation of the problem

The formulation of the problem follows closely its original derivation (GK0, GK1) and is as follows. If the asymmetric unit of a target crystal structure with \( s \) crystallographic symmetry operators comprises \( m \) different search models, each of which is present in \( c_m \) copies, with each model \( m \) consisting of \( i_m \) atoms, then the structure factor corresponding to the Bragg reflection \( h \) (and omitting for brevity the required orthogonalization and de-orthogonalization matrices) is

\[
F_{h} = \sum_{s} \sum_{m} \sum_{i_m} f_{m,i_m,h} \exp[-B_m \sin \theta_h / \lambda^2] \
\times \exp[2\pi i h (R_{m,c_m} x_{i_m} + T_{m,c_m}) + T_s],
\]

where, \( f_{m,i_m} \) is the atomic scattering factor of the atom \( i_m \) of the model \( m \), \( B_m \) is an overall temperature factor assigned to model \( m \), \( \theta_h \) is the Bragg angle corresponding to the reflection \( h \), \( R_s \) and \( T_s \) are the rotation and translation matrices corresponding to the crystallographic symmetry operator \( s \), \( x_{i_m} \) are the orthonormal coordinates of the atom \( i_m \) of the model \( m \) in a standard (fixed) molecular reference frame, and finally, \( R_{m,c_m} \) and \( T_{m,c_m} \) are the rotation and translation matrices describing the orientation and position of the \( c_m \) copy of the \( m \) model in the orthonormal frame.

Given a set of trial \( R_{m,c_m} \) and \( T_{m,c_m} \) (and possibly after correcting \( F_s \) values with an overall scale and temperature factor), we can calculate the value of a target function (in this case the linear correlation coefficient between the observed and calculated structure-factor amplitudes) as follows:
the same molecule. In other words, if there is a total of for each and every of the search models, even if these correspond to considerably different size (or proportions) one or more of the schedule, move-size control and other implementation details have been extensively discussed in §2 and 3 of GK1. Here we will only discuss features and limitations of the program that arose as a result of implementing support for multiple different models. As previously, we traded physical memory requirements for speed of execution by calculating the molecular transforms of all models on the same grid (see §2.2 of GK1). The implication of this choice is that for models of considerably different size (or proportions) one or more of the molecular transforms may by unnecessarily over-sampled. The advantage is that the inner loop of calculating the structure factors of the current trial structure (which is where the program spends more than 85% of its time) can proceed via one and the same interpolation function. A second limitation arising for similar reasons is that in the multi-model mode the program will calculate a molecular transform for each and every of the search models, even if these correspond to the same molecule. In other words, if there is a total of \( n \) copies of \( m \) different models, the current implementation of the program will calculate and store in memory \( n \) (and not \( m \)) molecular transforms. For \( n \gg m \) this choice can become rather wasteful with respect to physical memory resources. One other important difference from previous versions of the program concerns the annealing schedule and target function selected by default. This version uses a Boltzmann annealing schedule with a fixed (resolution-dependent) move-size, a fixed starting temperature, and a target function corresponding to the unweighted linear correlation coefficient between the observed and calculated structure-factor amplitudes. These choices bring the program more in-line with the current thinking in the field, reflect the strong theoretical foundation of Boltzmann annealing and of the correlation-based targets, and, finally, are identical to the choices made in all successful high-dimensionality applications of previous versions of the program.

### Figure 1
Evolution of the average values of the target function \([1.0 - \text{Corr}(F_c, F_o)]\) versus Monte Carlo moves for the three successful *Queen of Spades* minimizations. To minimize overlap, successive graphs have been translated by 0.1 unit along y, and by five million Monte Carlo steps along x. Each graph includes both the target function (foreground, dark colour) and its cross-validated value (background, light colour).

### Table 1
Final statistics for the five independent *Queen of Spades* minimizations.

<table>
<thead>
<tr>
<th>Minimization</th>
<th>(1.0 - \text{Corr}(F_c, F_o))</th>
<th>Free value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5215</td>
<td>0.5061</td>
</tr>
<tr>
<td>2</td>
<td>0.5641</td>
<td>0.5639</td>
</tr>
<tr>
<td>3</td>
<td>0.5215</td>
<td>0.5092</td>
</tr>
<tr>
<td>4</td>
<td>0.5600</td>
<td>0.5707</td>
</tr>
<tr>
<td>5</td>
<td>0.5210</td>
<td>0.5190</td>
</tr>
</tbody>
</table>

where \( F_{o,h} \) and \( F_{c,h} \) are the observed and calculated structure-factor amplitudes of the reflection \( h \), \( \bar{F}_c \), and \( \bar{F}_o \) are their respective means, and the sums are taken over all Miller indices (\( h \)).

The aim of our algorithm is the unconditional global optimization of \( C \) as a function of \( R_{m,c} \) and \( T_{m,c} \). The matrices \( R_{m,c} \) and \( T_{m,c} \) for which the value of the target function is optimized are the solution of our method.

### 3. Method of solution and implementation
The unconditional global optimization of the target function is based on a modified reverse Monte Carlo algorithm (McGreery & Pusztai, 1988; Keen & McGreevy, 1990). A complete account of the method has been presented previously (see §2.1 of GK0 and §§2.2–2.3 of GK1) and will not be repeated here.

The algorithm was implemented by modifying the program *Queen of Spades* (see §6 for program availability details). The annealing schedules, move-size control and other implementation details have been extensively discussed in §§2 and 3 of GK1. Here we will only discuss features and limitations of the program that arose as a result of implementing support for multiple different models. As previously, we traded physical memory requirements for speed of execution by calculating the molecular transforms of all models on the same grid (see §2.2 of GK1). The implication of this choice is that for models of considerably different size (or proportions) one or more of the molecular transforms may by unnecessarily over-sampled. The advantage is that the inner loop of calculating the structure factors of the current trial structure (which is where the program spends more than 85% of its time) can proceed via one and the same interpolation function. A second limitation arising for similar reasons is that in the multi-model mode the program will calculate a molecular transform for each and every of the search models, even if these correspond to the same molecule. In other words, if there is a total of \( n \) copies of \( m \) different models, the current implementation of the program will calculate and store in memory \( n \) (and not \( m \)) molecular transforms. For \( n \gg m \) this choice can become rather wasteful with respect to physical memory resources. One other important difference from previous versions of the program concerns the annealing schedule and target function selected by default. This version uses a Boltzmann annealing schedule with a fixed (resolution-dependent) move-size, a fixed starting temperature, and a target function corresponding to the unweighted linear correlation coefficient between the observed and calculated structure-factor amplitudes. These choices bring the program more in-line with the current thinking in the field, reflect the strong theoretical foundation of Boltzmann annealing and of the correlation-based targets, and, finally, are identical to the choices made in all successful high-dimensionality applications of previous versions of the program.

### 4. Application to a 17-dimensional 3-model problem
The structure chosen to illustrate the application of the program is the complex of NFAT, Fos and Jun with DNA, corresponding to the PDB entry 1A02 (Chen et al., 1998). To reduce the dimensionality of the problem we have treated Fos and Jun as one (rigid) search model, leaving us with three models (NFAT, Fos-Jun and the DNA molecule). We used real data deposited with the PDB (entry r1a02s.cent), and to make the example more realistic we modified the deposited coordinates by subjecting (without experimental restraints) the starting models to energy minimization with *CNS* (Brünger et al., 1998) for 500 cycles. The resulting models deviated from the deposited structures with r.m.s. deviations of 1.1, 1.5 and 2.2 Å for the NFAT, Fos-Jun and DNA, respectively, and were used as search models in the ensuing molecular replacement calculations. The *Queen of Spades* run was performed in its fully automatic mode (simply by entering \( \text{Qs-} \text{auto } 3 \)), requiring only the PDB files containing the search models and a free-format ASCII file containing the observed data that the program should use for the calculation (all data between 19.5 and 4 Å resolution for this example). In this default mode, the program performed five independent minimizations, each lasting 30 million Monte Carlo moves and taking approximately 73 h of CPU time on a personal computer equipped with a 1.8 GHz Pentium IV processor, 1 GByte of random access memory and a proper operating system (GNU/Linux, RedHat distribution v.7.3). The total physical memory requirements of the program amounted to 196 MBytes (a significant proportion of which corresponds to high resolution volumes of the molecular transforms and, thus, does not have to be resident in memory). All minimizations used the strongest 70% of all reflections with \( \text{Fo}(F) > 2.0 \) (4943 reflections in total), with 10% of these being reserved for statistical cross-validation (Brünger, 1997). The target function for the minimization was \( [1.0 - \text{Corr}(F_c, F_o)] \), where \( \text{Corr}(F_c, F_o) \) is the linear correlation coefficient between the observed and calculated struc-
problem. The method uses all information available at hand for a given problem (both with respect to the number and type of the search models, but also with respect to the measured crystallographic data) without resorting to Patterson-function-based heuristics. The algorithm only demands that the true crystal structure is the one for which the agreement between the observed and calculated structure-factor amplitudes is maximized. As usually is the case, this generality of treatment comes at a very high computational cost. The calculations described in section §4 could have been performed equally well with classical molecular replacement methods at a computational cost three orders of magnitude lower than the one reported here. To reiterate what has already been said, it is only when the assumptions behind the traditional methods break, that this multi-dimensional multi-model molecular replacement search could provide a useful alternative.

6. Program and data availability

The program Queen of Spades is free open-source software which is immediately available for download via http://origin.imbb.forth.gr/~glykos/ or from the various mirrors generously provided by the Collaborative Computational Project, Number 14. All files from the application of the program described in section §4 can be obtained via http://origin.imbb.forth.gr/~glykos/Qs_1A02.tar.gz.

We should like to thank Professor Jorge Navaza for his useful comments on the manuscript and for suggesting that it would be a worthwhile exercise writing the code for a multi-model mode of Qs.

References