Journal of Biomolecular NMR, **13:** 325–335, 1999.
KLUWER/ESCOM

Automated detection of problem restraints in NMR data sets using the FINGAR genetic algorithm method

David A. Pearlman
Vertex Pharmaceuticals Inc., 130 Waverly Street, Cambridge, MA 02139-4242, U.S.A.

Received 28 September 1998; Accepted 14 December 1998

Key words: error restraints, FK506, genetic algorithm, refinement

Abstract

The recently described FINGAR genetic algorithm method for NMR refinement [D.A. Pearlman (1996) *J. Biomol. NMR*, **8**, 67–76] has been extended so that it can be used to detect problem restraints in an NMR-derived set of data. A problem restraint is defined as a restraint in a generally well-behaved set where the associated target value is in error, due to inaccuracies in the data, misassignment, etc. The method described here, FINGAR.RWF, locates problem restraints by finding those restraints that, if removed from the data set, result in a disproportionate improvement in the scoring function. The method is applied to several test cases of simulated data, as well as to real data for the FK506 macrocycle, with excellent results.

Introduction

Typically, determination of a structure from NMR data proceeds through a three-step process (Wüthrich, 1990): data collection, followed by assignment of the data and conversion into specific structural restraints (distances, torsions), and finally generation of one or more three-dimensional structures from the restraints. In order to generate good, reliable 3D structures during the third phase (refinement), it is necessary that one have an accurate, properly assigned set of NMR-derived restraints. The availability of such a restraint set is critically dependent on steps one and two. Experimental errors during data collection can lead to inaccuracy in the distance or angle associated with a particular atom sequence. Misassignment can result in an NOE-derived distance being associated with the wrong pair of atoms. Inclusion of even a small number of improperly defined restraints can lead to refinement that either fails to converge, or else converges to a distorted structure (Zhao and Jardetzky, 1994; Adler, 1996). In practice, the initial restraint set produced in steps one and two will usually contain several such problem restraints, which need to be identified and either corrected or removed from the data set (by reanalyzing data from steps one and two) before structure generation can properly be completed. The rapid identification of these problem restraints during phase three is the focus of this work. The issues related to steps one and two have been reviewed elsewhere (Bax, 1989; Clore and Gronenborn, 1991; Hoch, 1991; Bain et al., 1994).

Traditionally, refinement has been performed using either distance geometry (Crippen and Havel, 1988) or a molecular dynamics (MD) based approach (Brunger and Karplus, 1991). In both cases, the goal is to produce a structure, or ensemble of structures, that best satisfies the restraint set while maintaining reasonable covalent and non-bonded interactions. Unfortunately, when problem restraints are included in the restraint set, it is frequently difficult or impossible to examine the results from refinement and determine which restraints are leading to poor results. Because numerous restraints can apply to the same region of a molecule, the effects of a bad restraint can be manifest as increased restraint violations among numerous restraints and can potentially spread far from the atoms that represent the actual problem. (Laskowski et al., 1996) NMR-derived structure determination therefore typically entails a trial-and-error process of examining the refined structures, guessing if any restraints are in error, removing or correcting these restraints, and

re-refining. When a reasonably low energy set of conformations can acceptably explain the restraint data, it is assumed that the restraint set is acceptable and the refinement cycle is terminated (Brunger and Nilges, 1993).

This procedure is unsatisfactory on three counts. First, it is time-consuming and inefficient. Secondly, it can be particularly difficult to pinpoint problem restraints when several are included. And finally, there is no absolute set of criteria against which the results of any particular refinement can be compared to determine if the set still contains problem restraints. For example, a seemingly well-refined structure can actually be moderately high in energy and distorted, even though the restraints appear reasonably well satisfied and there are no obvious geometric problems (Van Gunsteren, 1990).

For these reasons, it is desirable to have an automated objective method for helping to flag bad restraints during refinement. In essence, what is desired is a method that would check for restraints that – if removed from the refinement set – result in an inordinately large reduction in the energetic scoring function for the system. The method should be able to flag bad restraints even when several are present in the data set. Standard distance geometry and MD-based refinement methods do not easily lend themselves to this type of analysis. In principle, one could systematically remove all possible combinations of restraints, rerefine for each reduced data set, and compare the results. But a brute force approach such as this quickly becomes prohibitively time-consuming, especially if several bad restraints are included in the data set.

Recently, we described a powerful new genetic algorithm (GA) based method, FINGAR, for refinement of structures from NMR-derived data (Pearlman, 1996). The essence of FINGAR is that a basis set of potential molecular conformations is generated, and FINGAR refines the relative weights of these conformations in order to optimize the following function

$$E_{tot} = K_{pot} < E_{pot} > + K_{bond} E_{bond} + K_J E_J + K_{\sigma E} E_{\sigma E} \quad (1)$$

$<E_{pot}>$ is the weighted average of the energy over the ensemble, E_{bond} is derived from the differences between the weighted average distances in the ensemble and the target NOE-derived distances, E_J is derived from the differences between the weighted average J-coupling values for the ensemble and the target NMR-derived J-coupling values, and $E_{\sigma E}$ is a term that ensures the rms fluctuation of the weighted averaged potential energy is physically reasonable. The ensemble of conformations that serve as a basis set can be generated using distance geometry or unrestrained or restrained molecular dynamics at the relevant temperature.

Key advantages of the FINGAR approach over traditional NMR refinement methods include: FINGAR can fit the NMR data without introducing restraint-based distortions into the molecule; FINGAR easily allows for systems where the NMR data reflects multiple conformations; the results from FINGAR are a set of individual basis set weights which enumerate exactly which conformations are most important; and FINGAR is very fast. The superiority of FINGAR refinement to traditional refinement for several small molecule test cases has been demonstrated (Pearlman, 1996).

These advantages and features coupled with the intrinsic strength of the genetic algorithm as an optimizing technique suggest that FINGAR might be extended to also address the issue of flagging bad restraints. To extend FINGAR in this direction, we have introduced a new set of variables into refinement. These are the relative weights, $K_{x(i)}$, of the individual restraint terms contributing to E_{tot}. In doing so, we add NREST additional variables into refinement, corresponding to the NREST NMR-derived restraints. The contribution to E_{tot} from restraint i is given by

$$E_{rest} = K_{x(i)}(< x(i) > - x_{NMR}(i))^2 \quad (2)$$

where $<x(i)>$ is the weighted averaged value of the parameter (r_{NOE}, ^3J) for restraint i over the basis set ensemble, and $x_{NMR}(i)$ is the NMR-derived target value for this restraint.

Clearly, if no other terms are added to E_{tot} then all values of $K_{x(i)}$ will tend to zero, since this will minimize the otherwise positive E_{rest}. To counteract this effect, so that only a minimal number of $K_{x(i)}$ substantially decrease, additional terms are added to E_{tot} which disfavor reductions in the $\{K_{x(i)}\}$ unless these reductions result in a substantial improvement in the overall fitness function E_{tot}. We term the resulting procedure, where both the basis set weights $w(i)$ and the restraint weights $K_{x(i)}$ can vary, FINGAR.RWF for 'FIt NMR data using a Genetic Algorithm with Restraint Weight Floating'.

FINGAR.RWF has been applied to several sets of synthesized test data. The program has also been applied to the refinement of FK506 using an experimentally derived data set (Lepre et al., 1992). FK506 is a 107 atom macrocyclic inhibitor of FKBP-12 (Rosen

and Schreiber, 1992). As will be seen, the method is very capable of flagging problem restraints, even when a data set contains several such restraints or is relatively large.

Methods

Refinements were performed using a version of the FINGAR program modified to allow restraint weight floating, FINGAR.RWF. FINGAR has been described previously (Pearlman, 1996). Briefly, FINGAR refines the relative weights, $w(i)$, for a set of basis structures using a genetic algorithm to obtain a minimum value for the fitness function in Equation 1. The various terms in the fitness function are calculated as:

$$< E_{pot} >= \sum_{NBASIS} w(i) E_{pot}(i) \tag{3}$$

$$E_{bond} = \sum_{NRESTr} K_{r(i)} (< r(n) > - r_{NMR}(n))^2 \tag{4}$$

$$E_J = \sum_{NRESTj} K_{J(i)} (< J(n) > - J_{NMR}(n))^2 \tag{5}$$

$$E_{\sigma E} = (\sigma(E_{pot}) - \sigma(E_{pot0}))^2 \tag{6}$$

$E_{pot}(i)$ is the potential energy of basis set (i). *NBASIS* is the number of basis sets. *NRESTr* and *NRESTj* are the numbers of NMR-derived distance and torsional restraints, respectively. $K_{r(i)}$ is the force constant for the *i*th distance terms. $K_{J(i)}$ is the force constant for the *i*th *J*-coupling term. $< r(n) >$ is the weighted average value of restraint *n* over all basis sets *i*:

$$< r(n) >= (\sum_{NBASIS} w(i) r(i,n)^{-N})^{-1/N} \tag{7a}$$

where $r(i,n)$ is the value of distance restraint *n* for basis set *i*. *N* is typically 6. The reciprocal weighting reflects the relationship between the measured NOE-derived distance $r_{NMR}(n)$ and the experimental ensemble of distances contributing to $r_{NMR}(n)$ (Wagner and Wüthrich, 1979; Tropp, 1980). $< J(n) >$ is the weighted average value of *J*-coupling restraint *n* over all basis sets *i*:

$$< J(n) >= \sum_{NBASIS} w(i) J(i,n) \tag{7b}$$

$J(i,n)$ is the value of the *n*th restrained *J* for basis set *i*. $J(i,n)$ can be related to the underlying torsion angle τ through a Karplus relationship (Karplus, 1959). $J_{NMR}(n)$ is the experimentally measured ^3J coupling constant. The variables in FINGAR refinement are the basis set weights $w(i)$.

FINGAR.RWF differs from FINGAR in that we also allow the weight factors for the individual restraints to vary during refinement. That is, $K_{r(i)}$ and $K_{J(i)}$ in Equations 4 and 5 are made additional variables of the refinement. The expressions in Equations 4 and 5 are always ≥ 0, so if the values $K_{r(i)}$ and $K_{J(i)}$ were simply allowed to float, they would al1 tend towards 0. To counteract this and instead focus only on potentially 'problem' restraints, two terms are added to the expression in Equation 1:

$$E_{ksum} = E_{ksum} \sum_{NREST} (K_{x(i)}/K_{x0(i)} - 1)^2 \tag{8}$$

ensures that the total number of restraints with reduced values of $K_{x(i)}$ is modest. $K_{x(i)}$ represents the set of force constants $\{K_{r(i)}, K_{J(i)}\}$, and *NREST* represents the sum of *NRESTr* and *NRESTj*. $K_{x0(i)}$ is the starting (input) value of $K_{x(i)}$. K_{ksum} is a weighting coefficient. The second term,

$$E_{kzer} = \begin{cases} K_{kzer}((NREST - \sum_{NREST} K_{x(i)}/K_{x0(i)}) - S_0)^2 \\ \qquad \text{for } NREST - \sum_{NREST} K_{x(i)}/K_{x0(i)} > S_0 \\ 0 \qquad \text{for } NREST - \sum_{NREST} K_{x(i)}/K_{x0(i)} \leq S_0 \end{cases} \tag{9}$$

ensures that the number of force constants $K_{x(i)}$ that tend towards zero will be no greater than roughly S_0. If no force constants are reduced, i.e. $K_{x(i)} = K_{x0(i)}$ for all restraints, then $\sum_{\text{NREST}} K_{x(i)} = K_{x0(i)} = NREST$. Thus, $NREST - \sum_{NREST} K_{x(i)}/K_{x0(i)}$ gives a rough approximation of the number of restraints whose force constants have appreciably decreased during refinement. When this number is greater than S_0, the harmonic penalty function is applied. The user specifies K_{ksum}, K_{kzer} and S_0. The effects of the terms in Equations 8 and 9 are complementary. The total fitness function is then given by

$$\begin{aligned} E_{tot} \;=\; & K_{pot} < E_{pot} > + K_{bond} E_{bond} + K_J E_J \\ & + K_{\sigma E} E_{\sigma E} + E_{ksum} + E_{kzer} \end{aligned} \tag{10}$$

Changes in the restraint weights can be performed either continuously, or intermittently and either concurrent with changes in the basis weights or in a leapfrog fashion with those changes. In the simulations reported here, every 30 consecutive GA generations consisted of 15 generations where only the basis set

weights changed followed by 15 generations where both basis set weights and restraint weights changed. We have found this type of leap-frogging with respect to the restraint weights is more efficient and less prone to getting stuck in local minima than refinement where both basis set and restraint weights vary at every step.

Two types of tests have been performed to characterize the performance of FINGAR.RWF in flagging problem restraints. In the first, the 'exact test', refinement is performed against a set of restraints which are generated from a randomly assigned set of weights and a fabricated set of basis sets. For the 'exact test', we begin by generating a series of basis sets. A basis set is defined by the values for the restrained variables of that set (e.g. a list of distances and ^3J coupling constants). Here, values were randomly assigned from the interval [2.5,5.5]Å to represent restrained distances (for simplicity, no J-coupling restraints were used in these tests). To each member of the simulated basis set, a weight is then randomly assigned. These are initially chosen on the interval [0,1], and then 95% of the basis set weights are reduced by a random factor between 10 and 200. After assignment, the sum of weights is normalized to 1.0. This approximates real data, where the ensemble is typically dominated by a small subset of the conformations in the basis set. Potential energies are assigned to each basis set, based on the Boltzmann factor that would result in the assigned weight. A modest amount of noise is then introduced to the synthesized restraint set, again to simulate real data. To introduce this noise, the target value of each restraint is translated by a distance T_{res} chosen randomly from a Gaussian distribution centered on a specified value T_{ne} with a standard deviation of σ_{ne}. A 'flat well' region is allowed for each restraint, $\pm T_{fw}$, where T_{fw} is a specified fraction of the standard deviation in the restraint σ_r calculated from the weighted distribution of basis sets. Finally, a specified number of restraints, N_{err}, are chosen at random from the restraint set and these are translated by a distance $+T_{err}$ or $-T_{err}$. These are the 'problem' restraints, which the simulation will attempt to identify. Including the errors, the restraints are still required to fall within the interval [2.5,5.5]Å. If the error introduced to a restraint would place the target value of the restraint outside this range, the sign of the error is reversed before addition.

The second set of simulations run to characterize FINGAR.RWF were of the 'experimental test' variety. In these, a set of experimentally derived restraints for the FK506 macrocycle are used. The basis set is generated using both unrestrained and restrained molecular dynamics, followed by clustering, as previously described (Pearlman, 1996). Minimization and dynamics were performed using the program Amber/Sander, version 4.1 (Pearlman et al., 1995), and energies were evaluated using a previously described force field (Pearlman, 1994a,b). N_{err} restraints are chosen at random from the restraint set and the upper bound for each of these is reduced by the specified distance T_{err} (subject again to a minimum distance of 2.5 Å). Note that the experimental data set used is characterized by restraints that only have an upper bound specified (resulting in a half-harmonic restraint potential), which is why the restraint distance is always reduced in this case.

Each test simulation consists of a suite of 100 FINGAR.RWF runs. For the 'exact test' simulations, each run corresponds to a completely new set of simulated basis sets and restraints. For the 'experimental test' simulations, each run corresponds to a different randomly chosen set of restraints to which errors are applied. The initial values $\{K_{x0(i)}\}$ are all set to 1.0 kcal/mol. At the end of each run in a suite, the refined values of $\{K_{x(i)}\}$ are examined. Any restraint with $K_{x(i)} \leq 0.70$ kcal/mol (the tag threshold, K_{tag}) is tagged as a potential problem restraint. This value of K_{tag} was empirically chosen to optimize the correct flagging of bad restraints while minimizing the improper tag rate. The list of tagged restraints is compared to actual list of error restraints. The numbers of properly tagged error restraints and improperly tagged non-error restraints are stored. At the end of the 100 run suite, the averaged numbers of properly tagged error and improperly tagged correct restraints are reported.

All GA simulations have been run with a population of 500 members and for 180 generations. For each, $K_{pot} = 1.0$, $K_{bond} = 25.0$, and $K_{\sigma E} = 1.0$. For each GA mutation phase, 0.50% of the codons in the system (basis set weight and/or restraint weight variables), randomly chosen, are allowed to mutate. Changes in variables are restricted to multiples of 0.001. During replication, members of the population are replicated based on Boltzmann factors evaluated using Equation 10 at $T = 300$ K, and a maximum of 20% of the system can be replicated from any single parent. Each FINGAR refinement simulation takes approximately 2.4 min on an SGI R10000 workstation.

To compare the FINGAR.RWF approach to traditional MD-based refinement approaches in flagging problem restraints, a series of MD simulations has also been performed for FK506 using the experimentally-

derived 66 NOE restraint set. By direct analogy to the FINGAR.RWF runs, a set of $N_{err} = 5$ restraints are randomly selected for each simulation and errors of magnitude T_{err}. are applied to these restraints. Each MD test consists of a suite of 100 individual 400 ps simulations with different randomly selected sets of $N_{err} = 5$ restraints to which errors havc been applied. At the end of each MD run, those restraints whose contribution to the total energy is ≥ 1.0 kcal/mol are tagged as bad. The numbers of properly and improperly tagged restraints are averaged over all 100 independent MD simulations in a suite to give averaged tag rates for that suite. Each MD simulation consists of 400 ps of sampling at a temperature of 300 K, in vacuo, using a timestep of 2 fs. The force constant K_r used in these simulations is 20 kcal/mol. MD test suites have been generated for a variety of values of T_{err}, again by analogy to the 'experimental test' FINGAR.RWF tests. All test suites have been performed twice, using each of two refinement protocols: standard refinement, with restraint penalties of the form

$$E_{distance} = \sum_{measured\ NOEs} K_r (r_{MD} - r_{NOE})^2 \qquad (11)$$

and time-averaged refinement (Torda et al., 1989; Pearlman and Kollman, 1991), with restraint penalties of the form

$$E_{distance} = \sum_{measured\ NOEs} K_r (< r_{MD}^{-6} >^{-1/6} - r_{NOE})^2 \qquad (12)$$

r_{MD} is the value of the restrained NOE distance at each step of MD, and the '<>' notation indicates we determine the time-averaged value of the quantity enclosed by these brackets over an ensemble of steps during the MD simulation. An exponential decay function (Torda et al., 1990) with a time-constant τ of 10 ps is used so that the average in Equation 12 is weighted towards the more recent MD steps.

Results

As an initial test of the FINGAR.RWF method, a series of 'exact test' simulations were performed where the magnitude of the error in the problem restraints was varied. Eight separate sets of 100 runs were performed corresponding to error in the problem restraints of magnitude $T_{err} = 0.25 \rightarrow 2.00$ Å, in 0.25 Å

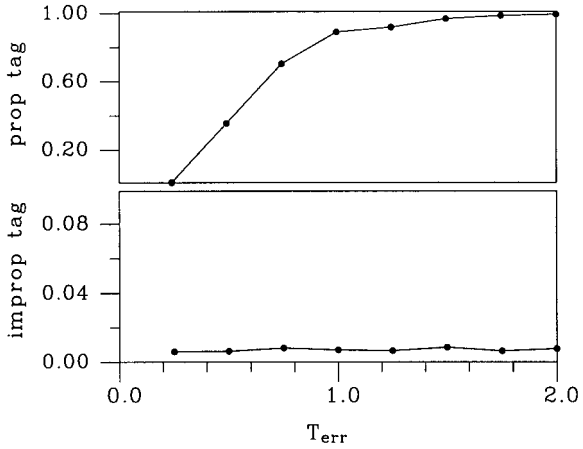

Figure 1. Ability of FINGAR.RWF to flag problem restraints, as a function of the magnitude (Å) of the error in those restraints, T_{err}. Results are for 'exact test' simulations on synthesized data (see text). The curve in the upper frame represents the fractions of the problem restraints tagged ($K_{r(i)} \leq 0.70$ kcal/mol) during the refinement. The curve in the bottom frame represent the fractions of the good restraints incorrectly tagged as problem restraints. All data are averages over 100 separate simulations with different synthesized data and different randomly chosen problem restraints. $K_{kzer} = 0$ for these simulations. All simulations were performed with 100 basis sets, 100 distance restraints, $N_{err} = 5$, and $K_{ksum} = 3.0$ kcal/mol.

increments (and either + or − sign). 100 fabricated basis sets and restraints were used, and the number of randomly chosen problem restraints, N_{err} was 5. For the remaining restraints, the magnitude of the error ('noise') in the target value, T_{res}, was chosen randomly from a Gaussian distribution centered on 0.5 σ_i with a standard deviation of 0.5 σ_i, subject to a maximum of $T_{res} = 0.5$ Å. T_{res} was randomly assigned either a + or − sign after the magnitude was chosen. For each restraint a 'flat well' region of $\pm T_{fw} = 0.5\ \sigma$ was used. Thus, a realistic set of data is simulated, where even the non-problem restraints will contain a modest amount of error and will be associated with a flat-welled restraint that reflects this error. For these simulations, K_{ksum} was set to 3.0 kcal/mol, and K_{kzer} was set to 0 (Equations 8–9). Double-sided wells were used for all restraints.

The results of these simulations are presented in Figure 1. Expectedly, the ability to tag the problem restraints increases with the amount of error in these restraints. For small values of T_{err}, the error in the problem restraints is comparable to the noise in the remaining restraints, which accounts for the relative inability to identify the problems. But the tag rate increases rapidly with the magnitude of the error, and

when the error is 1.0 Å, the tag rate is well over 80%. As T_{err} approaches 2.0 Å, the tag rate is essentially 100%. Regardless of the value of T_{err}, the rate at which non-problem restraints are incorrectly tagged remains an acceptably low < 1%.

Having demonstrated FINGAR.RWF quite capable of tagging problem restraints, two additional sets of simulations were performed to determine how the program's ability to flag these restraints varies with the size of the data set and with the number of problem restraints in the system. Conditions for these simulation series were as described above, with these changes: (1) The magnitude of the error in the problem restraints was fixed at 1.0 Å; (2) Ten separate simulations were performed corresponding to, respectively, $N_{err} = 1, 2, 3, 4, 5, 8, 10, 13, 16$ and 20 randomly chosen 'error restraints'; (3) Two simulation series were performed. In the first, the number of fabricated basis sets/restraints was 100, as before. In the second, the number of fabricated basis sets/restraints was 400. As in the previous simulations, a σ-dependent amount of error was added to the remaining (non-problem) restraints to reflect the error inherent in all restraints of a real data set.

The results of these two series of simulations are presented in Figure 2. The thin solid lines in the upper and lower frames represent the average fractions of problem restraints properly tagged and non-problem restraints improperly tagged, respectively, during the refinement series where the numbers of basis sets and restraints were 100. The dashed lines in the upper and lower frames represent the analogous results for refinement series where the numbers of basis sets and restraints were 400. As can be seen, for both simulation sets, there is only a very modest decrease in the ability of FINGAR.RWF to tag the bad restraints as the number of these restraints increases. The tagging rate remains above 80% in all cases. The tagging rate is actually higher for the larger data set simulations, probably reflecting the fact that for these simulations, there is more total data defining the correct answer. At any rate, FINGAR.RWF is certainly as capable of finding problem restraints in large data sets as small. The fractions of problem restraints mis-flagged are almost identical for the 100 and 400 basis set/restraint systems. In both cases, the fraction remains below 1.5% for all values of N_{err}. There is a small increase in the improper tag rate with N_{err}.

The ability of FINGAR to flag problem restraints can be made even better by decreasing K_{ksum}, the weighting constant for the term that acts to minimize

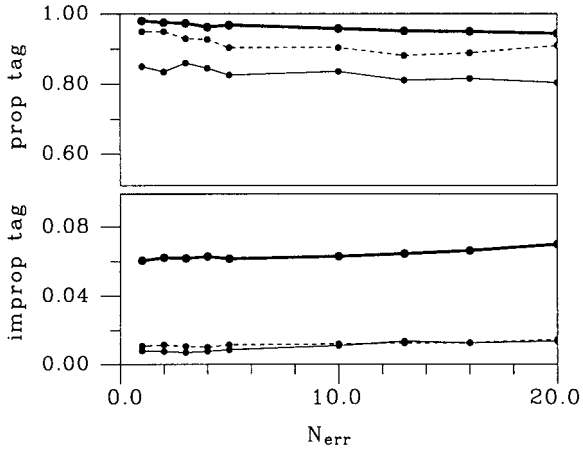

Figure 2. Ability of FINGAR.RWF to flag problem restraints, as a function of the number of randomly chosen problem restraints ($T_{err} = -1.0$ Å) in the data set. Results are for 'exact test' simulations on synthesized data (see text). The curves in the upper frame represent the fractions of the problem restraints tagged ($K_{r(i)} \leq 0.70$ kcal/mol) during the refinement. The curves in the bottom frame represent the fractions of the good restraints incorrectly tagged as problem restraints. All data are averages over 100 separate simulations with different synthesized data and different randomly chosen problem restraints. $K_{kzer} = 0$ for these simulations. Thin solid lines: runs with 100 basis sets, 100 distance restraints and $K_{ksum} = 3.0$ kcal/mol; thin dashed lines: runs with 400 basis sets, 400 distance restraints and $K_{ksum} = 3.0$ kcal/mol; thick solid lines: runs with 400 basis sets, 400 distance restraints and $K_{ksum} = 1.0$ kcal/mol.

the number of tagged (weight reduced) restraints. The thick solid curves in the upper and lower frames of Figure 2 correspond to a simulation with $K_{ksum} = 1.0$ kcal/mol and 400 fabricated basis sets/restraints. By lowering K_{ksum}, we have been able to increase the tagging rate for bad restraints by several percent, to well over 90%. The cost of this is that the percentage of restraints improperly tagged has increased to 6% from around 1%.

Having demonstrated that the method can very effectively tag problem restraints for a simulated data set, a second set of simulations was then carried out to determine how the method performs with a 'real' data set. The 'experimental test' protocol described above was used. The restraint set consisted of 66 properly assigned, low error distances derived for FK506 (Lepre et al., 1992). Each restraint was defined by only an upper distance bound. For each simulation, restraints from this set were selected at random, and the target distances of these restraints were reduced by T_{err}. A total of 16 sets of 100 runs each were performed, corresponding to, respectively, $T_{err} = -0.25$ to -4.00 Å, in increments of 0.25 Å. By performing a series of

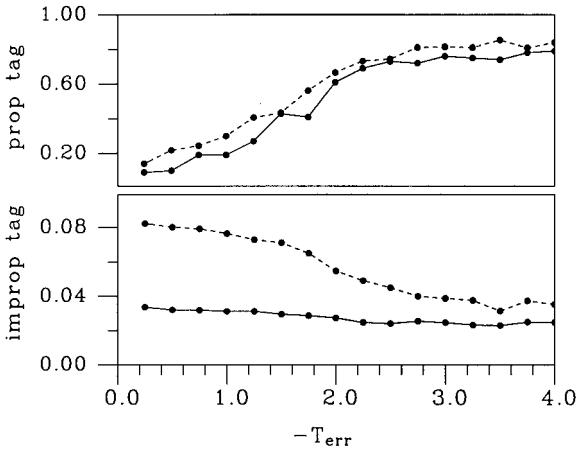

Figure 3. Ability of FINGAR.RWF to flag problem restraints, as a function of the magnitude of the error (Å) in the problem restraints. Results are for 'experimental test' simulations using an NMR-derived set of 66 distance restraints for the FK506 macrocycle. For each simulation a specified error $-T_{err}$ is applied to the target values of a randomly chosen set of restraints. The curves in the upper frame represent the fractions of the problem restraints properly tagged during the refinement ($K_{r(i)} \leq 0.70$ kcal/mol). The curves in the lower frame represent the fractions of the good restraints incorrectly tagged as problem restraints. All data are averages over 100 separate simulations with different randomly chosen problem restraints. $K_{ksum} = 12.0$ kcal/mol, $K_{kzer} = 10.0$ kcal/mol and $S_0 = N_{err}$ for these simulations. Solid lines: runs with the number of randomly chosen problem restraints $N_{err} = 1$; dashed lines: runs with $N_{err} = 5$.

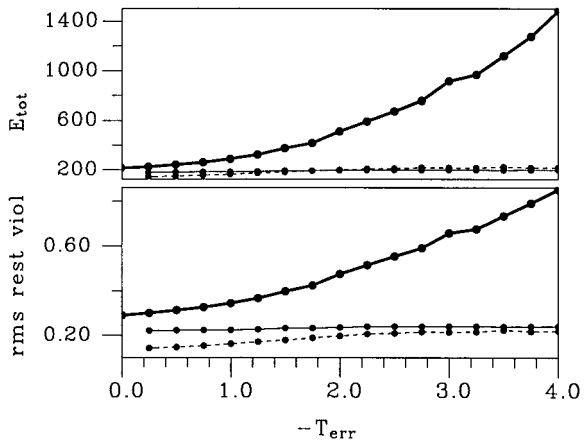

Figure 4. Ensemble averaged fitness function value E_{tot} (kcal/mol; upper frame) and rms restraint violations (Å; lower frame) as a function of the magnitude of the error in the problem restraints. Simulation conditions are the same as in Figure 2, and the thin solid and dashed lines represent data for the same simulations as in that figure. Thin solid lines: runs with the number of randomly chosen problem restraints $N_{err} = 1$; thin dashed lines: runs with $N_{err} = 5$; thick solid lines: runs where the restraint weights $\{K_{x(i)}\}$ were not allowed to change.

simulations with varying values of T_{err}, the ability to tag problem restraints can be correlated with the magnitude of the error. Two full arrays of 16 simulation sets were performed, one in which only one restraint was reduced ($N_{err} = 1$), and a second in which five restraints were reduced ($N_{err} = 5$). For both, S_0 was set to N_{err} (Equation 9), K_{ksum} was set to 12.0 kcal/mol, and K_{kzer} was set to 10.0 kcal/mol (values found, by trial and error, to yield reasonable results).

Figure 3 presents the results of the experimental test simulations. Data from the simulations with $N_{err} = 1$ are presented in thin lines, and those from $N_{err} = 5$ are presented in dashed lines. Again, the upper frame represents the fractions of problem restraints correctly tagged, while the bottom frame represents the fractions of non-problem restraints improperly tagged. Predictably, the ability to tag problem restraints increases substantially with the magnitude of the error in the problem restraints, reaching over 60% for restraints in error by 2.0 Å. At the same time, the tendency to mistakenly tag good restraints – while larger than in the exact case – remains modest (2–8%). While the positive tag rate is smaller here than in the exact test case above, this is to be expected. The restraints in this case are specified by only an upper bound. If some of the restraints are looser than they need to be (and certainly some will be), then we will observe no effect from reducing those restraints until the target value has been sufficiently reduced. If the 'true' restraint distance is near 2.5 Å or smaller, which is our self-imposed lower bound on a restraint, then of course we will never tag the restraint as a problem. An idea of the percentage of restraints that will never get tagged as problems can be ascertained by reference again to Figure 3. When T_{err} is 4.0 Å, any of the problem restraints will be set at $r_0 = 2.5$ Å. The fraction of hits in these cases is essentially the maximum percentage we can hope to find. As can be seen, this maximum is roughly 80%.

Figure 4 presents the averaged values of E_{tot} (Equation 10) and the average rms restraint violation after accounting for changes in the $K_{x(i)}$, for the refinements above (thin solid and dashed curves). These are compared with the analogous values in the case where no restraint weights are allowed to change (thick solid curves). The upper frame represents the total fitness function score (E_{tot}), while the bottom frame represents the rms restraint violation. As is seen, the total fitness score and restraint violations increase substantially with the magnitude of the errors when restraint weights are not allowed to change. When these re-

straint weights are allowed to float, however, we find that the averaged energy and restraint violations increase only modestly with the violations – evidence that the refinement is indeed isolating and removing (by reduction of $K_{x(i)}$) problem restraints. These plots also indicate that the less than 100% hit rate for problem restraints is not arising from a failure to identify restraints which negatively impact refinement. That is, the sub-100% hit rate is not due to inadequacies in the search algorithm. Rather, this sub-100% hit rate reflects the fact that there are certain problem restraints that can be left in the data set without appreciably changing the target function which is used to evaluate the quality of refinement.

Clearly, FINGAR.RWF can efficiently tag bad restraints. But how much better is FINGAR.RWF at pinpointing bad restraints than traditional MD refinement methods? Specifically, if we attempt to identify bad restraints in the traditional manner – by examining the restraint violations at the end of an MD simulation – how well will we do relative to FINGAR.RWF? To answer this question, suites of 100 independent MD simulations have been run for FK506 using the same set of 66 experimentally derived NOE restraints as before. Each suite corresponds to $N_{err} = 5$ randomly chosen restraints and a fixed value of T_{err} in the range [0.25,4.0]Å, in direct analogy with the FINGAR.RWF 'experimental test' simulations. Errors are applied to N_{err} randomly chosen restraints in each simulation. At the end of each refinement, restraints whose contributions to the energy are $\geq$1.0 kcal/mol are tagged as 'problems'. A restraint force constant K_r of 20.0 kcal/mol is used for all MD simulations. Other control parameters for the MD simulations are described under the Methods section.

Figure 5 presents the tagging efficiencies of the MD simulations as a function of T_{err}. This figure should be compared to the dashed line plots of Figure 3, which present the analogous data for the FINGAR.RWF runs with $N_{err} = 5$. In Figure 5, the solid lines represent data from standard MD refinement, and the dashed lines represent data from time-averaged refinement. It is immediately apparent that the proper tagging rate using MD is considerably lower (a maximum tagging rate of 0.6 versus a rate of 0.8 using FINGAR.RWF), while at the same time the improper tagging rate is roughly twice as large (note the different scales for the lower plots in Figures 3 and 5). In other words, MD-based tagging is considerably less reliable and efficient than FINGAR.RWF. In addition, if the tag energy threshold is increased to reduce the improper tag rate, the proper tag rate decreases even more quickly (not shown). For example, if we look at only the five largest contributions to the restraint penalty, the proper tag rate decreases to between 0.10 and 0.23, for values of T_{err} between 0.25 and 4.0 Å.

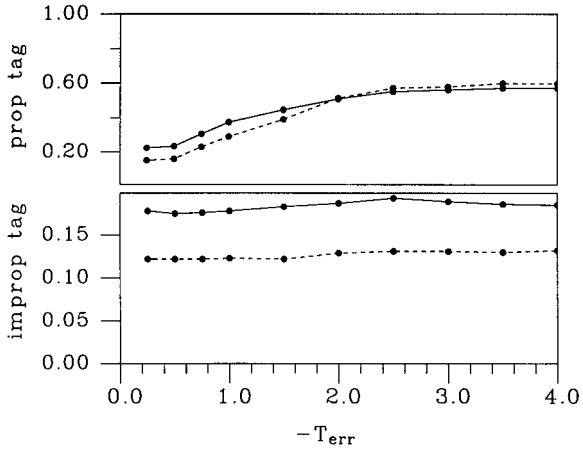

Figure 5. Ability of molecular dynamics (MD) to flag problem restraints, as a function of the magnitude of the error (Å) in the problem restraints. Results are for simulations using an NMR-derived set of 66 distance restraints for the FK506 macrocycle, and are analogous to the dashed-line simulations presented in Figure 3, but using MD. For each simulation a specified error $-T_{err}$ is applied to the target values of a randomly chosen set of $N_{err} = 5$ restraints. A restraint is tagged as 'bad' if its contribution to the restraint energy at the end of refinement is $\geq$1.0 kcal/mol. The curves in the upper frame represent the fractions of the problem restraints properly tagged during the refinement. The curves in the lower frame represent the fractions of the good restraints incorrectly tagged as problem restraints. All data are averages over 100 separate simulations with different randomly chosen problem restraints. $K_r = 20.0$ kcal/mol for these simulations. Solid lines: runs using standard refinement (Equation 11); dashed lines: run using time-averaged refinement (Equation 12).

It is interesting to ask how the presence of bad restraints affects the final averaged refined structures. To answer this question, averaged structures were generated for each simulation in both the MD and FINGAR.RWF suites. In the case of MD, the mean structure for each run was generated by first superimposing on the initial structure the snapshot conformation at every picosecond (discarding the first 40 ps, to allow for equilibration and thermalization) and then averaging. For FINGAR.RWF, the refined basis set weights at the end of each simulation were used to create a weighted archive file, then the structures in the archive file were superimposed on the the initial conformation and averaged. After each averaged structure was generated, it was superimposed on a target 'correct' averaged structure that had been determined from a

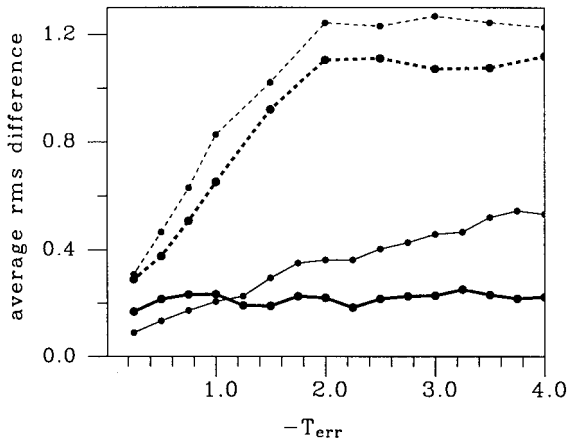

Figure 6. Ensemble averaged rms deviations (Å) from the 'correct' structure for FINGAR and MD-based refinements, as a function of magnitude (Å) of the error in the problem simulations. All refinements are of FK506 using the NMR-derived set of 66 distance restraints. Each plotted value corresponds to a suite of 100 separate simulations, each with a randomly chosen set of 5 restraints to which the given error $-T_{err}$ has been applied. The final average refined structure for each of the 100 simulations in a suite is determined and the rms deviation between this average and the 'correct' average (refined average with no errors applied to the restraint set) is calculated. The average of these 100 rms values gives the value that is plotted in Figure 6. Thin solid line: FINGAR refinement, no restraint weight floating; thick solid line: FINGAR.RWF refinement; thin dashed line: standard MD refinement; thick dashed line: time-averaged MD refinement.

refinement where no errors were introduced into the restraint set. The target structure for the MD refinements was generated from a MD refinement (standard or time-averaged, depending on the run), while the target structure for the FINGAR.RWF simulations was generated from a FINGAR refinement. The root-mean-squared (rms) difference between the averaged structure and the target structure was determined for each refinement in a suite, and these rms values were averaged to give an mean rms difference for that suite.

These mean rms differences are plotted in Figure 6. The thin solid line represents FINGAR runs where no restraint weight floating was allowed; the thick solid line represents FINGAR.RWF runs with restraint weight floating; the thin dashed line represents standard MD; and the thick dashed line represents time-averaged MD. From this plot it is clear that the MD refined structures are in appreciably worse agreement with the correct structures than are the FINGAR refined structures. It can also be seen that the deviations of the averaged structures from the FINGAR.RWF runs are essentially invariant with T_{err}. This is to be expected if the method is correctly identifying and effectively eliminating these restraints. The modest ~0.2 Å rms difference between the refined structures and the 'correct' structures reflects the fact that when we eliminate bad restraints during a single FINGAR.RWF run, they are not replaced by the correct ones. The replacement must be done by the user 'by hand'. Therefore, potentially valuable information has, in fact, been removed from the restraint set and the resulting structure may be somewhat different than if these restraints, without errors, had been included.

This same explanation applies to the question of why the FINGAR refinement without restraint weight floating actually results in somewhat lower rms deviations from the 'correct' structure for small values of T_{err}. Even a restraint that is somewhat in error can impart important information to the refinement. Obviously, if the error is too large, the effect on refinement will be detrimental. But if the error is small, it may be more valuable to keep the restraint than to simply remove it. Note that this is not a problem with the FINGAR.RWF refinement method; flagging restraints during FINGAR.RWF is a way of identifying bad restraints, not a way of generating the final solution. The user should intervene once the bad restraints have been tagged and fix or remove them, as warranted. Once the user is satisfied that all bad restraints have been fixed and/or permanently removed from the set, then final refinement without restraint weight floating can proceed.

Discussion

It is clear from the results present herein that FINGAR.RWF is very capable of flagging problem restraints in a dataset. For a well-defined set of double-sided restraints, the method is able to flag problem restraints with a success rate approaching 90% for errors of moderate magnitude (1.0 Å), and with a success rate of essentially 100% for larger errors (2.0 Å). For real data, using more error-forgiving single-sided restraints, the method is still quite able to find problem restraints, with the success rate proportional to the magnitude of the inherent error. The ability to flag problem restraints must be balanced with an attempt to minimize the number of correct restraints tagged as problematic. In all cases, the ability of FINGAR.RWF to reliably flag bad restraints is superior to that of traditional MD approaches.

By reducing K_{ksum} and K_{kzer} we can make it easier for refined restraint weights to decrease, thereby

increasing the chances of tagging problem restraints. But this will also increase the chances of improperly tagging restraints which are not in error: in the absence of the penalty functions defined by Equations 8 and 9, the natural tendency would be for all restraint weights to decrease, since the restraint term always makes a positive/unfavorable contribution to the total energy. The most appropriate values of K_{ksum}, K_{kzer} and S_0 will likely depend on the quality of the restraint set being examined. Improperly tagged restraints are not a problem, per se, but the user will have to examine both the correctly and incorrectly tagged restraints after the simulation. As the correct/incorrect ratio decreases, the amount of time required in this post simulation analysis increases.

It should be noted that the 'failure' of FINGAR.RWF to flag 100% of the problem restraints arises not from a failure of the GA global optimization method. Rather, this is due to the availability of solutions to the optimization that give equivalently low values for the target function E_{tot}, but that do not require the elimination of certain problem restraints. This can be seen by reference to Figure 4, where despite a 20–80% hit rate for problem restraints, depending on T_{err}, E_{tot} is nearly constant. In practice, this suggests that FINGAR.RWF should have no difficulty in isolating restraints that – if not removed – would have the greatest influence in misdirecting the results of the refinement.

The importance of FINGAR.RWF is in quickly identifying and correcting problem restraints in an experimentally-derived set of data. As such, it is not critically important that every bad restraint be identified in a single FINGAR.RWF simulation. Rather, it is expected that FINGAR.RWF may be run several times. In the first, the worst restraints will be found and corrected. Then the program can be re-run to determine if there are other bad restraints still in the data set. FINGAR.RWF can be re-run in such an iterative fashion until the user is satisfied the data set contains no restraints that are unduly affecting the results of the refinement. Once the user is satisfied that the restraint set is correct, final refinement is run using either FINGAR without restraint weight floating or MD.

The FINGAR.RWF method (and FINGAR itself) is general, and in principle should be applicable to any system. However, to user FINGAR, it is necessary that one be able to generate a representative set of basis structures containing all the important conformers. While this is generally a straightforward task for small molecules, it presents a more difficult problem for large molecules such as proteins. Additional work will likely be necessary to derive methods for generating the necessary basis sets in these cases. This caveat applies to determining of the structure of the large molecule itself. The structure of a small molecule bound to a larger, relatively fixed molecule (such as a ligand bound to a protein) should in general present no challenges beyond those of the system described herein.

It is interesting to consider the complementarity between the FINGAR.RWF approach and the recently described 'self correcting distance geometry' (SCDG) method (Hanggi and Braun, 1994). SCDG is an interactive approach to detecting inconsistent distance constraints during a distance geometry run. Each iteration of the SCDG approach consists of generating an ensemble of DG structures. Restraint violations in this ensemble are determined, and restraints that are violated a large percentage of the time are modified. The process is re-run several times until a restraint set is evolved that can be reasonably satisfied by a DG ensemble. This search for bad restraints is iterative and is not based on a more global optimization method like GA, so solutions far from the starting point (e.g. where several constraints that affect the same part of the molecule must be changed to achieve the proper answer) may be missed. Otherwise, SCDG is characterized by the same strengths and weaknesses as standard DG. Namely, DG is extremely fast and capable of generating structures that satisfy a set of distance constraints, but DG accounts for potential energy in only a very crude manner. Therefore, one is likely to identify misassigned constrants that are very geometrically inconsistent with the remainder of the data set, but not those that may be geometrically feasible but energetically unlikely. So the strength of SCDG will be in such tasks as cleaning up restraints derived from automated assignment or homology modeling (Mumenthaler and Braun, 1995a,b). Fine tuning the restraint set and finding restraints that are geometrically possible but energetically implausible will still need to be performed by a more sensitive energy-based method like FINGAR.RWF.

Ultimately, proof of the value of FINGAR.RWF must come from applications to real experimental data. In a recent study (Fejzo et al., 1999), FINGAR.RWF was used to analyze an experimentally derived NOE data set for protein-bound SLB506, an acyclic derivative of the FK506 macrocycle that binds to the FKBP protein. Initial refinement results in that study were unsatisfactory, as they yielded structures

with numerous modest restraint violations, and relatively large associated potential energies. Inspection of the results did not indicate any specific restraints that might be responsible for the unsatisfactory restraint violations and energies. FINGAR.RWF was applied and quickly identified the two misassigned restraints that were responsible for the problematic results. Once these restraints were corrected, refinement proceeded smoothly, yielding an excellent rms restraint violation along with a low potential energy.

In all, FINGAR.RWF appears to be a powerful new tool for decreasing the time and effort required to go from an initial set of NMR-derived distance assignments to a final refined structure.

Acknowledgements

Thanks to Dr Jasna Fejzo for helpful comments and to Dr Christopher Lepre for providing the original experimental FK506 data set used in performing some of the tests described.

References

Adler, M. (1996) *J. Biomol. NMR*, **8**, 404–416.
Bain, A.D., Burton, I.W. and Reynolds, W.F. (1994) *Prog. NMR Spectrosc.*, **26**, 59–89.
Bax, A. (1989) *Annu. Rev. Biochem.*, **58**, 223–256.
Brunger, A.T. and Karplus, M. (1991) *Acc. Chem. Res.*, **24**, 54–61.
Brunger, A.T. and Nilges, M. (1993) *Quart. Rev. Biophys.*, **26**, 49–125.
Clore, G.M. and Gronenborn, A.M. (1991) *Annu. Rev. Biophys. Biophys. Chem.*, **20**, 29–63.
Crippen, G.M. and Havel, T.F. (1988) *Distance Geometry and Molecular Conformation*, Wiley, New York, NY.
Fejzo, J., Pearlman, D.A., Morikawa, K. and Moore, J.M. (1999) *NMR studies of the structure of an acyclic variant of FK506 (SLB506) bound to FKBP-12*, to be submitted.
van Gunsteren, W.F. (1990) In *Modeling of Molecular Structures and Properties* (Ed., Rivail, J.-L.), Studies in Physical and Theoretical Chemistry, Vol. 71, Elsevier; Amsterdam, pp. 463–478.
Hanggi, G. and Braun, W. (1994) *FEBS Lett.*, **344**, 147–153.
Hoch, J.C. (1991) In *Computational Aspects of the Study of Biological Macromolecules by Nuclear Magnetic Resonance Spectroscopy* (Eds., Hoch, J.C., Poulsen, F.M. and Redfield, C.), Plenum Press, New York, NY, pp. 253–267.
Karplus, M. (1959) *J. Chem. Phys.*, **30**, 11–15.
Laskowski, R.A., Rullmann, J.A.C., MacArthur, M.W., Kaptein, R. and Thornton, J.M. (1996) *J. Biomol. NMR*, **8**, 477–486.
Lepre, C.A., Thomson, J.A. and Moore, J.M. (1992) *FEBS Lett.*, **302**, 89–96.
Mumenthaler, C. and Braun, W. (1995) *Protein Sci.*, **4**, 863–871.
Mumenthaler, C. and Braun, W. (1995) *J. Mol. Biol.*, **254**, 465–480.
Pearlman, D.A. (1996) *J. Biomol. NMR*, **8**, 49–66.
Pearlman, D.A., Case, D.A., Caldwell, J.C., Ross, W.S., Cheatham III, T.E., DeBolt, S., Ferguson, D.M., Seibel, G.L. and Kollman, P.A. (1995) *Comput. Phys. Commun.*, **91**, 1–41.
Pearlman, D.A. (1994a) *J. Biomol. NMR*, **4**, 1–16.
Pearlman, D.A. (1994b) *J. Biomol. NMR*, **4**, 279–299.
Pearlman, D.A. and Kollman, P.A. (1991) *J. Mol. Biol.*, **220**, 457–479.
Rosen, M.K. and Schreiber, S.L. (1992) *Ang. Chem., Int. Ed. Engl.*, **31**, 384–400.
Torda, A.E., Scheek, R.M. and van Gunsteren, W.F. (1989) *Chem. Phys. Lett.*, **157**, 289–294.
Torda, A.E., Scheek, R.M. and van Gunsteren, W.F. (1990) *J. Mol. Biol.*, **214**, 223–235.
Tropp, J. (1980) *J. Chem. Phys.*, **72**, 6035–6043.
Wagner, G. and Wüthrich, K. (1979) *J. Magn. Reson.*, **33**, 675–680.
Wüthrich, K. (1990) *J. Biol. Chem.*, **265**, 22059–22062.
Zhao, D. and Jardetzky, O. (1994) *J. Mol. Biol.*, **239**, 601–607.