DINOSAUR

Direct NOe Simulation Approach for Unbelievable structure Refinement

Alexandre Bonvin
Bijvoet Center for Biomolecular Research,
Utrecht University,
Padualaan 8,
3584 CH Utrecht,
The Netherlands
Telephone: 31 30 532652
Telefax: 31 30 537623

Version 2.0
December 28, 1993
1. Contents

1. Contents .................................................................................................................2

2. Introduction ...........................................................................................................6

3. Theory ....................................................................................................................8

  3.1. The nuclear Overhauser effect
       3.1.1. Relaxation matrix theory .................................................................8
       3.1.2. Dynamical averaging .................................................................10
       3.1.3. Computational aspects ..............................................................11
       3.1.4. From NOE's to distances
              3.1.4.1. Two-spin approximation ....................................................12
              3.1.4.2. Iterative relaxation matrix approach ..................................13
       3.1.5. Direct use of NOE's as structural constraints
              3.1.5.1. 2D NOE ...................................................................14
              3.1.5.2. 3D NOE-NOE ...............................................................17
              3.1.5.3. Time- and ensemble-averaged NOE restraints ..............18

  3.2. Dihedral angles ...............................................................................................19

  3.3. Structure calculations
       3.3.1. Distance geometry (DG) .................................................................20
       3.3.2. Simulated annealing (SA) ...............................................................21
       3.3.3. Restrained Molecular Dynamics and Energy Minimisation .......22
              (RMD, REM)
       3.3.4. A protocol for structure determination and refinement ..........23
       3.3.5. Quality of the structures ..............................................................25

4. Implementation .....................................................................................................27

  4.1. Methods for the computation of the theoretical NOE intensities $A_{ij}$ .........27

  4.2. Scaling of the NOE intensities .....................................................................29

  4.3. NOE restraining potentials $V_{NOE}$ ..........................................................30

  4.4. Methods for the computation of the NOE forces and of the gradient of the theoretical NOE intensities $\#A_{ij}$ ..........................................................33

  4.5. $R$ factor definitions ......................................................................................34

  4.6. Architecture of the DINOSAUR routines ..................................................36

  4.7. Proton names conventions in DINOSAUR .............................................38

  4.8. Implementation in an "all atoms" refinement program ...........................39
4.9. **Installation under UNIX** .................................................................40

5. **DINOSAUR procedures** .................................................................45

5.1 **Set-up**
5.1.1 dinoset (and filenames conventions) .............................................46
5.1.2 dinodir ..................................................................................47
5.1.3 dinoin ..................................................................................47
5.1.4 mkhco ..................................................................................47
5.1.5 mkhtyp ..................................................................................48

5.2 **Refinement**
5.2.1 dinodd ..................................................................................49
5.2.2 dinoem ..................................................................................50
5.2.3 dinomd ..................................................................................50

5.3 **Analysis**
5.3.1 rfacav ..................................................................................51
5.3.2 rfacdg ..................................................................................52
5.3.3 rfacgr ..................................................................................53
5.3.4 rfacdls ..................................................................................54
5.3.5 noise ....................................................................................54
5.3.6 selpks ..................................................................................54
5.3.7 ananoe ..................................................................................55
5.3.8 cvnoe ..................................................................................56
5.3.9 cvhco ..................................................................................56
5.3.10 chkass ..................................................................................56

5.4 **Special for ensemble refinement** ...............................................57
5.4.1 genens ..................................................................................57
5.4.2 splitco ..................................................................................57

5.5 **Special for symmetrical dimers** ................................................57
5.5.1 seldim ..................................................................................58
5.5.2 chamb ..................................................................................58

6. **Important files in DINOSAUR (description and examples)** ......59

6.1 **2D DINOSAUR input file** ..........................................................59
6.1.1 File description ......................................................................59
6.1.2 Example of a 2D DINOSAUR input file with one data set ..........63
6.1.3 Example of a 2D DINOSAUR input file with two data sets .........64

6.2 **3D DINOSAUR input file** ..........................................................66
6.2.1 File description ......................................................................66
6.2.2 Example of a 3D DINOSAUR input file ....................................69

6.3 **2D NOE file (BUILDUP)** ..........................................................70
6.3.1 File description ...............................................................70
6.3.2 Example of a 2D BUILDUP NOE file ..................................71

6.4 2D NOE file (free) ...............................................................72
6.4.1 File description ...............................................................72
6.4.2 Example of a 2D NOE file in free format .........................72

6.5 3D NOE-NOE file (REGINE) ..................................................73

6.6 3D NOE-NOE file (free) ..........................................................74
6.6.1 File description ...............................................................74
6.6.2 Example of a 3D NOE-NOE file in free format ..................74

6.7 Stereospecific assignments file ..............................................74
6.7.1 File description ...............................................................74
6.7.2 Example of a stereospecific assignment file ....................75

6.8 Chemical shift file (.shift) .....................................................75
6.8.1 File description ...............................................................75
6.8.2 Example of a chemical shift file .....................................75

6.9 Generalised order parameter matrix ......................................76

6.10 Generalised order parameter file (.s2) ......................................76
6.10.1 File description ...............................................................76
6.10.2 Example of an order parameter file ..................................76

6.11 Proton coordinates file (.mdh) ..............................................77
6.11.1 File description ...............................................................77
6.11.2 Example of a proton coordinates file ..............................77

6.12 GROMOS proton codes file (.htyp) ........................................77

7. Examples ..............................................................................80

7.1 2D DINOSAUR refinement example (crambin) .............................81
7.1.1 Getting started .................................................................81
7.1.2 Creating a proton only file (.mdh) (mkhco) .........................82
7.1.3 Creating the DINOSAUR input file (dinoi) .........................85
7.1.4 Creating the GROMOS proton code file (.htyp) (mkhtyp) ........91
7.1.5 NOE restrained energy minimisation (dinoem) ....................92
7.1.6 NOE restrained slow-cooling simulated annealing (dnomd) ...99
7.1.7 Calculating R factors from a single structure (rfacgr) ............108
7.1.8 Analysing NOE trajectories (.noetraj) (ananoe) ....................110
7.1.9 Analysing NOE files (.theno) (ananoe) ...............................116
7.1.10 Analysing selected NOE peaks (selpks & ananoe) .............119
7.1.11 Analysing the results of the dynamic assignment procedure (chkass) .......124
7.2 Ensemble-averaged DINOSAUR refinement example (crambin) ..........125
  7.2.1 Creating an ensemble coordinate file (genens) ......................125
  7.2.2 Creating the DINOSAUR input file (dinoin) .........................126
  7.2.3 Ensemble-averaged NOE restrained slow-cooling simulated ..........129
      annealing (dinomd)
  7.2.4 Splitting the ensemble coordinate file (splitco) ...................132
  7.2.5 Calculating R factors from an ensemble of structures .............133
      (mkhco & avrfac)

7.3 Time-averaged DINOSAUR refinement example (crambin) .............137
  7.3.1 Creating the DINOSAUR input file (dinoin) .........................137
  7.3.2 Starting a time-averaged NOE restrained MD (dinomd) ............139
  7.3.3 Continuing a time-averaged NOE restrained MD (dinomd) ..........148
  7.3.4 Calculating instantaneous and “true” time-averaged NOEs and ....153
      R factors from a time-averaged MD trajectory (mkhco & avrfac)

7.4 2D and 3D DINOSAUR refinement examples (helical peptide) ........157

8. Benchmarks ...........................................................................158

9. References .............................................................................160
2. Introduction

In the last decade, NMR spectroscopy has become a important alternative to X-ray crystallography for the structural studies of molecules of biological interest. The principal advantage of NMR is that it allows the study of biomolecules in solution, thus closer to their real physiological environment. The method is limited, however, to biomolecules of relatively small size due to limitations in spectral resolution and line broadening effects. The upper limit is presently situated around molecular weights of 30 kD. Knowledge of three-dimensional structures might lead, for example, to a better understanding of structure-function relationships or recognition processes. For this purpose, accurate structures are required and, as we shall see, the use of so-called relaxation matrix approaches provides a means to improve the accuracy of the method.

The process of structure determination based on NMR data can be divided into four stages:

- assignments of the proton resonances,
- determination of structural constraints from NOE and $J$-coupling data,
- generation of structures,
- refinement of these structures.

The first and laborious task of assigning proton resonances is still the main bottle-neck of the structure determination, but progress is made in this area by the development of new multidimensional NMR techniques (3D-4D) (Vuister and Boelens, 1987; Oschkinat et al., 1988; Griesinger et al., 1989; Kay et al., 1989; Fesik and Zuiderweg, 1990; Boelens et al., 1990; Clore and Gronenborn, 1991a; Bax and Grzesiek, 1993) and the use of labelled samples ($^{15}$N, $^{13}$C). Computational approaches are also being developed toward an automatic assignment of the NMR spectra (Billeter et al., 1988; Cieslar et al., 1988; Eads and Kuntz, 1989; Kleywegt et al., 1989, 1991; van de Ven, 1990; van de Ven et al., 1990; Catasti et al., 1990). We will not treat this aspect of the problem here, but concentrate on the last three stages involving the determination of structural constraints (distances, dihedral angles) and the generation and refinement of the structures.

Structure determination by NMR is based primarily on the nuclear Overhauser effect (NOE) (Nogge and Schirmer, 1971; Neuhaus and Williamson, 1989). Its origin is dipolar cross relaxation between protons, which is a function of distances and molecular motions. Because of the approximate $r^{-6}$ dependence of the NOEs on the interproton distances, these can only be measured between protons relatively close in space ($\leq 5\text{Å}$). NOEs are generally transformed into distance constraints which are then used, together with dihedral angle constraints obtained from $J$-coupling data, for the generation of structures using Distance Geometry (DG) (Crippen and Havel, 1978; Havel et al., 1983; Havel and Wüthrich, 1984; Braun and Go, 1985) or Simulated Annealing (SA) techniques (Clore et al., 1986a; Nilges et al., 1988; Scheek et al., 1989). These structures are then usually refined with a combination of restrained Energy Minimisation (EM) and Molecular Dynamics (MD) simulations (van Gunsteren et al., 1983; Kaptein et al., 1985; Clore et al., 1985). Genetic algorithms (Lucasius et al., 1991; Blommers et al., 1992) and heuristic methods based on Kalman filter techniques (Patchett et al., 1990) can also be used for this purpose. Several approaches have been proposed for the interpretation of NOEs in term of distances. The simplest one classifies the NOEs in strong, medium and weak peaks and attributes corresponding distance ranges to these three categories. However, more accurate distances can be obtained from relaxation matrix calculations which take into account all indirect magnetisation transfers ("spin diffusion") (Keepers and James, 1984; Olejniczak et al., 1986; Boelens et al., 1988, 1989a; Koning et al.,...
A more direct method for refinement was proposed by Yip and Case (1989), in which NOE intensities are directly used as structural constraints, thus avoiding the transformation into distances. In this method, the structures are directly refined against experimental NMR data by comparison of theoretical and experimental NOE intensities.

In addition to NOEs and $J$-couplings, chemical shifts can also provide structural information on biomolecules. Recent developments that permit the prediction of $^1$H, $^{13}$C, $^{15}$N and $^{19}$F nuclear magnetic resonance chemical shifts should open new avenues to the structure refinement of proteins (Ösapay and Case, 1991, 1993; Wishart et al., 1991; Williamson et al., 1992; de Dios et al., 1993; Harvey and van Gunsteren, 1993).

In the theory chapter, we will first discuss the underlying theory of the nuclear Overhauser effect, including a description of intramolecular motions. The use of NOE's as structural constraints, indirectly by conversion into distances or directly by comparison of experimental and theoretical intensities, will be presented together with the most common methods used to generate and refine structures. A protocol describing the use of relaxation matrix approaches for the determination of solution structures from high-resolution NMR data is proposed. In the following chapters, descriptions of the DINOSAUR routines and of their implementation in molecular dynamics and energy minimisation programs, of the various DINOSAUR procedures and examples of the DINOSAUR input and script files are given.
3. Theory

3.1. The nuclear Overhauser effect

3.1.1. Relaxation matrix theory

Longitudinal relaxation in a system of $N$ spins is governed by a set of coupled differential equations of the form (Solomon, 1955):

$$\frac{d}{dt} A = - R A$$  \hspace{1cm} (1)

where $A$ represents the $z$ components of the magnetisation and $R$ is the relaxation matrix. It comprises contributions from cross relaxation and from external relaxation. For a system of dipolar-coupled spins, neglecting cross-correlation, $R$ has the dimension of the number $N$ in the systems. Since the dipolar interaction is a function of time, the relaxation rates are intimately connected with the molecular motion. The elements of $R$ can be expressed as (Solomon, 1955; Tropp, 1980; Neuhaus and Williamson, 1989)

$$R_{ii} = r_{ii} = K \sum_{j \neq i} J_{ij}(0)! + 6J_{ij}(2w_o)! + R_{\text{leak}}$$  \hspace{1cm} (2)

and

$$R_{ij} = s_{ij} = K \sum_{j \neq i} J_{ij}(2w_o)!$$

where $w_o$ is the Larmor frequency and $K = (2\sqrt{5}) \sum_j J_{ij}^2$; $r_{ii}$ is the direct relaxation rate, with $R_{\text{leak}}$ as additional term to account for all non-dipolar relaxation mechanism, and $s_{ij}$ is the cross relaxation rate between spin $i$ and $j$. $J_{ij}(\omega)$ denotes the spectral densities at multiple values of the spin Larmor frequency ($\omega = 0, w_o, 2w_o$). Cross relaxation is normally measured by observing intensity changes upon saturation or inversion of spins. Two-dimensional NMR techniques have been developed to investigate the resulting nuclear Overhauser effects (NOE) (Macura and Ernst, 1980). The normalised intensities in a 2D NOE spectrum recorded with mixing time $t_m$, are given by the matrix equation (Macura and Ernst, 1980):

$$A = \exp(-R t_m)$$  \hspace{1cm} (3)

which is the formal solution of the differential Eq. 1 for $N$ orthogonal start conditions. The spectral densities $J_{ij}(\omega)$ in the definitions of the relaxation rates $r_{ii}$ and $s_{ij}$ are cosine Fourier transforms

$$J_{ij}(\omega) = \frac{1}{2\pi} \int_0^\infty C_{ij}(t) \cos(\omega t) \, dt$$  \hspace{1cm} (4)
where the $C(t)$ are correlation functions describing the time evolution of the dipolar interaction. Assuming that the molecule is a rigid body moving isotropically in solution, the correlation functions in Eq. 4 can be described by a simple exponential of the form $\exp(-t/\tau)$, $\tau$ being the correlation time for the overall rotation of the molecule. With this simple definition, the spectral density $J_{ij}(\omega)$ becomes:

$$ J_{ij}(\omega) = \frac{1}{4\pi r_{ij}^6} \left[ \frac{2}{1 + (\omega^2 r_{ij}^2)} \right] $$

(5)

If the assumption of a rigid body is not valid, which is often the case in biomolecules, the description of $C(t)$ by a simple exponential is no more correct. The correlation function should then be defined in a more general form as

$$ C_{ij}(t) = \left[ \frac{2m(\overline{\bar{I}}_{ij}^{\text{lab}}(t_0+t))!Y_{2m}^*(\overline{\bar{I}}_{ij}^{\text{lab}}(t_0))}{r_{ij}^3(t_0+t)!r_{ij}^3(t_0)} \right] $$

(6)

Here the angular brackets indicate a time average over $t_0$ (which is equivalent to an ensemble average); $Y_{2m}$ are the second order spherical harmonics and $r$ and $\overline{\bar{I}}_{ij}^{\text{lab}}$ denote the length and polar angles of the interproton vector in the laboratory frame of coordinates. The form of the spectral density functions $J_{ij}(\omega)$ could be obtained experimentally from relaxation measurements. The feasibility of this approach has been recently demonstrated by Peng and Wagner (1992a, 1992b) who mapped the spectral densities of N-H vectors using heteronuclear relaxation experiments. In theory, this function can also be computed from a long MD trajectory. With present day computational facilities $C(t)$ can only be computed with sufficient statistical accuracy for $t$-values of the order of 0.1 to 1.0 ns. Fortunately, for many interproton vectors $C(t)$ is observed to reach a plateau value after a few ps, indicating that fast picosecond motions are well separated from slower processes (Olejniczak et al., 1984; Koning et al., 1991).

Following the approach developed by Lipari and Szabo (1982a, 1982b), a “model-free” description of $C(t)$ can be set up in terms of two characteristic times, $\tau$, the time in which $C(t)$ decays to the initial plateau value due to the fast internal motions, and $\overline{\bar{I}}$, the correlation time for the overall rotation of the molecule. Assuming isotropic tumbling, and transforming to molecule fixed coordinates $\overline{\bar{I}}_{ij}^{\text{mol}}$ one has (Olejniczak et al., 1984)

$$ C_{ij}(t) = \frac{\exp(-t/\overline{\bar{I}})}{4\overline{\bar{I}}} C_{ij}^{\text{int}}(t) $$

(7)

where the internal motion correlation functions $C^{\text{int}}(t)$ are defined as

$$ C_{ij}^{\text{int}}(t) = \frac{4\overline{\bar{I}}_n}{5} \left[ \frac{2m(\overline{\bar{I}}_{ij}^{\text{mol}}(t_0+t))!Y_{2m}^*(\overline{\bar{I}}_{ij}^{\text{mol}}(t_0))}{r_{ij}^3(t_0+t)!r_{ij}^3(t_0)} \right] $$

(8)

According to the addition theorem $C_{ij}^{\text{int}}(0) = \overline{\bar{I}}_{ij}^{-6}$ The plateau value of the correlation function, $C_{ij}^{\text{int}}(t)$, can thus be defined quite generally as $S_{ij}^{\text{int}} = \overline{\bar{I}}_{ij}^{-6}$ where $S_{ij}$ is a generalised order parameter. It has a value between 0 and 1, and can be calculated from an MD trajectory by
using Eq. 8 and estimating the plateau value for each interproton vector. Within this simplified model the functions $C^{\text{int}}(t)$ can be written as

$$C^{\text{int}}_{ij}(t) = \frac{1}{6} \sum_{\ell} \left( \frac{S_{ij}^2}{1 + (I_{ij}s^2) + 1} \exp(-t I_{ij}s^2) + (1-S_{ij}^2) \exp(-t I_{ij}s^2) \right)$$

(9)

where the initial decay has been written as an exponential. Combining Eqs 4 and 6-8, one arrives at (Olejniczak et al., 1984; Koning et al., 1991)

$$J_i^1 = \frac{1}{4} \sum_{\ell} \sum_{ij} \left( \frac{S_{ij}^2}{1 + (I_{ij}s^2) + 1} \exp(-t I_{ij}s^2) + (1-S_{ij}^2) \exp(-t I_{ij}s^2) \right)$$

(10)

with $I_{ij} = I_{ij}^{-1} + I_{ij}^{-1}$. Assuming that $I_{ij} = <I>$, the second term in Eq. 10, related to the initial decay, vanishes and $J_i^1$ becomes similar to the definition of Eq. 5, with $S_{ij}$ as an additional scaling factor and $r_{ij}$ replaced by the average $<r_{ij}>$

### 3.1.2. Dynamical averaging

Besides the effects of fast local motions, other dynamic processes can affect the relaxation behaviour of biomolecules. Here a discrimination between processes which are either slow or fast relative to the overall molecular tumbling has to be made. Examples of such processes in proteins are the symmetrical aromatic rings flips and methyl groups rotations. Dynamic processes which occur at a rate slower than or equal to the overall motion of the molecule can be incorporated in two ways in the calculation of the NOE intensities. The first is by addition of a kinetic matrix $K$ which describes the internal motion to the calculated relaxation matrix. This resembles the way in which chemical exchange is usually treated (Jeener et al., 1979). The second method is based on an approach proposed by Landy and Rao (1989). The approximation here is that the relaxation matrix is a weighted average of the relaxation matrices of various exchanging conformations. Considering a system with $N$ multiple exchanging sites and $p_i$ the probability for each conformation $i$, and assuming that the exchange is faster than the cross relaxation rates the following relations hold

$$\sum_{i=1}^{N} p_i = 1 \quad \text{and} \quad \sum_{i=1}^{N} A_i = A \quad \text{and} \quad A_i = p_i A$$

and Eq. 3 becomes

$$A = \sum_{i=1}^{N} A_i = \exp\left( \sum_{i=1}^{N} p_i R_i \right)$$

(11)

This method corresponds to $<r^{-6}>$ averaging over the various allowed conformations. In the case of symmetrical exchange the averaged relaxation matrix can also be obtained by averaging.
the value of the cross relaxation rates in the matrix belonging to one conformation. This method can be used to describe aromatic ring flips (Koning et al., 1990b).

Some motions, such as methyl group rotation, which are faster than the overall tumbling of the molecule can be treated in a more specific way. The spectral density function is no longer the same and an $<r^{-6}>$ averaging is not correct. For a rotating methyl group, assuming a three sites jump model, the spectral density function takes the form (Tropp, 1980)

\[
J_{ij}(\hat{\mathbf{Q}}) = \frac{1}{5} \sum_{n=-2}^{2} S_{ij}^2 \left[ (\hat{\mathbf{Q}}^2 - 1)^2 + 3(\hat{\mathbf{Q}}^2 - 1)^2 \right] + \frac{3}{4} \left[ (\sin^2 \hat{\mathbf{Q}} + \sin^4 \hat{\mathbf{Q}}) \right] \left[ \hat{\mathbf{Q}}^2 \right]
\]

(12)

where $\hat{\mathbf{Q}}^{mol}$ represents the polar angles in the molecular frame of the internuclear proton vector of length $r_{ij}$ between a proton $j$ and each of the methyl protons. Using this spectral density function in Eq. 2 the cross relaxation rates and the NOE intensities can be calculated. This approach can be extended to treat the case of two interacting methyl groups, the second sum in Eq. 12 comprising then nine terms. The intramethyl relaxation depends on the correlation time for the fast motion $\hat{\mathbf{Q}}$ and is described by the spectral density function (Woessner, 1962)

\[
J_{ij}(\hat{\mathbf{Q}}) = \frac{1}{6} \left[ 1 + (\hat{\mathbf{Q}}_0^2 - 1)^2 \right] \left[ (\hat{\mathbf{Q}}^2 - 1)^2 + \frac{3}{4} (\sin^2 \hat{\mathbf{Q}} + \sin^4 \hat{\mathbf{Q}}) \right] \left[ \hat{\mathbf{Q}}^2 \right]
\]

(13)

where $\hat{\mathbf{Q}}$ is the angle between the rotation axis and the interproton vector and with

\[
\hat{\mathbf{Q}}_1 = \frac{1}{\hat{\mathbf{Q}}} + \frac{1}{\hat{\mathbf{Q}}}
\]

These spectral density functions can be used to calculate the extra diagonal relaxation contribution of the methyl spin interaction $R_{methyl}$, which is an extra contribution to $R_{leak}$, using the following equation

\[
R_{methyl} = \frac{\hat{\mathbf{Q}}^2}{6} \left[ 3J(\hat{\mathbf{Q}}) + 12J(2\hat{\mathbf{Q}}) \right]
\]

(14)

Taking into account methyl group rotation into the calculations allows to save computational time by considering the methyl protons as one entry in the relaxation matrix with an intensity of 3. The transformations suggested by Olejniczak (1989) to symmetrise the matrices have then to be applied. The dimension of the problem is particularly reduced in proteins where the number of methyl groups is relatively high.

3.1.3. Computational aspects

Basically two methods are available for the computation of theoretical NOE intensities. These have been recently reviewed by Forster (1990). The first involves finding the eigenvectors and eigenvalues of the real symmetrical matrix $\mathbf{R}$ by standard matrix techniques (Keeper and James, 1984):
\[ \square = X^{-1} R X \quad (15) \]

Combining this expression with Eq. 3 leads to

\[ A = \exp(-R\square_m) = X \exp(-\square_m) X^{-1} \quad (16) \]

The matrix diagonalisation scales as \( N^3 \) where \( N \) is the dimension of the matrix \( R \) and can become extremely time-consuming when \( N \) increases. The calculations can be speeded up by making use of the sparseness of the relaxation matrix; due to the \( r^{-6} \) dependence on the distances of the cross relaxation rates, only a few elements of \( R \) are significantly larger than zero. Thus, it is possible to apply a spherical cutoff around each proton pair defining a subset of NOEs with only the neighbours. In this way, for each peak, a small eigenvalue problem will then be solved. The second approach (Marion et al., 1987) for the calculation of theoretical intensities is based on a numerical integration of the differential equations (Eq. 1) describing the time dependence of the NOE intensities. Several integration schemes exist (Taylor series integration, Euler algorithm, various Runge-Kutta methods) and their accuracy and performance have been addressed by Forster (1990). A choice between the various methods has to be made depending on the dimension of the system and on the purpose of the NOE calculation; for example, numerical integration is no more suitable if eigenvalues and/or eigenvectors are required (e.g. for the analytical NOE gradient, see section 3.1.5.1.).

3.1.4. From NOE's to distances

3.1.4.1. Two-spin approximation

Proton-proton distance constraints are most conveniently derived from cross-peak intensities in 2D NOE spectra taken at various short mixing times. The initial build-up rate of these peaks is proportional to the cross relaxation rates \( \square_{ij} \) between protons \( i \) and \( j \) (Kumar et al., 1981). Therefore, these cross relaxation rates can be measured either from a single 2D NOE spectrum taken with a sufficiently short mixing time or, more accurately, from a build-up series recorded with various mixing times (Wagner and Wüthrich, 1979; Kumar et al., 1981; Dobson et al., 1982). For a large molecule, assuming isotropical tumbling and no internal motions, the \( J_{ij}(0) \) term in Eq. 2 dominates the cross relaxation rate \( \square_{ij} \) which becomes simply related to the distance \( r_{ij} \) and the correlation time \( t_c \):

\[ \square_{ij} \sim -k r_{ij}^{-6} \quad (17) \]

Therefore, with a known calibration distance \( r_{cal} \) the proton-proton distances follow from the relation

\[ r_{ij} = r_{cal} \left( \square_{cal} / \square_{ij} \right)^{1/6} \quad (18) \]

In practice Eqs 17 and 18 are only approximately valid. There are two main problems associated with accurate determination of proton-proton distances. The first is that proteins are not rigid
bodies, and intramolecular mobility leads to non-linear averaging of distances and to different effective correlation times for the different interproton vectors in the molecule. The second is that of indirect magnetisation transfer or "spin-diffusion" (Kalk and Berendsen, 1976). In reality the NOE cross-peaks are the result of multispin relaxation and only in the limit of extremely short mixing times (where the signal-to-noise ratio is poor) is the two-spin approximation of Eq. 17 valid. For this reasons, a common approach is that of translating the NOE intensity into distance ranges (e.g. 2-3, 2-4, 2-5 Å for strong, medium and weak NOE's, respectively) rather than attempting to obtain precise distances (Wüthrich, 1986).

It has become possible to circumvent the spin diffusion problem by the use of relaxation matrix calculations which take into account all indirect magnetization transfers. These methods will be developed in the next paragraph.

3.1.4.2. Iterative Relaxation Matrix Approach (IRMA)

When a model for the structure and dynamics of a molecule is available, the NOE's can be calculated from the spectral densities as shown above (see Eqs 2, 3, 15, 16). The opposite route from experimental NOE's to relaxation parameters is also possible. The relaxation matrix \( R \) can be obtained after diagonalisation of the NOE matrix \( A \) (Olejniczak et al., 1986)

\[
X^{-1} A X = D = \exp(-\frac{\hbar}{n} \ln) \\
R = -X \left[ \frac{n! D^t}{\ln} \right] X^{-1}
\]

Thus, theoretically the matrix \( R \) can be obtained directly from 2D NOE experiments taken at suitably chosen mixing times. In practice, this is not possible for large molecules, since the experimental NOE matrix is incompletely known due to overlap and missing assignments. We have proposed a method called IRMA to circumvent this problem (Boelens et al., 1988, 1989a; Koning, 1990a). In this procedure, the experimental data are supplemented by NOE's calculated from a model. An overview of IRMA is given in Fig. 1. The starting point to the calculations can be a single structure or an ensemble of structures, the relaxation matrix being built from the distances of the corresponding contributions averaged as \(<r^6>\) in the latter case. A theoretical NOE matrix is then computed using standard matrix techniques (see Eqs 15-16) and combined with the experimental data. The resulting matrix can then be transformed back to a corrected relaxation matrix from which new distances are calculated. These distances are used to generate an improved model. The whole process is repeated until convergence is obtained. To increase the accuracy, the calculations can be performed for a series of mixing times; upper and lower bound margins can then be related to the precision with which the elements of \(<R>\) can be calculated, i.e. their variation with the mixing time. Similar approaches have been developed by several other groups (Borgias et al., 1990; Koehl and Lefèvre, 1990; Madrid et al., 1991; van de Ven et al., 1991; Edmonson, 1992). It has been shown that distances obtained from similar procedures result in an increased accuracy of the structures (Thomas et al., 1991; Bonvin et al., 1993a).
3.1.5. Direct use of NOE's as structural constraints

3.1.5.1. 2D NOE

Yip and Case (1989) proposed a method for structure refinement based on a direct comparison of experimental and theoretical NOE intensities, avoiding the conversion to distance constraints. A NOE target function is defined which will be minimised in a refinement procedure. When defining such a function we have to remind the dependence of the NOE intensities on the distances, which results in a strong asymmetry of the function, and the large range of experimental intensities. We defined a NOE restraining potential function $V_{NOE}$ as follows (Bonvin et al., 1991a, 1993b):

$$V_{NOE} = \sum_{ij} \left( \frac{1}{w_{ij}} \right)^{1} \left( \frac{1}{w_{ij}} \right)^{-1} \left( \frac{1}{A_{ij}^{\text{theo}}} - A_{ij}^{\exp} \right)^{2}$$  \hspace{1cm} (21)

with

$$w_{ij} = \frac{1}{N! \cdot !^{A_{ij}^{\exp}}}$$  \hspace{1cm} (22)

The sum runs over all NOE peaks between protons $i$ and $j$, possibly for several mixing times; $N$ represents the experimental noise level and $\square$ a relative error on the experimental NOE.
intensities accounting for integration errors. A switching factor $\square_{ij}$ can be used to switch off the function when the theoretical NOE intensities are within the experimental errors:

$$\square_{ij} = 0 \quad \text{if} \quad |N_{ij}^\text{theo} - N_{ij}^\text{exp}| \leq \frac{1}{2} N_{ij}^\text{theo}$$

$$\square_{ij} = 1 \quad \text{otherwise.}$$

The scaling factor $f$ is chosen such as to minimise the function $V_{\text{NOE}}$ for a defined subset of peaks:

$$f = \frac{\sum_{ij=1}^{N_{\text{subset}}} w_{ij} N_{ij}^\text{theo} N_{ij}^\text{exp}}{\sum_{ij=1}^{N_{\text{subset}}} w_{ij} N_{ij}^\text{theo}^2}$$  \hspace{1cm} \text{(23)}$$

$f$ is allowed to vary during the refinement in a range defined by $f_{\text{max}}$ and $f_{\text{min}}$. An appropriate choice of $f_{\text{max}}$ and $f_{\text{min}}$ avoids large fluctuations in the simulations. The choice of scaling the theoretical intensities in Eq. 21 was guided by practical considerations for the implementation. Scaling of experimental intensities is also possible, the weighting factors $w_{ij}$ should then also be scaled accordingly. It can easily be demonstrated that both types of scaling result in identical NOE potential values. Several other NOE potential definitions have been implemented, but we obtained the best results with the potential of Eq. 21. The weighting factor in Eq. 21 can correct for the large range of intensities, but the function is still asymmetric. We will deal with this latter problem when computing the NOE forces.

**Calculation of the NOE forces.** Calculation of the NOE forces requires the calculation of the gradient of the NOE potential function of Eq. 21 and therefore the calculation of the gradient of the NOE intensities $#A^\text{theo}$. An analytical solution of this gradient has been proposed by Yip and Case (1989) which has been implemented since then by others as well Nilges et al., 1991; Mertz et al., 1991):

$$\#A_{ij}^\text{theo} = \prod_{rstu} X_{fr} X_{rs}^{-1} (#R)^{-1} X_{tn} X_{uj}^{-1} f_{nt}$$

$$f_{nt} = \frac{\exp(-\square_{\square_n}) - \exp(-\square_{\square_n})}{\exp(-\square_{\square_u})} \quad (r \neq u) ; \quad f_{ru} = -\square_n \exp(-\square_{\square_n})$$  \hspace{1cm} \text{(24)}$$

Another approach is based on a numerical evaluation of $#A^\text{theo}$ (Baleja et al., 1990; Baleja, 1992). Both methods are time consuming. An approximation for the calculation of $#A^\text{theo}$ can be based on the first two terms of the power series of the exponential of Eq. 3 (Bonvin et al., 1991a)

$$A = \exp(-\square_n R) = 1 - \square_n R + \frac{1}{2} \square_n^2 R^2 \ldots$$  \hspace{1cm} \text{(25)}$$

and the gradient becomes
with this simple approximation, the NOE forces $F_{\text{NOE}}$ can be computed rapidly as:

$$F_{\text{NOE}} = -k_{\text{NOE}} \#V_{\text{NOE}}$$

With this definition, each NOE peak only gives rise to forces on the corresponding protons.

The computational costs are sensibly reduced in comparison with those for an analytical evaluation of the gradient, the time consuming part becoming the computation of the theoretical NOE intensities. The entire procedure scales as a $N^3$ problem if a diagonalisation method is used for calculating the NOE’s. This can be reduced by the use of a spherical cutoff around each proton pair $ij$ defining an experimental NOE peak.

Recently, new methods have been proposed for an efficient calculation of the NOE gradient (Nesterova and Chuprina, 1993; Yip, 1993) that should allow for a further speed-up of the computations. Particularly promising is the method proposed by Yip (1993) based on an integral expression of the NOE gradient using standard perturbation expansion techniques. It has been shown to be equivalent to the analytical expression proposed by Yip and Case (1989) (Eq. 24) and, from preliminary results, appears to be significantly faster than the expression in Eq. 24 (Yip, 1993).

**Treatment of unassigned prochiral groups.** When no stereospecific assignment is available for diastereotopic methylene protons or methyl groups of leucine and valine, distance constraints are usually defined to the corresponding pseudo atoms with an additional correction term to account for the maximum possible error (Wüthrich et al., 1983). For NOEs, pseudo atoms can not be introduced in the same way as for distances since relaxation pathways are defined from the explicit protons. For a peak involving unassigned diastereotopic protons, the NOE intensity is computed for all possible assignments and the assignment giving the best fit with the experimental data is chosen. However, due to the form of the NOE function which increases extremely fast when distances are underestimated, a new assignment is allowed only if the absolute difference in intensity between experimental and theoretical NOEs is less than the absolute experimental value. The assignment probability, defined as the difference in the number of times each possible assignment is found divided by the total number of refinement steps, is monitored during the refinement and an assignment can be “frozen” if its probability reaches a user-defined threshold. This approach is not limited to diastereotopic protons but can be extended to aromatic ring protons and peaks with multiple assignments. It can also be applied in the case of symmetrical dimer molecules to distinguish between intra- or intermonomer NOEs (Nilges and Brünger, 1991b).
3.1.5.2. 3D NOE-NOE

For larger molecules considerable overlap can exist in 2D spectra. This overlap can be overcome by spreading the peaks in a third dimension and new multidimensional NMR techniques were developed therefore (Oschkinat et al., 1988; Griesinger et al., 1989; Kay et al., 1989; Fesik and Zuiderweg, 1990; Boelens et al., 1990; Clore and Gronenborg, 1991). A similar procedure as with the refinement of NOE data from 1D and 2D spectra can be followed for the treatment of NOE intensities from 3D spectra.

The intensities of cross-peaks in a 3D spectrum are proportional to the transfer efficiencies of the two mixing periods of the experiment. The net transfer can be described in general as:

\[
T_{ijk}^{3D} = T_{ij}(1) T_{jk}(2)
\]

(28)

where \(T_{ij}(1)\) and \(T_{jk}(2)\) describe the magnetisation transfer efficiencies for each separate mixing period and correspond to NOE transfer, exchange or \(J\) transfer. The use of cross-peak intensities in a structure refinement procedure requires calculation of the gradients of these 3D intensities with respect to the coordinates of the protons in a molecule. The gradient of a 3D peak can be written as the sum of two terms containing the separate gradients of \(T_{ij}(1)\) and \(T_{jk}(2)\):

\[
\#T_{ijk}^{3D} = T_{jk}(2) \#T_{ij}(1) + T_{ij}(1) \#T_{jk}(2)
\]

(29)

These equations can be generalized for even higher-dimensional experiments. Thus, given a valid model for the calculation of the magnetisation transfer efficiencies and their derivatives, peaks from multi-dimensional spectra can be used as constraints in a direct refinement procedure. In particular, 3D NOE-NOE spectra appear to be suitable for this purpose. This latter technique was developed for the direct observation of indirect NOE magnetisation transfer (“spin-diffusion”) as occurs in larger biomolecules (Boelens et al., 1989b) and has been applied to oligonucleotides (Boelens et al., 1989b) and proteins (Breg et al., 1990a). The quantitative use of 3D NOE-NOE cross-peaks has been described by Kessler et al. (1991) and Habazettl et al. (1992a). These NOE’s can be translated into constraints, which are used by distance geometry of restrained molecular dynamics computations. We have shown that it is also possible to use such 3D intensities as direct constraints for structure refinement (Bonvin et al., 1991b).

In a 3D NOE-NOE spectrum the 3D cross-peak intensities \(A_{ijk}^{3D}\) can be calculated from their separate 2D NOE transfer efficiencies using Eq. 28. The computation of the 3D NOE-NOE intensity \(A_{ijk}^{3D}\) then amounts to the multiplication of the calculated 2D contributions \(A_{ij}\) and \(A_{jk}\). A similar NOE restraining potential function \(V_{3DNOE}\) can be defined as for the refinement with 2D data. Combining Eqs 26 and 29 and extending the NOE potential of Eq. 21 to the 3D case, one obtains for the 3D NOE-NOE forces:

\[
F_{3DNOE} = -k_{3DNOE} \#V_{3DNOE}
\]

(30)
The summation over the time in Eq. 31 which becomes:

$$\sqrt{f (D_{n1} \circ A_{jk} (D_{m2}) \# R_{ij}^1 + D_{m2} \circ A_{ij} (D_{n1}) \# R_{jk})}$$

3.1.5.3. Time- and ensemble-averaged NOE restraints

One of the most claimed advantage of NMR upon X-ray crystallography is that it allows the study of biomolecules and of their dynamics in solution, close their physiological environment. NMR data are collected as time- and ensemble-averaged quantities. In the commonly used structure determination procedures, however, structures are generated that are required to satisfy on an instantaneous basis or as single structure constraints obtained typically from NOE and J-coupling data. In some cases, a structure might fail to satisfy all the NMR constraints at a same time or might be forced into an unrealistic conformation. Solutions to this problem have been proposed by the introduction of time- (Torda et al., 1989, 1990) and ensemble-averaged NMR constraints (Scheek et al., 1991; Kemmink et al., 1993). Time-averaging and its implementation in restrained Molecular Dynamics simulations has been described and applied first for NOE-derived distances (Torda et al., 1989, 1990; Pearlman and Kollman, 1991; Schmitz et al., 1992) and more recently for J-coupling data (Torda et al., 1993). Here we extend the application of time- and ensemble-averaged restraints for the refinement directly against the experimental NOE data.

When averaging is introduced, the theoretical intensities in Eq. 21 should be replaced by their time- or ensemble-averages, the averaged intensities, and no longer those calculated from a single static structure, being now required to satisfy the experimental constraints. In the case of time-averaging, for which the time course of a MD simulation can be used, the averaged NOEs, are given by:

$$\overline{A_{ij}^{\text{theo}}}(t) = \frac{1}{t} \int_0^t A_{ij}^{\text{theo}}(t') \, dt' \quad (31)$$

Eq. 31 corresponds to the true average, which, as the time increases, might become less sensitive to instantaneous fluctuations in the MD simulation (Torda et al., 1989). To avoid this problem, Torda et al. (1989) introduced an exponential decaying memory function with time constant $t_0$ in the summation over the time in Eq. 31 which becomes:

$$\overline{A_{ij}^{\text{theo}}}(t) = \int_0^t [1 - \exp(-t/t_0)]^{-1} \exp(-t/t_0) A_{ij}^{\text{theo}}(t-t') \, dt' \quad (32)$$

If the simulation time $t$ is much longer than the time constant of the exponential $t_0$, a practical way to calculate Eq. 32, suitable form for implementation in MD algorithms, is given by (Torda et al., 1989):
\[
A_{ij}^{\text{theo}}(t) = \frac{1}{1 - \exp(-\tau/\Delta t)} A_{ij}^{\text{theo}}(t) + \exp(-\tau/\Delta t) A_{ij}^{\text{theo}}(t-\Delta t) \quad (33)
\]

where $\Delta t$ is the time step of the integrator in the MD simulation. The true-averaged defined in Eq. 31 can however be used for the analysis of the MD trajectories.

For ensemble-averaged NOE restraints, the theoretical intensities are given by

\[
\overline{A_{ij}^{\text{theo}}} = \prod_{k=1}^{n_{\text{conf}}} p_k A_{ij}^{\text{theo}}(k) \quad \text{with} \quad \prod_{k=1}^{n_{\text{conf}}} p_k = 1 \quad (34)
\]

$p_k$ gives the probability of the conformer $k$. Ideally a Boltzmann weighting should be chosen for the probabilities. It is, however, not possible to obtain the free energy to calculate the Boltzmann factor in the course of a simulation and some other weighting function has to be used. Eq. 34 requires the computation of the theoretical NOE intensities for all conformers in the ensemble, which is one of the time-consuming part of direct NOE refinement. However, if we assume that the various conformers are in slow exchange, the formalism of Landy and Rao (1989) can be applied and the averaged NOEs are then given by

\[
\overline{A_{ij}^{\text{theo}}} = \prod_{k=1}^{n_{\text{conf}}} p_k A_{ij}^{\text{theo}}(k) = \frac{1}{n_{\text{conf}}} \prod_{k=1}^{n_{\text{conf}}} A_{ij}^{\text{theo}}(k) \quad (35)
\]

With this formalism, the computation of the theoretical NOE intensities is only performed once for the ensemble, which allows a reduction of the computational time.

Both time- and ensemble-averaging options are implemented into the DINOSAUR routines.

### 3.2. Dihedral angles

The $J$-coupling constant is another useful NMR parameter. Via relationships such as those first proposed by Karplus (1959) (cf. Pardi et al. (1984)) information is obtained about dihedral angles, which provide valuable information for refining protein structures, particularly at the local level on the protein surface where fewer NOEs are available. In favourable cases dihedral angle constraints can be obtained from three-bond $J$-couplings. These can be obtained from the fine structure of cross-peaks in COSY-like spectra recorded with high digital resolution (Hyberts et al., 1987; Griesinger et al., 1987). An important example is the three-bond coupling $^3J_{\text{H}^\beta\text{H}^\gamma}^\text{am}$ between amide and C=C protons in proteins, which gives a measure of the backbone torsion angles $\phi$. For helical regions $^3J_{\text{H}^\beta\text{H}^\gamma}^\text{am}$ is small (ca. 4 Hz), while for extended chain conformation such as in $\alpha$-sheets it takes large values (9-10 Hz). Another important example are the $^3J_{\text{H}^\beta\text{H}^\gamma}^\text{C}^\beta$ and $^3J_{\text{H}^\beta\text{H}^\gamma}^\text{C}^\gamma$ couplings, which, in combination with an analysis of NOE data, give a measure of the side chain torsion angle $\psi$ and allow the stereospecific assignment of the diastereotopic methylene protons $\text{H}^\beta$ and $\text{H}^\gamma$ (Zuiderweg et al., 1985; Hyberts et al., 1987). Usually the large $J$-couplings (8-10 Hz) are the most useful source of information, because $J$-coupling smaller than the linewidth (typically 5 Hz or larger) cannot be reliably measured due to cancellation effects in antiphase multiplets (Neuhaus et al., 1985).
Furthermore, the interpretation of large coupling constants in terms of dihedral angles is less ambiguous than for intermediate values for which uncertainties may arise, first, because several dihedral angles may belong to a certain $J$-coupling and, second, because they may be the result of motional averaging. The development of heteronuclear NMR techniques and the use of labelled samples ($^{15}$N,$^{13}$C) have expanded the possibilities for using dihedral angle constraints in structure determination (Kay and Bax, 1990; Wagner, 1990; Clore et al., 1991a; Archer et al., 1991; Grzesiek et al., 1992; Madsen et al., 1993; Sørensen et al., 1993; Powers et al., 1993). A more accurate characterisation of the $[\alpha]$ and $[\beta]$ dihedral angles has become possible by the determination of heteronuclear coupling constants like the nitrogen-protons coupling constants $^3J_{H\alpha^\beta}^\alpha$-$N_{i+1}^\alpha$, $^3J_{NH}^\alpha$ (Forman-Kay et al., 1990; Chary et al., 1991; Archer et al., 1991) or the carbon-proton coupling constants $^3J_{CO-H}^\alpha$ (Grzesiek et al., 1992). A characterisation of the $[\beta]$ dihedral angle in leucine residues has also become possible from the measurement of carbon-carbon couplings $^3J_{C-C}^\alpha$ (Bax et al., 1992; Powers et al., 1993). The $J$-couplings are usually introduced as constraints in structure refinement programs after transformation into dihedral angles (de Vlieg et al., 1986; Clore et al., 1986b; Kline et al., 1988; Moore et al., 1988). Recently, new methods have been proposed for the direct use of $J$-couplings as restraints, without transformation into dihedral angles (Kim and Prestegard, 1990; Mierke and Kessler, 1992; Torda et al., 1993). As with NOE data, these direct methods allow a better description of the experimental NMR data.

### 3.3. Structure calculations

#### 3.3.1. Distance geometry (DG)

The metric matrix distance-geometry (DG) algorithm (Blumenthal, 1970; Crippen and Havel, 1978) was known well before structure determination by NMR became possible (Havel et al., 1983; Havel and Wüthrich, 1984). This method does not rely on a starting conformation, is in this respect free from operator bias and therefore very attractive for structure determination. The DG procedure amounts to the following. First, upper and lower bound matrices $U$ and $L$ are set up for all atom-atom distances of the molecule. Some of the elements $u_{ij}$ and $l_{ij}$ follow from standard bond lengths and bond angles of the covalent structure and from experimentally found distance ranges from NOE’s and $J$-coupling constants. A bound smoothening procedure using triangle inequalities extends the constraints to all elements of $U$ and $L$. This procedure can even be extended to satisfy tetriangle inequalities as for example implemented in the DG-II package (Havel , 1991). Then, a distance matrix $D$ is set up with distances chosen randomly between upper and lower bounds, $l_{ij} \leq d_{ij} \leq u_{ij}$. The elements of $D$ can be chosen in such a way that they not only lie between their respective upper and lower bounds, but also themselves obey the triangle inequalities; this latter technique is known as random metrization. The so-called "embedding" algorithm then finds the best 3D structure consistent with the distances $d_{ij}$. This structure must be optimized with an error function consisting of chirality and distance constraints. This forces the amino acids side chains to adopt the correct chirality and the distances to satisfy the upper and lower bound criteria, although usually the DG structures still contain a number of violations of the distance bounds. By repetition of the DG calculations with randomly chosen distance matrices $D$, families of structures can be obtained which allow to judge how uniquely the structure is determined by the constraints.
Another method, also termed distance geometry but using a quite different mathematical procedure, was suggested by Braun and Go (1985). Here, the biomolecular conformation is calculated by minimizing a distance constraint error function. Special features of the method are that dihedral angles are used as independent variables rather than Cartesian coordinates and that it uses a variable target function, first satisfying local constraints (between amino acids nearby in the polypeptide chain) and later including long-range constraints. This approach avoids becoming trapped in a local minimum of the target function. Usually one starts with various initial conformations obtained by taking random values for the dihedral angles. This method, which has been implemented since in various programs, e.g. DISMAN (Braun and Go; 1985) and more recently DIANA (Güntert et al., 1991), has rather similar efficiency and convergence properties as the metric matrix distance-geometry algorithm.

3.3.2. Simulated annealing (SA)

Although the distance-geometry methods do not need starting structures and are therefore not subject to operator bias, this does not mean that they sample the allowed conformational space (consistent with the bounds) in a truly random fashion. Simplified molecular dynamics calculations with only geometric constraints were developed to increase the sampling of the DG methods. These are termed in general as simulated annealing (SA) procedures. Although these latter can be directly used to generate structures starting from random coordinates, they are most commonly combined with distance-geometry calculations, the starting structures for the simulated annealing stage being obtained by distance geometry embedding. Simulated annealing involves raising the temperature of the system followed by slow cooling in order to overcome local minima and locate the region of the global minimum of the target function $V_{tot}$ (Nilges et al., 1988). This is achieved by solving Newton's equations of motion

$$m_i \ddot{r}_i = F_i$$

with the forces given by

$$F'_i = -\partial V_{tot} / \partial r_i$$

The potential $V_{tot}$ usually contains the following terms:

$$V_{tot} = V_{covalent} + V_{repel} + V_{disre} + V_{dihre} + V_{NOE}$$

$V_{covalent}$ is a potential to maintain correct bond lengths, angles, planes and chirality, $V_{repel}$ is a simple repulsion term to prevent unduly close non-bonded contact, $V_{disre}$ and $V_{dihre}$ represent the experimental NOE distance and dihedral angles constraints, respectively, and $V_{NOE}$ is an extra term for direct comparison of theoretical and experimental NOE intensities as defined, for example, in Eq. 21. Usually the NOE distance constraints are represented by a flat well potential of the form
where \( l_{ij} \) and \( u_{ij} \) are the value of lower and upper limits of the target distances, respectively. Other forms that contains a linear part at long distances are also used (Kaptein et al., 1985) to avoid too large energies which may give computational problems. A similar definition as in Eq. 39 can be used for dihedral angle constraints, but other forms have also been proposed as (de Vlieg et al., 1986):

\[
V_{\text{dihre}} = K_{\text{dihre}} \left[ 1 - \cos \left( \frac{\theta - \theta_0}{\theta_1 - \theta_0} \right) \right] 
\]

where \( \theta_0 \) represents the dihedral angle value obtained from \( J \)-coupling data. The various terms in Eq. 38 have variable force constants which can be progressively increased during the annealing.

The combination of distance geometry, for generating starting structures with an approximate folding by embedding a subset of atoms, and simulated annealing calculations provides a powerful and efficient tool for the structure determination of biomolecules (Nilges et al., 1988; Holak et al., 1989). This hybrid method should be particularly useful to solve the structure of large proteins for which computational costs may become a limiting factor.

### 3.3.3. Restrained molecular dynamics and energy minimisation (RMD, REM)

The quality of biomolecular structures based on geometric constraints can be improved by taking energy considerations into account. For instance, in DG structures amino acids side chains often adopt eclipsed conformations. Also, hydrogen bonds and salt bridges may not be formed unless they are specifically introduced as constraints. In restrained molecular dynamics refinement structures are optimized simultaneously with respect to a potential energy function and a set of experimental restraints. Of course the success of this method now depends to a large extent on the quality of the force field used. It is therefore important to realize the limitations and approximate nature of this force field, especially when the calculations do not include solvent molecules.

In RMD calculations, Eqs 36 and 37 are integrated with the potential function given by

\[
V_{\text{tot}} = V_{\text{bond}} + V_{\text{angle}} + V_{\text{torsion}} + V_{\text{vdW}} + V_{\text{Coulomb}} + V_{\text{disre}} + V_{\text{dihre}} + V_{\text{NOE}}
\]

The first two terms tend to keep bond lengths and angles at their equilibrium values. \( V_{\text{torsion}} \) is a sinusoidal potential describing rotations about bonds; for \( V_{\text{vdW}} \) (the van der Waals interaction) usually a Lennard-Jones potential is taken, and \( V_{\text{Coulomb}} \) describes the electrostatic interactions. The three last terms, similar to those in Eq. 38, distinguishes RMD from more conventional MD simulations.
A RMD simulation is usually preceded and followed by restrained energy minimisation (REM) using steepest descent or conjugated gradient methods to bring the energy down to an acceptable level. REM using the same potential energy function as RMD (Eq. 41) usually induces only small changes in the structures by driving them to the closest minimum and is not able to take them out of this. By contrast, RMD is able to overcome barriers of the order of kT because of the kinetic energy in the system and therefore has a much larger radius of convergence. RMD works as an efficient minimiser, since excess potential energy, converted to kinetic energy, is drained off by coupling the system to a thermal bath of constant temperature. The sampling of the conformational space consistent with the experimental constraints can be increased by using high temperature simulations (Bruccoleri and Karplus, 1990) or the recently proposed potential energy annealing conformational search method (PEACS) (van Schaik et al., 1992). In this latter the potential energy is coupled to a reference energy bath which energy level is progressively decreased over the run.

Further extensions of the method, especially useful for highly mobile structures, include time-averaging or ensemble-averaging of the experimental restraints. In these methods, $V_{\text{direct}}$ is calculated from time- or ensemble-averaged distances instead of instantaneous ones. Forces are thus applied on the corresponding protons only if the averaged distance violates the restraint. These methods allow a larger conformational freedom consistent with the NOE restraints. The application of time-averaged RMD to Tendamistat (Torda et al., 1990) suggested that some side-chains are more flexible than what conventional refinement had indicated. Similar procedures can be applied for direct NOE refinement as has been described in section 3.1.5.3.

### 3.3.4. A protocol for structure determination and refinement

Once NMR data (NOE and $J$-coupling) are obtained, the determination and refinement of solution structures can be divided into three stages:

1. **Generation of structures using DG and/or SA methods**, 
2. **Distance restrained refinement of the structures**, 
3. **Direct NOE restrained refinement**.

**i) Structure generation.** In this first stage, a set of structures is being generated using DG and/or SA techniques. The NOE distance constraints for these calculations can be obtained from an initial rate analysis according to Eq. 18 or preferably from relaxation matrix calculations since the distances obtained in the latter case reflect the effect of spin diffusion and are expected to be more accurate. The number of structures generated at this stage will depend on the computational resources available, but should not be too small to make sure that a good sampling of the conformational space consistent with the experimental constraints is obtained and to be able to assess the precision of the structure.

**ii) Distance restrained refinement.** This stage includes REM and RMD calculations and the advantages of the methods have already been described above. The ensemble of structures obtained from the DG/SA calculations can be refined in a parallel procedure combining REM and RMD steps. When IRMA is used, the refined structures can then serve as input for new relaxation matrix calculations which will result in a new set of distance constraints. An advantage of refining all structures in parallel is that NOE's can be computed from an average relaxation matrix, which better reflects the experimental situation (several conformers might be present in solution). Parallel refinement can then be pursued with the new constraints reflecting both spin diffusion effects and ensemble averaging. This "ensemble" IRMA protocol (Bonvin et al., 1993a) can be repeated until convergence is achieved. The length of the RMD simulations
should be adapted to the system under consideration, but, since the purpose of such computations is not reaching an equilibrium state but driving the structures towards a global minimum, short simulation times (5 to 20 ps) should already be sufficient.

**iii) Direct NOE refinement.** Here REM and RMD are performed with direct use of experimental NOE's as constraints in the extra potential $V_{NOE}$ of Eq. 41. Since this is a computationally expensive procedure, it should only be applied in the final stage, starting from already well refined structures. A slow-cooling simulated annealing protocol as proposed for crystallographic $R$ factor refinement (Brünger and Krukowski, 1990) seems to offer a good compromise between computational costs and effectiveness. The complete potential energy

---

*Figure 2:* Protocol for determination and refinement of biomolecular structures from NMR data using relaxation matrix calculations.
function of Eq. 41 is used in a MD simulation starting at high temperatures (typically 600 to 1000 K) and the temperature of the system is progressively decreased with a cooling rate in the order of 10 K every 25 fs. EM alone already improves the fit between simulated and experimental NOE's without inducing much changes to the structures. The annealing stage allows further improvement of the correspondence between simulated and experimental NOE data combined with a better sampling of the conformational space.

A refinement protocol making use of both the "ensemble" IRMA and direct NOE refinement (DINOSAUR) is presented in Figure 2.

3.3.5. Quality of the structures

Important factors for assessing the quality of the NMR structures are the number and distribution of the NOE and dihedral angle constraints and the number of stereospecific assignments. Several other criteria have also to be considered, but these will all depend, to some extent, on the factors mentioned above. The precision of the structure is generally given by the average root mean squared deviations (r.m.s.d.) (pairwise or from the average coordinates). These are usually calculated for the backbone and all atoms, respectively, and sometimes on a substructure. We have to keep in mind, however, that the r.m.s.d. gives an indication of the precision of the structure, but are not a direct indication of its accuracy. Structures with very low r.m.s.d. can still be wrong! To judge the quality, we have to take several other criteria into account: the stereochemical quality (e.g. via Ramachandran plots), the deviations from the idealized covalent geometry, the conformational energy, residual restraints violations, the presence of hydrogen bonds and a good packing of the structures are criteria commonly used for this purpose. Another criterion to assess the quality of the structures obtained from NMR data is to directly compare calculated and experimental NOE’s. The fit can be described by a residual index or R factor in a similar way as in X-ray crystallography. Several R factor definitions have been proposed (Thomas et al., 1991; Gonzales et al., 1991; Withka et al., 1992) and can all be represented by a general equation of the form:

\[
R = \frac{\sum w_i^1 (N_i^{\text{theo}} - N_i^{\text{exp}})}{\sum w_i^1 N_i^{\text{ref}}} \tag{42}
\]

where

\[N_i^\dagger = A_{i,i}^\dagger \quad \text{or} \quad N_i^\dagger = (A_{i,i}^\dagger)^{1/6}\]

with, in the denominator, \(N_i^{\text{ref}}\) defined as:

\[N_i^{\text{ref}} = N_i^{\text{exp}} \quad \text{or} \quad N_i^{\text{ref}} = \frac{1}{2} (N_i^{\text{exp}} + N_i^{\text{theo}})\]

\(w_i^1\) are weighting factors which can be chosen equal to the mixing times, since NOE’s measured at longer mixing times are expected to have a better signal-to-noise ratio. The simplest one, with \(N_{i,j} = A_{i,j}\) and \(N_i^{\text{ref}} = N_i^{\text{exp}}\), is similar to the X-ray definition (Gonzales et al., 1991). For NMR, however, other functions may be more appropriate since we are dealing principally
with two problems. First, the range of NOE intensities is quite extended (by more than a factor 1000) and $R$ might be dominated by strong peaks, corresponding to short distances in the structures which are expected to be less important for determining the three-dimensional structure. A sixth-root residual index, $R^{1/6}$, where $N_i = (A_i)_{1/6}$, allows for a more even contribution of weak and strong NOE’s in the $R$ factor (Thomas et al., 1991). The second problem inherent to the $R$ factor comes from the approximate $r^{-6}$ dependence of the NOE’s. Because of this extreme distance dependence, the $R$ factor function is asymmetric, the error increasing strongly when the distances are shorter than the target distances. Normalizing the errors by the sum of both experimental and theoretical NOE intensities, $N_i^{\text{ref}} = \frac{1}{2} (N_i^\text{exp} + N_i^\text{theo})$, should avoid this problem (Withka et al., 1992). To give a better picture of the quality of the structures, a distinction can be made between intra- and interresidue peaks and the $R$ factors can be calculated as function of the residue sequence which allows to recognize problem regions in the structures.

In summary, Table 1 lists the various criteria which could be used to assess the quality of the NMR structures.

**Table 1:** Criteria for assessing the quality of the NMR structures.

- number and distribution of the NOE and dihedral angle constraints
- average r.m.s.d. (pairwise or from the average coordinates)
- residual constraint violations: maximum and average ($\pm$ standard deviations) violations for distance bounds and dihedral angle constraints
- deviations ($\pm$ standard deviations) from idealized covalent geometry
- conformational energy (with indication of the energy potential used)
- hydrogen bonds
- Ramachandran ($\phi, \psi$) plot for backbone dihedral angles
- $R$ factors (with indication of the definition used)
4. Implementation

DINOSAUR consists of a set of routines for simulation of theoretical NOE intensities, comparison with experimental data via various NOE restraining potentials and derivation of forces, written in FORTRAN 77. These, grouped under the name DINOSAUR (Direct NOe Simulation Approach for Unbelievable structure Refinement), are implemented in three programs: a simplified simulated annealing program, distance-bound-driven dynamics (Kaptein et al., 1988) where only the covalent information is used in addition to the experimental constraints, and in the GROMOS programs for energy minimisation (PROEM) and molecular dynamics (PROMD) which make use of a complete force field (van Gunsteren and Berendsen, 1987). The programs can handle several sets of data corresponding to build-up series recorded in various solvents, H$_2$O or D$_2$O (with the possibility of skipping the exchangeable protons for the D$_2$O calculations), or at various spectrometer frequencies. The methods used for the computation of the theoretical NOE intensities, NOE forces and $R$ factors and the architecture of the DINOSAUR routines will be described in this chapter.

4.1 Methods for the computation of the theoretical NOE intensities

Nine methods for the computation of the theoretical (2D and 3D) NOE intensities are implemented in DINOSAUR. Their numbering is not an indication of the performance of the method but a result of their chronological implementation in DINOSAUR. For completeness we will describe all nine methods, but we give preference to two of them, the exact one based on the diagonalisation of the complete relaxation matrix (definition 1 below) and the method making use of spherical cut-offs around each proton pair defining a NOE peak (definition 9 below). The choice between these two methods depends on the dimension of the system and the number of experimental NOE peaks and will be discussed below.

1) **Exact NOE calculation** by diagonalisation of the complete relaxation matrix (see paragraph 3.1.3., Eqs 15 and 16) (subroutines sdiag**).

\[
A_{ij} = (\mathbf{X} \exp(-t I_n) \mathbf{X}^{-1})_{ij}
\]  

(43)

The diagonalisation is performed using an optimised version of the EISPACK routines. Its performance was found to be comparable to the newest LAPACK diagonalisation routines. If possible machine-optimised libraries should be used. This can result in a significant improvement of the performance. For example, on CONVEX computers a speed-up of a factor two can be obtained by using the CONVEX optimised diagonalisation routines in VECLIB. The subroutine sdiag should be modified accordingly. (a CONVEX version can be found in the noecvlib library, called sdiagcv). Note that all NOE calculations are performed in double precision. This is a requirement for accuracy, especially with large systems.

2) **Two-spin approximation** (subroutine s2spn), NOE intensities are calculated using a two-spin approximation correct for $I_n = 0$. 

\[ A_{ij} = \frac{R_{ij}}{D} \exp\left[ \Theta \left( R_{ii} + R_{jj} \right) \right] \sinh(D) \]  

with  
\[ D = \sqrt{\frac{1}{4} \left[ |\bar{R}_{ij}|^2 + |\bar{R}_{ji}|^2 \right]} \quad \text{and} \quad \bar{R}_{ij} = R_{ii} - R_{jj} \]  

3) **Simplified expansion of the relaxation matrix** (subroutine **sexp**). In this approximation, only the largest contributions to \( A_{ij} \), up to a third order expansion of the exponential (Eq. 25), are taken into account.

\[ A_{ij} = - \frac{1}{6} R_{ij} + \frac{1}{2} \left( R_{ii} R_{ij} + R_{jj} R_{jj} \right) \]  

\[ - \frac{1}{6} \left( R_{ii}^2 R_{ij} + R_{jj} R_{ij} R_{jj} + R_{ii} R_{ji} R_{jj} \right) \]  

(45)

4) **Second order expansion of the relaxation matrix** (subroutine **s2ord**).

\[ A_{ij} = - \frac{1}{6} R_{ij} + \frac{1}{2} \left( R_{ii} R_{ij} + R_{jj} R_{jj} \right) \]  

(46)

5) **Second order perturbation expansion** (Yip, 1989) (subroutine **syip**).

\[ A_{ij} = \frac{R_{ij}}{D} \exp\left[ \Theta \left( R_{ii} + R_{jj} \right) \right] \sinh(D) \]  

\[ + \sum_{k \neq ij} R_{ik} R_{kj} \left[ \exp(\Theta R_{ij}) \exp(\Theta R_{kij}) \exp(\Theta R_{kij}) \exp(\Theta R_{kij}) \right] \]  

(47)

The next three methods are combinations of definitions 1 to 4. The theoretical NOE intensities are computed using one of the three approximations (definitions 2, 3 and 4) and corrected for spin diffusion:

\[ A_{ij} = A_{ij}^{\text{approx}} + \Box A_{ij} \quad \text{(each step)} \]  

\[ \Box A_{ij} = A_{ij}^{\text{exact}} - A_{ij}^{\text{approx}} \quad \text{(each n^{th} step)} \]  

(48)

The correction term is given by the difference between the exact NOE intensities obtained with definition 1) and those obtained with the chosen approximation. This spin diffusion correction is held constant during a predetermined number of refinement steps. The diagonalisation of the relaxation matrix, which is the time consuming procedure, is thus no longer required at every refinement step. The three definitions are:

6) **Two-spin approximation with spin diffusion correction**

7) **Simplified expansion of the relaxation matrix with spin diffusion correction**

8) **Second order expansion of the relaxation matrix with spin diffusion correction** .

©DINOSAUR Software manual - 28 - DINOSAUR
The last implemented method (definition 9), subroutine `sdiasmc` makes use of the sparseness of the relaxation matrix. The theoretical NOE intensities are computed using an user-defined spherical cut-off around each proton pair defining a NOE, which allows a considerable speed-up of the computations. With this approximation, the spin diffusion contributions of all neighbours within two spheres of radius \( r_{\text{cut}} \) around the two protons defining a NOE peak are taken into account. For each NOE a small relaxation matrix is calculated.

![Figure 3: Use of spherical cut-offs for the computation of the theoretical NOE intensities.](image)

The use of spherical cut-offs in the order of 4.0 to 4.5 Å is found to have little effect on the calculated NOE intensities compared to the full relaxation matrix treatment of definition 1). In proteins, the average number of neighbours within two spheres of 4 to 4.5 Å radius is found to be approximately 20. The performance of this method compared to the full relaxation matrix treatment as in definition 1) will depend on the dimension (number of protons) of the system, on the number of experimental NOEs and on the choice of the spherical cut-off. As in definition 1), optimised EISPACK routines are used for the diagonalisation of the small relaxation matrices.

For systems with more than 200 protons the use of spherical cut-offs as implemented in definition 9) will give the best performance (in terms of cpu time) with only negligible effects on the computed NOEs.

### 4.2 Scaling of the NOE intensities

For practical consideration we chose to scale the theoretical intensities in DINOSAUR. The scaling factor \( f \) is chosen such as to minimise the NOE restraining function \( V_{\text{NOE}} \) for a defined (sub)set of peaks (see Eq. 23, paragraph 3.1.5.1). Several possibilities are implemented in DINOSAUR (subroutine `sscal`) to define this subset. These are:

0) user defined subset of reference peaks: an overall scaling factor is calculated from the reference peaks given in the DINOSAUR input file. Peaks between geminal \( \text{\textsuperscript{13}C} \)-methylene protons, between protons in aromatic rings and H\( \text{\textsuperscript{13}C} \)-methyl peaks in alanine residues can be
used for this purpose. It is recommended to choose peaks between protons that are part of a stable secondary structure elements and close to the backbone to avoid introducing scaling errors due to internal motions.

1) scaling per mixing time using all NOE peaks

2) overall scaling using all NOE peaks

3) scaling per mixing time using only intraresidue peaks

4) overall scaling using only intraresidue NOE peaks

5) scaling per mixing time using only interresidue peaks

6) overall scaling using only interresidue NOE peaks

7) scaling per mixing time, intra- and interresidue peaks scaled separately

8) overall scaling, intra- and interresidue peaks scaled separately

9) scaling per mixing time, separately for short-, medium- and long-range NOEs. The classification is based on the distances in the structure. The criteria are: short if \( d < 2.8 \, \text{Å} \), medium if \( 2.8 \, \text{Å} < d < 3.5 \, \text{Å} \) and long if \( d > 3.5 \, \text{Å} \).

10) overall scaling, separately for short-, medium- and long-range NOEs

In general, if build-up series, consisting of several NOESY spectra at various mixing times recorded in one measuring session, are available, an overall scaling should be preferred. A separate scaling per mixing time can be used if the spectra have been recorded and/or processed with different parameters. An overall scaling should also be preferred when the overall correlation time is not well known. Definitions 7 to 10, in which different scaling factors are used for various types of NOE peaks, have been implemented with the idea to correct for different motional behaviours, but have not been tested thus far. We recommend the use of definition 0 or 2 which have been extensively tested and have shown to give good results for the refinement of several proteins in our group.

For scaling of the 3D NOE-NOE intensities four methods are implemented:

0) with reference peaks

1) with all peaks

2) with intraresidue back transfer peaks

3) with intraresidue peaks.

4.3 NOE restraining potentials
Several NOE restraining potential are implemented in DINOSAUR. Although we obtained the best results with the potential of Eq. 21 (definition 6 below, which we recommend to use), other definitions have been implemented for testing and development purpose and are still available in the present version of DINOSAUR. The NOE restraining potentials can be represented in a general form by

\[ V_{NOE} = \frac{1}{\text{Norm}} \sum_{ij}^! w_{ij} \cdot A_{ij}^{\text{theo}} - A_{ij}^{\exp} \cdot y \]  \hspace{1cm} (49) \]

where \( A_{ij}^{\exp} \) and \( A_{ij}^{\text{theo}} \) represent the experimental and theoretical NOE intensities between protons \( i \) and \( j \), respectively, \( f \) is a scaling factor, \( w_{ij} \) a weighting function and \( \text{Norm} \) a normalisation factor. Often a quadratic potential is used (\( y=2 \)), directly with the NOE intensities (\( x=1 \)) (Yip and Case, 1989; Baleja et al., 1990; Bonvin et al., 1991a; Mertz et al., 1991) or with their sixth-root (\( x=1/6 \)) (Nilges et al., 1991a) or inverse sixth-root (\( x=-1/6 \)) (Stawarz et al., 1992), the last two definitions being closer to a standard distance based potential.

Ten different definitions are presently implemented in DINOSAUR. These are:

1) absolute error definition, normalised by the sum of the experimental NOE intensities, weighting factors equal to 1 or used-defined value:

\[ V_{NOE} = \sum_{ij}^! j w_{ij} \cdot A_{ij}^{\exp} - A_{ij}^{\text{theo}} \cdot y^2 \]  \hspace{1cm} (50) \]

\( \square_j \) is a switching factor used to switch of the function when the theoretical NOE intensities are within the experimental errors (see paragraph 3.1.5.1).

2) absolute error definition, normalised by the sum of the theoretical NOE intensities, weighting factors equal to 1 or used-defined value:

\[ V_{NOE} = \sum_{ij}^! j w_{ij} \cdot A_{ij}^{\text{theo}} - A_{ij}^{\exp} \cdot y^2 \]  \hspace{1cm} (51) \]

3) relative error definition with respect to the experimental NOE intensities, weighting factors equal to 1 or used-defined value:

\[ V_{NOE} = \sum_{ij}^! j w_{ij} \cdot A_{ij}^{\exp} - A_{ij}^{\text{theo}} \cdot y^2 \]  \hspace{1cm} (52) \]

4) relative error definition with respect to the theoretical NOE intensities, weighting factors equal to 1 or used-defined value:

\[ V_{NOE} = \sum_{ij}^! j w_{ij} \cdot A_{ij}^{\text{theo}} - A_{ij}^{\exp} \cdot y^2 \]  \hspace{1cm} (53) \]
5) relative error definition with respect to the sum of theoretical and experimental NOE intensities, weighting factors equal to 1 or used-defined value:

\[
V_{\text{NOE}} = \frac{1}{\sqrt{\sum (|A_{ij}^{\text{theo}}| - |A_{ij}^{\text{exp}}|)^2}}
\]

(54)

6) $\|e\|^2$ like definition with errors defined as the sum of a noise level and of a relative error on the experimental NOE intensities (same as Eq. 21)

\[
V_{\text{NOE}} = \frac{1}{\sqrt{\sum (|A_{ij}^{\text{exp}}| + \sqrt{\text{noise}})^2}}
\]

(55)

with

\[
w_{ij} = \frac{1}{\sqrt{N! + \sqrt{\text{noise}}}}
\]

$N$ is the experimental noise level and $\square_{\text{exp}}$ a relative error on the experimental NOE intensities to account for integration errors and T1 noise.

7) $\|e\|^2$ like definition with errors defined as the sum of the experimental (noise + relative error on $A^{\text{exp}}$) and theoretical (relative error on $A^{\text{theo}}$) errors:

\[
V_{\text{NOE}} = \frac{1}{\sqrt{\sum (|A_{ij}^{\text{exp}}| + \sqrt{\|e\|^2})}}
\]

(56)

with

\[
w_{ij} = \frac{1}{\sqrt{N! + \sqrt{\text{noise}}}}
\]

$\square_{\text{theo}}$ is a relative error on the theoretical NOE intensities.

8) $\|e\|^2$ like definition with errors defined as the sum of the experimental (noise + relative error on $A^{\text{exp}}$) and theoretical (absolute error on distances) errors:

\[
V_{\text{NOE}} = \frac{1}{\sqrt{\sum (|A_{ij}^{\text{exp}}| + \sqrt{\|e\|^2})}}
\]

(57)

with

\[
w_{ij} = \frac{1}{\sqrt{N! + \sqrt{\text{noise}}}}
\]

dr indicates the absolute error on the distances $r$ in the structure. NOE peaks corresponding to short distances in the structure will be more heavily weighted.

9) absolute error definition as in definition 1) but closer to a distance based potential by taking the sixth-root of the NOE intensities:
implemented in DINOSAUR. The two definitions are:

based on the power series of the exponential (Eq. 25) (first order approximation).

We chose therefore to use an analytical solution (see Eq. 26) but closer to a distance based potential by taking the sixth-root of the NOE intensities:

\[
V_{\text{NOE}} = \sum_{ij} \frac{1}{w_{ij}} \exp \left( - \frac{1}{A_{ij}} \right) \] 

(58)

10) \[\Box^2\] like definition as in definition 6) but closer to a distance based potential by taking the sixth-root of the NOE intensities:

\[
V_{\text{NOE}} = \sum_{ij} \frac{1}{w_{ij}} \exp \left( - \frac{1}{A_{ij}} \right) \] 

(59)

with

\[
w_{ij} = \frac{1}{N! + 1! A_{ij}} \exp \left( - \frac{1}{A_{ij}} \right)\]

Note: Similar restraining potential definitions are implemented in the 3D DINOSAUR version. Definitions 1 to 8 are the same as in the 2D version and can be derived from the above definitions by replacing the indexes \(ij\) by \(ijk\). In definition 8, the absolute error on the distances is the sum on the two distances defining a 3D NOE-NOE peak, i.e. \((dr_{ij} + dr_{jk})\). The potential definition 9 in the 3D DINOSAUR version corresponds to definition 10 above and the 3D potential definition 10 uses the twelfth-root of the NOE intensities in place of the sixth-root in definition 10 above.

4.4 Methods for the computation of the NOE forces and of the gradient of the theoretical NOE intensities \#A_{ij}

Calculation of the NOE forces (see Eq. 27) requires the calculation of the gradient with respect to the proton coordinates in the structure(s) (\[\Box\]=(d/dx, d/dy, d/dz)). This gradient can be expressed as a product consisting of the gradient of the NOE potential with respect to the theoretical NOE intensities and of the gradient of the theoretical NOE intensities with respect to the proton coordinates.

\[
\#\Box V_{\text{NOE}} = \frac{dV_{\text{NOE}}}{dA_{\text{theo}}} \frac{dA_{\text{theo}}}{dr_{\Box}}
\]

(60)

The first part can be easily derived from the NOE potential definitions given above (it is calculated in DINOSAUR in the subroutine \texttt{sgrad}). The second part is more complicated. An analytical solution (see Eq. 24) has been proposed (Yip and Case, 1989) but is extremely time consuming. We chose therefore to use a simple approximation based on the first two terms of the power series of the exponential (Eq. 25) (first order approximation). Another approximation based on the first three terms of the power series (second order approximation) has also been implemented in DINOSAUR. The two definitions are:

1) first order approximation (subroutine \texttt{sg2spin}), see Eq. 26, paragraph 3.1.5.1. With this approximation, one NOE peak gives only forces on its corresponding protons.
2) second order approximation (subroutine `sg2ord`)

\[
\#A = \#(1 - \mathbf{R} + \frac{1}{2} \mathbf{R}^2) = -\mathbf{R} + \frac{1}{2} \mathbf{R}^2 \quad (61)
\]

With this approximation, one NOE peak will generate forces on all protons in the molecule. For most of them, however, the contribution will be very small or zero, since the NOE gradient has an even stronger dependency on the distances \(r^7\) than the NOE intensities \(r^6\). This definition required more cpu time than the simple (and effective) approximation in definition 1).

The cpu time required for the computation of the NOE forces in definition 1) is proportional to the number of NOEs \(N_{\text{peaks}}\) and negligible compared to the time required for the computation of the theoretical intensities. In definition 2) however, the time needed is increased by a factor proportional to the number of protons in the system \(N_{\text{peaks}} \times N_{\text{protons}}\). Although the second order approximation of the NOE gradient is implemented in DINOSAUR, it has not been used thus far for the refinement of proteins with real data and we recommend the use of the simple first order approximation of definition 1).

4.5 \(R\) factor definitions

Several \(R\) factor definitions are implemented in DINOSAUR for the comparison of experimental and theoretical NOE data. These are:

1) and 2) absolute errors definition (Gonzales et al., 1991)

\[
R = \frac{\sum_{ij} w_{ij} |A_{ij}^{\text{theo}}| - |A_{ij}^{\text{exp}}|}{\sum_{ij} w_{ij} |A_{ij}^{\text{exp}}|} \quad (62)
\]

3) and 4) relative errors definition

\[
R = \frac{\sum_{ij} w_{ij} (|A_{ij}^{\text{theo}}| - |A_{ij}^{\text{exp}}|)}{\sum_{ij} w_{ij} |A_{ij}^{\text{exp}}|} \quad (63)
\]

5) and 6) absolute errors definition with the sixth-root of the NOE intensities (Gonzales et al., 1991; Thomas et al., 1991)
with $w_{ij} = \frac{1}{\text{Noise} + A_{ij}^{\text{exp}}}$
4.6 Architecture of the DINOSAUR routines and programs

The major part of the parameters and arrays are passed in common blocks between the different DINOSAUR routines and are declared in two include files: **noere.inc** and **noecom.inc** for 2D NOE refinement and **noe3re.inc** and **noe3com.inc** for 3D NOE refinement. The architecture for 2D and 3D refinement is the same with some changes in the codes. The names of the routines and programs are almost similar: a 3 in the name indicates a 3D NOE refinement routine or program. The DINOSAUR programs and routines have the following architecture:

- **DINOSAUR main routine fnoe** (time or ensemble averaging of distances or NOEs, calculation of theoretical NOE intensities, scaling, dynamic assignments, calculation of the forces). fnoe calls the following routines:
  * **readno**: reads DINOSAUR input file
  * **smkpoi**: creates a pointer list in case of matrix contraction (fast methyl rotation)
  * **rdshft**: reads chemical shifts and leakage rates file
  * **rds2mx**: reads the order parameters matrix
  * **srmatavd**: creates the relaxation matrix
  * **sdiag**: exact NOE calculation
    - EISPACK diagonalisation routines
    - CONVEX library diagonalisation routines
  * **s2spn**: NOE calculation with two spin approximation
  * **ssexp**: NOE calculation with simplified expansion of exp
  * **s2ord**: NOE calculation with second order expansion of exp
  * **syip**: NOE calculation with second order perturbation expansion of exp
  * **sfndh**: finds neighbours within two spheres around the protons defining a NOE peak
  * **sdiasmc**: calculation of NOE intensities using spherical cut-offs
  * **sster**: dynamic assignments
  * **sscalt**: calculates scaling factor with reference peaks
  * **sggrad**: calculates gradient and forces. This routine calculates only the first term in Eq. 60 ($dV_{NOE}/dA^{\text{theo}}$) and calls itself two routines for the calculation of the gradient of the theoretical NOE intensities:
    * **sg2spn**: calculates the derivative of the NOE intensities using two-spin approximation.
    * **sg2ord**: calculates the derivative of the NOE intensities using a second order approximation.

- **Interface routines** (extract and generate hydrogen atoms if necessary, call the DINOSAUR main routine fnoe and set back forces on real atoms if necessary):
  * **noegro**: (GROMOS interface), calls subroutines:
    - **ixh**: (generate hydrogen atoms)
    - **stbgra**: (set back forces on real atoms)
  * **noedg**: DDD interface, calls subroutine:
    - **stbgrd**: (set back gradient on real atoms if necessary)
* **noedisco**: DISCOVER interface (this routine can be used as interface for implementation of DINOSAUR in "all-atoms" molecular dynamics programs).

Note: The routines listed above are only those required for direct NOE refinement. Several other DINOSAUR routines, which are used in set-up and analysis procedures, have not been listed here.

### Refinement programs:
- **dinodd**: Distance bound Driven Dynamics (DDD)
- **dinoem**: Energy minimisation (GROMOS)
- **dinomd**: Molecular Dynamics (GROMOS)

These programs make use of specific GROMOS or DDD routines that will not be distributed with DINOSAUR without the specific agreement of the respective authors.

For information on GROMOS contact: BIOMOS b.v., Biomolecular Software Laboratory of Physical Chemistry University of Groningen Nijenborgh 16 9747 AG Groningen The Netherlands

For information on DDD contact: Dr. R.M. Scheek Laboratory of Physical Chemistry University of Groningen Nijenborgh 16 9747 AG Groningen The Netherlands

### Other DINOSAUR programs (set-up, analysis):
- **ananoe**: analysis in terms of R factors and energies of the NOE trajectories and theoretical NOE files.
- **avrfac**: calculation of R factors from an ensemble of structures or proton trajectories.
- **chamb**: In case of a symmetrical dimer chamb checks the assignments (inter- or intra monomer) by comparing the distances and the NOE energies for both possible assignment.
- **chkass**: analyses the results of the dynamic assignments procedure
- **cvhco**: conversion binary <-> formatted of proton trajectory files
- **cvnoe**: conversion binary <-> formatted of NOE trajectory files
- **dgrfac**: R factors calculations from a DG/DDD coordinates file
- **dinoin**: creates of modifies the DINOSAUR input file. Contains several routines for reading various coordinates files, NOE result files, stereospecific assignment files. (For a description of these files see
chapter 6)

* **discorf**: \( R \) factors calculations from a DISCOVER coordinates file
* **genens**: combines several structures or generates multiple copies of one structure and shifts them for ensemble-averaged NOE refinement
* **grrfac**: \( R \) factors calculations from a GROMOS coordinates file
* **mkemin**: generates an input file for program dinoem
* **mkhco**: generates and extract proton coordinates from GROMOS coordinates or trajectory files.
* **mkhcoin**: generates an input file for program mkhco
* **mkhtyp**: creates a proton codes file for use with the GROMOS based programs dinoem and dinomd.
* **mkmdin**: generates an input file for program dinomd
* **mkrfin**: generates the input file for program grrfac
* **seldim**: for symmetrical dimers: selects NOEs, duplicate them (both inter- and intra monomer) and creates a NOE file that can be read by dinoin for further analysis with chamb.
* **selpks**: selects NOEs from DINOSAUR input file for analysis with ananoe
* **splitco**: splits coordinates file from ensemble-averaged NOE refinement

### 4.7 Atom names conventions in DINOSAUR

Explicit proton names are only needed when creating the DINOSAUR input file (procedure **dinoin**), which is read by most of the DINOSAUR programs. Once the DINOSAUR input file has been created, the DINOSAUR programs and routines will not use anymore proton names but proton numbers. DINOSAUR uses an internal numbering, starting from 1 with sequential numbers. The name information is needed to recognise methyl groups, aromatic rings, diastereotopic proton pairs or groups (methylene proton pairs or methyl groups in Val and Leu) and polar protons.

Dynamical averaging (see section 3.1.2) can be used to describe fast methyl rotation and 180° aromatic ring flips common in proteins. For each proton, the related information is stored in the DINOSAUR input file. If DINOSAUR does not recognise a methyl or aromatic group in your molecule, you can always edit the input file and specify the type of averaging and the averaging partners. For a description of the corresponding parameters in the input file see section 6.1: "2D DINOSAUR input file". Information about methyl and aromatic groups in DINOSAUR can be found in the include file "metharo.inc". This file also contains the names of the residues with methyl and aromatic groups. You can edit this file and define new names specific for your molecule (in such a case, don't forget to also modify the dimensions of the corresponding arrays) and recompile DINOSAUR (see section 4.8).

A dynamic assignment procedure (see section 3.1.5.1) can be applied to the NOE peaks involving aromatic ring protons or diastereotopic protons for which no stereospecific assignment is available. DINOSAUR recognises such protons using the information found in the include file "protoninfo.inc". Like for methyl and aromatic groups it is possible to edit this file and modify it for your particular problem.
When working with D2O data, DINOSAUR offers the possibility to skip the exchangeable (polar) protons. Information on such protons is also found in the "protoninfo.inc" include file.

The present DINOSAUR implementation recognises standard DDD, GROMOS and DISCOVER (Biosym) proton names.

4.8 Implementation in an "all atoms" refinement program

The DINOSAUR routines can be easily implemented in "all atoms" refinement programs (EM, MD). An interface routine should be used to extract the proton coordinates and pass them, together with the number of molecules (> 1 for ensemble-averaged NOE restraints), to the DINOSAUR leading routine fnoe. The call to fnoe is:

```fortran
call fnoe(xh,yh,zh,nh,enoe,nmol)
```

where xh, yh and zh are arrays of dimension (max_number_of_molecules,max_number_of_protons) containing the x, y and z proton coordinates, respectively, for nmol molecules. nh is the number of proton per molecule and enoe will return the NOE restraint energy. The forces (and not the gradient!) are returned in common blocks in the arrays drdx, drdy and drdz. It is therefore necessary to include the DINOSAUR include files noere.inc and noecom.inc in the interface routine. Output control can also be implemented at this level like, for example, the writing of a NOE trajectory file. An example of an "all atoms" interface routine is given in noedisco.f.

You should also include in your main program at the end, before the stop statement, a call to the DINOSAUR routine modiin:

```fortran
call modiin(90)
```

This routine writes the results of the dynamic assignments procedure to the DINOSAUR input file and generates if necessary a file with time-averaged NOE intensities that is required for proper continuation of a MD simulation in case of time-averaged NOE restraints.

**Note:** DINOSAUR delivers NOE forces and not gradients. If your main program requires gradients change the sign of the arrays drdx, drdy and drdz.

In short, the interface routine should:
- include the DINOSAUR include files noere.inc and noecom.inc
- extract the proton coordinates for one or several molecules
- call the DINOSAUR main routine fnoe
- eventually correct NOE forces by a factor -1 to obtain the gradients
- control the output (e.g. NOE trajectory)
- set back the NOE forces on the real atoms (from DINOSAUR internal numbering to main program numbering)
4.9 Installation under UNIX

For setting up DINOSAUR in a UNIX environment you have to copy the files from tape to disk with the `tar xv` command. You will notice the existence of a file `readme` and it is advisable to read this file before you continue. The compilation and linking of the programs is performed by executing the `install` script found in the `dino` directory, but you will first have to modify a few things to adapt dinosaur to your environment. In a few files, directories are defined as `$(HOME)/dino/...` If you have copied dinosaur into your home directory things will work fine otherwise you will have either to modify the files where the `$(HOME)/...` construction is used (these are: `dino/src/makefile` `dino/bin/_rcs` and `dino/bin/_rcsinit`) or simply create a link in your home directory to the dinosaur directory with the command

```bash
% ln -s homedirectory_of_dinosaur/dino dino
```

The DINOSAUR source, script and include files are placed under Revision Control System (RCS). RCS is very useful tool for programmers to keep trace of older version and check differences between two versions. A few RCS scripts, customised for DINOSAUR, can be found in the `dino/bin` directory. A short description of RCS and how to work with it can be found in `dino/bin/README`. For a more complete description refer to your manual pages. The file revisions are stored in three RCS directories (`dino/src/RCS`, `dino/csh/RCS` and `dino/include/RCS`). The `install` script will ask if you want to check that the most recent versions of the DINOSAUR files are used for compilation. This is done with RCS. If you don't want to use RCS you can just delete the RCS directories to save memory.

You can now install DINOSAUR by executing the `install` script:

```bash
% install
```

---

 DINOSAUR installation script
 ----------------------------------

 Reset DINOSAUR (will delete all executable and libraries) (y,n) ?
 Your choice ?
 y

 Make update using RCS (revision control system) to make sure that DINOSAUR will be compiled with the most recent versions (y,n) ?
 (only valid if DINOSAUR files placed under RCS control)
 Your choice ?
 y
 ......

 Modify precision of GROMOS part in DINOSAUR version (y,n)?
 y
 Single (s) or double precision (d) version (applies to GROMOS part)?
 d
This will only affect the GROMOS codes and the interface routine between DINOSAUR and GROMOS. The NOE calculations in DINOSAUR are all performed in double precision.

Next step is compiling:  
1) interactively  
2) in background  
Your choice ?  
1

Compile:  
1) 2D and 3D DINOSAUR versions  
2) only 2D version  
3) only 3D version  
Your choice ?  
1

DINOSAUR installed

The DINOSAUR programs will be compiled interactively. The script first checks if the directory is present. The GROMOS and DDD codes are found in this directory. If you didn't receive the GROMOS and DDD parts, compilation will still take place except for the refinement programs based on DDD (dinodd) and on the GROMOS molecular dynamics and energy minimisation codes (dinomd and dinoem). Note that make will ignore error messages and you should check the output to make sure that everything has been compiled properly. The makefile in dino/src will work on Silicon Graphics computers. Modify it if necessary before compiling on other machines.

Before using DINOSAUR you need to modify the DINOSAUR set-up file dino/csh/dinosaur to define the directories of DINOSAUR executables and scripts:

```
setenv  DINOSAURSRC   homedirectory_of_dinosaur/dino/src
setenv  DINOSAURCSH   homedirectory_of_dinosaur/dino/csh
```

Your are now ready to use DINOSAUR:

to set-up:
```
% alias dinosaur  source homedirectory_of_dinosaur/dino/csh/dinosaur
```
to start:
```
% dinosaur       ---- directories and aliases defined  
% dinoset        ---- define molecule name  
% readme         ---- general information  
```

A short description of the DINOSAUR procedures and commands can be obtained with remind:

```
% remind
*******************************************************************************
#
# DINOSAUR  Alexandre Bonvin
#       University of Utrecht
#       The Netherlands
# $Date: 93/10/15 14:39:25 $  
#```
# Direct NOE Simulation Approach for Unbelievable Structure Refinement
#
#*****************************************************************************
#
# set-up :
# ********
# - source dinosaur  (set-up DINOSAUR procedures)
# - dinoset         (defines molecule name)
# - dinodir         (defines working directory)
# - dinoin          (creates or modifies input file)
# - mkhco           (generates proton only file (.mdh) or proton
#                     trajectory file from GROMOS coordinates or
#                     trajectory files)
# - mkhtyp          (generates proton code file (.htyp) for
#                     refinement with dinoem or dinomd)
#
# refinement procedures/programs :
# *******************************
# - dinodd           (NOE driven dynamics)
# - dinomd           (GROMOS molecular dynamics)
# - dinoem           (GROMOS energy minimisation)
#
# analysis :
# ***********
# - rfacav
# or rfacdg
# or rfacgr
# or rfacdis        (analysis R-factors)
# - rnoise          (R-factors as function of experimental error)
# - selpks          (selects peaks for analysis with ananoe)
# - ananoe          (analysis NOE trajectory)
# - cvnoe           (conversion bin<-->form of NOE trajectories)
# - cvhco           (conversion bin<-->form of proton
#                     trajectories)
# - chkass          (analysis result from dynamic assignment)
#
# special for dimer molecule :
# *******************************
# - seldim          (selects NOEs and duplicates them to generate
#                     both intra and inter monomers peaks)
# - chamb           (checks assignments by analysis of distances
#                     and NOE energies for the two possible
#                     assignments (intra or inter monomers)
#
# special for ensemble refinement :
# *******************************
# - genens          (generates an ensemble by duplicating one
#                     molecule or combining several different
#                     ones).
# - splitco         (inverse of GENENS, splits ensemble
coordinates files into several files with each one solute molecule.)

# aliases:
# *********
#
# alias readme 'more $DINOSAURCSH/readme'
# alias remind 'more $DINOSAURCSH/remind'
# alias dinoset 'source $DINOSAURCSH/dinoset.csh'
# alias dinodir 'source $DINOSAURCSH/dinodir'
# alias mkhtyp '$DINOSAURSRC/mkhtyp'
# alias mkhco 'source $DINOSAURCSH/mkhco.csh'
#
# 2D NOE refinement
# ---------------
#
# alias dinoin '$DINOSAURSRC/dinoin'
# alias dinodd '$DINOSAURSRC/dinodd.csh'
# alias dinomd '$DINOSAURSRC/dinomd.csh'
# alias dinoem '$DINOSAURSRC/dinoem.csh'
# alias dinodisco '$DINOSAURSRC/dinodisco.csh'
# alias rfacav '$DINOSAURSRC/rfacav.csh'
# alias rfacdg '$DINOSAURSRC/rfacdg.csh'
# alias rfacgr '$DINOSAURSRC/rfacgr.csh'
# alias rfacdis '$DINOSAURSRC/rfacdis.csh'
# alias selpks '$DINOSAURSRC/selpks'
# alias seldim '$DINOSAURSRC/seldim'
# alias chamb '$DINOSAURSRC/chamb'
# alias ananoe '$DINOSAURSRC/ananoe'
# alias chkass '$DINOSAURSRC/chkass'
# alias genens '$DINOSAURSRC/genens'
# alias splitco '$DINOSAURSRC/splitco'
# alias cvnoe '$DINOSAURSRC/cvnoe'
# alias cvhco '$DINOSAURSRC/cvhco'
#
# 3D NOE-NOE refinement
# ---------------------
#
# alias dinoin '$DINOSAURSRC/dino3in'
# alias dinodd '$DINOSAURSRC/dino3dd.csh'
# alias dinomd '$DINOSAURSRC/dino3md.csh'
# alias dinoem '$DINOSAURSRC/dino3em.csh'
# alias dinodisco '$DINOSAURSRC/dino3disco.csh'
# alias rfacav '$DINOSAURSRC/rfac3av.csh'
# alias rfacdg '$DINOSAURSRC/rfac3dg.csh'
# alias rfacgr '$DINOSAURSRC/rfac3gr.csh'
# alias rfacdis '$DINOSAURSRC/rfac3dis.csh'
# alias ananoe '$DINOSAURSRC/ananoe3'
# alias chkass '$DINOSAURSRC/chkass3'
#
#*******************************************************************************

Before you start you have to create several files in your current directory. You need:
- the experimental NOEs. DINOSAUR can handle BUILDUP type NOE files, free format NOE files and BIOSYM format files (peaks and assignments files) for 2D data and REGINE type and free format NOE files for 3D data.
- a proton coordinates file. This can be a file in GROMOS/DDD format with only the protons coordinates or a DISCOVER file.
- a chemical shift file (option)
- an order parameter file (option)
- a stereoassignment file (option)
A description of these files can be found in chapter 6 "Important files in DINOSAUR".

**Note:** If you use a GROMOS program, you need a proton codes files. GROMOS uses united atoms and contains thus only polar protons, the apolar protons will be generated at every step during the refinement and the forces must be set back on the real atoms. You need therefore this file. It can be created with program mkhtyp. This program requires a standard GROMOS coordinates file and a proton only files.
5. DINOSAUR procedures

This chapter gives a description of the procedures in DINOSAUR (examples will be given in chapter 7). After defining your DINOSAUR environment with

```
% source homedirectory_of_dinosaur/dino/csh/dinosaur
```

aliases are defined for the different procedures. These are:

**set-up:**
- `dinoset` (defines molecule name)
- `dinodir` (defines working directory)
- `dinoins` (creates or modifies input file)
- `mkhco` (generates proton only file (.mdh) or proton trajectory file from GROMOS coordinates or trajectory files)
- `mkhtyp` (generates proton code file (.htyp) for refinement with GROMOS)

**refinement:**
- `dinodd` (NOE driven dynamics)
- `dinoem` (GROMOS energy minimisation)
- `dinoemd` (GROMOS molecular dynamics)

**analysis:**
- `rfacav` (R-factors calculation from a proton trajectory file)
- `rfacdg` (R-factors calculation from a DDD coordinates file)
- `rfacgr` (R-factors calculation from a GROMOS coordinates file)
- `rfacdis` (R-factors calculation from a DISCOVER coordinates file)
- `rnoise` (R-factors as function of experimental error)
- `selpks` (selects peaks for analysis with ananoe)
- `ananoe` (analyses a NOE trajectory)
- `cvnoe` (conversion bin<-->form of NOE trajectories)
- `cvhco` (conversion bin<-->form of proton trajectories)
- `chkass` (analyses results from dynamic assignments)

**special for ensemble refinement:**
- `genens` (generates an ensemble by duplicating one molecule or combining several different ones).
- `splitco` (inverse of GENENS, splits ensemble coordinates files into several files with each one solute molecule.)

**special for symmetrical dimers:**
- `seldim` (selects NOEs and duplicates them to generate both intra- and intermonomers peaks).
- `chamb` (checks inter/intramonomer assignments by analysis of distances and NOE energies)
5.1 Set-up

5.1.1 dinoset (and filenames conventions)

This procedure defines the molecule name. This name is used as default in the other procedures. The various DINOSAUR scripts will ask for filenames and the default name plus an extension will be used if you just type return (<CR>). When starting DINOSAUR the default name is set to "Flintstone". You can change it with dinoset. The extension usually indicates the type of file (NOEs, coordinates, order parameters,...). The following conventions are used as default for the extensions:

DINOSAUR related files:
- Flintstone.dis --- file with NOE distances from structure
- Flintstone.hco --- proton coordinates trajectory/ensemble file
- Flintstone.hcoin --- input file for program mkhco that generates a proton coordinates trajectory/ensemble file
- Flintstone.inp --- DINOSAUR input file
- Flintstone.mdh --- proton only coordinates file (GROMOS format)
- Flintstone.noeavin --- averaged NOEs/distances input file for time-averaged NOE refinement
- Flintstone.noeavout --- averaged NOEs/distances output file for time-averaged NOE refinement
- Flintstone.noetraj --- NOE trajectory file
- Flintstone.rfac --- R factors file
- Flintstone.rfin --- R factors calculations input file (procedure rfacgr)
- Flintstone.s2 --- order parameter file
- Flintstone.shift --- file with chemical shifts and additional leakage rates
- Flintstone.theno --- file with theoretical NOE intensities

GROMOS related files:
- Flintstone.dihre --- dihedral restraints file
- Flintstone.disre --- distance restraints file
- Flintstone.em --- GROMOS EM coordinates file
- Flintstone.emin --- GROMOS energy minimisation input file
- Flintstone.emout --- GROMOS energy minimisation output file
- Flintstone.htyp --- file with GROMOS protons codes for setting back the NOE forces on real atoms
- Flintstone.md --- GROMOS start coordinates file
- Flintstone.mdin --- GROMOS molecular dynamics input file
- Flintstone.mdnew --- GROMOS MD coordinates file
- Flintstone.mdout --- GROMOS molecular dynamics output file
- Flintstone.posre --- position restraints file
- Flintstone.topo --- molecular topology
- Flintstone.traj --- coordinates trajectory file

Other files:
- Flintstone.car --- DISCOVER coordinates file
- Flintstone.dg --- DDD start coordinates file
5.1.2 dinodir

Dinodir defines or modifies the DINOSAUR working directory which is be used when creating the script files for the different procedures to include a “chdir” command at the beginning of the file.

5.1.3 dinoin

Dinoin creates or modifies the DINOSAUR input file used in almost all DINOSAUR programs. If you create a new file, you need a NOE file (BUILDUP, free or BIOSYM format), a proton only file (GROMOS/DDD format) or DISCOVER structure and optionally a stereospecific assignments file otherwise you can just modify an old input file. For a description of the mentioned files see chapter 6 “Important files in DINOSAUR”.

5.1.4 mkhco

Mkhco creates a proton only file (.mdh) and a proton trajectory file (.hco) from GROMOS coordinates/trajectory files. The input file required by the program can be created with the program mkhcoin which is called within the script. To execute this script you need GROMOS coordinates/trajectory files and a GROMOS molecular topology file. A job file called “mkhco.job” will be created that you can execute directly or in background.
5.1.5 mkhtyp

Mkhtyp creates a proton codes file (.htyp). This file is only used in the GROMOS programs (dinoem and dinomd). GROMOS uses united atoms and contains thus only polar protons. The apolar protons are generated at every step during the refinement and the forces must be set back on the real atoms. The proton codes file (.htyp) contains the information about the existing and virtual protons in GROMOS. The codes are the same as those used in GROMOS for distance restraining. Each proton is characterised by five numbers (codes): the first four correspond to the real GROMOS atoms (i,j,k,l0 of which the proton position is a function and the last one defines the type of proton. The proton types used in DINOSAUR are:

0 = real hydrogen atom (i) (e.g. polar protons)
1 = virtual H, aliphatic CH1 (i, j, k, l) (e.g. H1 protons)
2 = virtual H, aromatic CH1 (i, j, k) (e.g. H1 and H2 protons in Tyr)
4 = virtual H, aliphatic CH2 (i, j, k) (e.g. H1 and H2 in Cys)
5 = virtual H, CH3 groups (i, j)

(i) is the number of the atom to which the proton is attached (or the proton number itself for type 0). The other atoms (j, k, l) are the number of the other atoms of which the proton position is a function. For H1 (aliphatic CH1, type 1) in amino acids (except in Gly), for example, the proton position is calculated from the C, Cα, N and Cβ coordinates and the NOE forces need thus to be set back on these "real" GROMOS atoms. For a more complete description of the virtual atoms in GROMOS see de Vlieg et al. (1986).

To run mkhtyp you need a GROMOS coordinates file and a proton only file (.mdh).

![Diagram](image-url)  
*Figure 5: Overview of the mkhco procedure.*

![Diagram](image-url)  
*Figure 6: Overview of the mkhtyp procedure.*
5.2 Refinement

5.2.1 dinodd

Dinodd creates a script file called dinodd.job to run the NOE driven dynamics program. This program is based on the Distance bounds Driven Dynamics (DDD) (Kaptein et al., 1988), a Newtonian dynamics simulation using a chiral constraint function and the distance constraint function of Eq. (68)

\[ V_{dis} = K_{dis} \begin{cases} \frac{1}{2} (d_{ij}^2 - u_{ij}^2)^2 & \text{if } d_{ij} > u_{ij} \\ \frac{1}{2} (d_{ij}^2 - l_{ij}^2)^2 & \text{if } d_{ij} < l_{ij} \end{cases} \]  

(68)

where \( u \) and \( l \) represent upper and lower bounds, respectively, and \( d \) the actual distance. All information about the covalent structure is represented by upper and lower bounds between atoms separated by less than 4 bonds.

The standard DG files used in this procedure are:
- a binary file with upper and lower bounds (DGUL or DGSMUL)
- a coordinate files (ECOOR)
- a file with chiral constraints (DGCHIR)
For a description of these files refer to the DG manual.

An error function called NOERE has been defined in DDD which allows for direct NOE refinement. The DG coordinate file must be created with a library containing all protons. Dinodd will generate protons if a pseudo atom is encountered (Q3 or Q2). It uses therefore the information about the connectivities. This can give chirality problems for the methylene protons.

To avoid such problems use the consistent DG library, “DGlibrary.all” which can be found in “dino/data” and remove the pseudo atoms for which the corresponding protons are present. In addition to the DG files the standard DINOSAUR files are requested: the DINOSAUR input file, a chemical shift file (.shift) (option) and the order parameters file (.s2) (option).

**DG/DDD files:**
- dinodd input file (.dgin) defining:
  - DG input coordinates file (ECOOR)
  - DG output coordinates file
  - distance matrix of upper and lower bounds (DGSMUL)
  - chiral constraints file (DGCHIR)
  - simulation parameters

**DINOSAUR files:**
- DINOSAUR input file (.inp)
- chemical shift file (.shift) *
- order parameter file (.s2) *

**Figure 7:** Overview of the dinodd procedure. The file indicated by * are optional.
5.2.1 dinoem

Dinoem sets up a NOE restrained energy minimisation using the GROMOS force field (van Gunsteren et al., 1987). In this force field only polar protons are treated explicitly, the others are incorporated using the united atom technique. Dinoem creates or modifies the GROMOS input file and creates a script file called dinoem.job. This procedure is based on the corresponding GROMOS program (proem). It allows for NOEs, distances, dihedral angles and position restraining. For a description of the GROMOS part of the program see GROMOS manual. This procedure can also be used for standard distance restrained refinement or for free energy minimisation. The type of restraining is controlled by parameters in the GROMOS input file. (For a description of the minimisation parameters see the GROMOS manual).

Figure 8: Overview of the dinoem procedure. The file indicated by * are optional.

5.2.3 dinomd

Dinomd sets up a NOE restrained molecular dynamics simulation using the GROMOS force field (van Gunsteren et al., 1987). In this force field only polar protons are treated explicitly, the others are incorporated using the united atom technique. Dinomd creates or modifies the GROMOS input file and creates a script file called dinomd.job. This procedure is based on the corresponding GROMOS program (promd). It allows for NOEs, distances, dihedral angles and position restraining. This procedure can also be used for standard distance restrained refinement or for free energy minimisation. The type of restraining is controlled by parameters in the GROMOS input file. (For a description of the minimisation parameters see the GROMOS manual).
5.3 Analysis

5.3.1 rfacav

Rfacav creates a script file called avrfac.job for the calculation of $R$ factors from a proton trajectory (.hco). Several $R$ factors (.rfac) are calculated according to the definitions in section 4.5. and for uncorrected ($A^{\text{theo}}$) and corrected ($A^{\text{theo} \pm \text{exp. error}}$) theoretical NOE intensities. The experimental errors are defined as $(N + e A^{\text{exp}})$ where $N$ is a noise level and $e$ a relative error on the experimental intensities. These two parameters are defined in the DINOSAUR input file. Several type of averaging, defined by the parameter `itypav` in the DINOSAUR input file (line 10, see "Important files in DINOSAUR", section 6), are implemented:

- **1 NOE averaging**: the $R$ factors are calculated from the average NOE intensities.
  
  With this option two NOE trajectories will be generated (binary): one containing the time-average NOE intensities according to Eq. 31 (.noeav) and the other containing the instantaneous NOE intensities (.theno). These trajectories can be analysed with ananoe. This option requires the calculation of the theoretical NOE intensities for each record/structure in the proton trajectory and is therefore more time consuming than the following ones.

- **1 distance averaging as $<r>$**: the $R$ factors are calculated from an average relaxation matrix calculated from distances averaged as $<r>$. A theoretical NOE file (.theno)
and a distance file (.dis) containing the average distances in the structure corresponding to the experimental NOE intensities are generated. Since this option only requires one NOE calculation it is much more faster than the previous one. This type of averaging has however no real physical meaning and should preferably not be used.

3 distance averaging as \( <r^{-3}> \): the \( R \) factors are calculated from an average relaxation matrix calculated from distances averaged as \( <r^{-3}> \). Except for the type of distance averaging, this option is similar to the previous one. An \( <r^{-3}> \) averaging should be used in case of fast motions (faster than the overall correlation time of the molecules) and is appropriate for the analysis of an MD trajectory.

6 distance averaging as \( <r^{-6}> \): the \( R \) factors are calculated from an average relaxation matrix calculated from distances averaged as \( <r^{-6}> \). Except for the type of distance averaging, this option is similar to the two previous ones. An \( <r^{-6}> \) averaging should be used in case of slow motions (slower than the overall correlation time of the molecules) or for the analysis of an ensemble of structures.

The number of records in the proton trajectory (binary!) and the number of protons per record are given at the beginning of the trajectory file (.hco) (see Important files in DINOSAUR, section 6)

![Diagram](image)

**Figure 10:** Overview of the rfacav procedure. The file indicated by * are optional.

### 5.3.2 rfacdg

Rfacdg is the analogue of rfacav, but for a single structure in DG format. It creates a script file called dgrfac.job. If required (pseudo atoms encountered in the co-ordinates file) hydrogens will be generated. Rfacdg uses the standard DINOSAUR files. The theoretical NOE intensities and the corresponding distances are written to two output files (.theno and .dis).
5.3.3 rfacgr

Rfacgr is the analogue of rfacav, but for a single structure in GROMOS format. It creates a script file called grfacjob. Rfacgr needs a GROMOS topology file (.topo) and an input file containing the GROMOS codes of the atoms to which apolar (non-existing in GROMOS) hydrogens are attached (.rfin). This latter can be generated within the rfacgr script if necessary. These codes are used to generated the virtual hydrogens in GROMOS. The codes for five different types of protons are required:
- one (planar) hydrogen (e.g. aromatic protons)
- one (dihedral) hydrogen (e.g. H\(^{-}\) protons, except for Gly)
- two 180 degrees (dihedral) hydrogens (e.g. H\(_2\)C=\(\equiv\)C- ) (not in standard amino acids)
- three or two 120 degrees (dihedral) hydrogens (e.g. methylene and methyl protons)
- existing protons in GROMOS with a name not beginning with a H

Rfacgr requires further the standard DINOSAUR files. The theoretical NOE intensities and the corresponding distances are written to two output files (.theno and .dis).

5.3.4 rfacdis

Rfacdis is the analogue of rfacav, but for a single structure in DISCOVER format. It creates a script file called discorfac.job. Rfacdis requires the standard DINOSAUR files. The theoretical NOE intensities and the corresponding distances are written to two output files (.theno and .dis).

![Diagram](image)

**Figure 13:** Overview of the rfacdis procedure. The file indicated by * are optional.

5.3.5 rnoise

Rnoise is a program to check the effect of experimental errors on the R factors. Experimental errors on the NOE intensities set a lower limit on the R factors. Rnoise allows to calculates R factors as a function of the experimental errors defined as the sum of a noise level and a relative error on the experimental intensities. This is done by replacing the theoretical NOE intensities A\(^{\text{theo}}\) in the R factor definitions by A\(^{\text{exp}}\) ± exp. errors. Random noise is added as a Gaussian distribution by defining a noise level N and its standard deviation at half width SD\(_{\text{noise}}\) and the relative contribution is chosen randomly between ± \(\hat{\square}\) A\(^{\text{exp}}\) where \(\hat{\square}\) is the relative error. Rnoise runs only requires the DINOSAUR input file.

5.3.6 selpks

Selpks is used to select peaks from the DINOSAUR input file for analysis with ananoe (see below). Several selection criteria are implemented:
1: difference in residue numbers
2: proton type (name)
3: residue number
4: defined proton (proton name + residue number).
The selected peaks (only peak identification) are written to a file that can be read in ananoe.

5.3.7 ananoe

Ananoe analyses in term of NOE restraint energies and $R$ factors the NOE trajectories (.noetraj) (binary) created by the refinement procedures (dinodd, dinoom and dinomd) or the theoretical NOE files (.theno, .noeav) (text file in free format, see section 6) created by the procedures for the calculation of $R$ factors (rfacav, rfacdg, rfacgr and rfacdis). This is an interactive program that requires the DINOSAUR input file and theoretical NOE files (trajectory or simple NOE file). It can handle selected peaks or the entire set of experimental data. Peaks can be selected by giving peak identifications of by reading them from a file generated with selpks (section 5.3.6).

In the case of a NOE trajectory (several records), overall, inter- and intraresidue $R$ factors and NOE restraint energies per peak (only for 2D data) are calculated as function of the time and can be inspected within the program. Several output files can be generated. For the NOE restraint energies and $R$ factors results can be written as curves (text file) and as a table with tabs separator suitable for reading from graphic programs (e.g. Cricket Graph or Delta Graph on a Macintosh).

In the case of a single NOE file (only one record), overall, inter- and intraresidue $R$ factors and NOE restraint energies per peak are calculated. The NOE restraint energies can be analysed as function of the residue sequence in a one or two-dimensional way. NOE restraint energies per peak can also be obtained to identify NOE constraints with large violations.

**Figure 14:** Overview of the ananoe procedure.
5.3.8 cvnoe

Cvnoe is a simple interconversion program binary<-->formatted for the NOE trajectories generated with the refinement procedures. NOE trajectories are generated as binary files. Conversion to formatted files (intensities written with format: 8(1pe10.3)) can be useful for exporting data to other type of computers.

5.3.9 cvhco

Cvhco is a simple interconversion program binary<-->formatted for the proton trajectory files (.hco) generated with the procedure mkhco. As with the NOE trajectories, conversion to formatted files can be useful for exporting data to other types of computers. For a description of the proton trajectory file see section 6.

5.3.10 chkass

Chkass allows one to analyse the results of the dynamic assignment procedure implemented in DINOSAUR (see section 3.1.5.1). The assignment probabilities are monitored during the refinement and written to the DINOSAUR output file at the end. Chkass reads these probabilities and generates an output file containing the assignment probability expressed in percentage. The present assignment (the one in the input file) is given first and then all other possible assignments. The program always compares the possible assignment two by two, i.e. the given percentage for one assignment is always given with respect to the previously chosen one. For example, the following results mean that the first assignment was preferred in 83.2 % of the time during the refinement.

Peak #  304:
  present assignment :  35ILE  HCG11  35ILE  H
  alternate ass. #  1 :  35ILE  HCG12  35ILE  H  P =  16.8%

In another example with multiple assignments,

Peak #  38:
  present assignment :  5PRO  H  CD1  44TYR  HCE11
  alternate ass. #  1 :  5PRO  H  CD1  44TYR  HCE21  P = 100.0%
  alternate ass. #  2 :  5PRO  H  CD2  44TYR  HCE11  P =   0.2%
  alternate ass. #  3 :  5PRO  H  CD2  44TYR  HCE21  P =  34.3%

the first alternate assignment was preferred to the present assignment (the one in the DINOSAUR input file) with a probability of 100 %. Then the second alternate assignment (#2) was compared to the first alternate assignment (#1) and its probability was found to be only 0.2 %; the alternate assignment #1 was thus kept and compared to the third possible assignment. This latter has a 34.3 % probability. Assignment #1 was thus preferred during the refinement. From these results one can say DINOSAUR excluded the initial assignment and the alternate assignment #2 and preferred the two others (#1 and #3) with probabilities of 65.8 % and 34.3 %, respectively.
5.4 Special for ensemble refinement

DINOSAUR allows for ensemble-averaged NOE refinement (see section 3.1.5.3). This means that the NOE restraints energies will be calculated from an ensemble of structures with averaging at the relaxation matrix level. In the present implementation all members of the ensemble received a same weight; no Boltzmann weighting is implemented. The two following procedures allows to generate (genens) or split (splitco) an ensemble of structures in GROMOS format.

5.4.1 genens

Genens generates an ensemble of structures by creating several copies of a same structure or by combining several different structures. The program will shift each structure in the three dimensions (x,y,z) by user-defined values. This is required for the GROMOS implementation of DINOSAUR. GROMOS allows the refinement of several solutes at a same time, but we have however to make sure that the molecules don't see each other during the refinement. This can be done by shifting them in 3D space so that the distance between two structures exceeds the cut-off used for the evaluation of the non-bonded terms in the GROMOS potential. When doing this it is better to spread the structures in a symmetrical way around the origin and use small shifts to avoid numerical problems during the simulations. For example, for an ensemble-averaged NOE refinement with eight structures, these can be put on the edge of a cubic box with its centre placed at the origin. The minimal length of the box should be at least superior to the sum of the longest axis of the molecule (diameter in case of a spherical shape) and of the non-bonded cut-off used in the MD simulations.

5.4.2 splitco

Splitco will splits a GROMOS ensemble of structures into separate coordinates files.

5.5 Special for symmetrical dimers

The two programs described in this section are aimed to the assignment of inter and intra monomers NOEs in case of a symmetrical dimer. Seldim will selects peaks and duplicate them. With these peaks a new DINOSAUR input file should be created. Then the corresponding theoretical NOEs and distances should be calculated with one of the R factors calculation procedures (rfavav, rfacgr or rfadg) and a NOE energy file per peak should be created with ananoe. Chamb will use this energy file and the distance file (.dis) generated by the R factor procedures to compare the two possible assignment.
5.5.1 seldim

Seldim generates NOE files with duplicated inter/intra monomer peaks from the DINOSAUS input file. Several selection criteria are implemented:

1: difference in residue numbers
2: proton type (name)
3: residue number
4: defined proton (proton name + residue number).

The selected peaks are duplicated and written to a NOE file that can be read in dinoin to create a new input file.

5.5.2 chamb

Chamb compares inter- and intra monomer peak assignments using a NOE energy file generated with ananoe and the corresponding distance file (.dis). The user defines a minimal distance difference between the two possible assignments. If the absolute difference is larger than the defined minimal distance chamb will chose the assignment corresponding to the shortest distance. If the distance difference is smaller than the user defined value, the assignment is considered ambiguous. Ambiguous assignments and swapped assignments are written to files together with the corresponding NOE energies. A typical output of chamb looks like:

<table>
<thead>
<tr>
<th>Ambiguous assignments :</th>
<th>dis</th>
<th>Enoe</th>
<th>peak ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>present :</td>
<td>3.13</td>
<td>6.318E+00</td>
<td>10 PHE  HCE11 12 LEU HCD21</td>
</tr>
<tr>
<td>swapped :</td>
<td>3.84</td>
<td>3.285E+01</td>
<td>10 PHE  HCE11 65 LEU HCD21</td>
</tr>
<tr>
<td>present :</td>
<td>4.81</td>
<td>1.778E+01</td>
<td>10 PHE  H C2 12 LEU HCD11</td>
</tr>
<tr>
<td>swapped :</td>
<td>4.78</td>
<td>3.665E+01</td>
<td>10 PHE  H C2 65 LEU HCD11</td>
</tr>
<tr>
<td>present :</td>
<td>7.24</td>
<td>5.462E+00</td>
<td>12 LEU  HCD11 14 TRP HCZ31</td>
</tr>
<tr>
<td>swapped :</td>
<td>7.01</td>
<td>5.971E-01</td>
<td>12 LEU  HCD11 67 TRP HCZ31</td>
</tr>
<tr>
<td>present :</td>
<td>5.76</td>
<td>2.055E+01</td>
<td>12 LEU  HCD11 14 TRP HCH21</td>
</tr>
<tr>
<td>swapped :</td>
<td>6.43</td>
<td>2.041E+01</td>
<td>12 LEU  HCD11 67 TRP HCH21</td>
</tr>
<tr>
<td>present :</td>
<td>7.24</td>
<td>5.462E+00</td>
<td>12 LEU  HCD11 14 TRP HCZ31</td>
</tr>
<tr>
<td>swapped :</td>
<td>7.01</td>
<td>5.971E-01</td>
<td>12 LEU  HCD11 67 TRP HCZ31</td>
</tr>
<tr>
<td>present :</td>
<td>5.76</td>
<td>2.055E+01</td>
<td>12 LEU  HCD11 14 TRP HCH21</td>
</tr>
<tr>
<td>swapped :</td>
<td>6.43</td>
<td>2.041E+01</td>
<td>12 LEU  HCD11 67 TRP HCH21</td>
</tr>
</tbody>
</table>

Note: Chamb assumes that the alternate (intra/inter) assignments are sequential in the input file. The first one should always correspond to the initial (present) assignment.
6. Important files in DINOSAUR

In this section the important files in DINOSAUR are described. These are:

- 2D DINOSAUR input file for refinement with 2D NOE data
- 3D DINOSAUR input file for refinement with 3D NOE data
- 2D NOE file in BUILDMAN format
- 2D NOE file in free format
- 3D NOE file in REGINE format
- 3D NOE file in free format
- stereospecific assignments file
- chemical shift file
- order parameter matrix
- order parameter file
- proton coordinates file
- proton codes file for refinement with GROMOS

6.1 2D DINOSAUR input file

This file contains the experimental NOE intensities and all the necessary information for the calculations of the 2D NOE intensities and forces. It can be created or modified with the program dinoin. This input file is used by almost every program in the 2D version of DINOSAUR.

6.1.1 File description

General parameters:

- (a80) TITLE
  If the title begin with "DIMER" the molecule is assumed to be a symmetrical dimer and the NOE intensities will be symmetrized (i.e. NOE forces will be calculated from the averaged NOE intensities from the two monomers)
- (1[free]) NDSET: number of data sets.
  (Data obtained at various spectrometer frequencies and/or in different solvents are considered as various data sets.)
- (1[free]) SCALFAC: coordinates scaling factor.
  (The NOE calculation in DINOSAUR require distances in Angström.)
- (1[free]) SSHIFT: integer controlling the inclusion of chemical shifts in the calculations
  1: chemical shift file will be read
  0: no
- (1[free]) SPARA: integer controlling the inclusion of order parameters in the calculations
  1: generalised order parameters will be read
  0: no
- (1[free]) LS2:integer controlling the type of file containing the order parameters:
  0: matrix, binary
  1: matrix, formatted
  2: order parameter file
- (8[free]) NOTYP: method used for the calculation of the theoretical NOE intensities
see section 4.1 for the definitions)

RFACTYP: NOE restraining potential definition (see section 4.3 for the definitions)

NFLAT: parameter defining the form of the NOE potential:
0: harmonic
1: harmonic with flat bottom (NOE forces set to 0 if the difference between experimental and theoretical intensities is within the experimental error defined as the sum of an experimental noise level and a relative error on the experimental intensity (see section 3.1.5.1).

GRADTYP: integer defining the type of NOE gradient used for the calculation of the forces (see section 4.4 for the definitions)

DXYZMAX: proton displacement in Å upon which the NOE forces will be updated
This is used to speed-up the calculations: only if the maximal proton displacement becomes larger than the value defined by DXYZMAX will the NOE forces be recalculated. If DXYZMAX is set to 0.0 then the NOE forces will be calculated at every steps, otherwise small values of DXYZMAX are recommended in the order of 0.01 to 0.05 Å to avoid becoming trapped in a local minimum.

NSTEP: control parameter for the NOE calculation with spin diffusion correction (definitions 6, 7 and 8) or spherical cut-offs (definition 9). The spin diffusion correction or neighbour list will be updated every NSTEP refinement steps (see section 4.1)

DISCUT: distance cut-off in Å for NOE calculations with spherical cut-off (definition 9, see section 4.1)

DISMHH: minimal distance in Å between two protons. If some interproton distance during the refinement becomes very small, the NOE forces will blow up. DISMHH can be used to impose a lower limit on the distances but should not exceed the sum of the van der Waals radii of two protons.

- (2\ff) ISKEXP: parameter controlling which experimental data will be skipped
  -1: negative peaks skipped
  0: none
  1: positive peaks skipped

ISYEXP: parameter for symmetrisation of the experimental intensities
0: no symmetrisation
1: peaks on both sides of the diagonal will be symmetrized

- (1\ff) DEXPAV: exponential decay constant for time-averaged NOE refinement
The value of DEXPAV should correspond to \exp(-D/t) (see Eq. 33 section 3.1.5.3).
For example, for an exponential memory function with a time constant \[ \text{of 10 ps and a MD time step } \tau \text{ of 0.001 ps, DEXPAV takes a value of 0.9999.}

- (1\ff) AVEDIS: type of averaging used for ensemble- or time-averaged NOE refinement and also for the calculation of \( R \) factors from a proton trajectory (procedure rfacav section 5.3.1)
0: none
-1: NOE averaging
1: distance averaging as \(<r>\)
3: distance averaging as \(<r^-3>\)
6: distance averaging as \(<r^-6>\)
For time-averaged NOE refinement both DEXPAV and AVEDIS must be different from 0. For ensemble-averaged NOE refinement the parameter DEXPAV is not used and only distance averaging (options 1, 3, or 6) is presently implemented.

Parameters specific for each data set i (repeated NDSET times):

- (80) TITLE(i) characterising the data set
The first three characters are used to recognise a "H2O" or "D2O" data set, respectively. In the case of a D2O data set, the exchangeable protons (information stored in column 9 of ITYPAV (see below proton information) will be skipped for the NOE calculations.

- (1\ff) FREQ(i): spectrometer frequency in Hz
- (1\ff) TAUC(i): correlation time for overall tumbling of the molecule in seconds
- (1\ff) ROSTARO(i): additional leakage rate (R_{\text{leak}}) in s^{-1} accounting for external relaxation contributions. ROSTARO will be added to the diagonal elements of the relaxation matrix.
Typical value of $R_{\text{leak}}$ can be obtained from the inverse of the average proton $T_1$.

- (1 free) $T_{\text{METH}}(i)$: methyl group rotational correlation time in seconds.
  $T_{\text{METH}}$ is used to calculate the additional relaxation contribution on methyl protons
  ($R_{\text{methyl}}$) due to fast rotation (see Eq. 14 section 3.1.2). This contribution is added to the
diagonal elements of the relaxation matrix.

- (1 free) $K_{\text{NOE}}(i)$: NOE force constant
- (2 free) $C_{\text{UTMAX}}(i)$, $C_{\text{UTMIN}}(i)$: cutoffs for maximum and minimum NOE forces in Å$^{-1}$
- (1 free) $A_{\text{NOISE}}(i)$: experimental noise level
- (2 free) $A_{\text{EXPER}}(i)$, $A_{\text{THEER}}(i)$: experimental and theoretical relative errors on the NOE intensities.
  (ATHEER should be given as an absolute error on distances in Å if the NOE restraining
  potential definition 8 is used).
- (3 free) $L_{\text{SCAL}}(i)$: parameter defining the type of scaling of the theoretical NOE intensities. (for the
  various definitions see section 4.2)

$S_{\text{CALMIN}}(i)$, $S_{\text{CALMAX}}(i)$: minimum and maximum scaling factors.
  These two parameter can be used to limit the scaling factors between reasonable values.
  This should avoid too large fluctuations during the refinement. The minimum and
  maximum scaling factors can be chosen on the base of the scaling factors obtained from
  $R$ factor calculations (rfacav, rfacgr, rfacdg).

- (1 free) $N_{\text{REF}}(i)$: number of reference peaks

and for each reference peak $j$:

- (2 free) $N_{\text{REF}1}(j,i)$, $N_{\text{REF}2}(j,i)$: proton numbers defining a NOE used as a reference peak for scaling.
  The proton numbers must correspond to the DINOSAUR internal numbering (column 4 of
  ITYPAV, see below proton information)

Protons information:

- (2 free) $I_{\text{AVMETH}}, I_{\text{AVARO}}$: parameters defining the type of dynamical averaging for methyl groups
  and aromatic rings, respectively. These are only given as information and are only used
  when modifying the DINOSAUR input file with dinoin. The dynamical averaging
  information for the protons is stored in ITYPAV below. The various type of averaging are:
  0: no averaging
  1: $\langle r^{-6} \rangle$ averaging for motions slower than the overall tumbling of the molecule,
     e.g. aromatic ring flips
  3: $\langle r^{-3} \rangle$ averaging for motions faster than the overall tumbling of the molecule,
     e.g. methyl group rotation. After averaging, the averaging partners are described by
     only one entry in the relaxation matrix with intensity corrected by the number of
     partners ("matrix contraction").
  4: $\langle r^{-3} \rangle$ averaging as for option 3 but without matrix contraction

- (1 free) $N_{\text{HTOT}}$: total number of protons in the molecule

For each proton $k$:

- (9i5,4x,23a) ITYPAV(1...9,k) proton information about dynamical averaging
  Nine parameters are given for each proton. The first three indicates the type of averaging
  (motion) and the following five give the proton number (DINOSAUR internal numbering)
  and the numbers of the averaging partners if present. The last parameter is used in case of
  D2O data sets and indicate if the proton should be skipped (exchangeable or skipped
  because of matrix contraction).
  ITYPAV(1,k): type of averaging
  0: no averaging
  1: $\langle r^{-6} \rangle$ averaging (aromatic flip or slow motions)
  2: same as option 1, but for this proton an additional number will be given after the
     averaging partners indicating the number of the other proton also present on the same
     dynamical structure element. This is used for aromatic protons to define the other
     protons on the same ring that are not averaging partner but undergo the same
motions. Relaxation rates between these protons should not be averaged
3: $<r^3>$ averaging with matrix contraction (methyl rotation or fast motions)
4: $<r^3>$ averaging without matrix contraction (methyl rotation or fast motions)

ITYPAV(2,k): proton status
0: not yet encountered for averaging
1: already encountered, is part of an averaging process
2: same as 1, in addition will be skipped for matrix contraction

ITYPAV(3,k): number of averaging partners which will be read
ITYPAV(4,k): proton number (DINOSAUR internal numbering)
ITYPAV(5...8,k): averaging partners
ITYPAV(9,k): proton status used in case of D2O data sets. Contains the same information as ITYPAV(2,k) with in addition a code 2 for the exchangeable protons that will be skipped for the NOE calculations

PRTID(k): proton identification string.
This string should contain as first information on the residue number which is used to determine the intra and inter residue peaks. The residue number is read in free format from this string and is the only information used.

Experimental NOE data for each data set i:
- (1\[free\]) NTM(i): number of mixing times
- (NTM\[free\]) TM(1...NTM): mixing times in seconds
- (1\[free\]) NEXP(i): number of experimental data (per mixing time)

For each experimental peak j:
- ((2+2\[NTM+1]\[free\]) NOE1(j), NOE2(j): proton numbers defining the NOE peak (DINOSAUR numbering)
AEXP(1,j),W(1,j): peak intensity and weighting factor for the first mixing time
......
AEXP(NTM,j),W(NTM,j): peak intensity and weighting factor for the last mixing time
PNUM(j): peak identification number (sequential numbering on all data sets)

Information for dynamic assignment procedure:
- (NDSET\[free\]) NSTE(1...NDSET) number of peaks for dynamic stereospecific assignments for the various data sets. Note that for NDSET > 1, NSTE(k) gives the sum of all dynamic assignment peaks for the data sets 1 to k. For example, if for the first data set there are 100 peaks with alternate assignment and 200 for the second data set, NSTE(1) is 100 and NSTE(2) will be 300 (100+200).
- (1\[free\]) DSAUP: maximum probability value above which the assignments will be frozen

For each alternate assignment j:
- (5\[free\]) NSFLAG(j): assignment status:
  0: not assigned, dynamic assignment will be performed
  1: assigned: no swap (i.e. the initial assignment was correct), no more dynamic assignment (assignment is frozen)
  2: assigned: swap (i.e. the initial assignment was not correct), no more dynamic assignment (frozen)
PSWAP(j): assignment probability: swap: > 0, no swap < 0
Every time an assignment if performed a number comprised between 0.5 (swap) and -0.5 (no swap) is added to the assignment probability. If all mixing times give the same assignment, the maximal value of ±0.5 is obtained. If there is discrepancy between the assignments obtained at various mixing times, the value decreases and become 0 if no decision can be made. If the absolute assignment probability reaches the
6.1.2 Example of a 2D DINOSAUR input file with one data set

```
Crambin DINOSAUR annealing
1 number of data sets
10.0000 coordinates scaling factor
1 l:chemical shift file will be read, 0:no
1 l:order parameters S2 will be read, 0:no
9 6 1 0.01000 1.450 1.650 NOTYP FACTYP NFLAT GRADTYP DXZMAX NSTEP DISCU T DISMH H
-1 1 skip exp NOEs (-1,0,+1) and symmetrise (0:n,1:y)
0.00000 Exponential decay constant for time aver.
6 time averaging -1:NOEs 0:none 1:r 3:1/r**3 6:1/r**6
H2O NOEs from rescram7di.dat, stereo from cram.stereo
5.0000E+08 Spectrometer frequency (Hz)
2.0000E-09 overall correlation time (s)
0.80000 Additional leakage rate (1/s)
1.0000E-10 methyl group correlation time (s)
4.0000E+01 NOE force constant
1.0000E+01 -1.0000E+01 max and min cutoffs for NOE forces
1.0000E+04 experimental noise level
0.05000 0.10000 experimental and theoretical errors
0 3.1000E+08 3.6000E+08 type of scaling, scalmin and scalmax
6 number of reference peaks
58 59 9 ALA H CA1 9 ALA H CB1
59 58 9 ALA H CB1 9 ALA H CA1
169 170 24 ALA H CA1 24 ALA H CB1
170 169 24 ALA H CB1 24 ALA H CA1
189 190 27 ALA H CA1 27 ALA H CB1
190 189 27 ALA H CB1 27 ALA H CA1
3 1 averaging types for methyl and aromatic groups
315 total number of protons
3 0 3 1 2 3 0 0 0 2 1 THR H1 1
3 2 2 2 3 0 0 0 0 2 1 THR H2 2
3 2 1 3 0 0 0 0 0 2 1 THR H3 4
0 0 1 4 0 0 0 0 0 0 1 THR H CA1 6
0 0 1 5 0 0 0 0 0 0 1 THR H CB1 8
0 0 1 6 0 0 0 0 0 0 2 1 THR HG1 10
3 0 3 7 8 9 0 0 0 0 1 THR HCG21 12
3 2 2 8 9 0 0 0 0 2 1 THR HCG22 13
3 2 1 9 0 0 0 0 0 2 1 THR HCG23 14
0 0 1 10 0 0 0 0 0 2 2 THR H CA1 20
0 0 1 12 0 0 0 0 0 0 2 2 THR H CB1 22
0 0 1 13 0 0 0 0 0 0 2 2 THR HG1 24
3 0 3 14 15 16 0 0 0 0 2 2 THR HCG21 26
3 2 2 15 16 0 0 0 0 2 2 THR HCG22 27
3 2 1 16 0 0 0 0 0 0 2 2 THR HCG23 28
... 0 0 1 87 0 0 0 0 0 0 13 PHE H CA1 173
0 0 1 88 0 0 0 0 0 0 13 PHE H CB1 175
0 0 1 89 0 0 0 0 0 0 13 PHE H CB2 176
2 0 2 90 91 92 0 0 0 0 13 PHE HCD11 179
1 1 1 91 0 0 0 0 0 0 13 PHE HCD21 181
2 0 2 92 93 90 0 0 0 0 13 PHE HCE11 183
1 1 1 93 0 0 0 0 0 0 13 PHE HCE21 185
0 0 1 94 0 0 0 0 0 0 13 PHE H CZ1 187
... 3 number of mixing times for data set # 1
0.08000 0.16000 0.25000
369 number of exp. NOEs for data set # 1
4 7 3.7900E+06 1.000 5.2600E+06 1.000 4.5800E+06 1.000 1
```
6.1.3 Example of a 2D DINOSAUR input file with two data sets

DIMER Arc DINOSAUR H2O and D2O data sets

2 number of data sets

10.00000 coordinates scaling factor

0 1:chemical shift file will be read, 0:nono

1 1:order parameters S2 will be read, 0:nono

0 S2 read from 0:matrix(bin) 1:maxtrix(form) 2:file

9 6 1 1 0.01000 1 4.25 1.65 NOTYP RFACTYP NPLAT GRAD Typ DX2MAX NSTEP DISCUT DISMHH
-1 1 skip exp NOEs (-1,0,+1) and symmetrise (0:n,1:y)

0.00000 Exponential decay constant for time.aver.

6 time averaging -1:NOEs 0:none 1:r 3:1/r**3 6:1/r**6

H2O NOEs from arch2o_di.noe

6.000E+08 Spectrometer frequency (Hz)

4.250E-09 overall correlation time (s)

2.00000 Additional leakage rate (1/s)

1.000E-10 methyl group correlation time (s)

4.000E+01 NOE force constant

1.000E+01 -1.000E+01 max and min cutoffs for NOE forces

5.000E+04 experimental noise level

0.05000 0.05000 experimental and theoretical errors

1 5.000E+06 5.000E+08 type of scaling, scalmin and scalmax

0 number of reference peaks

D2O NOEs from arcd2o_di.noe

6.000E+08 Spectrometer frequency (Hz)

4.250E-09 overall correlation time (s)
Additional leakage rate (1/s)  
1.000E-10  methyl group correlation time (s)  
4.000E+01  NOE force constant  
1.000E+01 -1.000E+01  max and min cutoffs for NOE forces  
5.000E+03  experimental noise level  
0.05000  0.05000  experimental and theoretical errors  
1 5.000E+05  5.000E+07  type of scaling, scalmin and scalmax  
0 number of reference peaks  
3 0  averaging types for methyl and aromatic groups  
896 total number of protons  
3 0 3 1 2 3 0 0 2 1 MET  H1 1  
3 2 2 2 3 0 0 0 2 1 MET  H2 2  
3 2 1 3 0 0 0 0 2 1 MET  H3 4  
0 0 1 4 0 0 0 0 0 1 MET  H CA1 6  
0 0 1 5 0 0 0 0 0 1 MET  H CB1 8  
0 0 1 6 0 0 0 0 0 1 MET  H CB2 9  
0 0 1 7 0 0 0 0 0 1 MET  H CG1 11  
0 0 1 8 0 0 0 0 0 1 MET  H CG2 12  
3 0 3 9 10 11 0 0 0 1 MET  H CE1 15  
3 2 2 10 11 0 0 0 2 1 MET  H CE2 16  
3 2 1 11 0 0 0 0 2 1 MET  H CE3 17  
... 3 number of mixing times for data set # 1  
0.10000  0.15000  0.20000  
1408 number of exp. NOEs for data set # 1  
123 766 1.033E+06  1.000  1.310E+06  1.000  1  
314 318 2.156E+06  1.000  3.051E+06  1.000  2  
762 766 2.156E+06  1.000  3.051E+06  1.000  3  
122 766 9.253E+05  1.000  1.205E+06  1.000  4  
125 766 1.185E+06  1.000  1.762E+06  1.000  5  
124 766 8.298E+05  1.000  1.324E+06  1.000  6  
318 572 8.298E+05  1.000  1.324E+06  1.000  7  
318 762 9.253E+05  1.000  1.762E+06  1.000  8  
318 572 9.253E+05  1.000  1.762E+06  1.000  9  
318 331 1.213E+06  1.000  1.691E+06  1.000  10  
... 3 number of mixing times for data set # 2  
0.12000  0.16000  0.20000  
1140 number of exp. NOEs for data set # 2  
119 122 9.855E+05  1.000  9.656E+05  1.000  1.000  1409  
567 570 9.855E+05  1.000  9.656E+05  1.000  1.000  1410  
318 570 8.854E+05  1.000  9.094E+04  1.000  1.091E+04  1.000  1411  
320 570 3.274E+04  1.000  3.660E+04  1.000  6.352E+04  1.000  1412  
318 573 1.204E+05  1.000  1.578E+05  1.000  1.356E+05  1.000  1413  
318 572 1.280E+05  1.000  1.374E+05  1.000  1.751E+05  1.000  1414  
320 572 7.017E+04  1.000  8.616E+04  1.000  1.004E+05  1.000  1415  
85 572 8.867E+04  1.000  8.383E+04  1.000  6.795E+04  1.000  1416  
374 377 8.872E+05  1.000  6.582E+05  1.000  1.110E+06  1.000  1417  
822 825 8.872E+05  1.000  6.582E+05  1.000  1.110E+06  1.000  1418  
... 629 1377 number of stereo peaks  
1.000E+05  cutoff for freezing dynamic assignment  
0 -2.835E+02  123 767  1  
0 -2.735E+02  314 319  2  
0 -2.789E+02  762 767  3  
0 -2.835E+02  122 767  4  
0 -2.835E+02  125 767  5  
0 -2.835E+02  124 767  6  
0 -2.835E+02  319 572  7  
0 -2.835E+02  319 573  8  
0 -2.835E+02  319 570  9  
0 2.835E+02  319 331 10  
...
6.2 3D DINOSAUR input file

This file contains the experimental NOE intensities and all the necessary information for the calculations of the 3D NOE-NOE intensities and forces. It can be created or modified with the program dinoin. This input file is used by almost every program in the 3D version of DINOSAUR.

6.2.1 File description

General parameters:

- (a80) TITLE
- (1 free) SCALFAC: coordinates scaling factor.
  (The NOE calculation in DINOSAUR require distances in Angström.)
- (1 free) SSHIFT: integer controlling the inclusion of chemical shifts in the calculations
  1: chemical shift file will be read
  0: no
- (1 free) SPARA: integer controlling the inclusion of order parameters in the calculations
  1: generalised order parameters will be read
  0: no
- (1 free) LS2: integer controlling the type of file containing the order parameters:
  0: matrix, binary
  1: matrix, formatted
  2: order parameter file
- (8 free) NOTYP: method used for the calculation of the theoretical NOE intensities
  see section 4.1 for the definitions)
RFACTYP: NOE restraining potential definition (see section 4.3 for the definitions)
NFLAT: parameter defining the form of the NOE potential:
  0: harmonic
  1: harmonic with flat bottom (NOE forces set to 0 if the difference between experimental and theoretical intensities is within the experimental error defined as the sum of an experimental noise level and a relative error on the experimental intensity
  (see section 3.1.5.1).
GRADTYP: integer defining the type of NOE gradient used for the calculation of the forces
  (see section 4.4 for the definitions)
DXYZMAX: proton displacement in Å upon which the NOE forces will be updated
  This is used to speed-up the calculations: only if the maximal proton displacement becomes larger than the value defined by DXYZMAX will the NOE forces be recalculated. If DXYZMAX is set to 0.0 then the NOE forces will be calculated at every steps, otherwise small values of DXYZMAX are recommended in the order of 0.01 to 0.05 Å to avoid becoming trapped in a local minimum.
NSTEP: control parameter for the NOE calculation with spin diffusion correction (definitions 6, 7 and 8) or spherical cut-offs (definition 9). The spin diffusion correction or neighbour list will be updated every NSTEP refinement steps (see section 4.1)
DISCUT: distance cut-off in Å for NOE calculations with spherical cut-off (definition 9, see section 4.1)
DISMHH: minimal distance in Å between two protons. If some interproton distance during the refinement becomes very small, the NOE forces will blow up. DISMHH can be used to impose a lower limit on the distances but should not exceed the sum of the van der Waals radii of two protons.
- (1 free) ISKEXP: parameter controlling which experimental data will be skipped
  -1: negative peaks skipped
  0: none
  1: positive peaks skipped
- (1 free) DEXPAV: exponential decay constant for time-averaged NOE refinement

  The value of DEXPAV should correspond to \( \exp(-\frac{t}{\tau}) \) (see Eq. 3.3 section 3.1.5.3).

  For example, for an exponential memory function with a time constant \( \tau \) of 10 ps and a
  MD time step \( \Delta t \) of 0.001 ps, DEXPAV takes a value of 0.9999.

- (1 free) AVEDIS: type of averaging used for ensemble- or time-averaged NOE refinement and also

  for the calculation of \( R \) factors from a proton trajectory (procedure rfacav section 5.3.1)

  0: none

  -1: NOE averaging

  1: distance averaging as \( \langle r \rangle \)

  3: distance averaging as \( \langle r^{-3} \rangle \)

  6: distance averaging as \( \langle r^{-6} \rangle \)

  For time-averaged NOE refinement both DEXPAV and AVEDIS must be different from 0.

  For ensemble-averaged NOE refinement the parameter DEXPAV is not used and only

  distance averaging (options 1, 3, or 6) is presently implemented.

- (a80) SUBTITLE characterising the data set

  The first three characters are used to recognise a "H2O" or "D2O" data set, respectively. In

  the case of a D2O data set, the exchangeable protons (information stored in column 9 of

  ITYPAV (see below proton information) will be skipped for the NOE calculations.

- (1 free) FREQ: spectrometer frequency in Hz

- (1 free) TAUC: correlation time for overall tumbling of the molecule in seconds

- (1 free) ROSTARO: additional leakage rate \( (R_{\text{leak}}) \) in s\(^{-1}\) accounting for external relaxation

  contributions. ROSTARO will be added to the diagonal elements of the relaxation matrix.

  Typical value of \( R_{\text{leak}} \) can be obtained form the inverse of the average proton \( T_1 \).

- (1 free) TMETH: methyl group rotational correlation time in seconds.

  TMETH is used to calculate the additional relaxation contribution on methyl protons

  \( (R_{\text{methyl}}) \) due to fast rotation (see Eq. 14 section 3.1.2). This contribution is added to the

  diagonal elements of the relaxation matrix.

- (1 free) KNOE: NOE force constant

- (2 free) CUTMAX, CUTMIN: cutoffs for maximum and minimum NOE forces in \( \text{Å}^{-1} \)

- (1 free) ANOISE: experimental noise level

- (2 free) AEXPER, ATHEER: experimental and theoretical relative errors on the NOE intensities.

  (ATHEER should be given as an absolute error on distances in \( \text{Å} \) if the NOE restraining

  potential definition 8 is used).

- (3 free) LCSAL: parameter defining the type of scaling of the theoretical NOE intensities. Four types

  of scaling are actually implemented:

  0 : with reference peaks

  1 : with all peaks

  2 : with intraresidue back transfer peaks

  3 : with intraresidue peaks

  These two parameter can be used to limit the scaling factors between reasonable values.

  This should avoid too large fluctuations during the refinement. The minimum and

  maximum scaling factors can be chosen on the base of the scaling factors obtained from

  \( R \) factor calculations (rfacav, rfacgr, rfacdg).

- (1 free) NREF: number of reference peaks

  and for each reference peak j:

- (3 free) NREF1(j), NREF2(j), NREF3(j): proton numbers defining a 3D NOE used as a reference peak

  for scaling. The proton numbers must correspond to the DINOSAUR internal numbering

  (column 3 of ITYPAV; see below proton information)

Protons information:

- (2 free) IAVMETH, IAVARO: parameters defining the type of dynamical averaging for methyl groups

  and aromatic rings, respectively. These are only given as information and are only used

  when modifying the DINOSAUR input file with dinoin. The dynamical averaging
information for the protons is stored in ITYPAV below. The various type of averaging are:

0: no averaging
1: $<r^6>$ averaging for motions slower than the overall tumbling of the molecule, e.g. aromatic ring flips
3: $<r^3>$ averaging for motions faster than the overall tumbling of the molecule, e.g. methyl group rotation. After averaging, the averaging partners are described by only one entry in the relaxation matrix with intensity corrected by the number of partners (“matrix contraction”).
4: $<r^3>$ averaging as for option 3 but without matrix contraction

- (1[free]) NHTOT: total number of protons in the molecule

For each proton $k$:

- (9i,4x,23a) ITYPAV(1...9,k) proton information about dynamical averaging

Nine parameters are given for each proton. The first three indicates the type of averaging (motion) and the following five give the proton number (DINOSAUR internal numbering) and the numbers of the averaging partners if present. The last parameter is used in case of D2O data sets and indicate if the proton should be skipped (exchangeable or skipped because of matrix contraction).

ITYPAV(1,k): type of averaging
0: no averaging
1: $<r^6>$ averaging (aromatic flip or slow motions)
2: same as option 1, but for this proton an additional number will be given after the averaging partners indicating the number of the other proton also present on the same dynamical structure element. This is used for aromatic protons to define the other protons on the same ring that are not averaging partner but undergo the same motions. Relaxation rates between these protons should not be averaged
3: $<r^3>$ averaging with matrix contraction (methyl rotation or fast motions)
4: $<r^3>$ averaging without matrix contraction (methyl rotation or fast motions)

ITYPAV(2,k): proton status
0: not yet encountered for averaging
1: already encountered, is part of an averaging process
2: same as 1, in addition will be skipped for matrix contraction

ITYPAV(3,k): number of averaging partners which will be read
ITYPAV(4,k): proton number (DINOSAUR internal numbering)
ITYPAV(5...8,k): averaging partners
ITYPAV(9,k): proton status used in case of D2O data sets. Contains the same information as ITYPAV(2,k) with in addition a code 2 for the exchangeable protons that will be skipped for the NOE calculations

PRTID(k): proton identification string.
This string should contain as first information on the residue number which is used to determine the intra and inter residue peaks. The residue number is read in free format from this string and is the only information used.

Experimental 3D NOE data:

- (1[free]) NTM: number of mixing times (thus 2!)
- (NTM[free]) TM(1...NTM): mixing times in seconds
- (1[free]) NEXP: number of experimental data

For each experimental peak $j$:

- (6[free]) NOE1(j), NOE2(j), NOE3(j): proton numbers defining the NOE peak (DINOSAUR numbering)
        AEXP(j): 3D peak intensity
        W(j): weighting factor
Information for dynamic assignment procedure:

- (1\text{free})\quad \text{NSTE: number of peaks for dynamic stereospecific assignments}
- (1\text{free})\quad \text{DSAUP: maximum probability value above which the assignments will be frozen}

For each alternate assignment $j$:

- (5\text{free})\quad \text{NSFLAG}(j): assignment status:
  0: not assigned, dynamic assignment will be performed
  1: assigned: no swap (i.e. the initial assignment was correct), no more dynamic assignment (assignment is frozen)
  2: assigned: swap (i.e. the initial assignment was not correct), no more dynamic assignment (frozen)

\text{PSWAP}(j): assignment probability: swap: > 0, no swap < 0

Every time an assignment is performed a number comprised between 0.5 (swap) and -0.5 (no swap) is added to the assignment probability. If all mixing times give the same assignment, the maximal value of ±0.5 is obtained. If there is discrepancy between the assignments obtained at various mixing times, the value decreases and become 0 if no decision can be made. If the absolute assignment probability reaches the maximum value defined by DSAUP, the assignment is frozen.

\text{NSTE1}(j), \text{NSTE2}(j), \text{NSTE23}(j): proton numbers defining the alternate assignment (DINOSAUR numbering)

\text{NSTP}(j): peak number to which the alternate assignment corresponds

5.2.2 Example of 3D DINOSAUR input file

Parvalbumine 3D DINOSAUR with 3D NOE-NOE

10.00000 \quad \text{coordinates scaling factor}

0 \quad 1:\text{chemical shift file will be read, 0:no}
0 \quad \text{order parameters S2 will be read, 0:no}
2 \quad \text{S2 read from 0:matrix(bin) 1:maxtrix(form) 2:file}
9 6 1 1 0.0250 1 4.00 1.50 \text{NOTYP RFACTYP NFLAT GRADTYP DXYZMAX NSTEP DISCUT DISMHH}
-1 \quad \text{skip exp NOEs (-1:neg,0:none,+1:pos)}
0.00000 \quad \text{Exponential decay constant for time aver.}
0 \quad \text{time averaging -1:NOEs 0:none 1:r 3:1/r**3 6:1/r**6}

H2O3D NOE-NOE data (volumes)
5.001E+10 \quad \text{Spectrometer frequency (Hz)}
1.200E-08 \quad \text{overall correlation time (s)}
2.00000 \quad \text{Additional leakage rate (1/s)}
1.000E-10 \quad \text{methyl group correlation time (s)}
1.000E+03 \quad \text{NOE force constant}
1.000E+00 -1.000E+00 \quad \text{max and min cutoffs for NOE forces}
1.000E+04 \quad \text{experimental noise level}
0.50000 0.05000 \text{ experimental and theoretical errors}
2 \quad 1.000E+09 1.000E+10 \quad \text{type of scaling, scalmin and scalmax}
5 \quad \text{number of reference peaks}
67 68 67 67 8 \text{ALA H CA1 8 ALA H CB1 8 ALA H CA1}
118 117 118 14 \text{ALA H CB1 14 ALA H CA1 14 ALA H CB1}
654 655 654 84 \text{ALA H CA1 84 ALA H CB1 84 ALA H CA1}
693 692 693 88 \text{ALA H CB1 88 ALA H CA1 88 ALA H CB1}
697 698 697 89 \text{ALA H CA1 89 ALA H CB1 89 ALA H CA1}
3 0 \quad \text{averaging types for methyl and aromatic groups}
839 \quad \text{total number of protons}

...
... 2 number of mixing times for 3D data set
0.20000 0.20000
7461 number of exp. NOEs for 3D data set
439 434 439 7.996E+06 1.00 1
729 725 729 1.045E+07 1.00 2
439 442 439 1.210E+07 1.00 3
729 732 729 1.296E+07 1.00 4
439 440 439 9.883E+06 1.00 5
729 730 729 1.045E+07 1.00 6
729 731 729 1.678E+07 1.00 7
439 441 439 1.322E+07 1.00 8
429 434 439 4.814E+05 1.00 9
429 430 439 5.715E+05 1.00 10
732 731 729 8.538E+05 1.00 11
722 725 729 6.492E+05 1.00 12
....
2724 number of dynamic assignment peaks
1.000E+06 cutoff for freezing dynamic assignment
0 0.000E+00 756 777 756 29
0 0.000E+00 759 757 451 32
0 0.000E+00 774 777 756 35
0 0.000E+00 385 386 451 38
0 0.000E+00 758 746 451 45
0 0.000E+00 758 757 756 46
0 0.000E+00 775 777 756 47
0 0.000E+00 777 756 451 51
0 0.000E+00 776 777 756 55
0 0.000E+00 451 747 753 7459
....

6.3 The BUILDUP result file

The result file contains the information on experimental 2D NOE intensities in form of a 2D buildup series. It contains a header section, followed by peak information. For each peak two lines (i.e. two FORTRAN read/write statements) are used. Note that the length of the first of these two lines depends on the number of mixing times.

6.3.1 File description

- (a80) title
- (i5) file identification number (not used in DINOSAUR)
- (i5) NTAPE = number of mixing times = number of NMR files
- (16x1,1pe10.3) mixing times
- (2x1,1pe10.3) reference rate, reference distance (not used in DINOSAUR)

Fore each mixing time:

- (a80) name of first NMR file (not used in DINOSAUR)

For each experimental peak I:
- (5(1x),17(1x,1pe10.3))

I: peak number (not checked on reading)
IARRAY(I), JARRAY(I), IARRY2(I), JARRY2(I): channel numbers defining rectangle around peak (not used in DINOSAUR)
RATE(I): rate defined by fitted buildup curve (not used in DINOSAUR)
YV(I...NTAPE,I): array of peak volumes (function of mixing time)
SUMMY(I), DELLY(I), SUMMX(I), YFIRR(I): data defining fitted buildup curve (not used in DINOSAUR)
- (1x,11,a1,2(1x,4),1x,2(5,2a5,i5))

HANDLD(I): logical indicating whether the buildup curve has been fitted or not (not used in DINOSAUR)
PKFLAG(I): character indicating quality of buildup fit; set by user. (Checked by DINOSAUR, if bad ('b' or 'B') these buildup data will be skipped)
IPEAK(I), IPEAK2(I): two identification numbers (not used in DINOSAUR)
RESNUM1(I), RESNAM1(I), ATNAM1(I), ATNUM1(I)
RESNUM2(I), RESNAM2(I), ATNAM2(I), ATNUM2(I): peak identification containing the residue numbers and names and the proton names and numbers for both proton defining a peak. Only proton numbers and names are used in DINOSAUR. In the name, the last number is important.
If a name ends with a 0 the corresponding peak is assumed to consist of two overlapping peaks and the intensity of these peaks will be divided by two when entering DINOSAUR.
If the proton number is 0, the peak is considered as unassigned and will be skipped when entering DINOSAUR.

Note: The proton numbers in this file must correspond with the numbers in the proton only file (.mdh)

6.3.2 Example of a 2D NOE BUILDUP file

crambin BUILDUP result file
1 (Identification number of result file)
6 (Number of mixing times)
2.000E-02 4.000E-02 8.000E-02 1.200E-01 1.600E-01 2.500E-01
2.100E+07 3.500E+00
$disk3: [guest2.2dnmr.data]rlcrd20c2.dat
$disk3: [guest2.2dnmr.data]rlcrd40c2.dat
$disk3: [guest2.2dnmr.data]rlcrd80c2.dat
$disk3: [guest2.2dnmr.data]rlcrd120c2.dat
$disk3: [guest2.2dnmr.data]rlcrd160c2.dat
$disk3: [guest2.2dnmr.data]rlcrd250c2.dat
1 799 512 804 516 5.545E+00 9.310E+05 1.470E+06 2.430E+06 2.030E+06
1.140E+06 1.370E+06 -5.554E+00 2.596E+06 2.600E-01 0.000E+00
TB 1 23 1THR H CB1 8 1THR HCG21 12
2 798 517 803 524 5.200E+07 1.490E+06 1.560E+06 3.790E+06 5.980E+06
5.260E+06 4.580E+06 1.500E+00 1.456E+07 4.200E-01 0.000E+00
T 1 24 1THR H CA1 6 1THR HCG21 12
3 481 799 488 806 9.053E+07 9.763E+05 7.794E+05 6.769E+05 2.317E+06
5.092E+06 4.580E+06 -1.500E+00 1.456E+07 4.200E-01 0.000E+00
T 10 18 1THR HCG21 12 39THR H CA1 543
4 554 634 563 645 2.832E+08 5.580E+06 9.920E+06 1.374E+07 2.040E+07
2.360E+07 2.680E+07 0.000E+00 0.000E+00 0.000E+00 0.000E+00
T 8 10 4CYS1 H CB0 46 5PRO H CDO 61
5 293 680 298 688 3.484E+07 9.763E+05 7.794E+05 6.769E+05 2.317E+06
2.092E+06 2.154E+06 3.554E+00 2.549E+06 2.600E-01 0.000E+00
T 16 21 5PRO H CB2 56 44TYR HCE11 610
6 502 457 511 471 2.162E+08 1.880E+06 7.320E+06 1.280E+07 1.600E+07
2.040E+07 2.420E+07 -5.204E+00 2.783E+07 6.700E-01 0.000E+00
T 4 5 NOASSNOASS 0 NOASSNOASS 0...
6.4 The 2D NOE file (free)

This file contains the experimental 2D NOE intensities in a free format.

6.4.1 File description

- (a80) TITLE
- (1¶free) NTM: number of mixing times
- (NTM¶free) TM(1...NTM): mixing times in seconds
- (1¶free) NEXP: number of peaks

For each peak i:

- ((2+NTM¶free) NOE1(i), NOE2(i): proton numbers identifying the peak (DINOSAUR internal numbering!)
  AEXP(1...NTM,i): NOE intensities for the various mixing times

6.4.2 Example of 2D NOE file

Test Helix 2D NOE file
6
2.0000000E-02 4.0000000E-02 8.0000000E-02 0.1200000 0.1600000
0.2500000
260
4       5 1.2038938717272649E-02 2.2457957830652079E-02
3.8857825760890261E-02 5.0108252127324801E-02 5.7225506354384801E-02
6.264922671912127E-02
4       6 1.5955495203074918E-02 2.8671011925645743E-02
5.7236338243967326E-02 6.2859198712322177E-02
4       7 7.5357439291456440E-02 1.8972523208111137E-02
4.6616942556614002E-02 5.7236338243967326E-02 6.2859198712322177E-02
4       8 1.8758956447183535E-02
4       9 1.4324948616282555E-03 3.3091440329761616E-03
7.6022275535372427E-03 1.1690122543746284E-02 1.5064230896918262E-02
1.9742085902122095E-02
4       9 5.0907531082544563E-03 9.7160336386135686E-03
1.7284579792887482E-02 2.2388175412513455E-02 2.5243264457905789E-02
2.5901921706183449E-02
4       10 3.638307233494379E-04 7.2759045552081826E-04
1.8009680872243796E-03 3.007497279303161E-03 4.182933107325244E-03
6.2340816228749105E-03 ...

...
6.5 The 3D NOE-NOE file (REGINE)

This file contains the experimental 3D NOE-NOE intensities in a REGINE format. The header of the file contains a format description. For more information refer to the REGINE manual.

```
1 239943 3 99 1 1 LMAX3D former_Alison_peak
6.3310E+07 1.3680E+07

2 239945 3 99 1 1 LMAX3D former_Alison_peak
7.7760E+07 2.2730E+07
1:GLY_95:H                 1:GLY_95:H                 1:GLY_95:H

3 239946 3 99 1 1 LMAX3D former_Alison_peak
7.9960E+06 1.8810E+06

4 239947 3 99 1 1 LMAX3D former_Alison_peak
1.0450E+07 2.9730E+06
1:GLY_95:H                 1:ASP_94:H                 1:GLY_95:H

Examples of peaks with multiple assignments:

97 240044 3 -2 1 2 LMAX3D former_Alison_peak
4.5220E+06 6.6860E+05
1:GLY_98:HCA1              1:GLY_98:HCA1              1:GLY_98:HCA1
1:GLY_98:HCA1              1:GLY_98:HCA1              1:GLY_98:HCA1

109 240058 3 -4 1 2 LMAX3D former_Alison_peak
7.9890E+05 1.3180E+05
1:GLU_101:HCB1             1:GLY_98:H                 1:ILE_58:H
1:GLU_101:HCB1             1:GLY_98:H                 1:ILE_58:H
```

6.6 The 3D NOE-NOE file (free)

This file contains the experimental 3D NOE intensities in a free format.

6.6.1 File description

- (1 free) NTM: number of mixing times
- (NTM free) TM(1...NTM): mixing times in seconds
- (1 free) NEXP: number of peaks

For each peak i:

- (4 free) NOE1(i), NOE2(i), NOE3(i): proton numbers identifying the 3D peak (DINOSAUR internal numbering!)
  AEXP(i): 3D NOE-NOE intensities

6.6.2 Example of 3D NOE-NOE file

```
2
0.1200000  0.1600000
3932
4  5  4  2.877E-03
4  5  6  1.123E-02
4  5  7  3.121E-03
4  5  8  2.032E-03
4  5  9  2.266E-04
.....
```

6.7 Stereospecific assignments file (.stereo)

This file contains stereospecific assignments and can be used in the procedure dinoin to determine the stereospecifically unassigned protons that will be submitted to dynamic assignment.

6.7.1 File description

- (a80) TITLE
- (a1,i5,a5,i5,NST(1x,a5,i2))
  STATUS: assignment status
  ?: not assigned
  S: assignment should be swapped
  N: no swap
  RESNUM, RESNAM: residue number and residue name
  NST: number of stereospecific protons
  (HNAME(k),HNUM(k),k=1,NST): protons names and new numbering of the corresponding stereospecific protons. The new numbering will determine the proton names after applying the assignments to the experimental data and is not used in DINOSAUR.
6.7.2 Example of a stereospecific assignment file

Stereo assignments crambin
S 3CYS1 2 H CB1 2 H CB2 1
S 5PRO 2 H CB1 2 H CB2 1
? 5PRO 2 H CD1 0 H CD2 0
? 6SER 2 H CB1 0 H CB2 0
? 7ILE 2 HCG1 0 HCG1 2
S 8VAL 2 HCG1 1 HCG2 2
? 10ARG 2 H CG1 0 H CG2 0
? 10ARG 2 H CD1 0 H CD2 0
? 10ARG 2 HH1 0 HH2 0
N 12ASN 2 H CB1 1 H CB2 2
? 12ASN 2 HD2 0 HD2 2
...

6.8 Chemical shift file (.shift)

This file contains information about the chemical shifts and the additional leakage rates added on the diagonal elements of the relaxation matrix to account for external relaxation contributions (Rleak, see Eq. 2 section 3.1.1). If the latter are equal to 0, the default value given in the input file by ROSTARO will be used. Only protons for which information is available are found in this file.

6.8.1 File description

- (a80) TITLE
- (1[free]) NPROT: number of protons for which information will be read from this file

For each proton i:

- (3[free]) HNUM(i): proton number (DINOSAUR internal numbering)
SHIFT(i): chemical shift
ROSTAR(i): additional diagonal relaxation rate

6.8.2 Example of a chemical shift file

Crambin Pro-Leu chemical shifts
288
4 4.190000 0.0000000E+00 0.0000000E+00
5 4.270000 0.0000000E+00 0.0000000E+00
7 1.190000 0.0000000E+00 0.0000000E+00
8 1.190000 0.0000000E+00 0.0000000E+00
9 1.190000 0.0000000E+00 0.0000000E+00
10 8.620000 0.0000000E+00 0.0000000E+00
...
6.9 Generalised order parameter matrix

This matrix has the size of the relaxation matrix after contraction, *i.e.* each methyl group is treated as one entry, and contains generalised order parameters $S^2$ values for each proton-proton vector. These values vary between 1 and 0 and can be obtained from a free molecular dynamics simulation in water (see section 3.1.1). The matrix can either be formatted (8(1pe10.3)) or unformatted.

6.10 Generalised order parameter file (.s2)

This file contains only generalised order parameters $S^2$ values that differ from 1. All other values not found in this file are assumed to be equal to 1.

6.10.1 File description

- (a80) TITIE
- (1[]free) NS2: number of order parameter values to be read

For each order parameter:

- (3[]free) NH1,NH2: corresponding proton numbers
S2: order parameter value

6.10.2 Example of an order parameter file

```
S2 order parameters Crambin from free MD with Xray structure
3016
  1  2  0.4670000
  1  3  0.4160000
  1  4  0.6990000
  1  5  0.7420000
  1  6  0.7750000
  1  7  0.6150000
  1  8  0.6150000
  1  9  0.6150000
  1 10  0.8840000
  1 11  0.9370000
  1 14  0.9090000
  1 15  0.9090000
  1 16  0.9090000
  1 235 0.8180000
  1 245 0.9050000
  1 246 0.8960000
  1 249 0.8840000
  1 253 0.8360000
  1 254 0.8360000
  1 255 0.8360000
```

....
6.11 The proton coordinates file (.mdh)

This file contains the proton identifications. The format of this file is strict. Names are only used in the procedure dinoin. Methyl and aromatic groups will only be recognised in dinoin if the corresponding names are defined in the “metharo.inc” include file. Protons coordinates are not used. The three protons of the methyl groups and the two equivalent protons of the aromatic groups are assumed to be in following order in this file. If methyl rotation and aromatic ring flip are not included in the calculations the order of these protons is no longer important. This file can be created from a GROMOS structure with the procedure mkhco (see section 5.1.4).

6.11.1 File description

- (a80) TITLE
- (i5) NHTOT: number of protons in the file

For each proton i:

- (i5,2a5,i5,3f8.3) RESNUM(i): residue number
  RESNAM(i):residue name
  HNAME(i): proton name
  HNUM(i): proton number
  X,Y,Z: proton coordinates

6.11.2 Example of proton coordinates file (.mdh)

Crambin Pro/Leu based on X-ray, EM1, TR, Jan. 89

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1THR</td>
<td>H1</td>
<td>1</td>
<td>1.793</td>
<td>1.409</td>
</tr>
<tr>
<td>1THR</td>
<td>H2</td>
<td>2</td>
<td>1.630</td>
<td>1.419</td>
</tr>
<tr>
<td>1THR</td>
<td>H3</td>
<td>4</td>
<td>1.708</td>
<td>1.489</td>
</tr>
<tr>
<td>1THR</td>
<td>CA1</td>
<td>6</td>
<td>1.695</td>
<td>1.200</td>
</tr>
<tr>
<td>1THR</td>
<td>CB1</td>
<td>8</td>
<td>1.818</td>
<td>1.358</td>
</tr>
<tr>
<td>1THR</td>
<td>G1</td>
<td>10</td>
<td>2.015</td>
<td>1.291</td>
</tr>
<tr>
<td>1THR</td>
<td>HC21</td>
<td>12</td>
<td>1.922</td>
<td>1.122</td>
</tr>
<tr>
<td>1THR</td>
<td>HC22</td>
<td>13</td>
<td>1.768</td>
<td>1.179</td>
</tr>
<tr>
<td>1THR</td>
<td>HC23</td>
<td>14</td>
<td>1.767</td>
<td>1.067</td>
</tr>
<tr>
<td>2THR</td>
<td>H</td>
<td>18</td>
<td>1.547</td>
<td>1.080</td>
</tr>
<tr>
<td>2THR</td>
<td>CA1</td>
<td>20</td>
<td>1.344</td>
<td>1.242</td>
</tr>
<tr>
<td>2THR</td>
<td>CB1</td>
<td>22</td>
<td>1.192</td>
<td>1.054</td>
</tr>
<tr>
<td>2THR</td>
<td>G1</td>
<td>24</td>
<td>1.261</td>
<td>0.891</td>
</tr>
<tr>
<td>2THR</td>
<td>HC21</td>
<td>26</td>
<td>1.137</td>
<td>1.119</td>
</tr>
<tr>
<td>2THR</td>
<td>HC22</td>
<td>27</td>
<td>1.240</td>
<td>1.255</td>
</tr>
<tr>
<td>2THR</td>
<td>HC23</td>
<td>28</td>
<td>1.307</td>
<td>1.127</td>
</tr>
</tbody>
</table>

6.12 The GROMOS proton codes file (.htyp)

This file contains GROMOS proton codes. This file is only used in the GROMOS programs (dinoem and dinomd). GROMOS uses united atoms and contains thus only polar
protons. The apolar protons are generated at every step during the refinement and the forces must be set back on the real atoms. The proton codes file (.htyp) contains the information about the existing and virtual protons in GROMOS. The codes are the same as those used in GROMOS for distance restraining. Each proton is characterised by five numbers (codes): the first four correspond to the real GROMOS atoms (i,j,k,l0 of which the proton position is a function and the last one defines the type of proton. The proton types used in DINOSAUR are:

0 = real hydrogen atom (i) (e.g. polar protons)
1 = virtual H, aliphatic CH1 (i, j, k, l) (e.g. H[1] protons)
2 = virtual H, aromatic CH1 (i, j, k) (e.g. H[2] and H[3] protons in Tyr)
4 = virtual H, aliphatic CH2 (i, j, k) (e.g. H[1] in Tyr and H[2] in Cys)
5 = virtual H, CH3 groups (i, j)

(i) is the number of the atom to which the proton is attached (or the proton number itself for type 0). The other atoms (j, k, l) are the number of the other atoms of which the proton position is a function. For H[1] (aliphatic CH1, type 1) in amino acids (except in Gly), for example, the proton position is calculated from the C, Cα, N and Cβ coordinates and the NOE forces need thus to be set back on these "real" GROMOS atoms. For a more complete description of the virtual atoms in GROMOS see de Vlieg et al. (1986).

This file can be created from a GROMOS structure and a proton only file (.mdh) with the procedure mkhtyp (see section 5.1.5). The file contains the following information:

- (a80) TITLE
- (i5) NHTOT: number of protons

For each proton five codes are read:

- (5i5) I,J,K,L: atom number of which the proton position is a function
  CODE: code defining the type of proton

Conventions for the ordering of the atoms (I,J,K,L)

Type 0 (real hydrogen atom):
  i = proton number (existing proton in GROMOS)
  j,k,l = 0

Type 1 (virtual H, aliphatic CH1):
  i = number of the atom to which the proton is attached
  j,k,l = number of the other three atoms attached to that atom

Type 2 (virtual H, aromatic CH1):
  i = number of the atom to which the proton is attached
  j,k = number of the two other atoms attached to that atom
  l = 0

Type 4 (virtual H, aliphatic CH2):
  i = number of the atom to which the proton is attached
  j,k = number of the two other atoms attached to that atom
  If j > i and k < i proton 1 (e.g. HCB1) generated
  If j < i and k > i proton 2 (e.g. HCB2) generated
l = 0

Type 5 (virtual H, CH3 groups)
i = number of the methyl carbon to which the proton is attached
j = number of the atom to which the methyl carbon is attached
k,l = 0

Example

To the following GROMOS structure (GROMOS numbering)

13PHE  N  109  -0.539  -0.337  0.038
13PHE  H  110  -0.453  -0.389  0.043
13PHE  CA  111  -0.521  -0.195  0.002
13PHE  CB  112  -0.374  -0.155  -0.029
13PHE  CG  113  -0.350  -0.008  -0.047
13PHE  CD1  114  -0.382  0.063  -0.163
13PHE  CD2  115  -0.290  0.049  0.065
13PHE  CE1  116  -0.362  0.206  -0.161
13PHE  CE2  117  -0.262  0.186  0.063
13PHE  C2  118  -0.299  0.263  -0.047
13PHE  C  119  -0.616  -0.159  -0.116
13PHE  O  120  -0.696  -0.066  -0.103

corresponds the following proton file (.mdh) (all atoms numbering)

13PHE  H  171  -0.453  -0.389  0.043
13PHE  H  CA1  173  -0.547  -0.136  0.090
13PHE  H  CB1  175  -0.312  -0.189  0.053
13PHE  H  CB2  176  -0.344  -0.206  -0.121
13PHE  HCD11  179  -0.420  0.013  -0.251
13PHE  HCD21  181  -0.266  -0.011  0.151
13PHE  HCE11  183  -0.394  0.268  -0.244
13PHE  HCE21  185  -0.212  0.232  0.147
13PHE  HCZ1  187  -0.279  0.369  -0.046

and the following proton codes file (.htyp) (GROMOS numbering)

110  0  0  0  0
111  112  109  119  1
112  113  111  0  4
112  111  113  0  4
114  113  116  0  2
115  113  117  0  2
116  114  118  0  2
117  115  118  0  2
118  116  117  0  2
7. Examples

This chapter gives examples of DINOSAUR procedures for refinement with 2D NOE and 3D NOE-NOE data. The 2D examples are based on experimental data for crambin, a small 46-amino acid protein the structure of which has been determined in our laboratory (Bonvin et al., 1993a,c). The 3D DINOSAUR example is based on synthetic NMR data for a small helical nonapeptide corresponding to the second helix of crambin. The corresponding files required to run DINOSAUR as well as refinement results can be found in the dino/examples directory. This directory contains five subdirectories giving examples of various type of DINOSAUR refinement. This should allow you to play around with these data and get acquainted with the DINOSAUR procedures. Each directory contains a README file that we recommend to read before starting. The five example subdirectories are:

- dino/examples/2D : 2D DINOSAUR for a small helical peptide
- dino/examples/3D : 3D DINOSAUR for a small helical peptide
- dino/examples/cram : 2D DINOSAUR refinement of crambin with experimental NMR data using a slow-cooling simulated annealing
- dino/examples/cram_ens : 2D ensemble-averaged DINOSAUR refinement of crambin with experimental NMR data
- dino/examples/cram_tav : 2D time-averaged DINOSAUR refinement of crambin with experimental NMR data

All five examples use the GROMOS implementation of DINOSAUR. The GROMOS topology files in binary format (.topo) and formatted (.topofmt) are given. The binary files were created on a Silicon Graphics workstation with a double precision version of GROMOS (real*8). If you run DINOSAUR on another platform or use a single precision version of GROMOS, the binary topology files (.topo) should be created from the formatted files (.topofmt) using the GROMOS program PRORMT that converts a formatted topology into a binary file. The script file for doing this should look like:

```
l -s My_Molecule_Name.topo fort.20
prormt < My_Molecule_Name.topofmt
/bin/rm fort.20
```

In most of the DINOSAUR programs and scripts, default parameters are defined. Their values are normally given when a program is asking for a parameter. If you do not want to modify the parameter value you can just type return (<cr>) and the default value (or the value previously read from an old file will be kept). The same applies to DINOSAUR scripts. When starting DINOSAUR you can define a default molecule name with dinoset (the default name at the beginning is Flintstone). When creating a script file (.job) DINOSAUR will ask you several file names; by typing <cr> the default molecule name with default extensions for the various file types will be used (see section 5.1.1).

In the following examples, a few conventions are used: everything appearing on your screen is given in Courier fonts, a command is always preceded by a % sign and commands, text and parameters supplied by the user are indicated in bold.
7.1 2D DINOSAUR refinement example (crambin)

Crambin is a small 46-amino acid protein. It contains a total of 735 atoms, 315 of which are protons. The GROMOS coordinate file contains however only 396 atoms since GROMOS uses united atoms and only polar protons are explicitly included in the calculations. The following files given in the dino/examples/cram directory will be used in the following examples:

- **cram.md**: GROMOS coordinate file
- **cram.topo**: GROMOS topology file (binary)
- **cram.noe**: Experimental 2D NOE buildups measured at 500 MHz for six mixing times (20, 40, 80, 120, 160 and 250 ms) including stereospecific assignments. Peaks involving stereospecifically unassigned protons were treated as overlapping and the intensities of the peaks for the two diastereotopic protons were summed.
- **cram.shift**: experimental chemical shifts file
- **cram.stereo**: stereospecific assignment file
- **cram.disre**: qualitative distance constraints in GROMOS format corresponding to peaks with bad buildup series or close to the water frequency
- **cram.dihre**: dihedral angle constraints file in GROMOS format
- **cram.s2**: file with generalised order parameter calculated from a free MD simulation in water of crambin.

For a more complete description of the experimental NMR data for crambin see Bonvin et al., 1993a. In the following section we will illustrate the consecutive steps for creating the necessary files for DINOSAUR, for refining crambin with a combination of NOE restrained energy minimisation and a slow-cooling simulated annealing and finally for analysing the results in terms of R factors.

7.1.1 Getting started

After having defined the dinosaur alias (in your .cshrc file for example) with

```bash
% alias dinosaur source homedirectory_of_dinosaur/dino/csh/dinosaur
```

you start dinosaur with

```bash
% dinosaur<cr>
```
NOE refinement using 2D or 3D data (2/3) ?

Actual DINOSAUR working directory : /usr/ruuci5/albo/dino/examples/cram

New DINOSAUR working directory ?
/usr/ruuci5/albo/dino/examples/cram <cr>

New DINOSAUR working directory : /usr/ruuci5/albo/dino/examples/cram

*** For some information type: readme or remind ***

Have fun !

By just typing return <cr> for the new working directory, DINOSAUR will use the directory your are in as working directory. This also applies to the procedure dinodir. We can now first define the default molecule name with:

```
% dinoset <cr>
molecule name  :
cram <cr>
%
```

### 7.1.2 Creating a proton only file (.mdh) (mkhco)

```
% mkhco <cr>
! GENERATION OF PROTON COORDINATES OR TRAJECTORY FILES
! FROM GROMOS FILES

Creates or modify the MKHCO input file (y/n) ?
y <cr>

*** Program M K H C O I N *** Version 08-Jan-92 ***

Laboratory of NMR spectroscopy

Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994
```
This program creates or modifies the MKHCO input file for creation of all proton files for DINOSAUR.

Creates new file (0) or modify old one (1)? ( 1)

0 <cr>

The following options are implemented:
- 1: quit
- 0: write file
- 1: check file formats
- 2: check H codes for NOE refinement

Your choice? ( 0)

1 <cr>

number of coordinates files to be read? ( 1)

1 <cr>

number of records to skip at begin? ( 0)

<cr>

Record are used in case of MD trajectories. Typically a trajectory file contains coordinates saved at fixed time intervals during the simulation. A record corresponds thus to a structure at a defined time during the MD simulation.

number of records per file? ( 1)

<cr>

from every N records read, only the 1st one is used, N= ( 1)

<cr>

number of coordinates per record? ( 0)

1188 <cr>

The following format are implemented:
- 1: input coordinates and title are read from normal gromos coordinates files
- 0: input coordinates and title record are unformatted
- 1: input coordinates have been packed
  (see subr. pack), title record is unformatted
- 2: input coordinates are formatted
  (see subr. pack), title record is formatted
- 3: input coordinates are in standard formatted form

format input coordinates? ( -1)

<cr>

number of digits behind decimal? ( 3)

<cr>

sequence number of the molecule to be analysed? ( 1)

<cr>

number of solute atoms? ( 0)

396 <cr>

box coordinates are read {no:0,yes:2}? ( 0)

<cr>

(used is case of a MD trajectory in water with constant pressure: the box dimensions are also written to the trajectory file)

output H trajectory form(0) or unform(1)(IRMA/DINOSAUR)? ( 1)

<cr>

The following options are implemented:
- 1: quit
- 0: write file
- 1: check file formats
- 2: check H codes for NOE refinement

Your choice? ( 1)

2 <cr>
These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated in mkhco for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.

<table>
<thead>
<tr>
<th>number of atom codes to be read</th>
<th>NIAT ? (3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>one (planar) H</td>
<td>? (15 16 0 0)</td>
</tr>
<tr>
<td>one (dihedral) H</td>
<td>? (12 29 0 0)</td>
</tr>
<tr>
<td>two 180 degrees (dihedral) H</td>
<td>? (0 0 0 0)</td>
</tr>
<tr>
<td>three or two 120 degrees (dihedral) H</td>
<td>(13 14 32 0)</td>
</tr>
</tbody>
</table>

Give now the integer atom codes of existing protons in GROMOS whose names do not begin with an H

| codes for protons in GROMOS without H ? (0 0 0 0) |

The following options are implemented:
-1 : quit
0 : write file
1 : check file formats
2 : check H codes for NOE refinement

Your choice ? (2)

0

File name ? ()
cram.hcoin

The following options are implemented:
-1 : quit
0 : write file
1 : check file formats
2 : check H codes for NOE refinement

Your choice ? (0)
-1

*** End program MKHCoin ***

STOP statement executed

MKHCoin input file name ?
cram.hcoin
MKHCoin output file name ?
cram.hcoout
Molecular topology file name ?
cram.topo
GROMOS H coord. output file name (form.)?
cram.mdh
GROMOS H coord. output file name (bin.)?
cram.hco
Number of input coord./traj. files ?
1
Coordinates/trajectory file number 1 ?
cram.md
mkhco.job created!
execute or submit it
%
Two files have been generated at this point:

- **cram.hcoin** (the mkhco input file):

```
1         0         1         1      1188        -1         3
1       396         0
3
15        16         0
12        29         0
0         0         0
13        14        32
0         0         0
1
```

- **mkhco.job** (DINOSAUR script file for running mkhco):

```
# ! CONVERSION OF GROMOS FILES TO PROTONS COORDINATES
chdir /usr/ruuci5/albo/dino/examples/cram
ln -s  cram.topo   fort.10
ln -s  cram.mdh    fort.92
ln -s  cram.hco    fort.91
ln -s  cram.md     fort.11
(/bin/time /usr/ruuci5/albo/dino/src/mkhco <cram.hcoin) > & cram.hcoout
/bin/rm fort.10
/bin/rm fort.11
/bin/rm fort.92
/bin/rm fort.91
```

By executing mkhco.job two files are created: cram.mdh, the proton only file, and cram.hco, a proton trajectory file with only one structure/record in this particular case. This latter file can be discarded at this point. We now have all necessary files to create the DINOSAUR input file.

### 7.1.3 Creating the DINOSAUR input file (dinoin)

% dinoin <cr>

*** Program D I N O I N *** Version 14–Oct–93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991–1994

This program creates or modifies the input file for the DINOSAUR refinement programs.

Options: 0=create DINOSAUR input 1=modify old (  0)
0 <cr>
Title of the DINOSAUR input file ? ()
Crambin DINOSAUR <cr>
Symmetrical dimer (0=n,1=y) ? (  0)
0 <cr>
Number of NOE data sets ? (  1)
1 <cr>
H coord. scaling factor (--> Angstrom) ? (  1.000E+01)
10.0 <cr>
A chemical shift list will be read (y=1,n=0) ? (  0)
1 <cr>
Order parameters S2 are included (y=1,n=0) ? (  0)
1 <cr>
S2 read from matrix (0:bin,1:form) or file(2)? (  2)
Nine different methods can be used to calculate the theoretical NOE intensities:
1: exact calculation via diagonalisation of the relaxation matrix
2: two spins approximation
3: simplified expansion of \( \exp(-\tau m R) \)
4: 2nd order approximation (expansion of \( \exp \))
5: Yip approximation
6: two spins approximation with correction for spin diffusion updated every N steps
7: simplified expansion with correction for spin diffusion updated every N steps
8: 2nd ord. exp. with correction for spin diffusion updated every N steps
9: same as 1 but with use of a spherical cutoff around the corresponding proton pair

NOE definition: ? (9)

Distance cutoff [Å] ? (4.500E+00)

Neighbours list updated every N steps N = ? (1)

Minimal H-H distance [Å] ? (0.000E+00)

Time (-1), ensemble(+1) or no averaging (0) ? (0)

Which experimental data do you want to skip:
-1: negative peaks
0: none
1: positive peaks

Exp. NOEs: (-1,0,1) ? (-1)

Symmetrise exp. NOEs (0:n,1:y)? (1)

Several NOE potential are implemented:
1: \( R = \sum w(\|I_{\text{exp}} - I_{\text{the}}\|^2) / \sum wI_{\text{exp}}^2 \)
2: \( R = \sum w(\|I_{\text{exp}} - I_{\text{the}}\|^2) / \sum wI_{\text{the}}^2 \)
3: \( R = \sum w(\|I_{\text{exp}} - I_{\text{the}}\|^2 / I_{\text{exp}}^2) / \sum w \)
4: \( R = \sum w(\|I_{\text{exp}} - I_{\text{the}}\|^2 / I_{\text{the}}^2) / \sum w \)
5: \( R = \sum w(\|I_{\text{exp}} - I_{\text{the}}\|^2 / 0.5(I_{\text{exp}} + I_{\text{the}})^2) / \sum w \)
6: \( R = \sum w(I_{\text{exp}} - I_{\text{the}})^2 / \text{daexp} / \sum w \)
   \[ \text{daexp} = (A0 + C I_{\text{exp}})^2 \]
   \[ A0 = \text{experimental noise level} \]
   \[ C = \text{experimental relative error} \]
7: \( R = \sum w(I_{\text{exp}} - I_{\text{the}})^2 / \text{det} / \sum w \)
   \[ \text{det} = \text{daexp} + \text{datheo} \]
   \[ \text{datheo} = \text{relative error on distances} \]
8: \( R = \sum w(I_{\text{exp}} - I_{\text{the}})^2 / \sum w \)
   \[ \text{datheo} = \text{absolute error on distances} \]
9: \( R = \sum w((1/I_{\text{exp}}^{1/6} - 1/I_{\text{theo}}^{1/6}))^2 / \sum w((1/I_{\text{exp}}^{1/3} - 1/I_{\text{theo}}^{1/3}))^2 \)
10: \( R = \sum w((1/I_{\text{exp}}^{1/6} - 1/I_{\text{theo}}^{1/6})^2 / \text{daexp} / \sum w \)
    \[ \text{daexp} = (A0 + C I_{\text{exp}})^{1/3} \]
    \[ A0 = \text{experimental noise level} \]
    \[ C = \text{experimental relative error} \]

NOE potential: ? (6)

Flat well potential (0:n,1:y)? (1)

Two gradient type are implemented:
1: \( dA = -T_{\text{mix}}dR \)
Gradient definition ? ( 1)

1 <cr>

The NOE gradient will be updated if the proton displacements > dxyzmax

Maximum proton displacement in A DXYZMAX = ( 1.00E-02)

0.01 <cr>

**************************************************************

Check these parameters (T) or continue (F) ? (T)

F <cr>

**************************************************************

*** Now determine protons information ***

**************************************************************

H names/numbers from 0=GROMOS/DG 1=DISCOVER 2=none ? ( 0)

0 <cr>

File name ? ()
cram.mdh <cr>

Proton file read from:
Crambin average DINOSAUR 1000 steps EM steepest dist+dihed+NOEs

Different dynamical process can be described by averaging the corresponding elements of the relaxation matrix:

0 : no internal motion (no averaging)
1 : slow motion (<r**-6> averaging)
3 : fast motion (<r**-3> averaging with matrix contraction)
4 :fast motion (<r**-3> averaging without matrix contraction)

type of averaging for the methyl protons ? ( 3)

3 <cr>
type of averaging for the aromatic protons ? ( 1)

0 <cr>

Now checking proton list for methyls and aromatics
28 methyl groups found
3 aromatic groups found

**************************************************************

Check protons & dynamics (T) or continue (F) ? (T)

F <cr>

**************************************************************

Data set number 1 ***

**************************************************************

H2O (0) or D2O (1) data set ? ( 0)

0 <cr>

Identification string ? ( parameters for data set # 1)

Experimental NOEs from cram.no <cr>

Spectrometer frequency (Hz) ? ( 5.000E+08)

5.e+8 <cr>

Overall correlation time (s) ? ( 2.000E-09)

2.e-9 <cr>

Methyl correlation time (s) ? ( 1.000E-10)

1.e-10 <cr>

Additional leakage rate on diagonal(~1/T1)(s-1)? ( 8.000E-01)

0.8 <cr>

NOE force constant ? ( 4.000E+01)

40 <cr>

Maximal force ? ( 1.000E+01)
Minimal force ? \((-1.000E+01)\)

Experimental noise level ? \((1.000E+04)\)

Experimental relative error ? \((5.000E-02)\)

11 types of scaling are actually implemented:

0 : with reference peaks
1 : with all peaks, separately for each mixing time
2 : with all peaks, averaged on all mixing times
3 : with intraresidue peaks, separately
4 : with intraresidue peaks, averaged
5 : with interresidue peaks, separately
6 : with interresidue peaks, averaged
7 : separately for intra- and interresidue peaks and for each mixing time separately
8 : separately for intra- and interresidue peaks and averaged on all mixing times
9 : separately for short, medium and long range NOEs and for each mixing time separately
10 : separately for short, medium and long range NOEs and averaged on all mixing times

Type of scaling ? ? \((0)\)

Number of reference peaks ? \((0)\)

For each reference peak give:
atom number 1, atom number 2
(same numbers as on the proton (mdh) file)
Proton numbers ? \((0 \ 0)\)

9 ALA H CA1 9 ALA H CB1

Proton numbers ? \((0 \ 0)\)

9 ALA H CB1 9 ALA H CA1

Proton numbers ? \((0 \ 0)\)

24 ALA H CA1 24 ALA H CB1

Proton numbers ? \((0 \ 0)\)

24 ALA H CB1 24 ALA H CA1

Proton numbers ? \((0 \ 0)\)

27 ALA H CA1 27 ALA H CB1

Proton numbers ? \((0 \ 0)\)

27 ALA H CB1 27 ALA H CA1

Minimum scaling factor ? \((-1.000E+20)\)

Maximum scaling factor ? \((1.000E+20)\)

NOEs from 0:BUILDUP 1:NOEfile 2:BIOSYM 3:none ? \((0)\)

File name ? ()
cram.noe <cr>
Overlapping peaks will be skipped (T/F) ? (F)

f <cr>

How to treat overlapping peaks ?

0: treats the corresponding protons as if they were in slow exchange: averages distributes the NOE forces on both protons according to $1/r^{**6}$.  
1: divides their intensity by two (should be used in combination with dynamic assignments)

Your choice ? (0)

1 <cr>

Experimental NOEs from:
crambin  experimental buildup series at 500 MHz

(now reading the data)

735 peaks read
182 bad peaks
9 unassigned peaks
544 peaks left for processing
120 peaks with unassigned or overlapping stereospecific protons

The mixing times are:
1: 0.0200  2: 0.0400  3: 0.0800  4: 0.1200  5: 0.1600  6: 0.2500

Number of mixing times to be used? (6)

4 <cr>
Mixing time number ? (1)

2 <cr>
Mixing time (s) ? (4.000E-02)
<cr>
Weighting factor ? (1.000E+00)
<cr>
Mixing time number ? (2)

3 <cr>
Mixing time (s) ? (8.000E-02)
<cr>
Weighting factor ? (1.000E+00)
<cr>
Mixing time number ? (3)

5 <cr>
Mixing time (s) ? (1.600E-01)
<cr>
Weighting factor ? (1.000E+00)
<cr>
Mixing time number ? (4)

6 <cr>
Mixing time (s) ? (2.500E-01)
<cr>
Weighting factor ? (1.000E+00)
<cr>

******************************************************************************

Check again or continue (T/F) ? (T)

F <cr>

******************************************************************************

** Select now the NOEs for the refinement.  **
*** the selection criterion is the difference ***
*** in residue numbers.  ***
******************************************************************************

Give diff. min and diff. max ? (0 1000)

0 1000 <cr>
Data set # 1 :  544 selected NOEs

New selection (T) or continue (F) ? (F)
f <cr>

Keep NOEs between H bound to the same C and intra aromatic(Y:1,N:0)? ( 0)
0 <cr>

Data set # 1 :  31 peaks skipped because
H bound to the same atom or intra aromatic

Skip intra proline peaks ?(Y:1,N:0) ( 0)
l <cr>

Data set # 1 :  29 intraproline peaks skipped

Symmetrise exp. NOEs and skip redundant ?(Y:1,N:0) ( 0)
l <cr>

Data set # 1 :  117 redundant peaks skipped
367 peaks left for refinement

Add random noise to the data (T/F)? (F)
f <cr>

Randomly delete peaks (T/F)? (F)
f <cr>

Use stereo file (T/F)? (T)
t <cr>
File name of STEREO-ASSIGNMENT file? ()
cram.stereo <cr>

Stereo assignments read from file :
Stereo assignments J-COUPLING and Pro-data

        number of assignment read : 40
        number of stereoassigned proton pairs : 18

Apply stereospecific assignments (1:y,0:n) ?
WARNING: only if new NOEs read from result file(s)
or old input file without stereoass. !

Your choice ? ( 0)
0 <cr>

Dynamic stereospecific assignment is implemented.
The options are :
    0 : no dynamic assignment
    1 : dynamic assignment using new data
    2 : dynamic assignment using data from
        the old input file

Dynamic stereospecific assignments ? ( 0)
l <cr>

Upper limit for dynamic assignment ? ( 1.000E+04)
10000.0 <cr>

Include aromatic protons (T/F)? (T)
t <cr>

Include VAL/LEU methyls (T/F)? (T)
t <cr>

Data set # 1 :  dynamic assignment for 156 peaks.

Dir+file name of the output file? ()
cram.inp <cr>

Write exp. NOE constraints to separate file (y:1,n:0) ? ( 0)
<cr>

*** End program D I N O I N ***
7.1.4 Creating the GROMOS proton codes file (.htyp) (mkhtyp)

Before starting the refinement with NOE restrained EM or MD we need another file, the proton codes file (see section 6.11.2). It can be created from a GROMOS coordinates and proton only file with **mkhtyp**.

% mkhtyp <cr>

*** Program M K H T Y P *** Version 26-Mar-91 ***

Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

*************************************************

Name of GROMOS coordinates file ? ()
cram.md <cr>

Atom names read from file :
Crambin average DINOSAUR 1000 steps EM steepest dist + dihed + NOEs
Name of GROMOS proton coordinates file ? ()
cram.mdh <cr>

Atom names read from file :
Crambin average DINOSAUR 1000 steps EM steepest dist + dihed + NOEs

How to treat aromatic proton HCD11 in residue 13PHE ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? ( 1)
1 <cr>

How to treat aromatic proton HCD21 in residue 13PHE ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? ( 1)
1 <cr>

How to treat aromatic proton HCE11 in residue 13PHE ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? ( 1)
1 <cr>

How to treat aromatic proton HCE21 in residue 13PHE ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? ( 1)
1 <cr>

How to treat aromatic proton HCD11 in residue 29TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? ( 1)
1 <cr>

How to treat aromatic proton HCD21 in residue 29TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)
Your choice ? (1)

1 <cr>
How to treat aromatic proton HCE11 in residue 29TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? (1)

1 <cr>
How to treat aromatic proton HCE21 in residue 29TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? (1)

1 <cr>
How to treat aromatic proton HCD11 in residue 44TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? (1)

1 <cr>
How to treat aromatic proton HCD21 in residue 44TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? (1)

1 <cr>
How to treat aromatic proton HCE11 in residue 44TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? (1)

1 <cr>
How to treat aromatic proton HCE21 in residue 44TYR ?
0: pseudo (force set on CG or CZ)
1: explicit (if separately observed from partner or 1/r**6 averaging or dynamic assignment)

Your choice ? (1)

1 <cr>
Name of GROMOS proton types ? ()
cram.htyp <cr>

Title for the new file ? (cram.htyp)
Crambin GROMOS proton codes, all aromatic protons explicit <cr>

*** End program M K H T Y P ***

7.1.5 NOE restrained energy minimisation (dinoem)

This gives an example of the set-up of restrained energy minimisation with NOE, distance and dihedral restraints using the steepest descent algorithm in the GROMOS implementation of DINOSAUR.

% dinoem <cr>
! NOE's RESTRAINT ENERGY MINIMALIZATION

Creates or modify the EM input file (y/n) ?
y <cr>
This program creates or modifies the GROMOS input file for restrained energy minimisation.

Creates new file (0) or modify old one (1)? ( 0)

0 <cr>

The following options are implemented:

-1 : quit
0 : write file
1 : check file formats
2 : check box
3 : check minimisation parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice? ( 0)

1 <cr>

number of solute molecules NPM ? ( 1)

1 <cr>

number of solvent molecules NSM ? ( 0)

0 <cr>

NTX = 0 : coord. x are read from tape21 (unform.)
= 1 : coord. x are read from tape21 (form.)
= 2 : coord. x are read from tape21 (unform.)
= 3 : coord. x and box are read from tape21 (form.)
= 4 : coord. x and box are read from tape21 (unform.)

NTX = ? ( 1)

1 <cr>

NTCX = 0 : no constraints are read from tape23 (NTC>1)
= 1 : solute constraints are read from tape23 (NTC>1)

NTCX = ? ( 0)

0 <cr>

NTXO = 0 : final coord. are written to tape31 (unform.)
= 1 : final coord. are written to tape31 (form.)
= 2 : final coord. are written to tape31 (unform.)
= 3 : final coord. x and box are written to tape21 (form.)
= 4 : final coord. x and box are written to tape21 (unform.)

NTXO = ? ( 1)

1 <cr>

The following options are implemented:

-1 : quit
0 : write file
1 : check file formats
2 : check box
3 : check minimisation parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement
Your choice ? (  1)
2 <cr>

NTB < 0 : periodicity is applied
   box is truncated octahedron (BETA=90)
   = 0 : no periodicity is applied
   > 0 : periodicity is applied
      box is rectangular or monoclinic depending on beta
   if abs(NTB)=2 the virial is calculated (BETA=90)

NTB =          ? (  0)
0 <cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (   2)
3 <cr>
EM Algorithm 1=Steep.Desc., 2=Conj. Grad.? (    1)
1 <cr>
maximum number of steps NSTLIM   ? (   300)
300 <cr>
new conj. grad. started after x steps NCYC? (    1)
1 <cr>
SHAKE initial coordinates 1=yes 2=no INIT? (    1)
1 <cr>
stop if energy change less than DELE? ( 1.000E-05)
<cr>
initial step size or unit-length DX0 ? ( 1.000E-02)
<cr>
maximal step size or unit-length DMX ? ( 5.000E-02)
<cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (    3)
4 <cr>

NTC = 1 : SHAKE is not performed, NITER=1
   = 2 : constraint atom pairs are taken from IBH-JBH
      if NTCC=0
   = 3 : in addition, constraint atom pairs are taken
      from IB--JB if NTCC=0

NTC =            ? (    1)
1 <cr>

NTCC = 0 : constraint lengths are taken from B0 using
  IBH JBH ICBH IB JB and ICB
 = 1 : constraint LENGTHS are taken sequentially
      from CONSTR using ICO AND JCO

NTCC =         ? (    0)
relative tolerance for coord. resetting ? ( 1.000E-04)

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 4)

5

NTF = 1 : complete interaction is calculated
  = 2 : bond interactions involving h-atoms are omitted
  = 3 : in addition bond interactions involving no h-atoms
       are omitted
  = 4 : in addition bond-angle interactions involving
       h-atoms c are omitted
  = 5 : in addition bond-angle interactions involving no
       h-atoms are omitted
  = 6 : in addition dihedral interactions involving h-atoms
       are omitted
  = 7 : in addition dihedral interactions involving no
       h-atoms are omitted
  = 8 : in addition non-bonded interactions are omitted
  = 12: only bond interactions involving h-atoms are
       calculated
  = 13: in addition bond interactions involving no h-atoms
       are calculated

NTF = ? ( 1)

1

NTID = 0 : improper dihedral interaction is skipped
  = 1 : improper dihedral interactions involving no
       h-atoms are calculated
  = 2 : in addition, improper dihedral interactions
       involving h-atoms are calculated

NTID = ? ( 2)

2

NTN = 1 : non-bonded interaction is calculated using an atom
       pair-list see subrs. nbpal and nonbal
  = 2 : non-bonded interaction is calculated using a
       molecular pair list
  = 3 : non-bonded interaction is calculated using a grid
       see subr. nonbb
  = 4 : non-bonded interaction is calculated directly by
       scanning all pairs see subr. nonbp

NTN = ? ( 2)

2

NRE(1..2..) = atom sequence numbers of the last atoms of each
             group (.ie. NRP) NRE(3) is set equal to the last
             solute atom and NRE(4) to the last atom

NRE(1..4) = ? ( 0 0 0 0)

0 0 396

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice? ( 5)
6 <cr>

NTNB = 0 : no non-bonded pair list is made in the first step
= 1 : a non-bonded pair list is made in the first step
(if NTN=1 or 2)
<cr>
NTNB = ? ( 1)
<cr>
NSNB : after NSNB steps the non-bonded pair list will
be updated (if NTN=1 or 2)
<cr>
NSNB = ? ( 10)
<cr>
cut-off radius (for select. pairs) RCUTP ? ( 8.000E-01)
<cr>
switching radius (--- S(R)=1) RSWI2 ? ( 1.000E+01)
<cr>
cut-off radius (--- S(R)=0)(>=RSWI2)RCUT2 ? ( 1.000E+01)
<cr>
min. length of an edge of a grid cell CELM? ( 2.000E-01)
<cr>
RCUTL = radius within which all neighbour grid-cells are
searched for charge groups or solvent molecules
must be smaller than BOX(M)-CEL(M)
or (BOX(M)-CEL(M))*SQRT(3)/2
<cr>
RCUTL = ? ( 1.200E+00)
<cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice? ( 6)
7 <cr>
print data every NTPR steps
<cr>
NTPR = ? ( 10)
10 <cr>

Note: this parameter also controls the output frequency of the theoretical NOE intensities
written to the NOE trajectory file. Do not choose a too small value to avoid generating huge
NOE trajectories in case of large experimental data sets.

monitor dihedral transitions (1=yes 0=no)? ( 0)
0 <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? ( 7)

8 <cr>
position restraining (0=no 1 or 2=yes) NTR ? ( 0)

0 <cr>
NTDR = 0 : no restraining
  = 1 or 2 : standard distance restraining
  = 3 : distance and NOE restraining
  = 4 : NOE restraining

NTDR = ? ( 4)

3 <cr>
dis. restr. force constant CDIS ? ( 2.000E+03)

2000.0 <cr>
which part of the potential is harmonic ? ( 1.000E-01)

0.1 <cr>
carbon hydrogen distance DISH ? ( 1.090E-01)
<cr>
CH1-CH3 distance DISC ? ( 1.530E-01)
<cr>
dihedral restraining (0=no 1=yes) NTDLR ? ( 0)

1 <cr>
dih. restr. force constant CDLR ? ( 1.100E+02)

110 <cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check minimisation parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 8)

9 <cr>
number of atom codes to be read NIAT ? ( 3)
<cr>

These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated at every EM step for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.

one (planar) H ? ( 15 16 0 0)
<cr>
one (dihedral) H ? ( 12 29 0 0)
<cr>
two 180 degrees (dihedral) H ? ( 0 0 0 0)
<cr>
three or two 120 degrees (dihedral) H? ( 13 14 32 0)
<cr>

Give now the integer atom codes of existing protons in GROMOS whose names do not begin with an H codes for protons in GROMOS without H? ( 0 0 0 0)
<cr>

The following options are implemented :
-1 : quit
  0 : write file
1 : check file formats
2 : check box
3 : check minimisation parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? ( 9)
0 <cr>
File name ? ()
cram.emin <cr>
title ? ()
Crambin DINOSAUR 300 steps DINEM <cr>

The following options are implemented :
-1 : quit
 0 : write file
 1 : check file formats
 2 : check box
 3 : check minimisation parameters
 4 : check SHAKE parameters
 5 : check non-bonded interactions parameters
 6 : check cutoff parameters
 7 : check print parameters
 8 : check restraining parameters
 9 : check H codes for NOE refinement

Your choice ? ( 0)
-1 <cr>

*** End program M K E M I N ***

Now give GROMOS specific files:
*********************************
GROMOS input coordinates file name ? :
cram.md <cr>
GROMOS output coordinates file name ? :
cram.em <cr>
Standard output file name ? :
cram.emout <cr>
Molecular topology file name ? :
cram.topo <cr>
GROMOS input file name ? :
cram.emin <cr>
Distance constraints file name ? :
cram.disre <cr>
Dihedral restraints file name ? :
cram.dihre <cr>
Position restraining file name ? :
<cr>

Now give DINOSAUR specific files:
*********************************
Proton codes file name ? :
cram.htyp <cr>
DINOSAUR input file name ? :
cram.inp <cr>
Order parameters file name ? :
cram.s2 <cr>
Chemical shift file name ? :
cram.shift <cr>
dinoem.job created !
NOE trajectory output file name ? :
Two files have been created at this point:

- the GROMOS input file for the energy minimisation (`cram.emin`):

```
cram_em.noetraj <cr>
dinoem.job created !
execute or submit it

Crambin DINOSAUR 300 steps DINOEM
 1 0 1 0 1
 0 0.00000 0.00000 90.00000
 1 300 1 1 0.00010 0.01000 0.05000
 1 0 0.00010
 1 2 2 0 0
 1 10 0.80000 10.00000
 10 0
 0 19000.00000
 32000.00000 0.10000 0.10900 0.15300
 1 110.00000
 3
 15 16 0
 12 29 0
 0 0 0
 13 14 32
 0 0 0
```

- and the DINOSAUR script to run the EM job (`dinoem.job`):

```
# ! NOEs RESTRAINT ENERGY MINIMALIZATION
chdir /usr/ruuci5/albo/dino/examples/cram
/bin/rm FLnoeav MXavedis
ln -s  cram.md  fort.21
ln -s  cram.em  fort.31
ln -s  cram.topo  fort.20
ln -s  cram.disre  fort.26
ln -s  cram.dihre  fort.28
ln -s  cram.posrel  fort.24
ln -s  cram.posre2  fort.25
ln -s  cram.htyp  fort.89
ln -s  cram.inp  fort.90
ln -s  cram.s2  fort.92
ln -s  cram.shift  fort.94
ln -s  cram_em.noetraj  fort.98
(/bin/time /usr/ruuci5/albo/dino/src/dinoem) <cram.emin >& cram.emout
/usr/ruuci5/albo/dino/csh/neatout cram.emout dinoem.job cram.emin
/bin/rm fort.21
/bin/rm fort.20
/bin/rm fort.24
/bin/rm fort.25
/bin/rm fort.26
/bin/rm fort.28
/bin/rm fort.31
/bin/rm fort.89
/bin/rm fort.90
/bin/rm fort.92
/bin/rm fort.94
/bin/rm fort.98
```

7.1.6 NOE restrained slow-cooling simulated annealing (dinomd)

This gives an example of the set-up of a restrained slow-cooling simulated annealing from 1000 K to 1 K (cooling rate 10 K / 25 fs) with NOE, distance and dihedral restraints with the GROMOS implementation of DINOSAUR.
% dinomd <cr>
! NOES RESTRAINT MOLECULAR DYNAMICS

Creates or modify the MD input file (y/n) ? y <cr>

*** Program M K M D I N *** Version 07-Jan-93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

This program creates or modifies the GROMOS input file for restrained molecular dynamics.

Creates new file (0) or modify old one (1)? ( 0)
0 <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 0)
1 <cr>

number of solute molecules NPM ? ( 1)
1 <cr>

number of solvent molecules NSM ? ( 0)
0 <cr>

NTX = 0 : coordinates x are read from tape21 (unform.)
  1 : coordinates x are read from tape21 (form.)
  2 : coordinates x are read from tape21 (unform.)
  3 : x and f (predicted positions) are read from tape21 (unform.)
  4 : x and v are read from tape21 (unform.)
  5 : x and v are read from tape21 (form.)
  6 : x,v and box(1...3) are read from tape21 (unform.)
  7 : x,v and box(1...3) are read from tape21 (form.)
  8 : x,v, box and xc are read from tape21 (unform.)
  9 : x,v, box and xc are read from tape21 (form.)

NTX =

1 <cr>

NTCX = 0 : no constraints are read from tape23 (NTC>1)
  1 : solute constraints are read from tape23 (NTC>1)

NTCX =

0 <cr>

NTXO = 0 : final coordinates are written to tape31 (unform.)
  1 : final coordinates are written to tape31 (form.)

NTXO =

1 <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
2 : check box
3 : check MD parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? (      1)
2 <cr>

NTB < 0 : periodicity is applied
  box is truncated octahedron (BETA=90)
 = 0 : no periodicity is applied
> 0 : periodicity is applied
  box is rectangular or monoclinic depending on beta
  if abs(NTB)=2 the virial is calculated (BETA=90)

NTB =                          ? (      0)
0 <cr>

The following options are implemented :
-1 : quit
  0 : write file
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (      2)
3 <cr>

Initial temperature                TEMPI ? (  3.000E+02)
1000 <cr>
Random number generator seed(TEMPI>1.e-6)? (  60589)
<cr>
HEAT : if > 1.e-6 all solute and solvent velocities are
multiplied by HEAT
if <-1.e-6 all solute velocities are multiplied by
abs(HEAT), solvent velocities not changed

HEAT =                          ? (  0.000E+00)
<cr>

NTT = 0 : classical MD
= 1 : MD with velocity scaling (constant temperature)
= 2 : MD with velocity scaling (constant temperature)
  separate scaling for solute and solvent

NTT =                           ? (      1)
1 <cr>

Reference temperature              TEMP0 ? (  3.000E+02)
1.0 <cr>
T deviat. for rescaling of the velocities? (  1.000E+01)
10.0 <cr>
Temperature relaxation time        TAUTP ? (  1.000E-01)
0.01 <cr>

NTP = 0 : classical MD
= 1 : MD with isotropic position scaling
  (constant pressure, abs(NTB)) = 2
= 2 : MD with anisotropic diagonal (X-,Y-,Z-) position
  scaling (constant pressure, abs(NTB)) = 2

NTP =                           ? (      0)
0 <cr>

Number of submolecules forming 1 solute mol.? (      1)
1 <cr>
Numbers of last atom of the submolecules ? (      0)
396 <cr>

NDFMIN : number of degrees of freedom that will be
subtracted from the total number of degrees of freedom

NDFMIN = ? (6)

6 <cr>

NTCM = 0 : translational motion of and rotational motion about the centre of mass is not removed from the initial velocities (INIT=4)

= 1 : if is removed and NTCM is set equal to 0 (INIT<4)

NTCM = ? (1)

1 <cr>

Remove this motion (again) after NSCM steps ? (10000)

10000 <cr>

Two protocols are implemented:
- if NRUN (number of MD run) = 1 : standard MD
- if NRUN > 1 : annealing procedure; the temperature is then varied in NRUN runs from the initial value given by TEMPI to the final value given by TEMP0.

This works for both cooling and heating.

Number of MD runs NRUN ? (1)

100 <cr>

Number of steps NSTLIM ? (5000)

25 <cr>

INIT induces different starting procedures
- 1 : X(t) and V(t-dt/2) will be shaken initial centre of mass motion can be stopped
- 2 : as for INIT=1, except X(t) is not shaken
- 3 : as for INIT=2, except V(t-dt/2) is not shaken options 1-3 are meant for initialising a MD run
- 4 : as for INIT=4, except centre of mass motion is not changed; INIT=4 is required to continue a MD simulation without a discontinuity in the trajectory

INIT = ? (1)

1 <cr>

NTU = 1 : energies in KCAL/ MOLE
- 2 : energies in KJ/ MOLE; coordinates are multiplied by 0.1 and velocities by sqrt(4.184)
- 3 : energies in KJ/ MOLE

NTU = ? (3)

3 <cr>

Initial time [ps] ? (0.000E+00)

0.0 <cr>

Time step [ps] ? (2.000E-03)

0.001 <cr>

The following options are implemented:
-1 : quit
0 : write file
1 : check file formats
2 : check box
3 : check MD parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? (3)

4 <cr>

NTC = 1 : SHAKE is not performed, NITER=1
- 2 : constraint atom pairs are taken from IBH-JBH if NTCC=0
- 3 : in addition, constraint atom pairs are taken from IB-JB if NTCC=0

NTC = ? (3)

3 <cr>

NTCC = 0 : constraint lengths are taken from B0 using IBH JBH ICBH IB JB and ICB
= 1 : constraint LENGTHS are taken sequentially from CONSTR using ICO AND JCO

NTCC = ? ( 0)

0 <cr>
relative tolerance for coord. resetting ? ( 1.000E-04)

1.e-4 <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 4)

5 <cr>

NTF = 1 : complete interaction is calculated
  = 2 : bond interactions involving h-atoms are omitted
  = 3 : in addition bond interactions involving no h-atoms are omitted
  = 4 : in addition bond-angle interactions involving h-atoms are omitted
  = 5 : in addition bond-angle interactions involving no h-atoms are omitted
  = 6 : in addition dihedral interactions involving h-atoms are omitted
  = 7 : in addition dihedral interactions involving no h-atoms are omitted
  = 8 : in addition non-bonded interactions are omitted
  = 12 : only bond interactions involving h-atoms are calculated
  = 13 : in addition bond interactions involving no h-atoms are calculated
  = 14 15 16 17 : analogue to 4 5 6 7

NTF = ? ( 3)
3 <cr>

NTID = 0 : improper dihedral interaction is skipped
  = 1 : improper dihedral interactions involving no h-atoms are calculated
  = 2 : in addition, improper dihedral interactions involving h-atoms are calculated

NTID = ? ( 2)
2 <cr>

NTN = 1 : non-bonded interaction is calculated using an atom pair-list see subs. nbpal and nonbal
  = 2 : non-bonded interaction is calculated using a molecular pair list
  = 3 : non-bonded interaction is calculated using a grid see subr. nonbb
  = 4 : non-bonded interaction is calculated directly by scanning all pairs see subr. nonbp

NTN = ? ( 2)
2 <cr>

NRE(1..2..) = atom sequence numbers of the last atoms of each group (.i.e. NRP) NRE(3) is set equal to the last solute atom and NRE(4) to the last atom

NRE(1..4) = ? ( 0 0 0 0)
0 0 396 <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
2 : check box
3 : check MD parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? ( 5)
6 <cr>

NTNB = 0 : no non-bonded pair list is made in the first step
  = 1 : a non-bonded pair list is made in the first step
      (if NTN=1 or 2)
NTNB = ? ( 1)
<cr>
NSNB : after NSNB steps the non-bonded pair list will
      be updated (if NTN=1 or 2)
NSNB = ? ( 10)
<cr>
cut-off radius (for select. pairs) RCUTP ? ( 8.000E-01)
<cr>
switching radius (--> S(R)=1) RSWI2 ? ( 1.000E+01)
<cr>
cut-off radius (--> S(R)=0) RSWI2 ? ( 1.000E+01)
<cr>
min. length of an edge of a grid cell CELM? ( 2.000E-01)
<cr>
RCUTL = radius within which all neighbour grid-cells are
        searched for charge groups or solvent molecules
        must be smaller than BOX(M)-CEL(M)
        or (BOX(M)-CEL(M))*SQRT(3)/2
RCUTL = ? ( 1.200E+00)
<cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 6)
7 <cr>

print data every NTTPR steps NTTPR = ? ( 100)
25 <cr>

Note: this parameter also controls the output frequency of the theoretical NOE intensities
written to the NOE trajectory file. Do not choose a too small value to avoid generating huge
NOE trajectories in case of large experimental data sets.

monitor dihedral transitions (1=yes 0=no)? ( 1)
0 <cr>
write atomic coordinates (1=yes 0=no) ? ( 0)
<cr>
write atomic velocities (1=yes 0=no) ? ( 0)
<cr>
write various energies (1=yes 0=no) ? ( 0)
<cr>
write data unformatted(0) or formatted(2)? ( 2)
<cr>
The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 7)
8 <cr>
position restraining (0=no 1(or2)=yes) NTR ? ( 0)
<cr>
NTDR = 0 : no restraining
   = 1 or 2 : standard distance restraining
   = 3 : distance and NOE restraining
   = 4 : NOE restraining
 NTDR = ?
3 <cr>
dis. restr. force constant CDIS ? ( 4.000E+03)
2000 <cr>
which part of the potential is harmonic ? ( 1.000E-01)
 0.1 <cr>
carbon hydrogen distance DISH ? ( 1.090E-01)
<cr>
CH1-CH3 distance DISC ? ( 1.530E-01)
<cr>
dihedral restraining (0=no l=yes) NTDLR ? ( 0)
1 <cr>
dih. restr. force constant CDLR ? ( 1.100E+02)
110 <cr>
a perturbation potential is applied (0=no,1=yes)?
     ( 0)
<cr>
The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 8)
9 <cr>

These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated at every MD step for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.

number of atom codes to be read NIAT ? ( 3)
<cr>
one (planar) H ? ( 15 16 0 0)
<cr>
one (dihedral) H ? ( 12 29 0 0)
<cr>
two 180 degrees (dihedral) H ? ( 0 0 0 0)
three or two 120 degrees (dihedral) H? ( 13 14 32 0)

Give now the integer atom codes of existing protons in GROMOS whose names do not begin with an H:
codes for protons in GROMOS without H? ( 0 0 0 0)

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 9)
0 <cr>

File name ? ()
cram.anin <cr>
title ? ()
Crambin DINOSAUR annealing 1000 --> 1K in 2.5 ps <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 0)
-1 <cr>

*** End program M K M D I N ***

Now give GROMOS specific files:
*******************************
GROMOS input coordinates file name ? :
cram.em <cr>
GROMOS output coordinates file name ? :
cram.an <cr>
Standard output file name ? :
cram.anout <cr>
GROMOS coord. trajectory file name ? :
<cr>
Molecular topology file name ? :
cram.topo <cr>
GROMOS input file name ? :
cram.anin <cr>
Distance constraints file name ? :
<cr>
Dihedral restraints file name ? :
<cr>
Position restraining file name ? :
<cr>

Now give DINOSAUR specific files:
******************************************************************************

Protons codes file name ? :
cram.htyp <cr>
DINOSAUR input file name ? :
cram.inp <cr>
Order parameters file name ? :
cram.s2 <cr>
Chemical shift file name ? :
cram.shift <cr>
NOE trajectory output file name ? :
cram_an.noetraj <cr>
Run with time-averaged restraints (y,n)?
n <cr>
dinomd.job created ! (RENAME TO dinoan.job)
execute or submit it

Two files have been created at this point:
- the GROMOS input file for the restrained MD simulation (cram.anin):

```
Crambin DINOSAUR annealing 1000 --> 1K in 2.5 ps
  1         0         1         0     605891000.00000   0.00000         1
  0   0.00000   0.00000   0.00000  90.00000     100
  0         1   1.00000  10.00000   0.01000   0.10000
  0         1   0.06102 0.0007476   0.50000
396
6         1     10000
25         1   3     0.00000   0.00100
  3         0   0.00010
  3         2   2     0         0
1         10   0.80000 10.00000  10.00000   0.20000   1.20000
25         0
25         0     25     0     25     0     2
  0   19000.00000
32000.00000   0.10000   0.10900   0.15300
  0         0     2     0.00000   0.05000
  1 110.00000
3
15        16     0
12        29     0
  0         0     0
13        14     32
  0         0     0
```

- and the DINOSAUR script to run the MD job (dinomd.job (renamed to dinoan.job)):

```
# ! NOEs RESTRAINT MOLECULAR DYNAMICS
chdir /usr/ruuci5/albo/dino/examples/cram
ln -s cram.em fort.21
ln -s cram.an fort.31
ln -s cram.traj fort.12
ln -s cram.topo fort.20
ln -s cram.disre fort.26
ln -s cram.dihre fort.28
ln -s cram.posrel fort.24
ln -s cram.posre2 fort.25
ln -s cram.htyp fort.89
ln -s cram.inp fort.90
ln -s cram.s2 fort.92
ln -s cram.shift fort.94
ln -s cram_an.noetraj fort.98
(/bin/time /usr/ruuci5/albo/dino/src/dinomd) <cram.anin >& cram.anout
```
7.1.7 Calculating $R$ factors from a single GROMOS structure (rfacgr)

```
% rfacgr <cr>
  ! CALCULATION OF $R$-FACTORS

Creates or modify the GRRFAC input file (y/n) ? y <cr>

*** Program M K R F I N *** Version 07-Jan-93 ***

Laboratory of NMR spectroscopy

Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

*************************************************

This program creates or modifies the GRRFAC input file for $R$ factor calculation from GROMOS structures.

Creates new file (0) or modify old one (1)? (      1)
0 <cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check GROMOS proton codes

Your choice ? (   0)
1 <cr>

number of atom codes to be read NIAT ? (    3)
<cr>
```

These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.

```
one (planar) H  ? (    15    16    0    0)
<cr>
one (dihedral) H  ? (    12    29    0    0)
<cr>
two 180 degrees (dihedral) H  ? (    0    0    0    0)
<cr>
three or two 120 degrees (dihedral) H? (    13    14    32    0)
<cr>
```

Give now the integer atom codes of existing protons
in GROMOS whose names do not begin with an H codes for protons in GROMOS without H? (0 0 0 0)

The following options are implemented:
-1 : quit
  0 : write file
  1 : check GROMOS proton codes

Your choice? (1)
0 <cr>

File name? ()
cram.rfin <cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check GROMOS proton codes

Your choice? (0)
-1 <cr>

*** End program M K R F I N ***

GROMOS coordinates file name?:
cram.anem <cr>
Molecular topology file name?:
cram.topo <cr>
GROMOS input file name?:
cram.rfin <cr>
DINOSAUR input file name?:
cram.inp <cr>
R-factor output file name?:
cram_anem.rfac <cr>
Theo. NOEs output file name?:
cram_anem.theno <cr>
Distances output file name?:
cram_anem.dis <cr>
Order parameters file name?:
cram.s2 <cr>
Chemical shift file name?:
cram.shift <cr>
grrfac.job created!
execute or submit it

Two files have been created at this point:
- the grrfac input file (cram.rfin):

    3
   15  16  0
   12  29  0
    0  0  0
   13  14 32
    0  0  0

- and the DINOSAUR script to run the rfacgr job (grrfac.job):

    # ! CALCULATION OF R-FACTORS
cd /usr/ruuci5/albo/dino/examples/cram
ln -s cram.anem fort.91
ln -s cram.topo fort.10
ln -s cram.rfin fort.11
ln -s cram.inp fort.90
ln -s cram.s2 fort.92
ln -s cram.shift fort.94
ln -s cram_anem.theno fort.97
ln -s cram_anem.dis fort.98
(/bin/time /usr/ruuci5/albo/dino/src/grrfac) >& cram_anem.rfac
/bin/rm fort.91
/bin/rm fort.10
/bin/rm fort.11
/bin/rm fort.90
/bin/rm fort.92
/bin/rm fort.94
/bin/rm fort.97
/bin/rm fort.98

7.1.8 Analysing NOE trajectories (ananoe)

% ananoe <cr>

*** Program A N A N O E *** Version 14-Oct-93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994
******************************************************************************
This program analyses an NOE trajectory in terms of R-factor

Name of the DINOSAUR input file ? ()
cram.inp <cr>

Now reading the NOE constraints file:
Crambin DINOSAUR
Number of protons : 315
Coordinates scaling factor (-->A) : 10.00000

NOTYP  RFACYP  GRADYP  DXYZMAX  NSTEP  DISCUT
9       6       1       0.01000  1     4.500
Minimum H-H distance = 1.50 A
Flat well NOE potential: Vnoe = 0 if abs(Iexp-Itheo) < exp. error
Inclusion of the order parameters
Negative experimental NOEs will be skipped
Experimental NOEs will be symmetrized if possible

******************************************************************************
*** Parameters for NOE data set # 1 ***
******************************************************************************
H2O experimental NOEs from cram.noe, stereo from cram.stereo

FREQ    TAUC    TMETH    ROSTARO
5.000E+08 2.000E-09 1.000E-10 0.80000
-1.000E+20 < scaling factor < 1.000E+20

Force constant : 4.000E+01

Cutoffs max and min for forces : 1.000E+01-1.000E+01

Experimental noise level : 1.000E+04

Experimental relative error: 5.0 %

Mixing times : 0.04000  0.08000  0.16000  0.25000

Number of NOE constraints : 367

-----------------------------------------------

Dynamic stereospecific assignments for 156 peaks

Upper limit to switch off dynamic assignment : 1.000E+05

The following R factor definitions correspond to the DINOSAUR NOE potentials :

1 : R = sum[w*(Iexp-Ithe)**2] / sum[w*Iexp**2]
2 : R = sum[w*(Iexp-Ithe)**2] / sum[w*Ithe**2]
3 : R = sum[w*(Iexp-Ithe)**2] / Iexp**2 / sum w
4 : R = sum[w*(Iexp-Ithe)**2] / Ithe**2 / sum w
5 : R = sum[w*(Iexp-Ithe)**2] / 0.5*(Iexp + Ithe)**2 / sum w
6 : R = sum[w*(Iexp-Ithe)**2/ daexp] / sum w
daexp = (A0 + C Iexp)**2
A0 = experimental noise level
C = experimental relative error
7 : R = sum[w*(Iexp-Ithe)**2/ det] / sum w
det = daexp + datheo
datheo = absolute error on distances
8 : R = sum[w*(Iexp-Ithe)**2/ det] / sum w
datheo = absolute error on distances
9 : R = sum[w*(1/Iexp**((1/6)-1/Itheo**((1/6)))**2] / sum[w*(1/Iexp**((1/3))]
10 : R = sum[w*(Iexp**((1/6) - Itheo**((1/6)))**2/ daexp] / sum w
daexp = (A0 + C Iexp)**(1/3)
A0 = experimental noise level
C = experimental relative error

Four additional R factor definition are implemented for analysis

11 : R = sum[tm abs(Iexp-Itheo)] / sum [tm Iexp]
12 : R = sum[tm abs(Iexp**1/6-Itheo**1/6)] / sum[tm Iexp**1/6]
13 : R = sum[tm abs(Iexp-Itheo)] / sum[0.5 tm (Iexp+Itheo)]
14 : R = sum[tm abs(Iexp**1/6-Itheo**1/6)] / sum[0.5 tm (Iexp**1/6+Itheo**1/6]

NOE R factor/potential definition ? (6)
<cr>

*******************************
*** Data set number 1 ***
*******************************

Experimental noise level ? (1.000E+04)
<cr>
Experimental relative error ? (5.000E-02)
<cr>
NOE force constant ? (4.000E+01)
<cr>

The following options are implemented:
1: analyse all peaks
2: analyse selected peaks

Your choice ? (1)
NOEtraj. file(0) or form. NOE file(1) ? (  0)

Name of the NOE trajectory file ? (cram.inp)

cram_an.noetraj

Number of records to be skipped at begin ? (   0)

Number of records left to be read        ? (  2600)

Rescale NOE intensities (0:no,1:yes)? (   0)

ANANOE can calculate uncorrected R-factors or correct them
to account for the experimental errors defined as (noise + % Aexp);
in the latter case the difference between theo. and exp. NOEs is set to 0
if it is within the experimental error otherwise Atheo = Atheo +- exp. error.

Calculate uncorrected (0) or corrected (1) R-factors ? (   0)

Cluster size for averaging ? (    1)

Do you want to inspect some R-factors (T/F)? ( T)

The standard R-factor definitions are:

\[ DX = |THE-EXP| \]
\[ AX = |EXP| \]
\[ SX = 0.5(|EXP| + |THE|) \]
\[ AX6 = |EXP|^{1/6.} \]
\[ AT6 = |THE|^{1/6.} \]

1: \[ \frac{\sum(DX)}{\sum(AX)} \]
2: \[ \frac{\sum(TM*DX)}{\sum(TM*AX)} \]
3: \[ \frac{\sum(DX/AX)}{\sum(1)} \]
4: \[ \frac{\sum(TM*DX/AX)}{\sum(TM)} \]
5: \[ \frac{\sum((AX6-AT6))}{\sum(AX6)} \]
6: \[ \frac{\sum(TM*(AX6-AT6))}{\sum(TM*AX6)} \]
7: \[ \frac{\sum(DX)}{\sum(SX)} \]
8: \[ \frac{\sum(TM*DX)}{\sum(TM*SX)} \]
9: \[ \frac{\sum((AX6-AT6))}{0.5*\sum(AX6+AT6)} \]
10: \[ \frac{\sum(TM*(AX6-AT6))}{0.5*\sum(TM*(AX6+AT6))} \]
11: \[ \left[ \frac{\sum(WW*DX**2)}{\sum(WW*AX**2)} \right]^{1/2} \]

WW = 1/(NOISE+AX)

Give definition number : (    2)

R-factors: all, intra or interresidue peaks(0,1,2)? (   0)

(Entering plot subroutine SCURV)

Type: NWID,'FMA'

NWID = number of points on ordinate, default value  50
FMA = data format descriptor, default G11.4
(NWID + 2*FIELD-WIDTH) should not exceed 75 (terminal) or 128 (printer)
NWID should be < 107

Type / for use of default parameters

/ 0.3246 1.000
0.3367 I
0.3548 I
0.3593 I
0.3416 I
0.3486 I
0.3392 I
0.3532 I
0.3242 I
0.3256 I
0.3099 11.00 I
0.3202 I
0.3091 I
Write R-factors to a file ? :
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)
Your choice ? ( 1 )
0 <cr>
Do you want to inspect more results (T/F)? (T)
t <cr>

The standard R-factor definitions are:

DX  = ABS(THE-EXP)
AX  = ABS(EXP)
SX  = 0.5*(ABS(EXP)+ABS(THE))
AX6 = ABS(EXP)**(1./6.)
AT6 = ABS(THE)**(1./6.)

1: SUM(DX) / SUM(AX)
2: SUM(TM*DX) / SUM(TM*AX)
3: SUM(TM**2) / SUM(TM)
4: SUM(TM**2)/SUM(TM**2)
5: SUM((AX6-AT6) / SUM(AX6)
6: SUM(TM*(AX6-AT6)) / SUM(TM*AX6)
7: SUM(DX) / SUM(SX)
8: SUM(TM*DX) / SUM(TM*AX)
9: SUM((AX6-AT6) / 0.5*SUM(AX6+AT6)
10: SUM(TM*(AX6-AT6)) / 0.5*SUM(TM*(AX6+AT6))
11: [SUM(WW*DX**2)/SUM(WW*AX**2)]**1/2
   WW = 1/(NOISE+AX)
Give definition number : ( 2)

10 <cr>
R-factors: all, intra or interresidue peaks(0,1,2)? ( 0)

0.7539E-01 1.000  
0.7491E-01  I  I  
0.7785E-01  I  I  
0.8033E-01  I  I  
0.7842E-01  I  I  
0.7511E-01  I  I  
0.7523E-01  I  I  
0.8027E-01  I  I  
0.7115E-01  I  I  
0.7507E-01  I  I  
0.7238E-01  I  I  
0.7490E-01  I  I  
0.6888E-01  I  I  
0.7272E-01  I  I  
0.7036E-01  I  I  
0.7451E-01  I  I  
0.6922E-01  I  I  
0.6991E-01  I  I  
0.7299E-01  I  I  
0.7370E-01  I  I  
0.6975E-01 11.00  I  I  
0.7221E-01  I  I  
0.7344E-01  I  I  
0.6813E-01  I  I  
0.6804E-01  I  I  
0.6810E-01  I  I  
0.6808E-01  I  I  
0.6818E-01  I  I  
0.6724E-01  I  I  
0.6838E-01  I  I  
0.6571E-01 31.00  I  I  
0.6707E-01  I  I  
0.6408E-01  I  I  
0.6403E-01  I  I  
0.6374E-01  I  I  
0.6752E-01  I  I  
0.6451E-01  I  I  
0.6397E-01  I  I  
0.6591E-01  I  I  
0.6292E-01  I  I  
0.6428E-01 41.00  I  I  
0.6169E-01  I  I  
0.6452E-01  I  I  
0.6095E-01  I  I  
0.6065E-01  I  I  
0.6109E-01  I  I  
0.6118E-01  I  I  
0.5975E-01  I  I  
0.5875E-01  I  I  
0.5882E-01 50.00  I  I  

Write R-factors to a file ? :
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice ? ( 0)
1 <cr>
Name of the file ? ()
cram_an.R10 <cr>
Do you want to inspect more results (T/F)? (T)
f <cr>
Write R-factors to a file?:
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice? (1)
0 <cr>

Do you want to inspect some NOE energy functions (T/F)? (F)
t <cr>

Give selected peak number: (1)
183 <cr>

NOE restrained energy for peak 21THR HCG21 22PRO H CD2

<table>
<thead>
<tr>
<th>Energy</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>30.34</td>
<td>1.000</td>
</tr>
<tr>
<td>29.38</td>
<td>1</td>
</tr>
<tr>
<td>33.18</td>
<td>*</td>
</tr>
<tr>
<td>33.35</td>
<td>*</td>
</tr>
<tr>
<td>35.92</td>
<td>*</td>
</tr>
<tr>
<td>34.99</td>
<td>*</td>
</tr>
<tr>
<td>28.53</td>
<td>*</td>
</tr>
<tr>
<td>29.97</td>
<td>*</td>
</tr>
<tr>
<td>36.12</td>
<td>*</td>
</tr>
<tr>
<td>38.39</td>
<td>*</td>
</tr>
<tr>
<td>36.69</td>
<td>11.00</td>
</tr>
<tr>
<td>25.62</td>
<td>I</td>
</tr>
<tr>
<td>23.48</td>
<td>*</td>
</tr>
<tr>
<td>29.03</td>
<td>*</td>
</tr>
<tr>
<td>31.95</td>
<td>*</td>
</tr>
<tr>
<td>26.63</td>
<td>I</td>
</tr>
<tr>
<td>10.10</td>
<td>*</td>
</tr>
<tr>
<td>22.52</td>
<td>I</td>
</tr>
<tr>
<td>27.28</td>
<td>I</td>
</tr>
<tr>
<td>30.08</td>
<td>I</td>
</tr>
<tr>
<td>19.93</td>
<td>21.00</td>
</tr>
<tr>
<td>26.95</td>
<td>I</td>
</tr>
<tr>
<td>27.12</td>
<td>I</td>
</tr>
<tr>
<td>31.66</td>
<td>I</td>
</tr>
<tr>
<td>17.05</td>
<td>I</td>
</tr>
<tr>
<td>25.31</td>
<td>I</td>
</tr>
<tr>
<td>18.09</td>
<td>I</td>
</tr>
<tr>
<td>10.40</td>
<td>I</td>
</tr>
<tr>
<td>2.548</td>
<td>I</td>
</tr>
<tr>
<td>4.664</td>
<td>I</td>
</tr>
<tr>
<td>3.417</td>
<td>31.00</td>
</tr>
<tr>
<td>5.245</td>
<td>I</td>
</tr>
<tr>
<td>4.299</td>
<td>I</td>
</tr>
<tr>
<td>9.930</td>
<td>I</td>
</tr>
<tr>
<td>6.922</td>
<td>I</td>
</tr>
<tr>
<td>4.840</td>
<td>I</td>
</tr>
<tr>
<td>4.036</td>
<td>I</td>
</tr>
<tr>
<td>4.629</td>
<td>I</td>
</tr>
<tr>
<td>4.305</td>
<td>I</td>
</tr>
<tr>
<td>4.593</td>
<td>I</td>
</tr>
<tr>
<td>9.882</td>
<td>41.00</td>
</tr>
<tr>
<td>6.313</td>
<td>I</td>
</tr>
<tr>
<td>4.095</td>
<td>I</td>
</tr>
<tr>
<td>4.782</td>
<td>I</td>
</tr>
<tr>
<td>3.592</td>
<td>I</td>
</tr>
<tr>
<td>3.420</td>
<td>I</td>
</tr>
<tr>
<td>4.036</td>
<td>I</td>
</tr>
<tr>
<td>3.452</td>
<td>I</td>
</tr>
<tr>
<td>3.602</td>
<td>I</td>
</tr>
<tr>
<td>3.504</td>
<td>50.00</td>
</tr>
</tbody>
</table>

Write NOE energies to a file?:
0: none
1: curves
2: Cricket graph format
3: Plotref format (HP plotter)

Your choice? (0)
0 <cr>
Do you want to inspect more results (T/F)? (T)
f <cr>
New peak selection and/or new file (T/F)? (F)
f <cr>

Thank you for your visit!!!
See you...

*** End program A N A N O E ***

7.1.9 Analysing NOE files (.theno) (ananoe)

% ananoe <cr>

*** Program A N A N O E *** Version 14-Oct-93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

***************************************************************************
This program analyses an NOE trajectory in terms of R-factor
Name of the DINOSAUR input file? ()
cram.inp

Now reading the NOE constraints file:
Crambin DINOSAUR

Number of protons: 315
Coordinates scaling factor (-->A): 10.00000

NOTYP RFACYP GRADYP DXYZMAX NSTEP DISCUT
9 6 1 0.01000 1 4.500

Minimum H-H distance = 1.50 A
Flat well NOE potential: Vnoe = 0 if abs(Iexp-Itheo) < exp. error
Inclusion of the order parameters
Negative experimental NOEs will be skipped
Experimental NOEs will be symmetrized if possible

***************************************************************************
*** Parameters for NOE data set # 1 ***
***************************************************************************
H2O experimental NOEs from cram.noe, stereo from cram.stereo

FREQ TAUC TMETH ROSTARO
5.000E+08 2.000E-09 1.000E-10 0.80000
-1.000E+20 < scaling factor < 1.000E+20

Force constant : 4.000E+01

Cutoffs max and min for forces : 1.000E+01-1.000E+01

Experimental noise level : 1.000E+04

Experimental relative error: 5.0 %

Mixing times : 0.04000 0.08000 0.16000 0.25000

Number of NOE constraints : 367

***************************************

Dynamic stereospecific assignments for 156 peaks

Upper limit to switch off dynamic assignment : 1.000E+05

The following R factor definitions correspond to the DINOSAUR NOE potentials :

1 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2]}{\text{sum} [w*I_{\text{exp}}^2]} \]

2 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2]}{\text{sum} [w*I_{\text{theo}}^2]} \]

3 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2]}{\text{sum} [w*I_{\text{exp}}^2]} / \text{sum} w \]

4 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2]}{\text{sum} [w*I_{\text{theo}}^2]} / \text{sum} w \]

5 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2]}{0.5*(I_{\text{exp}} + I_{\text{theo}})^2} / \text{sum} w \]

6 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2/da_{\text{exp}}]}{\text{sum} w} \]

\[ da_{\text{exp}} = (A_0 + C I_{\text{exp}})^2 \]

7 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2/\text{det}]}{\text{sum} w} \]

\[ \text{det} = da_{\text{exp}} + datheo \]

\[ datheo = \text{relative error on distances} \]

8 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2]}{\text{sum} w} \]

\[ \text{datheo = absolute error on distances} \]

9 : \[ R = \frac{\text{sum}[w*(1/I_{\text{exp}} - 1/I_{\text{theo}})^2]}{\text{sum} [w*(1/I_{\text{exp}}^2)*1/3]} \]

10 : \[ R = \frac{\text{sum}[w*(I_{\text{exp}}-I_{\text{theo}})^2/da_{\text{exp}}]}{\text{sum} w} \]

\[ da_{\text{exp}} = (A_0 + C I_{\text{exp}})^{(1/3)} \]

\[ A_0 = \text{experimental noise level} \]

\[ C = \text{experimental relative error} \]

Four additional R factor definition are implemented for analysis

11 : \[ R = \text{sum} \left\{ \frac{\text{tm abs}(I_{\text{exp}}-I_{\text{theo}})}{\text{sum} \left[ \text{tm } I_{\text{exp}} \right]} \right\} \]

12 : \[ R = \text{sum} \left\{ \frac{\text{tm abs}(I_{\text{exp}}^{1/6} - I_{\text{theo}}^{1/6})}{\text{sum} \left[ \text{tm } I_{\text{exp}}^{1/6} \right]} \right\} \]

13 : \[ R = \text{sum} \left\{ \frac{\text{tm abs}(I_{\text{exp}} - I_{\text{theo}})}{\text{sum} \left[ \text{tm } I_{\text{exp}}^{1/6} \right]} \right\} \]

14 : \[ R = \text{sum} \left\{ \frac{\text{tm abs}(I_{\text{exp}}^{1/6} - I_{\text{theo}}^{1/6})}{\text{sum} \left[ \text{tm } I_{\text{exp}}^{1/6} + I_{\text{theo}}^{1/6} \right]} \right\} \]

NOE R factor/potential definition ? (6)

<cr>

*****************************

*** Data set number 1 ***

*****************************

Experimental noise level ? (1.000E+04)

Experimental relative error ? (5.000E-02)

NOE force constant ? (4.000E+01)

The following options are implemented:

1: analyse all peaks
2: analyse selected peaks

Your choice ? (1)
NOEtraj. file(0) or form. NOE file(1) ? (0)
1 <cr>
Name of the NOE file ? (cram.inp)
cram_anem.theno <cr>
Rescale NOE intensities (0:no,1:yes)? (0)
<cr>
ANAANOE can calculate uncorrected R-factors or correct them to account for the experimental errors defined as (noise + % \( \Delta \text{exp} \)); in the latter case the difference between theo. and exp. NOEs is set to 0 if it is within the experimental error otherwise \( \text{Atheo} = \text{Atheo} + \pm \text{exp. error} \).

Calculate uncorrected (0) or corrected (1) R-factors ? (0)
<cr>
Write exp + theo NOEs to file (Cricket)(1:y,0:n)? (0)
<cr>
Write NOE rest. en. matrix per residue (T/F)? (T)
f <cr>
Write residue NOE rest. en. (T/F)? (F)
t <cr>
Name of the file ? ()
cram_anem.rese noe <cr>
Title ? ()
Residue Enoe <cr>

The standard R-factor definitions are:

\[
\begin{align*}
\text{D} & = \text{ABS(\text{THE}-\text{EXP})} \\
\text{A} & = \text{ABS(\text{EXP})} \\
\text{S} & = 0.5*(\text{ABS(\text{EXP})+ABS(\text{THE})}) \\
\text{A}_6 & = \text{ABS(\text{EXP})}^{**(1./6.)} \\
\text{A}_6 & = \text{ABS(\text{THE})}^{**(1./6.)} \\
1: & \frac{\text{SUM(DX)}}{\text{SUM(AX)}} \\
2: & \frac{\text{SUM(TM*DX)}}{\text{SUM(TM*AX)}} \\
3: & \frac{\text{SUM(DX/AX)}}{\text{SUM(1)}} \\
4: & \frac{\text{SUM(TM*DX/AX)}}{\text{SUM(TM)}} \\
5: & \frac{\text{SUM((AX}_6^-\text{AT}_6)/\text{SUM(AX}_6}) \\
6: & \frac{\text{SUM(TM*(AX}_6^-\text{AT}_6)/\text{SUM(TM*AX}_6})} \\
7: & \frac{\text{SUM(DX)}}{\text{SUM(SX)}} \\
8: & \frac{\text{SUM(TM*DX)}}{\text{SUM(TM*SX)}} \\
9: & \frac{\text{SUM((AX}_6^-\text{AT}_6)/0.5*SUM(AX}_6+\text{AT}_6})} \\
10: & \frac{\text{SUM(TM*(AX}_6^-\text{AT}_6)/0.5*SUM(TM*(AX}_6+\text{AT}_6))} \\
11: & \left[\frac{\text{SUM(WW*DX**2)}}{\text{SUM(WW*AX**2))}}\right]^{*1/2} \\
\text{WW} & = \frac{1}{(\text{NOISE+AX})}
\end{align*}
\]

<table>
<thead>
<tr>
<th>R-factor</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.2624</td>
<td>0.2501</td>
<td>0.4993</td>
<td>0.3986</td>
<td>0.0626</td>
<td>0.0568</td>
<td>0.2712</td>
<td>0.2606</td>
<td>0.0630</td>
</tr>
<tr>
<td>0.0573</td>
<td>0.4430</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.2673</td>
<td>0.2562</td>
<td>0.5652</td>
<td>0.4276</td>
<td>0.0602</td>
<td>0.0547</td>
<td>0.2750</td>
<td>0.2657</td>
<td>0.0605</td>
</tr>
<tr>
<td>0.0550</td>
<td>0.4597</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.2531</td>
<td>0.2384</td>
<td>0.4334</td>
<td>0.3698</td>
<td>0.0653</td>
<td>0.0591</td>
<td>0.2640</td>
<td>0.2508</td>
<td>0.0658</td>
</tr>
<tr>
<td>0.0598</td>
<td>0.4094</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Write R-factors to a file ? : 
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice ? (1)
0 <cr>
Write NOE restraint energies per peak (1:y,0:n) ? (0)
1 <cr>
Name of the file ? ()
cram_anem.enoe <cr>
Title ? ()
Crambin peak NOE energies <cr>
    New peak selection and/or new file (T/F)? (T)
f <cr>

Thank you for your visit !!!
See you...

*** End program A N A N O E ***

Two files containing the NOE restrained energies per residue (cram_anem.resenoe) and per peak (cram_anem.enoe) were generated with ananoe. These files can be used to identify peaks or regions with high NOE restraint energies.

7.1.10 Analysing selected NOE peaks (selpks & ananoe)

% ananoe <cr>

*** Program A N A N O E *** Version 14–Oct–93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991–1994

********************************************************************
This program analyses an NOE trajectory in terms of R-factor

Name of the DINOSAUR input file ? ()
cram.inp <cr>

Now reading the NOE constraints file:
Crambin DINOSAUR
Number of protons : 315
Coordinates scaling factor (-->A) : 10.00000
NOTYP RFACYP GRADYP DXYZMAX NSTEP DISCUT
9 6 1 0.01000 1 4.500
Minimum H-H distance = 1.50 A
Flat well NOE potential: Vnoe = 0 if abs(Iexp-Itheo) < exp. error
Inclusion of the order parameters
Negative experimental NOEs will be skipped
Experimental NOEs will be symmetrized if possible

********************************************************************
*** Parameters for NOE data set # 1 ***
********************************************************************
H2O experimental NOEs from cram.noe, stereo from cram.stereo

<table>
<thead>
<tr>
<th>FREQ</th>
<th>TAUC</th>
<th>TMETH</th>
<th>ROSTARO</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.000E+08</td>
<td>2.000E-09</td>
<td>1.000E-10</td>
<td>0.80000</td>
</tr>
</tbody>
</table>

-1.000E+20 < scaling factor < 1.000E+20

Force constant : 4.000E+01

Cutoffs max and min for forces : 1.000E+01-1.000E+01

Experimental noise level : 1.000E+04

Experimental relative error: 5.0 %

Mixing times : 0.04000 0.08000 0.16000 0.25000

Number of NOE constraints : 367

***************************************

Dynamic stereospecific assignments for 156 peaks

Upper limit to switch off dynamic assignment : 1.000E+05

The following R factor definitions correspond to the DINOSAUR NOE potentials:

1 : R = sum[w*(Iexp-Ithe)**2] / sum [w*Iexp**2]
2 : R = sum[w*(Iexp-Ithe)**2] / sum [w*Ithe**2]
3 : R = sum[w*(Iexp-Ithe)**2 / Iexp**2] / sum w
4 : R = sum[w*(Iexp-Ithe)**2 / Ithe**2] / sum w
5 : R = sum[w*(Iexp-Ithe)**2 / 0.5*(Iexp + Ithe)**2] / sum w
6 : R = sum[w*(Iexp-Ithe)**2/ daexp] / sum w
   daexp = (A0 + C Iexp)**2
   A0 = experimental noise level
   C = experimental relative error

7 : R = sum[w*(Iexp-Ithe)**2/ det] / sum w
   det = daexp + datheo
   datheo = absolute error on distances

8 : R = sum[w*(Iexp-Ithe)**2/ det] / sum w
   datheo = relative error on distances

9 : R = sum[w*(1/Iexp**(1/6)-1/Ithe**(1/6))**2 / sum[w*(1/Iexp**(1/3))]

10 : R = sum[w*(Iexp**(1/6) - Ithe**(1/6))**2/ daexp] / sum w
    daexp = (A0 + C Iexp)**(1/3)
    A0 = experimental noise level
    C = experimental relative error

Four additional R factor definition are implemented for analysis

11 : R = sum[tm abs(Iexp-Ithe)] / sum [tm Iexp]
12 : R = sum[tm abs(Iexp**1/6-Itheo**1/6)] / sum[tm Iexp**1/6]
13 : R = sum[tm abs(Iexp-Itheo)] / sum[0.5 tm (Iexp+Itheo)]
14 : R = sum[tm abs(Iexp**1/6-Itheo**1/6)] / sum[0.5 tm (Iexp**1/6+Itheo**1/6)

NOE R factor/potential definition ? ( 6)

***************************

*** Data set number  1 ***
***************************

Experimental noise level ? ( 1.000E+04)

Experimental relative error ? ( 5.000E-02)

NOE force constant ? ( 4.000E+01)
The following options are implemented:
   1: analyse all peaks
   2: analyse selected peaks

Your choice ? ( 1)

2

You should give now the peak numbers which you want to analyse.
numbers read from file(0) or terminal(1)? ( 1)

0

File name ? (cram.inp)

selection

6 peak numbers read from file:
Crambin all NOEs involving 21 Thr
NOEtraj. file(0) or form. NOE file(1)? ( 0)

Name of the NOE trajectory file? (selection)
cram_em.noetraj

Number of records to be skipped at begin? ( 0)

Number of records left to be read? ( 2600)

Rescale NOE intensities (0:no,1:yes)? ( 0)

ANAANOE can calculate uncorrected R-factors or correct them to account for the experimental errors defined as (noise + % Aexp); in the latter case the difference between theo. and exp. NOEs is set to 0 if it is within the experimental error otherwise Atheo = Atheo + exp. error.

Calculate uncorrected (0) or corrected (1) R-factors? ( 0)

End of file detected during read, only 12 complete records for analysis

Continue with this reduced set (T/F)? (T)
t

Cluster size for averaging? ( 1)

Do you want to inspect some R-factors (T/F)? (T)

The standard R-factor definitions are:

\[
\begin{align*}
DX &= \text{ABS(} \text{THE-EXP}) \\
AX &= \text{ABS(} \text{EXP}) \\
SX &= 0.5(\text{ABS(} \text{EXP})+\text{ABS(} \text{THE})) \\
AX6 &= \text{ABS(} \text{EXP})^{1/6} \\
AT6 &= \text{ABS(} \text{THE})^{1/6}
\end{align*}
\]

1: \( \frac{\text{SUM(}DX)}{\text{SUM(}AX)} \)
2: \( \frac{\text{SUM(}TM*DX)}{\text{SUM(}TM*AX)} \)
3: \( \frac{\text{SUM(}DX/AX)}{\text{SUM(}1)} \)
4: \( \frac{\text{SUM(}TM*DX/AX)}{\text{SUM(}TM)} \)
5: \( \frac{\text{SUM(}AX6-AT6)}{\text{SUM(}AX6)} \)
6: \( \frac{\text{SUM(}TM*(}AX6-AT6)}{\text{SUM(}TM*AX6)} \)
7: \( \frac{\text{SUM(}DX)}{\text{SUM(}SX)} \)
8: \( \frac{\text{SUM(}TM*DX)}{\text{SUM(}TM*SX)} \)
9: \( \frac{\text{SUM(}AX6-AT6)}{0.5*\text{SUM}(AX6+AT6)} \)
10: \( \frac{\text{SUM(}TM*(}AX6-AT6)}{0.5*\text{SUM(}TM*(}AX6+AT6)} \)
11: \( \frac{\text{SUM(WW*DX**2)/SUM(WW*AX**2))**1/2}}{\text{WW} = 1/(\text{NOISE+AX})} \)

Give definition number: ( 2)

R-factors: all, intra or interresidue peaks(0,1,2)? ( 0)
(Entering plot subroutine SCURV)
Type: NWID,'FMA'
NWID = number of points on ordinate, default value 50
FMA = data format descriptor, default G11.4
(NWID + 2*FIELD-WIDTH) should not exceed 75 (terminal) or 128 (printer)
NWID should be < 107
Type / for use of default parameters

/ <cr>
0.3322 1.000
0.2966 I * I
0.2811 I * I
0.2735 I * I
0.2693 I * I
0.2660 I * I
0.2618 I * I
0.2601 I * I
0.2573 I* I
0.2557 * I
0.2550 11.00 * I
0.2550 12.00 * I
<cr>

Write R-factors to a file ? :
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)
Your choice ? (2)
0 <cr>

Do you want to inspect more results (T/F)? (T)

The standard R-factor definitions are:

\[
\begin{align*}
DX &= \text{ABS(THE-EXP)} \\
AX &= \text{ABS(EXP)} \\
SX &= 0.5\times(\text{ABS(EXP)}+\text{ABS(THE)}) \\
AX6 &= \text{ABS(EXP)}^{1/6.} \\
AT6 &= \text{ABS(THE)}^{1/6.}
\end{align*}
\]

1: \( \sum DX / \sum AX \)
2: \( \sum(TM\times DX) / \sum(TM\times AX) \)
3: \( \sum(DX/AX) / \sum(1) \)
4: \( \sum(TM\times DX/AX) / \sum(TM) \)
5: \( \sum((AX6-AT6) / \sum AX6) \)
6: \( \sum(TM\times(AX6-AT6)) / \sum(TM\times AX6) \)
7: \( \sum(DX) / \sum(SX) \)
8: \( \sum(TM\times DX) / \sum(TM\times SX) \)
9: \( \sum((AX6-AT6) / 0.5\times\sum(AX6+AT6)) \)
10: \( \sum(TM\times(AX6-AT6)) / 0.5\times\sum(TM\times(AX6+AT6)) \)
11: \( \left[ \sum(WW\times DX^2)/\sum(WW\times AX^2) \right]^{1/2} \)

WW = 1/(NOISE+AX)

Give definition number : (2)

10 <cr>
R-factors: all, intra or inter residue peaks(0,1,2)? (0)

/ <cr>
0.7841E-01 1.000
0.7027E-01 I * I
0.6762E-01 I * I
0.6602E-01 I * I
0.6472E-01 I * I
0.6385E-01 I * I
0.6308E-01 I * I
0.6258E-01 I * I
0.6194E-01 I * I
0.6153E-01 * I
0.6138E-01 11.00 * I
0.6138E-01 12.00 * I
Write R-factors to a file?:
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice? (0)

Do you want to inspect more results (T/F)? (T)

Write R-factors to a file?:
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice? (0)

Do you want to inspect some NOE energy functions (T/F)? (F)

Give selected peak number:

NOE restrained energy for peak 20GLY H 21THR H

15.83 1.000
13.11 I
11.38 I *
9.836 I *
8.593 I *
7.546 I *
6.713 I *
6.057 I *
5.692 I *
5.446 I *
5.394 11.00 *
5.394 12.00 *

Write NOE energies to a file?:
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice? (0)

Do you want to inspect more results (T/F)? (T)

Give selected peak number:

NOE restrained energy for peak 21THR H CA1 21THR HCG21

0.8162 1.000
0.7858 * I
0.7955 I * I
0.7997 I * I
0.7840 * I
0.8021 I * I
0.8226 I * I
0.8494 I * I
0.8670 I * I
0.8852 I *
0.8717 11.00 I *
0.8717 12.00 *

Write NOE energies to a file?:
0: none
1: curves
2: Cricket graph format
3: Plotrf format (HP plotter)

Your choice ? ( 0)
<cr>
Do you want to inspect more results (T/F)? (T)
f <cr>
New peak selection and/or new file (T/F)? (F)
f <cr>

Thank you for your visit !!!
See you...

*** End program A N A N O E ***

7.1.11 Analysing the results of the dynamic assignment procedure (chkass)

% chkass <cr>

*** Program C H K A S S *** Version 14-Oct-93 ***

Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

*****************************************************************
*                                                             *
* CHKASS analyses the results of the dynamic assignment *     *
* procedure implemented in DINOSAUR. The results are *       *
* read from a standard DINOSAUR input file.                   *
*                                                             *
*****************************************************************

Dir + file name for the DINOSAUR input file? ()
cram.inp <cr>

The results in the DINOSAUR input file correspond to the
sum over all calls to the stereoassignment routine
during the refinement. To calculate a probability
the program needs to know the total number of calls

Total number of calls for dynamic assignments = ( 2.654E+03)
2654 <cr>

Output file name for assignment results ()
cram.dynass <cr>

Title ? ()
Crambin dynamic assignment from EM, annealing, EM <cr>

Dynamic assignment results for data set # 1
*****************************************************************

Peak#  15:
  present assignment :  2THR H CB1  13PHE HCE11
  alternate ass. # 1 :  2THR H CB1  13PHE HCE21 P = 60.8%

Peak#  26:
  present assignment :  4CYS1 H CA1  4CYS1 H CB1
  alternate ass. # 1 :  4CYS1 H CA1  4CYS1 H CB2 P = 68.3%
Peak#    27:

present assignment :   4CYS1 H CA1   5PRO H CD1
alternate ass. # 1 :   4CYS1 H CA1   5PRO H CD2 P = 24.1%

Peak#    29:

present assignment :   4CYS1 H CB1  10ARG  H CD1
alternate ass. # 1 :   4CYS1 H CB1  10ARG  H CD2 P = 39.6%
alternate ass. # 2 :   4CYS1 H CB2  10ARG  H CD1 P = 6.3%
alternate ass. # 3 :   4CYS1 H CB2  10ARG  H CD2 P = 31.7%

....

Total number of peaks analysed = 156

*** End program C H K A S S ***

7.2 Ensemble-averaged DINOSAUR refinement example (crambin)

This section gives an example of ensemble-averaged NOE refinement for crambin (Bonvin et al., 1993e). The same files as previously used (section 7.1) are used therefore. Scripts examples and results can be found in the dino/examples/cram_ens directory.

7.2.1 Creating and ensemble coordinate file (genens)

An ensemble of eight structure is created by duplicating one crambin structure. The copies are placed on the edges of a cubic box of 100 Å length to make sure that the structures do not interact with each other during the refinement.

% genens <cr>

*** Program G E N E N S *** Version 06-Nov-92 ***

Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

GENENS generates a GROMOS coordinates file for use in the ensemble DINOSAUR refinement. Two option are implemented:
0: generate ensemble from one structure by duplicating and shifting this latter
1: generate ensemble by combining several different molecules and shifting them.

Your choice ? ( 0)
<cr>
How many copies of the same molecule ? ( 1)
8 <cr>
Input GROMOS coordinates file ? ()
../cram/cram.md <cr>
Output GROMOS coordinates file ? (cram.md)
cram_ens.md <cr>
Title output file ? (Crambin average DINOSAUR 1000 steps EM steepest dist + dihed.)
Crambin ensemble of 8 structures from cram.md (cubic box 100A)

Structure/copy # 1

X,Y,Z displacement ? (  5.000E+00   5.000E+00   5.000E+00)
<cr>

Note: GROMOS coordinates in nm!

Structure/copy # 2

X,Y,Z displacement ? (  5.000E+00   5.000E+00   5.000E+00)
-5.0  5.0  5.0 <cr>

Structure/copy # 3

X,Y,Z displacement ? ( -5.000E+00  5.000E+00  5.000E+00)
-5.0 -5.0  5.0 <cr>

Structure/copy # 4

X,Y,Z displacement ? ( -5.000E+00 -5.000E+00  5.000E+00)
 5.0 -5.0  5.0 <cr>

Structure/copy # 5

X,Y,Z displacement ? (  5.000E+00 -5.000E+00  5.000E+00)
 5.0  5.0 -5.0 <cr>

Structure/copy # 6

X,Y,Z displacement ? (  5.000E+00  5.000E+00 -5.000E+00)
-5.0  5.0 -5.0 <cr>

Structure/copy # 7

X,Y,Z displacement ? ( -5.000E+00  5.000E+00 -5.000E+00)
-5.0 -5.0 -5.0 <cr>

Structure/copy # 8

X,Y,Z displacement ? ( -5.000E+00 -5.000E+00 -5.000E+00)
 5.0 -5.0 -5.0 <cr>

*** End program G E N E N S ***

7.2.2 Creating the DINOSAUR input file

The only difference with a single structure DINOSAUR input file is the averaging type (ITYPAV). We can read the file created under section 7.1.3 and modify this parameter.

% dinoin <cr>

*** Program D I N O I N *** Version 14-Oct-93 ***

Laboratory of NMR spectroscopy

Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

******************************************************************************
This program creates or modifies the input file for the DINOSAUR refinement programs.

Options: 0=create DINOSAUR input 1=modify old (0)

1 <cr>

Dir + file name for the DINOSAUR input file? ()../cram/cram.inp <cr>
Title of the DINOSAUR input file? (Crambin DINOSAUR) <cr>
Symmetrical dimer (0=n, 1=y) ? (0)
Number of NOE data sets ? (1)

H coord. scaling factor (Angstrom) ? (1.00E+01) <cr>
A chemical shift list will be read (y=1, n=0) ? (1)
Order parameters S2 are included (y=1, n=0) ? (1)
S2 read from matrix (0:bin, 1:form) or file(2)? (2)

Nine different methods can be used to calculate the theoretical NOE intensities:
1: exact calculation via diagonalisation of the relaxation matrix
2: two spins approximation
3: simplified expansion of exp(-tm*R)
4: 2nd order approximation (expansion of exp)
5: Yip approximation
6: two spins approximation with correction for spin diffusion updated every N steps
7: simplified expansion with correction for spin diffusion updated every N steps
8: 2nd ord. exp. with correction for spin diffusion updated every N steps
9: same as 1 but with use of a spherical cutoff around the corresponding proton pair

NOE definition ? (9) <cr>
Distance cutoff [A] ? (4.500E+00) <cr>
Neighbours list updated every N steps N = ? (1) <cr>
Minimal H-H distance [A] ? (1.500E+00) <cr>

Here come the choice for the type of averaging. We assume that the various conformers are in slow exchange (see section 3.1.3.5) and use a distance averaging as <1/r^6>:

Time (-1), ensemble(+1) or no averaging (0) ? (0)
1 <cr>

Give now the type of averaging
-1: NOEs
  0 : none
  1 : averaging following <dis>
  3: averaging following <1/dis**3>
  6 : averaging following <1/dis**6>

Type of averaging (-1/0/1/3/6) ? (0) 6 <cr>

Which experimental data do you want to skip
-1 : negative peaks
  0 : none
1 : positive peaks

Exp. NOEs : (-1,0,1) ? (1)

Symmetrise exp. NOEs (0:n,1:y) ? (1)

Several NOE potential are implemented:
1 : \[ R = \sum [w*(I_{exp} - I_{the})^2] / \sum [w*I_{exp}^2] \]
2 : \[ R = \sum [w*(I_{exp} - I_{the})^2] / \sum [w*I_{the}^2] \]
3 : \[ R = \sum [w*(I_{exp} - I_{the})^2] / \sum w \]
4 : \[ R = \sum [w*(I_{exp} - I_{the})^2] / I_{the}^2 / \sum w \]
5 : \[ R = \sum [w*(I_{exp} - I_{the})^2] / 0.5*(I_{exp} + I_{the})^2 / \sum w \]
6 : \[ R = \sum [w*(I_{exp} - I_{the})^2/da_{exp}] / \sum w \]
   \[ da_{exp} = (A_0 + C*I_{exp})^2 \]
   \[ A_0 = \text{experimental noise level} \]
   \[ C = \text{experimental relative error} \]
7 : \[ R = \sum [w*(I_{exp} - I_{the})^2/ \text{det}] / \sum w \]
   \[ \text{det} = da_{exp} + datheo \]
   \[ datheo = \text{relative error on distances} \]
8 : \[ R = \sum [w*(I_{exp} - I_{the})^2 \text{det}] / \sum w \]
   \[ datheo = \text{absolute error on distances} \]
9 : \[ R = \sum [w*(l/I_{exp}^*(1/6) - l/I_{theo}^*(1/6))]^2 \]
10 : \[ R = \sum [w*(l/I_{exp}^*(1/6) - l/I_{theo}^*(1/6))]^2/da_{exp} / \sum w \]
   \[ da_{exp} = (A_0 + C*I_{exp})^{1/3} \]
   \[ A_0 = \text{experimental noise level} \]
   \[ C = \text{experimental relative error} \]

NOE potential ? (6)

Flat well potential (0:n,1:y) ? (1)

Two gradient type are implemented:
1 : \[ dA = -T_{mix}*dR \]
2 : \[ dA = -T_{mix}*dR + T_{mix}^2*(dR*R + R*dR)/2 \]

Gradient definition ? (1)

The NOE gradient will be updated if the proton displacements > dxyzmax

Maximum proton displacement in A DXYZMAX = 1.000E-02

Check these parameters (T) or continue (F) ? (T)

*****************************************************************

The following parameters and experimental data are defined as in section 7.1.3. The number of conformers in the ensemble will be passed from the GROMOS programs to the DINOSAUR routines and are defined in the GROMOS input files (see below).
7.2.3 Ensemble-averaged NOE restrained slow-cooling simulated annealing (dinomd)

This is basically the same protocol as described before (section 7.1.6) for a single structure. The only differences are the type of averaging in the DINOSAUR input file and the number of solute molecules in the GROMOS input files. We will first modify in the GROMOS input file created under section 7.1.6 the parameter defining the number of solute molecule (we do not need to modify other parameters in this file).

% dinomd <cr>
!

Creates or modify the MD input file (y/n) ? y <cr>

*** Program M K M D I N *** Version 07-Jan-93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

This program creates or modifies the GROMOS input file for restrained molecular dynamics.

Creates new file (0) or modify old one (1)? ( 0 ) 1 <cr>
File name ? ( ) .../cram/cram.anin <cr>

The following options are implemented :
-1 : quit 
 0 : write file 
 1 : check file formats 
 2 : check box 
 3 : check MD parameters 
 4 : check SHAKE parameters 
 5 : check non-bonded interactions parameters 
 6 : check cutoff parameters 
 7 : check print parameters 
 8 : check restraining parameters 
 9 : check H codes for NOE refinement

Your choice ? ( 0 ) 1 <cr>
number of solute molecules NPM ? ( 1 ) 8 <cr>
number of solvent molecules NSM ? ( 0 ) 0 <cr>

NTX = 0 : coordinates x are read from tape21 (uniform.)
  = 1 : coordinates x are read from tape21 (form.)
  = 2 : coordinates x are read from tape21 (uniform.)
  = 3 : x and f (predicted positions) are read from tape21 (uniform.)
  = 4 : x and v are read from tape21 (uniform.)
  = 5 : x and v are read from tape21 (form.)
  = 6 : x,v and box(1...3) are read from tape21 (uniform.)
  = 7 : x,v and box(1...3) are read from tape21 (form.)
  = 8 : x,v, box and xc are read from tape21 (uniform.)
  = 9 : x,v, box and xc are read from tape21 (form.)

NTX = ? ( 1 )
NTCX = 0 : no constraints are read from tape23 (NTC>1)
       = 1 : solute constraints are read from tape23 (NTC>1)
NTCX =                          ? (      0)
0 <cr>
NTXO = 0 : final coordinates are written to tape31 (unform.)
       = 1 : final coordinates are written to tape31 (form.)
NTXO =                          ? (      1)
1 <cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (      9)
0 <cr>
File name ? ()
cram_ens.anin <cr>
title     ? (Crambin DINOSAUR annealing 1000 --> 1K in 2.5 ps)
Crambin ensemble-average DINOSAUR annealing from 1000 to 1 K in 2.5 ps <cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (      0)
-1 <cr>

*** End program M K M D I N ***

Now give GROMOS specific files:

GROMOS input coordinates file name ? :
cram_ens.md <cr>
GROMOS output coordinates file name ? :
cram_ens.an <cr>
Standard output file name ? :
cram_ens.anout <cr>
GROMOS coord. trajectory file name ? :
<cr>
Molecular topology file name ? :
../../cram/cram.topo <cr>
GROMOS input file name ? :
cram_ens.anin <cr>
Distance constraints file name ? :
../../cram/cram.disre <cr>
Dihedral restraints  file name ? :
   ./.cram/cram.dihre <cr>
Position restraining  file name ? :
   <cr>

Now give DINOSAUR specific files:
 *********************************************
Protons codes  file name ? :
   ./.cram/cram.htyp <cr>
DINOSAUR input  file name ? :
   cram_ens.inp <cr>
Order parameters  file name ? :
   ./.cram/cram.s2 <cr>
Chemical shift  file name ? :
   ./.cram/cram.shift <cr>
NOE trajectory output file name ? :
   cram_ens_an.noetraj <cr>
Run with time-averaged restraints (y,n)?
   n <cr>
dinomd.job created !
execute or submit it

The modified input file (cram_ens.anin) (modified parameter with respect to the file created under section 7.1.6 are indicated in bold) for the ensemble-averaged NOE restrained slow-cooling simulated annealing looks like:

```
Crambin  ensemble-average  DINOSAUR  annealing  from  1000  to  1 K  in  2.5 ps

8   0   0   1   0   605891000.00000   0.00000   1
0   0.00000   0.00000   0.00000   90.00000
100   1   1.00000   10.00000   0.01000   0.10000
0   1   0.06102   0.0007476   0.50000
396
6   1   10000
25   1   3   0.00000   0.00100
3   0   0.00010
3   2   2   0   0
1   10   0.80000   10.00000   10.00000   0.20000   1.20000
25   0
25   0   25   0   25   0   2
3   19000.00000
32000.00000   0.10000   0.10900   0.15300
0   0   2   0.00000   0.05000
1   110.00000
3
15   16   0
12   29   0
0   0   0
13   14   32
0   0   0
```

and a new script file for running the slow-cooling annealing has been created (dinomd.job):

```
# ! NOEs RESTRAINT MOLECULAR DYNAMICS
chdir  /usr/ruuci5/albo/dino/examples/cram_ens
ln -s  cram_ens.md  fort.21
ln -s  cram_ens.an  fort.31
ln -s  cram_ens.traj  fort.12
ln -s  ../cram/cram.topo  fort.20
ln -s  ../cram/cram.disre  fort.26
ln -s  ../cram/cram.dihre  fort.28
ln -s  cram_ens.posrel  fort.24
ln -s  cram_ens.posre2  fort.25
ln -s  ../cram/cram.htyp  fort.89
ln -s  ../cram/cram.inp  fort.90
ln -s  ../cram/cram.s2  fort.92
```
A ensemble-averaged NOE restrained energy minimisation can be set up in the same way, the only difference in the GROMOS input file being the number of solute molecules.

### 7.2.4 Splitting an ensemble coordinate file (splitco)

After having refined structures as an ensemble it is necessary to split the ensemble in individual structures for analysis. This can be done with `splitco`.

```bash
% splitco <cr>
*** Program S P L I T C O *** Version 06-Nov-92 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

SPLITCO splits a GROMOS coordinates file containing several same solutes into single structure files.

Input GROMOS coordinates file ? ()
cram_ens.anem <cr>
How many copies of the same molecule ? (1)
8 <cr>

Structure # 1
Output GROMOS coordinates file ? (cram_ens.anem)
cram1.anem <cr>

Structure # 2
Output GROMOS coordinates file ? (cram1.anem)
cram2.anem <cr>

Structure # 3
Output GROMOS coordinates file ? (cram2.anem)
cram3.anem <cr>

Structure # 4
Output GROMOS coordinates file ? (cram3.anem)
cram4.anem <cr>

Structure # 5
```
7.2.5 Calculating R factors from an ensemble of structures (mkhco & rfacav)

In case of ensemble-averaged NOE refinement or simply to analyse a set of structures, \( R \) factors can be calculated from ensemble-averaged NOEs with rfacav. Rfacav read a proton trajectory/ensemble file created with mkhco. The type of averaging should be defined in the DINOSAUR input file. Various type of averaging are possible: NOE averaging (ITYPAV = -1) or distance averaging assuming slow (ITYPAV = 6) or fast exchange (ITYPAV = 3) between the various conformers.

First a proton ensemble coordinates file should be created with mkhco:

```bash
% mkhco <cr>
! GENERATION OF PROTON COORDINATES OR TRAJECTORY FILES
! FROM GROMOS FILES

Creates or modify the MKHCO input file (y/n) ? y <cr>

*** Program M K H C O I N *** Version 08–Jan–92 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991–1994

*******************************************************************************

This program creates or modifies the MKHCO input file for creation of all proton files for DINOSAUR.

Creates new file (0) or modify old one (1)? (   1)
0 <cr>

The following options are implemented:
-1 : quit
 0 : write file
 1 : check file formats
 2 : check H codes for NOE refinement

Your choice ? (   0)
1 <cr>
number of coordinates files to be read? (   1)
8 <cr>
```
Record are used in case of MD trajectories. Typically a trajectory file contains coordinates saved at fixed time intervals during the simulation. A record corresponds thus to a structure at a defined time during the MD simulation.

number of records to skip at begin ? ( 0)
0 <cr>

number of records per file ? ( 1)
1 <cr>

number of coordinates per record ? ( 0)
1188 <cr>

The following format are implemented:
-1: input coordinates and title are read from normal gromos coordinates files
0: input coordinates and title record are unformatted
1: input coordinates have been packed (see subr. pack), title record is unformatted
2: input coordinates are formatted (see subr. pack), title record is formatted
3: input coordinates are in standard formatted form

format input coordinates ? ( -1)
-1 <cr>

number of digits behind decimal ? ( 3)
<cr>

sequence number of the molecule to be analysed ? ( 1)
<cr>

number of solute atoms ? ( 0)
396 <cr>

box coordinates are read (no:0,yes:2) ? ( 0)
<cr>

(used is case of a MD trajectory in water with constant pressure: the box dimensions are also written to the trajectory file)

output H trajectory form(0) or unform(1)(IRMA/DINOSAUR) ? ( 1)
1 <cr>

The following options are implemented:
-1 : quit
0 : write file
1 : check file formats
2 : check H codes for NOE refinement

Your choice ? ( 1)
2 <cr>

These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated at every MD step for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.

number of atom codes to be read NIAT ? ( 3)
<cr>
one (planar) H ? ( 15 16 0 0)
<cr>
one (dihedral) H ? ( 12 29 0 0)
<cr>
two 180 degrees (dihedral) H ? ( 0 0 0 0)
<cr>
three or two 120 degrees (dihedral) H? ( 13 14 32 0)
<cr>
Give now the integer atom codes of existing protons in GROMOS whose names do not begin with an H
<cr>
codes for protons in GROMOS without H? ( 0 0 0 0 0)
<cr>
The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check H codes for NOE refinement
<cr>
Your choice? ( 2)
0 <cr>
File name? ()
cram_ens.hcoin <cr>
The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check H codes for NOE refinement
-1 <cr>
*** End program MKHCO ***

MKHCO input file name?
cram_ens.hcoin <cr>
MKHCO output file name?
cram_ens.hcoout <cr>
Molecular topology file name?
cram.topo <cr>
GROMOS H coord. output file name (form.)?
cram_ens.mdh <cr>
GROMOS H coord. output file name (bin.)?
cram_ens.hco <cr>
Number of input coord./traj. files?
8 <cr>
Coordinates/trajecory file number 1?
cram1.anem <cr>
Coordinates/trajecory file number 2?
cram2.anem <cr>
Coordinates/trajecory file number 3?
cram3.anem <cr>
Coordinates/trajecory file number 4?
cram4.anem <cr>
Coordinates/trajecory file number 5?
cram5.anem <cr>
Coordinates/trajecory file number 6?
cram6.anem <cr>
Coordinates/trajecory file number 7?
cram7.anem <cr>
Coordinates/trajecory file number 8?
cram8.anem <cr>
mkhco.job created!
execute or submit it

Two files have been created:
- the mkhco input file (cram_ens.hcoin):

8 0 1 1 1188 -1 3
1 396 0
3
15 16 0
- and the script file `mkhco.job`:

```bash
# ! CONVERSION OF GROMOS FILES TO PROTONS COORDINATES
chdir /usr/ruuci5/albo/dino/examples/cram_ens
ln -s ../cram/cram.topo fort.10
ln -s cram_ens.mdh fort.92
ln -s cram_ens.hco fort.91
ln -s cram1.anem fort.11
ln -s cram2.anem fort.12
ln -s cram3.anem fort.13
ln -s cram4.anem fort.14
ln -s cram5.anem fort.15
ln -s cram6.anem fort.16
ln -s cram7.anem fort.17
ln -s cram8.anem fort.18
/bin/time /usr/ruuci5/albo/dino/src/mkhco <cram_ens.hcoin) > & cram_ens.hcoout
/bin/rm fort.10
/bin/rm fort.11
/bin/rm fort.12
/bin/rm fort.13
/bin/rm fort.14
/bin/rm fort.15
/bin/rm fort.16
/bin/rm fort.17
/bin/rm fort.18
/bin/rm fort.92
/bin/rm fort.91
```

The proton ensemble coordinate file created by executing the `mkhco.job` script can be used in `rfacav` to calculate ensemble NOE intensities and $R$ factors:

```
% rfacav <cr>
! CALCULATION OF R-FACTORS FROM A PROTON TRAJECTORY

H coordinates (bin) file name ? : cram_ens.hco <cr>
2D DINOSAUR input file name ? : cram_ens.inp <cr>
R-factor output file name ? : cram_ens.rfac <cr>
Calculate NOE averages from proton trajectory (y,n)? n <cr>
Theo. NOEs output file name ? : cram_ens.theno <cr>
Distances output file name ? : cram_ens.dis <cr>
Order parameters file name ? : ../cram/cram.s2 <cr>
Chemical shift file name ? : ../cram/cram.shift <cr>
avrfac.job created !
execute or submit it
```

The `avrfac.job` script file has been created:

```
# ! CALCULATION OF R-FACTORS
chdir /usr/ruuci5/albo/dino/examples/cram_ens
ln -s cram_ens.hco fort.91
ln -s cram_ens.inp fort.90
```
7.3 Time-averaged DINOSAUR refinement example (crambin)

This section gives an example of time-averaged NOE refinement for crambin (Bonvin et al., 1993e). The same files as previously used (section 7.1) are used therefore. Scripts examples and results can be found in the `dino/examples/cram_tav` directory.

7.2.1 Creating the DINOSAUR input file (dinoin)

The only difference with a single structure DINOSAUR input file is the averaging type (ITYPAV) and the value of the exponential memory function (DEXPAV). We can read the file created under section 7.1.3 and modify these parameters.

```
% dinoin <cr>
*** Program D I N O I N *** Version 14-Oct-93 ***
Laboratory of NMR spectroscopy
Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991–1994
*************************************************
This program creates or modifies the input file for the DINOSAUR refinement programs.

Options: 0=create DINOSAUR input 1=modify old (0)
1 <cr>
Dir + file name for the DINOSAUR input file? ()
../cram/cram.inp <cr>
Title of the DINOSAUR input file ? (Crambin DINOSAUR)
<cr>
Symmetrical dimer (0=n,1=y) ? (0)
<cr>
Number of NOE data sets ? (1)
<cr>
H coord. scaling factor (--> Angstrom) ? (1.000E+01)
<cr>
A chemical shift list will be read (y=1,n=0) ? (1)
<cr>
Order parameters S2 are included (y=1,n=0) ? (1)
<cr>
S2 read from matrix (0:bin,1:form) or file(2)? (2)
<cr>
```
Nine different methods can be used to calculate the theoretical NOE intensities:
1: exact calculation via diagonalisation of the relaxation matrix
2: two spins approximation
3: simplified expansion of \( \exp(-tm*R) \)
4: 2nd order approximation (expansion of exp)
5: Yip approximation
6: two spins approximation with correction for spin diffusion updated every N steps
7: simplified expansion with correction for spin diffusion updated every N steps
8: 2nd order approximation with correction for spin diffusion updated every N steps
9: same as 1 but with use of a spherical cutoff around the corresponding proton pair

NOE definition ? (9)
Distance cutoff [A] ? (4.500E+00)
Neighbours list updated every N steps N = ? (1)
Minimal H-H distance [A] ? (1.500E+00)

Here come the choice for the type of averaging. We choose a NOE averaging:

Time (-1), ensemble(+1) or no averaging (0) ? (0)
-1

Give now the type of averaging
-1: NOEs
0: none
1: averaging following \(<\text{dis}>\)
3: averaging following \(<1/<\text{dis}^2>\)
6: averaging following \(<1/<\text{dis}^6>\)

Type of averaging (-1/0/1/3/6) ? (0)
-1

Time averaging uses an exponential memory function:

\[ \langle \text{NOE} \rangle = (1-\exp(-dt/tav))*\text{NOE}(t) + \exp(-dt/tav)*\text{NOE}(t-dt) \]
with \( dt = \) time step of MD simulation
\( tav = \) memory decay time

Time step of MD simulation [ps]? (1.000E-03)
0.001

Memory function decay time [ps]? (1.000E+00)
5.0
Exponential decay constant for time. aver. \( \exp(-dt/tav)= 9.998E-01 \)

Which experimental data do you want to skip
-1: negative peaks
0: none
1: positive peaks

Exp. NOEs: (-1,0,1) ? (-1)

Symmetrise exp. NOEs (0:n,1:y)? (1)

Several NOE potential are implemented:
1 : \( R = \sum[w*(I_{EOP} - I_{THE})^2] / \sum[w*I_{EOP}^2] \)
2 : \( R = \sum[w*(I_{EOP} - I_{THE})^2] / \sum[w*I_{THE}^2] \)
3 : \( R = \sum[w*(I_{EOP} - I_{THE})^2 / I_{EOP}^2] / \sum w \)
4 : \( R = \sum[w*(I_{EOP} - I_{THE})^2 / I_{THE}^2] / \sum w \)
5: \[ R = \sum w \frac{(I_{exp} - I_{the})^2}{0.5(I_{exp} + I_{the})^2} / \sum w \]

6: \[ R = \sum \frac{w(I_{exp} - I_{the})^2}{daexp} / \sum w \]
    \[ daexp = (A0 + C I_{exp})^2 \]
    \[ A0 = \text{experimental noise level} \]
    \[ C = \text{experimental relative error} \]

7: \[ R = \sum \frac{w(I_{exp} - I_{the})^2}{det} / \sum w \]
    \[ det = daexp + datheo \]
    \[ datheo = \text{relative error on distances} \]

8: \[ R = \sum \frac{w(I_{exp} - I_{the})^2}{det} / \sum w \]
    \[ datheo = \text{absolute error on distances} \]

9: \[ R = \sum \frac{w(I_{exp}^2 - 1/I_{the}^2)^2}{det} / \sum w \]
    \[ det = daexp + datheo \]
    \[ datheo = \text{relative error on distances} \]

10: \[ R = \sum \frac{w(I_{exp}^{1/6} - I_{the}^{1/6})^2}{daexp} / \sum w \]
    \[ daexp = (A0 + C I_{exp})^{1/3} \]
    \[ A0 = \text{experimental noise level} \]
    \[ C = \text{experimental relative error} \]

NOE potential \( ? (6) \)
Flat well potential \( (0:n,1:y)? (1) \)

Two gradient type are implemented :
1: \[ dA = -T_{mix}*dR \]
2: \[ dA = -T_{mix}*dR + T_{mix}^2*(dR*R + R*dR)/2 \]

Gradient definition \( ? (1) \)

The NOE gradient will be updated if the proton displacements \( > dxyzmax \)

Maximum proton displacement in A \( \text{DXYZMAX} = (1.000E-02) \)

Check these parameters (T) or continue (F) \( ? (T) \)

---

The following parameters and experimental data are defined as in section 7.1.3.

### 7.3.2 Starting a time-averaged NOE restrained MD (dinomd)

Initiating and continuing a time-averaged NOE restrained MD simulation are two different things. When initiating a MD run, initial velocities have to be randomly attributed to the atoms and the initial NOEs are set equal to the experimental data to avoid remaining trapped in the initial configuration. Then, continuing a run requires that the velocities and time-averaged NOE intensities of the previous run be used to avoid discontinuity in the simulation. This section describes the GROMOS input and the DINOSAUR script files for starting a time-averaged MD simulation at 300 K (first 10 ps). Examples of continuing a time-averaged MD run will be presented in the following section.

% dinomd <cr>
! NOEs RESTRRAINTE MOLECULAR DYNAMICS

Creates or modify the MD input file (y/n) ?
Program M K M D I N *** Version 07-Jan-93 ***

Laboratory of NMR spectroscopy

Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

*************************************************

This program creates or modifies the GROMOS input file for restrained molecular dynamics.

Creates new file (0) or modify old one (1)? (0)
1<cr>
File name ? ()
../cram/cram.anin <cr>
Crambin DINOSAUR annealing 1000 --> 1K in 2.5 ps

The following options are implemented:
-1: quit
  0: write file
  1: check file formats
  2: check box
  3: check MD parameters
  4: check SHAKE parameters
  5: check non-bonded interactions parameters
  6: check cutoff parameters
  7: check print parameters
  8: check restraining parameters
  9: check H codes for MOE refinement

Your choice ? (1)
1<cr>
number of solute molecules NPM ? (1)
<cr>
number of solvent molecules NSM ? (0)
<cr>
NTX = 0: coordinates x are read from tape21 (unform.)
  = 1: coordinates x are read from tape21 (form.)
  = 2: coordinates x are read from tape21 (unform.)
  = 3: x and f (predicted positions) are read from tape21 (unform.)
  = 4: x and v are read from tape21 (unform.)
  = 5: x and v are read from tape21 (form.)
  = 6: x, v and box(1...3) are read from tape21 (unform.)
  = 7: x, v and box(1...3) are read from tape21 (form.)
  = 8: x, v, box and xc are read from tape21 (unform.)
  = 9: x, v, box and xc are read from tape21 (form.)

NTX = ? (1)
<cr>
NTCX = 0: no constraints are read from tape23 (NTC>1)
  = 1: solute constraints are read from tape23 (NTC>1)

NTCX = ? (0)
<cr>
NTXO = 0: final coordinates are written to tape31 (unform.)
  = 1: final coordinates are written to tape31 (form.)

NTXO = ? (1)
<cr>

The following options are implemented:
-1: quit
  0: write file
  1: check file formats
  2: check box
  3: check MD parameters
4: check SHAKE parameters
5: check non-bonded interactions parameters
6: check cutoff parameters
7: check print parameters
8: check restraining parameters
9: check H codes for NOE refinement

Your choice? (1)

2 <cr>

NTB < 0 : periodicity is applied
   box is truncated octahedron (BETA=90)
   = 0 : no periodicity is applied
   > 0 : periodicity is applied
   box is rectangular or monoclinic depending on BETA
   if abs(NTB)=2 the virial is calculated (BETA=90)

   ? (0)

The following options are implemented:
-1: quit
0: write file
1: check file formats
2: check box
3: check MD parameters
4: check SHAKE parameters
5: check non-bonded interactions parameters
6: check cutoff parameters
7: check print parameters
8: check restraining parameters
9: check H codes for NOE refinement

Your choice? (2)

3 <cr>

Initial temperature TEMPI ? (1.000E+03)

300.0 <cr>

Random number generator seed(TEMPI>1.e-6)? (60589)

<cr>

HEAT : if > 1.e-6 all solute and solvent velocities are multiplied by HEAT
   if <-1.e-6 all solute velocities are multiplied by abs(HEAT), solvent velocities not changed

   ? (0.000E+00)

<cr>

NTT = 0 : classical MD
   = 1 : MD with velocity scaling (constant temperature)
   = 2 : MD with velocity scaling (constant temperature)
   separate scaling for solute and solvent

   ? (1)

<cr>

Reference temperature TEMP0 ? (1.000E+00)

300.0 <cr>

T deviat. for rescaling of the velocities? (1.000E+01)

<cr>

Temperature relaxation time TAUTP ? (1.000E-02)

<cr>

NTP = 0 : classical MD
   = 1 : MD with isotropic position scaling
   (constant pressure, abs(NTB)) = 2
   = 2 : MD with anisotropic diagonal (X-,Y-,Z-) position scaling (constant pressure, abs(NTB)) = 2

   ? (0)

<cr>

Number of submolecules forming 1 solute mol.? (1)

<cr>

Numbers of last atom of the submolecules? (396)

<cr>

NDFMIN : number of degrees of freedom that will be subtracted from the total number of degrees of freedom
NDFMIN =                        ? (   6)
<cr>
NTCM = 0 : translational motion of and rotational motion 
   about the centre of mass is not removed from the 
   initial velocities (INIT=4)
   = 1 : if is removed and NTCM is set equal to 0 (INIT<4)
NTCM =                          ? (   1)
<cr>
Remove this motion (again) after NSCM steps ? ( 10000)
<cr>
Two protocols are implemented:
- if NRUN (number of MD run) = 1 : standard MD
- if NRUN > 1 : annealing procedure; the temperature 
   is then varied in NRUN runs from the initial value 
   given by TEMPI to the final value given by TEMP0.
   This works for both cooling and heating.

Number of MD runs    NRUN ? (   100)
1<cr>
Number of steps     NSTLIM ? (    25)
10000<cr>
INIT induces different starting procedures
   = 1 : X(t) and V(t-dt/2) will be shaken
   initial centre of mass motion can be stopped
   = 2 : as for INIT=1, except X(t) is not shaken
   = 3 : as for INIT=2, except V(t-dt/2) is not shaken
   options 1-3 are meant for initialising a MD run
   = 4 : as for INIT=4, except centre of mass motion is not 
   changed; INIT=4 is required to continue a MD 
   simulation without a discontinuity in the trajectory
INIT =                          ? (   1)
<cr>
NTU = 1 : energies in KCAL/MOLE
   = 2 : energies in KJ/MOLE; coordinates are multiplied 
   by 0.1 and velocities by sqrt(4.184)
   = 3 : energies in KJ/MOLE
NTU =                           ? (   3)
<cr>
Initial time [ps]                     ? (  0.000E+00)
<cr>
Time step [ps]                         ? (  1.000E-03)
<cr>

The following options are implemented :
  -1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (   3)
4<cr>
NTC = 1 : SHAKE is not performed, NITER=1
   = 2 : constraint atom pairs are taken from IBH-JBH
      if NTCC=0
   = 3 : in addition, constraint atom pairs are taken 
      from IB-JB if NTCC=0
NTC =                          ? (   3)
<cr>
NTCC = 0 : constraint lengths are taken from B0 using 
   IB JBH ICBH IB JB and ICB
   = 1 : constraint LENGTHS are taken sequentially 
   from CONSTR using ICO AND JCO
NTCC =                                   ? (      0)
relative tolerance for coord. resetting ? (  1.000E-04)

The following options are implemented :
-1 : quit
 0 : write file
 1 : check file formats
 2 : check box
 3 : check MD parameters
 4 : check SHAKE parameters
 5 : check non-bonded interactions parameters
 6 : check cutoff parameters
 7 : check print parameters
 8 : check restraining parameters
 9 : check H codes for MOE refinement

Your choice ? (      4)
5 <cr>
NTF = 1 : complete interaction is calculated
= 2 : bond interactions involving h-atoms are omitted
= 3 : in addition bond interactions involving no h-atoms
  are omitted
= 4 : in addition bond-angle interactions involving
  h-atoms c are omitted
= 5 : in addition bond-angle interactions involving no
  h-atoms are omitted
= 6 : in addition dihedral interactions involving h-atoms
  are omitted
= 7 : in addition dihedral interactions involving no
  h-atoms are omitted
= 8 : in addition non-bonded interactions are omitted
= 12: only bond interactions involving h-atoms are
  calculated
= 13: in addition bond interactions involving no h-atoms
  are calculated
= 14 15 16 17 : analogue to 4 5 6 7
NTF =                                   ? (      3)
<cr>
NTID = 0 : improper dihedral interaction is skipped
= 1 : improper dihedral interactions involving no
  h-atoms are calculated
= 2 : in addition, improper dihedral interactions
  involving h-atoms are calculated
NTID =                                   ? (      2)
<cr>
NTN = 1 : non-bonded interaction is calculated using an atom
  pair-list see subrs. nbpal and nonbal
= 2 : non-bonded interaction is calculated using a
  molecular pair list
= 3 : non-bonded interaction is calculated using a grid
  see subr. nonbb
= 4 : non-bonded interaction is calculated directly by
  scanning all pairs see subr. nonbp
NTN =                                   ? (      2)
<cr>
NRE(1...2..) = atom sequence numbers of the last atoms of each
  group (.ie. NRP) NRE(3) is set equal to the last
  solute atom and NRE(4) to the last atom
NRE(1...4) =                             ? (      0       0       0       0)
<cr>

The following options are implemented :
-1 : quit
 0 : write file
 1 : check file formats
 2 : check box
 3 : check MD parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? ( 5)

6 <cr>

NTNB = 0 : no non-bonded pair list is made in the first step
   = 1 : a non-bonded pair list is made in the first step
      (if NTN=1 or 2)
NTNB = ? ( 1)
<cr>
NSNB : after NSNB steps the non-bonded pair list will
      be updated (if NTN=1 or 2)
NSNB = ? ( 10)
<cr>
cut-off radius (for select. pairs) RCUTP ? ( 8.00E-01)
<cr>
switching radius (---> S(R)=1) RSWI2 ? ( 1.00E+01)
<cr>
cut-off radius(---> S(R)=0)(-->RSWI2)RCUT2 ? ( 1.00E+01)
<cr>
min. length of an edge of a grid cell CELM? ( 2.000E-01)
<cr>
RCUTL = radius within which all neighbour grid-cells are
      searched for charge groups or solvent molecules
      must be smaller than BOX(M)-CEL(M)
      or (BOX(M)-CEL(M))*SQRT(3)/2
RCUTL = ? ( 1.200E+00)
<cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? ( 6)

7 <cr>

print data every NTPR steps      NTPR = ? ( 25)
100 <cr>

Note: this parameter also controls the output frequency of the time-averaged NOE intensities
written to the NOE trajectory file. Do not choose a too small value to avoid generating huge
NOE trajectories in case of large experimental data sets.

monitor dihedral transitions (1=yes 0=no)? ( 0)
<cr>
write atomic coordinates (1=yes 0=no) ? ( 0)
1 <cr>
every NTWX steps             NTWX = ? ( 25)
-50 <cr>

Coordinates will be written every 50 steps. The - sign indicates that only solute coordinates will
be written. This has no use in this case since no solvent is included.

as long as the step number<NTWXM+1  NTWXM = ? ( 0)
100000 <cr>
write atomic velocities (1=yes 0=no)  ? (  1)
0 <cr>
write various energies   (1=yes 0=no)  ? (  0)
0 <cr>
write data unformatted(0) or formatted(2)? (  2)
<cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (  7)
8 <cr>
position restraining (0=no 1(or2)=yes)  NTR ? (  0)
<cr>
NTDR = 0 : no restraining
       = 1 or 2 : standard distance restraining
       = 3 : distance and NOE restraining
       = 4 : NOE restraining
NTDR = ? (  3)
<cr>
dis. restr. force constant  CDIS ? ( 2.000E+03)
<cr>
which part of the potential is harmonic  ? ( 1.000E-01)
<cr>
carbon hydrogen distance  DISH ? ( 1.090E-01)
<cr>
CH1-CH3 distance  DISC ? ( 1.530E-01)
<cr>
dihedral restraining (0=no 1=yes)  NTDLR ? (  1)
<cr>
dih. restr. force constant  CDLR ? ( 1.100E+02)
<cr>
a perturbation potential is applied (0=no,1=yes)? (  0)
<cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement

Your choice ? (  8)
9 <cr>

These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated at every MD step for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.
number of atom codes to be read          NIAT ? ( 3)
<cr>
one (planar) H                     ? ( 15 16 0 0)
<cr>
one (dihedral) H                  ? ( 12 29 0 0)
<cr>
two 180 degrees (dihedral) H        ? ( 0 0 0 0)
<cr>
three or two 120 degrees (dihedral) H? ( 13 14 32 0)
<cr>
Give now the integer atom codes of existing protons in GROMOS whose names do not begin with an H codes for protons in GROMOS without H ? ( 0 0 0 0)
<cr>
The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement
<cr>
Your choice ? ( 9)
0<cr>
File name ? (cram.anin)
cram_tav.mdlin <cr>
title    ? (Crambin DINOSAUR annealing 1000 --> 1K in 2.5 ps)
Crambin time-averaged DINOSAUR tau=5ps 10ps MD <cr>
The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check box
  3 : check MD parameters
  4 : check SHAKE parameters
  5 : check non-bonded interactions parameters
  6 : check cutoff parameters
  7 : check print parameters
  8 : check restraining parameters
  9 : check H codes for NOE refinement
<cr>
Your choice ? ( 0)
-1 <cr>
*** End program M K M D I N ***
<cr>
Now give GROMOS specific files:
**********************************************************
GROMOS input coordinates file name ? :
../cram/cram.md <cr>
GROMOS output coordinates file name ? :
cram_tav.mdl <cr>
Standard output file name ? :
cram_tav.mdlout <cr>
GROMOS coord. trajectory file name ? :
cram_tav.traj1 <cr>
Molecular topology file name ? :
../cram/cram.topo <cr>
GROMOS input file name ?:
cram_tav.mdlin
Distance constraints file name ?:
../cram/cram.disre
Dihedral restraints file name ?:
../cram/cram.dihre
Position restraining file name ?:

Now give DINOSAUR specific files:
******************************************************************************
Protons codes file name ?:
../cram/cram.htyp
DINOSAUR input file name ?:
cram_tav.inp
Order parameters file name ?:
../cram/cram.s2
Chemical shift file name ?:
../cram/cram.shift
NOE trajectory output file name ?:
cram_tav.noetraj1
Run with time-averaged restraints (y,n)?
y
NOE (1) or distance (2) averaging ?
1
Time-averaged NOEs/distances input file name (only needed for proper continuation= averages from previous run), not required at start)?:

Time-averaged NOEs/distances output file name (needed for proper continuation of next run)?:
cram_tav.noeavl

Two files have been created at this point:
- the GROMOS input file for the restrained MD simulation (cram_tav.mdlin):

<table>
<thead>
<tr>
<th>Crambin time-averaged DINOsaur tau=5ps 10ps MD</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 1 0 60589 300.00000 0.00000 1</td>
<td></td>
</tr>
<tr>
<td>0 0.00000 0.00000 0.00000 90.00000</td>
<td></td>
</tr>
<tr>
<td>1 1 300.00000 10.00000 0.01000 0.10000</td>
<td></td>
</tr>
<tr>
<td>0 1 0.06102 0.0007476 0.50000</td>
<td></td>
</tr>
<tr>
<td>396</td>
<td></td>
</tr>
<tr>
<td>6 1 10000</td>
<td></td>
</tr>
<tr>
<td>10000 1 3 0.00000 0.00100</td>
<td></td>
</tr>
<tr>
<td>3 0 0.00010</td>
<td></td>
</tr>
<tr>
<td>3 2 2 0 0</td>
<td></td>
</tr>
<tr>
<td>1 10 0.80000 10.00000 10.00000 0.20000 1.20000</td>
<td></td>
</tr>
<tr>
<td>100 0</td>
<td></td>
</tr>
<tr>
<td>-50 100000 25 0 25 0 2</td>
<td></td>
</tr>
<tr>
<td>0 19000.00000</td>
<td></td>
</tr>
<tr>
<td>32000.00000 0.10000 0.10900 0.15300</td>
<td></td>
</tr>
<tr>
<td>0 0 2 0.00000 0.05000</td>
<td></td>
</tr>
<tr>
<td>1 110.00000</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>15 16 0</td>
<td></td>
</tr>
<tr>
<td>12 29 0</td>
<td></td>
</tr>
<tr>
<td>0 0 0</td>
<td></td>
</tr>
<tr>
<td>13 14 32</td>
<td></td>
</tr>
<tr>
<td>0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

- and the DINOSAUR script to run the MD job (dinomd_tav_start.job):

```
# ! NOEs RESTRAINT MOLECULAR DYNAMICS
chdir /usr/ruucis5/albo/dino/examples/cram_tav
```
7.3.3 Continuing a time-averaged NOE restrained MD (dinomd)

Continuing a time-averaged NOE restrained MD simulation requires that the velocities and time-averaged NOE intensities of the previous run be used to avoid discontinuity in the simulation. Therefore a few parameters in the GROMOS input file have to be modified.

% dinomd <cr>
! NOEs RESTRAINT MOLECULAR DYNAMICS

Creates or modify the MD input file (y/n) ?
y <cr>

Laboratory of NMR spectroscopy

Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

**********************************************************************

This program creates or modifies the GROMOS input file for restrained molecular dynamics.

Creates new file (0) or modify old one (1)? { 0 }
1 <cr>
File name ? ()
cram_tav.mdlin <cr>
Crambin time-averaged DINOSAUR tau=5ps 10ps MD

The following options are implemented :
-1 : quit
  0 : write file
1 : check file formats
2 : check box
3 : check MD parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? (  1)
1 <cr>

number of solute molecules NPM ? (  1)
<cr>

number of solvent molecules NSM ? (  0)
<cr>

NTX = 0 : coordinates x are read from tape21 (unform.)
   = 1 : coordinates x are read from tape21 (form.)
   = 2 : coordinates x are read from tape21 (unform.)
   = 3 : x and f (predicted positions) are read from tape21 (unform.)
   = 4 : x and v are read from tape21 (unform.)
   = 5 : x and v are read from tape21 (form.)
   = 6 : x,v and box(1...3) are read from tape21 (unform.)
   = 7 : x,v and box(1...3) are read from tape21 (form.)
   = 8 : x,v, box and xc are read from tape21 (unform.)
   = 9 : x,v, box and xc are read from tape21 (form.)

NTX = {  1}
5 <cr>

Coordinates and velocities (and box if water run) should be read!

NTCX = 0 : no constraints are read from tape23 (NTC>1)
   = 1 : solute constraints are read from tape23 (NTC>1)

NTCX = {  0}
<cr>

NTXO = 0 : final coordinates are written to tape31 (unform.)
   = 1 : final coordinates are written to tape31 (form.)

NTXO = {  1}
<cr>

The following options are implemented :
-1 : quit
0 : write file
1 : check file formats
2 : check box
3 : check MD parameters
4 : check SHAKE parameters
5 : check non-bonded interactions parameters
6 : check cutoff parameters
7 : check print parameters
8 : check restraining parameters
9 : check H codes for NOE refinement

Your choice ? (  1)
3 <cr>

Initial temperature TEMPI ? (  3.000E+02)
0.0 <cr>

The initial temperature must be equal to 0 to avoid giving new random velocities to the atoms.
HEAT = ? ( 0.000E+00)

NTT = 0 : classical MD
   = 1 : MD with velocity scaling (constant temperature)
   = 2 : MD with velocity scaling (constant temperature)
         separate scaling for solute and solvent

NTT = ? ( 1)

Reference temperature TEMP0 ? ( 3.000E+02)

T deviat. for rescaling of the velocities? ( 1.000E+01)

Temperature relaxation time TAU TP ? ( 1.000E-02)

NTP = 0 : classical MD
   = 1 : MD with isotropic position scaling
         (constant pressure, abs(NTB)) = 2
   = 2 : MD with anisotropic diagonal (X-,Y-,Z-) position
         scaling (constant pressure, abs(NTB)) = 2

NTP = ? ( 0)

Number of submolecules forming 1 solute mol.? ( 1)

Numbers of last atom of the submolecules ? ( 396)

NDFMIN : number of degrees of freedom that will be
         subtracted from the total number of degrees
         of freedom

NDFMIN = ? ( 6)

NTCM = 0 : translational motion of and rotational motion
         about the centre of mass is not removed from the
         initial velocities (INIT=4)
   = 1 : if is removed and NTCM is set equal to 0 (INIT<4)

NTCM = ? ( 1)

Translational rotational motions about the centre of mass are not removed.

Remove this motion (again) after NSCM steps ? ( 10000)

Two protocols are implemented:
- if NRUN (number of MD run) = 1 : standard MD
- if NRUN > 1 : annealing procedure; the temperature
   is then varied in NRUN runs from the initial value
   given by TEMPI to the final value given by TEMP0.
   This works for both cooling and heating.

Number of MD runs NRUN ? ( 1)

Number of steps NSTLIM ? ( 10000)

INIT induces different starting procedures
   = 1 : X(t) and V(T-dt/2) will be shaken
   initial centre of mass motion can be stopped
   = 2 : as for INIT=1, except X(t) is not shaken
   = 3 : as for INIT=2, except V(t-dt/2) is not shaken
   options 1-3 are meant for initialising a MD run
   = 4 : as for INIT=4, except centre of mass motion is not
   changed; INIT=4 is required to continue a MD
   simulation without a discontinuity in the trajectory

INIT = ? ( 1)

Must be 4 for proper continuation!

NTU = 1 : energies in KCAL/MOLE
   = 2 : energies in KJ/MOLE; coordinates are multiplied
The following options are implemented:

- quit file (0)
- write file (1)
- check file formats (2)
- check box (3)
- check MD parameters (4)
- check SHAKE parameters (5)
- check non-bonded interactions parameters (6)
- check cutoff parameters (7)
- check print parameters (8)
- check restraining parameters (9)
- check H codes for NOE refinement (10)

Your choice? (3)

File name? (cram_tav.md1in)
cram_tav.md2in

title? (Crambin time-averaged DINOSAUR tau=5ps 10-20ps)
Crambin time-averaged DINOSAUR tau=5ps 10-20ps

Your choice? (3)

- quit file (0)
- write file (1)
- check file formats (2)
- check box (3)
- check MD parameters (4)
- check SHAKE parameters (5)
- check non-bonded interactions parameters (6)
- check cutoff parameters (7)
- check print parameters (8)
- check restraining parameters (9)
- check H codes for NOE refinement (10)

*** End program M K M D I N ***

Now give GROMOS specific files:

******************************
GROMOS input coordinates file name?: cram_tav.md1
GROMOS output coordinates file name?: cram_tav.md2
GROMOS standard output file name?: cram_tav.md2out
GROMOS coord. trajectory file name?: cram_tav.traj2
Molecular topology file name?: ../cram/cram.topo
GROMOS input file name?: cram_tav.md2in
Distance constraints file name?: ../cram/cram.disre
Dihedral restraints file name?: ../cram/cram.dihre
Position restraining file name?:
Now give DINOSAUR specific files:
*********************************
Protons codes    file name ? :
   ./cram/cram.htyp  <cr>
DINOSAUR input   file name ? :
cram_tav.inp   <cr>
Order parameters file name ? :
   ./cram/cram.s2  <cr>
Chemical shift   file name ? :
   ./cram/cram.shift  <cr>
NOE trajectory output file name ? :
cram_tav.noetraj2  <cr>
Run with time-averaged restraints (y,n)?
y <cr>
NOE (1) or distance (2) averaging ?
1 <cr>
Time-averaged NOEs/distances input file name (only needed for proper
continuation=averages from previous run), not required at start) ?
cram_tav.noeav1 <cr>
Time-averaged NOEs/distances output file name (needed for proper
continuation of next run) ?
cram_tav.noeav2 <cr>
dinomd.job created ! (RENAME TO dinomd_tav_cont.job)
execute or submit it

Again two files have been created at this point:
- the GROMOS input file for continuing the restrained MD simulation (cram_tav.md2in) (the
differences with the input file for initiating a MD run in section 7.3.2 are indicated in bold):

Crambin  time-averaged DINOSAUR tau=5ps 10-20ps

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>60589 0.00000 0.00000 0.00000 1</td>
</tr>
<tr>
<td>0</td>
<td>0.00000 0.00000 0.00000 90.00000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>300.00000 10.00000 0.01000 0.10000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0.06102 0.0007476 0.50000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>396</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>10000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10000</td>
<td>4</td>
<td>3</td>
<td>10.00000 0.00100</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0.00010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>0.80000 10.00000 10.00000 0.20000 1.20000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-50</td>
<td>100000</td>
<td>25</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>0</td>
<td>19000.00000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32000.00000</td>
<td>0.10000</td>
<td>0.10000</td>
<td>0.15300</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0.00000</td>
<td>0.05000</td>
</tr>
<tr>
<td>110.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>16</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>29</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>14</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- and the DINOSAUR script to run the MD job (dinomd_tav_cont.job):

# ! NOEs RESTRAINT MOLECULAR DYNAMICS
chdir /usr/ruucgi5/albo/dino/examples/cram_tav
/bin/cp cram_tav.noeav1 FLnoeav
ln -s cram_tav.md1 fort.21
ln -s cram_tav.md2 fort.31
ln -s cram_tav.traj2 fort.12
ln -s ./cram/cram.topo fort.20
ln -s ./cram/cram.dihre fort.26
ln -s ./cram/cram.dihre fort.28
7.3.4 Calculating instantaneous and "true time-averaged" NOEs and $R$ factors from time-averaged MD trajectories (mkhco & rfacav)

During a time-averaged NOE restrained MD simulation averaged NOE intensities with exponential memory (Eq. 33, section 3.1.5.3) are written to a NOE trajectory file. We can however analyse the MD trajectories by calculating true NOE averages (Eq. 31, section 3.1.5.3) with the procedure rfacav, but, first, a proton trajectory file (.hco) should be generated from the MD trajectory files with mkhco.

```
% mkhco <cr>
! GENERATION OF PROTON COORDINATES OR TRAJECTORY FILES
! FROM GROMOS FILES

Creates or modify the MKHCO input file (y/n) ?
y <cr>

*** Program M K H C O I N *** Version 08-Jan-92 ***

Laboratory of NMR spectroscopy

Alexandre Bonvin
Utrecht University, The Netherlands
(c) DINOSAUR Software 1991-1994

***************************************************

This program creates or modifies the MKHCO input file for creation of all proton files for DINOSAUR.

Creates new file (0) or modify old one (1)? (    1)
0 <cr>

The following options are implemented :
-1 : quit
  0 : write file
  1 : check file formats
  2 : check H codes for NOE refinement

Your choice ? (    0)
1 <cr>
```
number of coordinates files to be read? (  1)
2 <cr>
number of records to skip at begin ? (  0)

Record are used in case of MD trajectories. Typically a trajectory file contains coordinates saved at fixed time intervals during the simulation. A record corresponds thus to a structure at a defined time during the MD simulation. In this example, coordinates have been written every 50 steps (0.05 ps), this corresponds to 200 records for 10 ps MD simulation

<cr>
number of records per file ? (  1)
200 <cr>
from every N records read, only the 1st one is used, N= (  1)
1 <cr>
number of coordinates per record ? (  0)
1188 <cr>
The following format are implemented:
-1: input coordinates and title are read from normal gromos coordinates files
  0: input coordinates and title record are unformatted
  1: input coordinates have been packed
     (see subr. pack), title record is unformatted
  2: input coordinates are formatted
     (see subr. pack), title record is formatted
  3: input coordinates are in standard formatted form

<cr> format input coordinates ? ( -1)
2 <cr>
number of digits behind decimal ? (  3)
3 <cr>
sequence number of the molecule to be analysed ? (  1)
1 <cr>
number of solute atoms ? (  0)
396 <cr>
box coordinates are read (no:0,yes:2) ? (  0)
0 <cr>

(used is case of a MD trajectory in water with constant pressure: the box dimensions are also written to the trajectory file)

output H trajectory form(0) or unform(1)(IRMA/DINOSAUR) ? (  1)
<cr>

The following options are implemented:
-1 : quit
  0 : write file
  1 : check file formats
  2 : check H codes for NOE refinement

Your choice ? (  1)
2 <cr>

These are the codes of the GROMOS atoms to which apolar (non-existing protons in GROMOS) are attached. Protons will be generated at every MD step for these atoms. The atom codes can be found in the topology file or in the GROMOS residue libraries used to generate the molecular topology file (e.g. rt37d.dat and rt37c.dat). The default codes below are those found in the rt37d.dat file for proteins and DNA.

<cr> number of atom codes to be read NIAT ? (  3)
one (planar) H ? (  15  16  0  0)
<cr> one (dihedral) H ? (  12  29  0  0)
<cr> two 180 degrees (dihedral) H ? (  0  0  0  0)
three or two 120 degrees (dihedral) H? ( 13 14 32 0)

Give now the integer atom codes of existing protons in GROMOS whose names do not begin with an H codes for protons in GROMOS without H? ( 0 0 0 0)

The following options are implemented:
-1: quit
0: write file
1: check file formats
2: check H codes for NOE refinement

Your choice? (2)
0 <cr>
File name? ()
cram_tav.hcoin <cr>

The following options are implemented:
-1: quit
0: write file
1: check file formats
2: check H codes for NOE refinement

Your choice? (0)
-1 <cr>

*** End program MKHCO ***

MKHCO input file name?
cram_tav.hcoin <cr>
MKHCO output file name?
cram_tac.hcoout <cr>
Molecular topology file name?
../cram/tav.topo <cr>
GROMOS H coord. output file name (form.)?
cram_tav.mdh <cr>
GROMOS H coord. output file name (bin.)?
cram_tav.hco <cr>
Number of input coord./traj. files?
2 <cr>
Coordinates/trajectory file number 1?
cram_tav.traj1 <cr>
Coordinates/trajectory file number 2?
cram_tav.traj2 <cr>
mkhco.job created!
execute or submit it

Two files have been created:
- the mkhco input file (cram_tav.hcoin):

- and the script file mkhco.job:
# ! CONVERSION OF GROMOS FILES TO PROTONS COORDINATES
chdir /usr/ruuci5/albo/dino/examples/cram_tav
ln -s ..../cram/cram.topo fort.10
ln -s cramb_tav.mdh fort.92
ln -s cramb_tav.hco fort.91
ln -s cramb_tav.traj1 fort.11
ln -s cramb_tav.traj2 fort.12
(/bin/time /usr/ruuci5/albo/dino/mkhco <cram_tav.hcoin) > & cramb_tac.hcoout
/bin/rm fort.10
/bin/rm fort.11
/bin/rm fort.12
/bin/rm fort.92
/bin/rm fort.91

After executing the mkhco.job script a proton trajectory file is created that can be used in rfacav to compute true-averaged NOEs and R factors. The input file created in section 7.3.1 is used therefore.

% rfacav <cr>
! CALCULATION OF R-FACTORS FROM A PROTON TRAJECTORY

H coordinates (bin) file name ? : cramb_tav.hco <cr>
2D DINOSAUR input file name ? : cramb_tav.inp <cr>
R-factor output file name ? : cramb_tav.rfac <cr>
Calculate NOE averages from proton trajectory (y,n)? y <cr>
Instantaneous NOEs output file name ? : cramb_tav.instnoe <cr>
Time-averaged NOEs output file name ? : cramb_tav.avnoe <cr>
Distances output file name ? : cramb_tav.dis <cr>
Order parameters file name ? : ../cram/cram.s2 <cr>
Chemical shift file name ? : ../cram/cram.shift <cr>
avrfac.job created!
execute or submit it

The avrfac.job script file looks like:

# ! CALCULATION OF R-FACTORS
chdir /usr/ruuci5/albo/dino/examples/cram_tav
ln -s cramb_tav.hco fort.91
ln -s cramb_tav.inp fort.90
ln -s ..../cram/cram.s2 fort.92
ln -s cramb_tav.avnoe fort.93
ln -s ..../cram/cram.shift fort.94
ln -s cramb_tav.instnoe fort.97
ln -s cramb_tav.dis fort.98
(/bin/time /usr/ruuci5/albo/dino/src/avrfac) >& cramb_tav.rfac
/bin/rm fort.91
/bin/rm fort.90
/bin/rm fort.92
/bin/rm fort.93
/bin/rm fort.94
/bin/rm fort.97
/bin/rm fort.98
Note that this procedure will take some time since NOEs will be calculated for every structure/record in the proton trajectory file. Two NOE trajectories will be generated, one containing the instantaneous NOEs for every structure/record in the trajectory (\texttt{cram_tav.instnoe}) and one with the true-averaged NOEs as function of the time (\texttt{cram_tav.avnoe}). These can be analysed with \texttt{ananoe} (see section 7.1.8).

7.4 2D and 3D DINOSAUR refinement examples (helical peptide)

In the \texttt{dino/examples/2D} and \texttt{dino/examples/3D} two examples of 2D and 3D DINOSAUR refinement are given for a small helical peptide with synthetic NOE data. The DINOSAUR procedures for these two examples are basically the same as described for crambin above and no additional example will be given.

The following files can be found in these directories:

- \texttt{helix.2Dnoe} : 2D NOE file in free format
- \texttt{helix.3Dnoe} : 3D NOE-NOE file in free format
- \texttt{helix.mdh} : proton only file
- \texttt{helix.shift} : chemical shift file
- \texttt{helix.disre} : qualitative distance constraints file in GROMOS format
- \texttt{helix.htyp} : proton codes file
- \texttt{helix.topo} : GROMOS molecular topology file
- \texttt{target.md} : target structure from which the NOE files were generated
- \texttt{start.md} : start structure generated with DG and a few qualitative distance constraints.

In addition for 3D DINOSAUR a file is required that controls the output frequency of the theoretical NOE intensities written to the NOE trajectory. The output frequency is no longer controlled by the print parameter in the GROMOS input file but should be defined in free format in the "PROUT" file. If this file is not present a default value of 10000 will be used.
8. Benchmarks

DINOSAUR benchmarks are given in the `dino/benchmark` directory. Two type are used. The first one will compute theoretical NOE intensities by diagonalisation of the entire relaxation matrix for systems of increasing size and a constant number of NOE peaks. The corresponding script files are named `bench***.csh` where *** indicates the dimension of the system. In the second benchmark, theoretical NOEs will be computed with a spherical cut-off of 4.5 Å around each proton pair defining a NOE for an increasing number of peaks. The script files in this case are named `bench_ct_***.csh` where *** indicates the number of experimental peaks. In both case, cpu will be monitored for ten calculations after setting-up the system. The average values are given in the table below. The source codes and a makefile are found in the `dino/benchmark/src` directory. Source codes are the same as those used in the DINOSAUR refinement programs.

**Benchmark 1 (bench***.csh)**

- NOEs + gradient calculation for 100 peaks, 1 mixing time, for various dimensions of the system. No dynamical averaging for the calculations of the NOEs.
- Method: diagonalisation of the entire relaxation matrix using the EISPACK routines included in DINOSAUR.
- Times (user + system) [s] obtained by averaging over 10 calls to fnoe

<table>
<thead>
<tr>
<th>System</th>
<th>dimension of the matrix (# of protons)</th>
<th>factors</th>
<th>compiler options</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>SGI Challenge(100MHZ)</td>
<td>0.057</td>
<td>0.336</td>
<td>2.248</td>
</tr>
<tr>
<td>SGI Indigo2 (100MHZ)</td>
<td>0.062</td>
<td>0.363</td>
<td>2.495</td>
</tr>
<tr>
<td>SGI Indigo (100MHZ)</td>
<td>0.060</td>
<td>0.359</td>
<td>2.478</td>
</tr>
<tr>
<td>SGI Crimson (50 MHZ)</td>
<td>0.061</td>
<td>0.392</td>
<td>2.675</td>
</tr>
<tr>
<td>SGI 4D/35 (36 MHZ)</td>
<td>0.123</td>
<td>0.785</td>
<td>5.553</td>
</tr>
<tr>
<td>SGI Indigo (33 MHZ)</td>
<td>0.134</td>
<td>0.954</td>
<td>6.805</td>
</tr>
<tr>
<td>Convex C220</td>
<td>0.185</td>
<td>0.927</td>
<td>4.799</td>
</tr>
<tr>
<td>Convex C220 (veclib)</td>
<td>0.086</td>
<td>0.466</td>
<td>2.591</td>
</tr>
<tr>
<td>AlphaDec7000(150MHZ)</td>
<td>0.020</td>
<td>0.152</td>
<td>0.981</td>
</tr>
<tr>
<td>SUN (40 MHZ)</td>
<td>0.055</td>
<td>0.444</td>
<td>3.084</td>
</tr>
<tr>
<td>SUN (36 MHZ)</td>
<td>0.063</td>
<td>0.545</td>
<td>3.790</td>
</tr>
<tr>
<td>HP750 (100 MHZ)</td>
<td>0.054</td>
<td>0.343</td>
<td>2.462</td>
</tr>
</tbody>
</table>

**Benchmark 2 (bench_ct_***.csh)**

- NOEs + gradient calculation for various number of peaks, 1 mixing time, no dynamical averaging. System with a total of 400 protons.
- Method: cutoff 4.5 Å around each NOE peak. For each peak a small matrix (~20) is diagonalised with the EISPACK routines
- Times (user + system) [s] obtained by averaging over 10 calls to fnoe

<table>
<thead>
<tr>
<th>System</th>
<th>number of NOE peaks</th>
<th>factors</th>
<th>compiler options</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>SGI Challenge (100MHZ)</td>
<td>1.263</td>
<td>1.386</td>
<td>1.840</td>
</tr>
<tr>
<td>SGI Indigo2 (100MHZ)</td>
<td>1.258</td>
<td>1.397</td>
<td>1.878</td>
</tr>
<tr>
<td>SGI Indigo (100MHZ)</td>
<td>1.190</td>
<td>1.327</td>
<td>1.772</td>
</tr>
<tr>
<td>SGI Crimsom (50 MHZ)</td>
<td>1.141</td>
<td>1.299</td>
<td>1.759</td>
</tr>
<tr>
<td>SGI 4D/35 (36 MHZ)</td>
<td>2.739</td>
<td>3.008</td>
<td>4.513</td>
</tr>
<tr>
<td>SGI Indigo (33 MHZ)</td>
<td>3.425</td>
<td>3.557</td>
<td>4.667</td>
</tr>
<tr>
<td>Convex C220</td>
<td>1.776</td>
<td>2.321</td>
<td>4.373</td>
</tr>
</tbody>
</table>

**Remarks**

1. It appears from the benchmark 1 that DINOSAUR with diagonalisation of the entire relaxation matrix scales as a $N^3$ problem where $N$ is the dimension of the matrix. The time consuming part of the calculation is the diagonalisation itself. The time increases only slowly with the number of NOEs as can be seen from the following results (system of 100 protons)

<table>
<thead>
<tr>
<th>#NOE peaks</th>
<th>100</th>
<th>200</th>
<th>400</th>
<th>800</th>
</tr>
</thead>
<tbody>
<tr>
<td>cpu</td>
<td>1.000</td>
<td>1.034</td>
<td>1.106</td>
<td>1.142</td>
</tr>
</tbody>
</table>

2. When a spherical cutoff is used, the time for the calculations depends on the number of peaks and on the cutoff. For a given cutoff, the time dependence on the number of peaks becomes linear above approximately 200 peaks. For a given number of peaks (800) and an increasing cutoff, the results are (system of 100 protons):

   | cut-off [Å] | 3.0 | 3.5 | 4.0 | 4.5 | 5.0 | 5.5 | 6.0 |
   | <# neighbours>(max) | 9(14) | 10(17) | 14(22) | 18(31) | 23(36) | 28(45) | 33(55) |
   | cpu        | 1.000 | 1.458 | 2.498 | 4.232 | 6.900 | 11.77 | 18.90 |

The time dependence with spherical cutoffs can be defined approximately as

$$N_{\text{peaks}} \times N_{\text{neighbours}}^{(2-2.5)}$$
9. References


