

---

## Including Novel Restraints Supplied by the User to the TNT Refinement Package

Dale. E. Tronrud,

*Howard Hughes Medical Institute and Institute of Molecular Biology University of Oregon Eugene, Oregon 97403, USA - Email : [dale@uoxray.uoregon.edu](mailto:dale@uoxray.uoregon.edu) ; WWW: <http://www.uoxray.uoregon.edu/dale/welcome.html>*

### Abstract

It is not unusual, when refining the structure of a macromolecule, to wish to incorporate some information which was not foreseen by the authors of the software being used. This article discusses the mechanisms in the TNT refinement package which allow this information to be utilized, without modifying the source code for TNT nor recompiling. A program which calculates the data that TNT requires to apply the new restraints must be written by the user.

### Introduction

The TNT refinement package (Tronrud et al., 1987) has been available for many years and is still being developed and used for a number of refinement problems. One of its unique features is its ability to accept information from other programs which allows a macromolecular model to be restrained by information not explicitly anticipated within the TNT software.

One example of the successful use of this feature of TNT is the maximum likelihood refinement package Buster (Bricogne, 1997). This package uses the TNT refinement package to perform the calculation for function optimization and stereochemical restraints but replaces TNT's least-squares diffraction residual function with a much more elaborate maximum likelihood target function.

TNT uses the preconditioned conjugate gradient method of function optimization (Tronrud, 1992). This algorithm uses a subset of the Normal matrix (and that subset must be easily invertible) to find an alternative coordinate system for the parameters of the model. The new, abstract, coordinate system is one where the conjugate gradient method (Fletcher & Reeves, 1964) will be more efficient. The conjugate gradient method requires the ability to evaluate the function being optimized given any set of values for the parameters of the model, as well as the gradient of this function. The preconditioned conjugate gradient method requires, in addition, the ability to calculate the diagonal of the Normal matrix.

If the new information being added to the refinement is statistically independent of the restraints implemented in TNT, the expanded restraint function will simply be the sum of the traditional restraints and the function based on the new restraints. Since the derivative operator is a linear operator, the first and second derivatives of the expanded restraint function will, likewise, be the sum of the derivatives of the "traditional" restraint function and the added restraint function. The TNT refinement package can read function values, first derivatives, and the diagonal of second derivatives from a formatted file and add these values to its own corresponding quantities.

To add restraints to a TNT refinement, the user must write a program which will read a TNT coordinate file and produce the value of the restraining function, its first derivative, and diagonal of its second derivative. Proper modification of the TNT refinement script will cause the program to run at the appropriate time, and instruct TNT's program Shift to read the supplied data files in addition to TNT's own.

## Choice of Function

TNT is a least-squares refinement package. If the added restraints are to be compatible with the other restraints anticipated by TNT the user should preferably cast them as least-squares restraints, or, failing this, in a negative log-likelihood expression.

The form of a least-squares residual function is

$$f = W \sum_{\text{Observations}} \frac{(Q_{\text{obs}} - Q_{\text{calc}})^2}{\sigma_{\text{obs}}^2} \quad (1)$$

where  $Q_{\text{obs}}$  is the target value of the restraint,  $\sigma_{\text{obs}}$  is the standard deviation of, or the confidence in, the target value, and  $Q_{\text{calc}}$  is the equivalent quantity calculated from the current model. The weight,  $W$ , is used to control the balance between the added restraints and those of TNT. Its value has to be determined empirically by running a series of refinement jobs with a range of values and selecting that value which results in either the lowest free R or all the categories of restraints being met to within the precision of their libraries. If the  $\sigma$ 's are accurate the value of the weight will be close to unity.

If the new restraints are cast in the form of a probability, the function will be a form incompatible with a least-squares residual function. This difference, along with the need to maximize probabilities and minimize residuals, makes it impossible to combine within TNT. Taking the negative logarithm of a probability distribution will make the function compatible with a leastsquares residual.

## Format of TNT's Data Files

Programs in the TNT refinement package pass data amongst themselves in free-format text files. Each line in one of these files contains a packet of information which can be interpreted independent of its context - The order of the lines in a file is of no importance. The type of data on a line is defined by a keyword written at its head.

To allow TNT to track which piece of information belongs to which part of the function being minimized, the user must choose a name for the information being supplied. You can choose whatever name you wish (with the exception of `RFACTOR`, `GEOMETRY`, and `NCS` which TNT uses) but only the first four letters are significant. In the examples that follow the name "MINE" is used.

The keywords used to pass information about restraints to TNT's optimizer program (named Shift) are

<code>FUNCVAL</code>	Provides the value of the function being minimized
<code>DRVC</code>	Provides the first derivatives of the function with respect to the parameters of a single atom
<code>CRVC</code> , <code>CRVAC</code> , and <code>CRVBC</code>	Provides the second derivatives of the function with respect to the parameters of a single atom

Following the keyword is the name chosen for the supplied function.

The `FUNCVAL` statement only needs the function name and the value of the function when evaluated for the current model coordinate file. An example of a `FUNCVAL` statement is

A DRVC statement contains the first derivative of the function for the parameters of a particular atom. There is one DRVC statement for each atom in the model. In this statement the keyword is followed by the name chosen by the user, and the derivatives with respect to the five parameters of the atom (X, Y, Z, B, and occupancy). These numbers are followed by the atom's name, the name of the residue containing it, and the name of the chain containing it. An example of a DRVC statement is

```
DRVC MINE 2.12E+02 3.62E+02 1.12E+02 -1.13E+00 7.22E+01 N 1 A
```

These are the five derivatives for the amide nitrogen of residue 1 in chain A.

A question which now must be answered is "Which coordinate system is being used?" TNT was created prior to the establishment of the current "PDB standard" Cartesian coordinate system. While the coordinate system used in TNT to describe the location of atoms is Cartesian, it is not that used in a PDB format coordinate file. To convert from crystallographic "fractional" coordinate system to TNT's Cartesian system use the conversion matrix

$$\mathbf{M}_{\mathbf{c} \rightarrow \mathbf{o}} = \begin{pmatrix} 1/a^* & 0 & 0 \\ -a \sin \beta \cos \gamma^* & b \sin \alpha & 0 \\ a \cos \beta & b \cos \alpha & c \end{pmatrix}. \quad (2)$$

If the residual function does not depend on the placement of the molecule in the crystal then this difference in convention is unimportant. The coordinates found in the TNT coordinate file will already be in an appropriate orthonormal coordinate system.

The diagonal second derivatives are defined with the CRVC, CRVAC, and CRVBC statements. In the most elaborate form available in TNT the diagonal of the second derivative matrix is defined as a symmetric 5x5 block for each atom. (The matrix blocks containing second derivatives for parameters in different atoms are always assumed to be zero in TNT.) The values on the diagonal of this 5x5 block (i.e.  $\partial^2 f / \partial X^2$ ,  $\partial^2 f / \partial Y^2$ ,  $\partial^2 f / \partial Z^2$ ,  $\partial^2 f / \partial B^2$ , and  $\partial^2 f / \partial Occ^2$ ) are almost always non-zero and significant. The off-diagonal elements of this block are almost always zero (except for  $\partial^2 f / \partial B \partial Occ$ ) and can usually be ignored.

Reflecting this pattern, the CRVC statement contains the values of the five diagonal elements of the block. If all the other elements are equal to zero there is no need to write CRVAC and CRVBC statements. Since the diagonal elements are given on the CRVC statement the order of values on the other statements is a little odd. The values are listed in "threaded order" which is best described by the figure

$$\begin{pmatrix} 1 & A1 & A5 & B3 & B5 \\ - & 2 & A2 & B1 & B4 \\ - & - & 3 & A3 & B2 \\ - & - & - & 4 & A4 \\ - & - & - & - & 5 \end{pmatrix} \quad (3)$$

The elements identified as 1, 2, 3, 4, and 5 are given (in that order) on the CRVC statement. The elements identified as A1, A2, A3, A4, and A5 are given on the CRVAC statement and the elements B1, B2, B3, B4, B5 are given on the CRVBC statement. Since the CRVBC statement contains only derivatives which

mix positional parameters with the B factor or occupancy it has contained only zeros in our experience. It is unlikely that you will ever write a CRVBC statement.

An example of a CRVC/CRVAC pair is

CRVC	MINE	5.90E+02	5.90E+02	642E+02	2.07E-01	8.37E+02	N	1	A
CRVAC	MINE	0.00E+00	0.00E+00	0.00E+00	-1.15E+01	0.00E+00	N	1	A

## Practical Considerations for Curvatures

The diagonal approximation to the second derivative matrix that is used in TNT is a poor one for many types of restraints. While it works reasonably well for restraints to diffraction amplitudes, geometrical restraints will always have large off-diagonal elements which are a consequence of the "ties" between various sets of atoms.

We have determined, empirically, that when refining with restraints which have large off-diagonal elements (which are always ignored in TNT) the off-diagonal elements in the diagonal 5x5 block should also be ignored. In TNT CRVAC and CRVBC statements are never written for stereochemical and noncrystallographic symmetry restraints.

The diagonal block of second derivatives which is calculated for restraints to diffraction amplitudes for most atoms contains zeros on all off-diagonal elements (except for the  $\partial^2 f / \partial B \partial \text{Occ}$  element which is not important if the occupancy of the atom is held fixed.) The exception to this pattern occurs when an atom is near a special position in the unit cell. In that situation, the off-diagonal elements of the positional second derivatives are non-zero and are critical to the correct refinement of the location of the atom.

It is recommended that any new restraint programs be written to produce CRVAC and CRVBC statements. Tests comparing the efficiency of the refinement with and without these statements, however, should be run to determine if the presence of this information is helpful or harmful.

## Editing the TNT Refinement Script

TNT consists of a number of programs which are coordinated by a script when running refinement. Several csh scripts are supplied for use on unixlike operating systems. The user will have to edit one of these scripts to cause the new program to be executed and its data to be read by TNT.

Regardless of the features of a particular script, all cycles of refinement in TNT have two parts. In the first part (called the "long loop") each helper program calculates the function value, gradient, and diagonal of the second derivative matrix for its restraints.

All this information (along with the previous cycle's shift vector) is read by the program Shift and three things are done:

- A new shift vector is determined by applying the preconditioned conjugate gradient method.
- An estimate of the fraction of this shift vector to be applied is made and written to a temporary file.
- A coordinate file (named shifted.cor) is written which has the trial shift applied.

In the second part of the cycle of refinement (called the "short loops"), the true fraction of the shift vector which optimizes the model is determined. This goal is accomplished by a one-dimensional search along the shift vector. In each iteration, each helper program evaluates its residual function for the trial coordinate file; no derivatives are required here. The program Shift reads the current function values and

its own temporary files containing the status of the optimization, and calculates a new estimate. If the new estimate is not significantly different the cycle ends and a new cycle begun.

Refinement of a model against diffraction data and stereochemical restraints is performed by the script `$tntbin/tnt` cycle. The simplest way to add new restraints to a cycle of TNT is to make a copy of this script and edit it. This script runs the programs Rfactor and Geometry (which are TNT's helper programs for restraints on diffraction data and ideal stereochemistry) and then runs Shift. The user should add the commands required to cause their program to calculate its function value and derivatives before the script runs Shift. Then modify the script to cause Shift to read the file containing the appropriate file.

The script commands which run Shift are

```
#
$tntbin/shift << $eof
INCLUDE init.cor
INCLUDE rfactor.dat
INCLUDE geometry.dat
INCLUDE olddir.dat
INCLUDE $control
$eof
if ($status >< 0) exit 1
if (-e tempparm.tmp) rm tempparm.tmp
#
rm rfactor.dat geometry.dat
```

These commands cause Shift to read the starting coordinates (init.cor), the output of the programs Rfactor and Geometry (rfactor.dat and geometry. dat), the previous shift vector (olddir.dat), and the TNT control file (whose name is stored in the symbol \$control). If the user's program writes its data in the file mine.dat add the statement

```
INCLUDE mine.dat
```

anywhere in the existing list of INCLUDE statements.

In the second half of the script Rfactor and Geometry are run once again. These programs are followed by Shift. The user will have to add commands to run the added program here (This time only calculating the function value) and add an INCLUDE statement to force Shift to read the user's new file.

## Example: Restraining to Torsion Angle Distributions

TNT, as well as other refinement packages (e.g. CNS (Brünger et al., 1998)) restrain molecular models to ideal torsion angles using a simplistic method. Each torsion angle can have several "ideal" values which represent the allowed rotomers for that group of atoms. These ideal values are required to be equally spaced within the circle of 360 degrees, and the width and height of the probability of each of these values are assumed to be identical. Priestle (2003) discusses many of these shortcomings.

To create a restraint on torsion angles which does not have these limitations, one could use the raw probability distributions for the  $\chi$  angles in place of the distilled list of most probable  $\chi$  angles usually listed in a rotomer library. Probability distributions suitable for this use are listed in Priestle (2003).

The target function of such a restraint would be

$$f = -W \ln \prod_{j=1}^n P_{\chi_j}(\chi_{c,j}(\mathbf{x})), \quad (4)$$

where  $W$  is the overall weight of this term relative to the term restraining the model to the diffraction data and the term restraining other stereochemistry items,  $j$  is the index of a particular side chain  $\chi$  angle,  $\chi_{c,j}$  is the value of this  $\chi$  angle calculated from the current model, and  $P_{\chi_j}$  is the probability distribution of all the values this  $\chi$  angle could adopt.

The probability of an entire model is the product of the probabilities of each of its  $\chi$  angles, assuming that the values of these angles are independent of each other. (They are, of course, not independent but to handle the relationship between  $\chi$  angles requires higher dimensional probability distributions which introduce their own problems. We will continue this example assuming independence.) The negation of the log of this product is calculated to ensure compatibility with TNT.

TNT requires the first, and diagonal of the second derivatives of this function. These derivatives must be with respect to the positional parameters of the atoms in the model. The equation for the first derivatives are

$$\frac{\partial f}{\partial \mathbf{x}_i} = - \sum_{k=\{a,b,\dots\}} \frac{1}{P_{\chi_k}(\chi_{c,k}(\mathbf{x}))} \frac{dP_{\chi_k}(\chi_{c,k}(\mathbf{x}))}{d\chi_{c,k}(\mathbf{x})} \frac{\partial \chi_{c,k}(\mathbf{x})}{\partial \mathbf{x}_i} \quad (5)$$

and the equation for diagonal of the second derivative matrix is

$$\frac{\partial^2 f}{\partial \mathbf{x}_i^2} = \sum_{k=\{a,b,\dots\}} \left( \frac{1}{P_{\chi_k}^2(\chi_{c,k}(\mathbf{x}))} \left( \frac{dP_{\chi_k}(\chi_{c,k}(\mathbf{x}))}{d\chi_{c,k}(\mathbf{x})} \right)^2 \left( \frac{\partial \chi_{c,k}(\mathbf{x})}{\partial \mathbf{x}_i} \right)^2 \right. \quad (6)$$

$$- \frac{1}{P_{\chi_k}(\chi_{c,k}(\mathbf{x}))} \frac{d^2 P_{\chi_k}(\chi_{c,k}(\mathbf{x}))}{d\chi_{c,k}^2(\mathbf{x})} \left( \frac{\partial \chi_{c,k}(\mathbf{x})}{\partial \mathbf{x}_i} \right)^2 \quad (7)$$

$$- \frac{1}{P_{\chi_k}(\chi_{c,k}(\mathbf{x}))} \frac{dP_{\chi_k}(\chi_{c,k}(\mathbf{x}))}{d\chi_{c,k}(\mathbf{x})} \frac{\partial^2 \chi_{c,k}(\mathbf{x})}{\partial \mathbf{x}_i^2} \left. \right) \quad (8)$$

$\mathbf{x}_i$  is the positional parameters (i.e.  $X$ ,  $Y$ , and  $Z$ ) for the  $i^{\text{th}}$  atom and  $\{a, b, \dots\}$  are the indices of the torsion angles which involve this atom.  $\partial \chi_{c,k}(\mathbf{x}) / \partial \mathbf{x}_i$  indicates how the torsion angle changes as the atom is moved. Values for these derivatives can be found in Tronrud *et al.* (1987).

The derivative of the probability distribution with respect to the  $\chi$  angle can either be calculated either by discrete differentiation of the histogram itself, or by taking the derivative of a curve fit to the histogram.

The second derivative contains three terms. While the first term is positive in all cases the second and third may be either positive or negative. The optimization method used in TNT requires that the second derivative matrix be positive definite. For a diagonal matrix this means that all the diagonal entries must be positive. The performance of TNT would likely be improved by only calculating the first term.

## Further Information

To find further information about the TNT refinement package visit the website <http://www.uoxray.uoregon.edu/tnt/>

## Acknowledgements

This work was supported in part by NIH grant GM20066 to B.W. Matthews.

## References

Bricogne, G. (1997). In *Macromolecular Crystallography*, Part A, edited by R. M. Sweet & C. W. Carter, Jr volume 276 of *Methods in Enzymology* pp. 361-423. New York: Academic Press, Inc.

Brünger, A. T., Adams, P. D., Clore, G. M., Gros, P., Grosse-Kunstleve, R. W., Jiang, J.-S., Kuszewski, J., Nilges, M., Pannu, N. S., Read, R. J., Rice, L. M., Simonson, T., & Warren, G. L. (1998). *Acta Cryst.* D54, 905-921.

Fletcher, R. & Reeves, C. (1964). *Computer Journal*, 7, 81-84.

Priestle, J. P. (2003). *J. Appl Cryst*, 36, 34-42.

Tronrud, D. E. (1992). *Acta Cryst.* A48, 912-916.

Tronrud, D. E., Ten Eyck, L. F., & Matthews, B. W. (1987). *Acta Cryst.* A43, 489-501.