

User Guide for



Modeling Package

(An user interface for EXDIS/DIAMOD/FANTOM/TRANSLATE)

Version 1.0

(C) The University of Texas System

by

Venkatarajan S. Mathura & Prof. Werner Braun

UTMB, Galveston

CONTENTS

1	Introduction	2
2	Protein structure prediction	2
3	Homology modeling	3
4	Steps involved in the homology modeling	5
4.1	Step 1: inferring homology	5
4.1.1	BLASTP/PSIBLAST	5
4.1.2	Fold-recognition servers	5
4.2	step 2: Alignment refinement & selection	6
4.2.1	Selection of alignment	7
4.3	step 3: Constructing 3d model	8
5	Integrated modeling package - mpack	8
6	Guide to use modeling programs	10
6.1	Tools available in our group	10
7	using Mpack (version 1.0)	13
7.1	Requirement	13
7.2	Installation	13
7.3	Features	13
7.4	Preparing input files for mpack	13
7.4.1	Alignment file (s)	13
7.4.2	Template pdb file(s)	14
7.4.3	Translate topology file	15
7.5	Executing mpack	16
7.6	Selecting crude model for energy minimization	19
7.7	Output of mpack	20
7.8	Running molmol to visualize model	21
7.9	Evaluating the model	22
8	Examples	22
8.1	Example 1: modeling c20 from chicken	23
8.2	Example 2: modeling igg 1 hil based on two templates	26
9	MPACK program details	33
10	Version	35
11	Author	35
12	Disclaimer	36

1 Introduction

With the completion of human genome projects, protein tertiary structure prediction from primary sequence is gaining tremendous importance. Structure prediction is helpful in obtaining structural insights and understanding complex biochemical function of proteins especially in a large-scale, where conventional experimental methods like NMR, X-Ray are limited by time and cost involved. Structural knowledge of protein function and interaction will help to accelerate drug-discovery and maximize the meaningful information content of numerous genomics and proteomics projects.

2 Protein Structure Prediction

Proteins can be modeled using comparative modeling (homology modeling), threading or *ab initio*. In homology modeling technique a closely related highly homologous structure is used to build the model for an unknown sequence. The success of homology modeling technique is very high especially if a clear homologue structure is already known. One cannot hope to have structure of closely related homologue protein for all unknown sequence. Difficult targets that do not show obvious homology for sequences with structure can be modeled by threading techniques. Threading techniques employ structural profiles or physico-chemical based profiles (one that is being developed in our group) for identifying how well the query sequence fits into a known fold. Threading can identify suitable targets to model even when the sequence alignment identity is as low as 15% (Twilight region). Success of threading depends on the type of method and also in many cases limited. With structural genomics in progress there will be higher number of structures determined per year basis. This will in turn aid homology modeling and threading techniques to perform

better. In cases where there is no way to infer homology to a known structure and if the unknown sequence is small enough (less than 150 residues) one can resort to purely physics based approach called as *ab initio* methods or use a combination of physics and knowledge based methods called new fold detection techniques.

3 Homology Modeling

By the term homology in biological context relates to an evolutionary relationship shared among proteins. In an evolutionary tree, daughter proteins are evolved by random mutation and selection, which creates a net drift at each stage acting as a flux, from parent or ancestral proteins. The daughter proteins to a large extent share identical sequence with few residue variants to the ancestral proteins. They share similar structure and function. The selection process allows residue variation only in those regions that are less important for the function of the protein (those residue changes don't affect the structure too, because the geometrical framework upon which the functional residues are anchored must not be disturbed). Due to degeneracy of codons, wobble in the triplet codes, all changes in nucleotides may not be reflected at the protein primary sequence level. The protein primary sequence can be represented by 20 unique alphabets. But studies have shown that amino acids can be exchanged with each other without compromising changes in the structure. Such exchanges are possible because amino acids share similar physico-chemical properties, and any changes within similar group will be tolerated. The degree of substitution at a particular residue position depends on the functional role and the environmental location of the residue in the folded form of the protein. Due to this a number of slightly different sequences may adopt similar structure (divergent evolution). Hence a many to one

relationship is possible between members in the sequence and the structure space (Note: Similar structure or geometry of a set of amino acids, may have been evolved independently due to convergent evolution). Experimental structure determination using X-ray, NMR have accumulated information in the form of a databank called PDB (protein databank). Now the task is to assign structural parent available in the PDB to an unknown sequence using some qualitative inference rules. Since homologous sequences share a high amount of sequence identity one can adopt this as an inference rule. As a rule of thumb one can model an unknown sequence (target) based on a sequence for which structure has been determined (template) if they share identical residues of greater than 40% along their length. Since amino acids that share similar physical-chemical properties can be exchanged without affecting the structure using a similarity exchange matrix like Blosum, PAM or Gonnet one can measure similarity index for comparison of two sequences and infer homology. Similarity matrices are derived using sequence, structure or physical-chemical property information. In constructing Blosum, PAM or Gonnet one uses sequences that belong to the same family (basically sequences that share very high sequence identity) to obtain frequency of amino acid exchange and obtain a quantitative value that describes the relationship between two amino acid alphabets. Structure based matrix uses more diverse sequences that adopt similar 3D structure, and calculates exchange frequency of amino acids. Once two sequences are compared and homology is inferred the next task will be to complete the unknown structure using the structural information from the template.

4 Steps involved in the homology modeling

1. Infer homology between target sequence and template sequence by comparing their sequence using a similarity matrix (and by deriving an alignment)
2. Once homology can be inferred using some inference rules align/refine alignments between two sequences
3. Construct a 3D model for the sequence based on the template:
 - a. Using alignment and template pdb file extract geometrical constraints
 - b. Apply these geometrical constraints to force the target to adopt similar structure.
 - c. Energy refine the crude model developed in 'b'

4.1 Step 1: Inferring homology

4.1.1 BLASTP/PSIBLAST:

BLASTP is a sequence search tool which uses a similarity matrix like BLOSUM, PAM to compare a query sequence with subject sequence using a Smith-Waterman algorithm. For every pair-wise comparison BLAST produces a Bit-score (by adding log-odd bit for every matching pairs during comparison) and an E-value (Expected value indicates the probability of a sequence to match with similar score randomly in a database of similar size). PSIBLAST is a modified version of BLAST in which a position-specific scoring matrix is created and iteratively updated as new sequences above a certain E-value threshold is found. A BLAST/PSIBLAST search may result in a hit that has an experimental structure. If it does then (with E value threshold <0.001 for BLAST, and <0.005 for PSIBLAST in non-redundant sequence database) the subject sequence hit can be used as a template for

modeling. If BLASTP/PSIBLAST fails to locate a suitable template one can use more effective fold-recognition servers that return template and alignments.

4.1.2 Fold-recognition servers

Fold-recognition in general tries to match a given sequence to encoded information about a structure or more specifically a fold, thus distinguishing proteins with different topology. Since structural information is encoded in this approach a remotely related proteins can be identified compared to BLAST/PSIBLAST. There are many different fold-recognition servers having their own algorithms that are based on structural parameters (or profiles and sometimes implements purely sequence based Hidden Markov Models like SAM-T99). The output of fold-recognition servers will be a score, templates ranked based on the score and corresponding alignments. A METASERVER is available to submit one or more sequences to different servers that have proven to perform better in previous CASP (Critical Assessment of Structure Prediction) experiments.

4.2 Step 2: Alignment refinement & Selection

The easiest case in modeling protein is the one with a unique template and unique alignment returned by alignment programs (mostly the case if the sequence identity is very high eg. 90%). More complicated cases will involve template being clear but the alignments may be different, or the servers return different templates with different alignments. Refining alignments may require additional biological knowledge about the protein, like conserved residues that must match both in the template and the target. Most servers return alignments that may require slight hand-improvement (one can use predicted secondary structure and template secondary structure as a guide). The protein may be consists of two domains or a

part of the protein may not have matching template. In those cases approaches like simple FASTA search in PDB may be helpful to select a local template. An overlapping segment that will match predicted secondary structure and have higher sequence identity to the fragment may be a good choice.

Some simple rules

- Do not allow gaps in the secondary structure region
- Try to match secondary structures between template and predicted target
- When selecting a small fragment from another template to match a small region for the template try to have an overlapping region so that the new fragment can be oriented.
- Extract more constraints if you are using smaller fragments

4.2.1 Selection of Alignment

A difficult situation is one where one has different folds (templates) scoring similar or the scores are not significant. In that one can look for a voting scheme in which a particular fold ranked or frequently occurs in the top 5 hits from different servers. Such voting schemes to select template and alignment has been implemented in metaservers like BIOINFO, BIOSERVER. One can also use ALIGNSCORER based on physical-chemical motifs to obtain the highest scoring alignment for each of the motif.

4.3 Step 3: Constructing 3D model

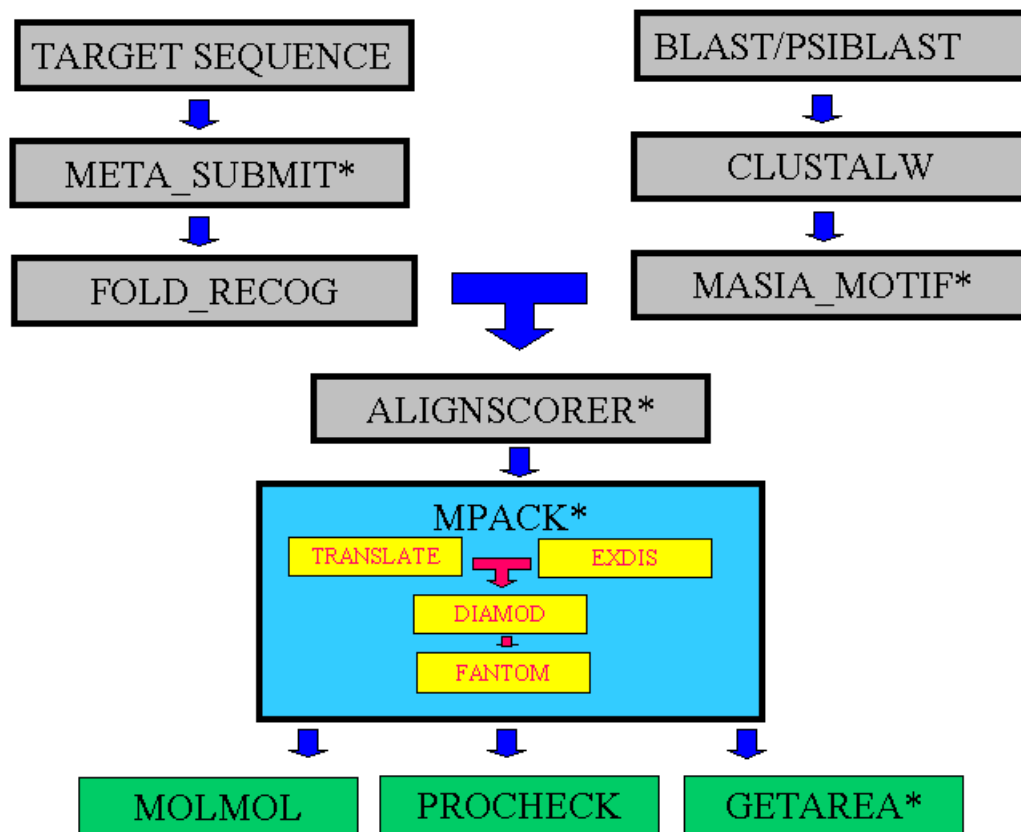
After suitable template and alignment is obtained one can extract geometrical constraints like distance between matching atoms in the template and target. The backbone dihedral angles like PHI, PSI and OMEGA (mostly set to 180 except when there is a CIS PROLINE) are also extracted to be used as geometrical information. A distance geometry program takes in distance constraints, and dihedral constraints to apply progressively on to a random structure such that the target protein optimally fits into the geometrical constraints.

EXDIS extracts distance and dihedral constraints that are used by DIAMOD to solve for a crude model using distance-geometry algorithm. At this stage one can develop several crude models that should be thoroughly inspected for consistency in the structure (like no sever clashes, mirror images etc). The crude model is subjected to energy minimization during which energetically unfavorable interactions are removed to have a refined structure.

5 Integrated Modeling Package - MPACK

MPACK (Modeling Package) is an integrated protein modeling suite that currently handles comparative and *ab initio* modeling procedures. The objective of this suite is to systematically bring different steps (or programs) under one roof in order to facilitate rapid model generation with minimal user effort and to create a biological data-flow pipeline for large scale-scale modeling of protein sequences from genomic projects. This integrated suite was created with the geometry extraction program EXDIS that extracts distance and dihedral constraints, specified using protein template for which experimental structure is known. Inside/outside, secondary structure predictions from MASIA and other knowledge based topological constraints can be used directly to convert into suitable distance constraints using TRANSLATE. Geometric constraints generated are directed as input into

the distance geometry program DIAMOD that optimally calculates structures by either starting from a random conformation or from approximate models produced by EXDIS. The program also has an option for generating models with disulphide bonds and switches for self-correcting distance geometry procedures when approximate constraints (like constraints from MASIA predictions) are used. The package is robust and efficient in handling situations wherein a user intends to model a target based on multiple template fragments.



Flowchart of the modeling procedure. Softwares ALIGNSCORER returns scores for the alignment fit in motif regions and used for the evaluation of the quality of the alignment. Current release of MPACK contains TRANSLATE, EXDIS, DIAMOD and FANTOM.

6 Guide to use modeling programs

6.1 Tools available in our group

MASIA*

Purpose: To create motifs, optimize motif parameters, secondary structure prediction, inside/outside prediction, measure degree of sequence conservation, match complex motifs/patterns (eg. Helix-turn-helix)

Availability: Web version at http://www.scsb.utmb.edu/comp_bio.html/masia

METASUBMIT

Purpose: To submit one or many sequences to different fold-recognition servers.

Availability: Through web at <http://129.109.73.53/venkat/metasubmit.html>

GETAREA

Purpose: To measure exposed surface area of proteins

Availability: Through web at http://www.scsb.utmb.edu/cgi-bin/get_a_form.tcl

MOTIFMAKER (included in PCPMer)

Purpose: To create motifs and physical-chemical property based profiles

Availability: Command line

PCPMer

Purpose: To create physical chemical property based motifs and search for related sequences

Availability: Command line

ALIGNMENTSCORER

Purpose: To score different motifs in an alignment

Availability: Command line

ALIGNREP

Purpose: To reformat alignment into CLUSTALW format, calculate %id and fragments

Availability: Command line, more information of usage available at:
<http://129.109.73.53/venkat/casp5/alignrep.html>

MPACK

Purpose: To provide an user interface to modeling programs – EXDIS, DIAMOD, FANTOM, TRANSLATE

Availability: As a package with EXDIS/DIAMOD/FANTOM and TRANSLATE. More information available at <http://129.109.73.53/MPACK/>

EXDIS*

Purpose: To extract geometrical constraints from template

Availability: As a modeling package EXDIS/DIAMOD/FANTOM

DIAMOD*\$

Purpose: To generate crude models using constraints from EXDIS/TRANSLATE using either a random structure as a start or an approximate structure

Availability: As a modeling package EXDIS/DIAMOD/FANTOM

TRANSLATE

Purpose: To generate constraints based on inside/outside or secondary structure predictions.

Availability: As a modeling package EXDIS/DIAMOD/FANTOM/TRANSLATE

FANTOM*

Purpose: To perform energy minimization, monte-carlo simulation etc.

Availability: As a modeling package EXDIS/DIAMOD/FANTOM/TRANSLATE

COFIMA

Purpose: To reformat DIAMOD/FANTOM files into regular PDB files, to renumber coordinates

Availability: Command line

MAKETOP

Purpose: To convert JPRED secondary structure and MASIA inside/outside files into topology files suitable for TRANSLATE input

Availability: Command line. More details available at the web site

<http://129.109.73.53/venkat/casp5/maketop.html>

DISPLAYSEC

Purpose: To display alignment with predicted secondary structure and template secondary structure.

Availability: Command line. More details available at

<http://129.109.73.53/venkat/casp5/displaysec.html>

***Printed detailed manual available.**

\$Refer to DYANA manual.

7 USING MPACK (VERSION 0.6)

7.1 Requirement

- Fortran compiler F77, G77
- PERL compiler
- OS: Linux, IRIX

7.2 Installation

1. Edit Makefile appropriate for your operating system (default is linux rh7.2)
2. Type at unix prompt: `sh install.sh`
3. The script will automatically create executables for EXDIS, DIAMOD, TRANSLATE, FANTOM and MPACK
4. If you want to reinstall type `make clean` and follow the installation procedure.

7.3 Features

- Script for MOLMOL visualization
- TRANSLATE included
- Multiple-templates can be used for modeling
- SS-Bond constraints can be added automatically
- LOG file contains details of the program execution


7.4 Preparing input files for MPACK

7.4.1 Alignment file (s)

The alignment obtained from Fold-recognition servers or hand edited alignments should be formatted in an appropriate format (CLUSTALW format). One can ALIGNREP to achieve

this from a crude alignment. ALIGNREP reformats the alignment into appropriate format which has to be renamed with an extension *.aln.

The first input file (called master alignment) should have the complete target residues. If one wants to use multiple templates or small fragments then appropriate alignments containing those regions should be given as an alignment (starting number of the



Alignment rules:

- Make sure no special characters in the template/target name like space, comma, or -
- Gaps should be indicated only by "-" and not by "."
- No space allowed in the alignment block
- Indicate "CYSS" in the alignment file (for the target) by 'c' instead of 'C'
- Only 20 capital alphabets and 1 small alphabet [c] are accepted. Alphabets like Z, J, X must be excluded.
- Sometimes selenomethionine will be indicated by 'X' edit this to 'M' and also you may be required to edit the template pdb file. Refer to Template PDB rules.
- Sometimes alignment (especially from BLAST) will have 'X' marked at the low complexity regions of the target. You must edit it back to the original residues.

target residue contained in these files should be mentioned separately). All alignment files must follow the alignment rules.

7.4.2 Template PDB file(s)

Template PDB files can be obtained from RCSB (<http://www.rcsb.org>). Almost in all cases to have consistency while modeling it is recommended to spend time to look into the header files of the PDB. In some cases it may be required to edit the file.



Template PDB rules:

- Read header carefully for SSBONDS. If you see a matching “C” in the template and the target, most probably the target may also have similar SSBONDS you may need to edit the alignment file. Refer to the “Alignment rules”.
- The template residues in the PDB file and in the alignment file should match EXACTLY otherwise the program will generate errors and stop. There is no set way to do this other manually inspect for the difference.
- In case of selenomethionine rename MES to MET and delete SE.
- In some cases mutations/clashes/insertions compared to homologus will be indicated by including same residue number with an alphabet A or B eg. In the pdb file 1EJG (crambin structure)

ATOM	101	CA	AILE	A	7	8.829	2.039	13.300	0.55	2.58	C
ATOM	102	CA	BILE	A	7	9.104	2.209	13.197	0.45	2.14	C
ATOM	103	C	AILE	A	7	7.559	2.141	12.460	0.55	2.53	C
ATOM	104	C	BILE	A	7	7.839	2.105	12.369	0.45	2.15	C
ATOM	105	O	AILE	A	7	7.573	2.124	11.205	0.55	2.78	O
ATOM	106	O	BILE	A	7	7.990	2.102	11.154	0.45	2.53	O

In this case A chain has residue ILE and has AILE and BILE. Delete one of them and rename it to ILE. Valid records include (ATOM, ATOM No., ATOM NAME, 3letter RES, Residue number, 3d co-ordinates). Also refer to pdb file 1QKG where homologus residues are represented by an alphabet following the residue number.

- PDB file should end with TER followed by END.
- The file should end with extension *.pdb
- If the pdb file consists only of C-alpha trace (only CA atoms) use program like MAXSPROUT (http://jura.ebi.ac.uk:8181/holm/dali_align.cgi?mode=maxsprout) to construct main chain atoms.

7.4.3 TRANSLATE Topology file

If you want to include additional constraints like dihedral constraints for secondary structure regions, inside/outside predictions you can directly input a topology file (*.top) that

includes the details. You can use MAKETOP program available to convert JPRED secondary structure, MASIA inside/outside to topology file. You can hand edit or create a new topology file following specific format.

```
#Topology created with masia io output and MAKETOP          (always include comment in the first line)
alph H1 20 39                                               (Helix 1 start and end residue)
alph H2 64 73
alph H3 80 94
alph H4 100 106
alph H5 110 117
alph H6 128 144
alph H7 158 164
alph H8 170 197
alph H9 211 222
alph H10225 238
alph H11249 266
alph H12270 273
turn T1 40 52                                               (Turn 1 start and end residue)
beta B1 53 57                                               (Beta strand 1 start and end residue)
beta B2 274 278
beta B3 302 304
shta B1 B2 54 277                                           (antiparallel Beta sheet between B1 and B2 including Hbond)
insd 9 26 34 37 42 44 52 53 54 65
insd 9 66 74 79 80 82 84 86 87 90
insd 9 91 98 99 109 112 115 116 119 126
insd 9 127 133 137 139 146 150 152 157 165
insd 9 169 171 173 175 176 179 182 184 188
insd 9 189 192 200 206 208 211 212 214 215
insd 9 221 226 228 229 230 231 235 240 247
insd 9 251 254 257 264 267 270 271 273 276
insd 6 282 288 295 296 297 298
outs 9 24 29 31 33 35 38 39 40 41
outs 9 43 45 46 47 50 56 59 60 67
outs 9 71 73 75 76 77 78 83 89 92
outs 9 93 96 97 100 101 102 104 106 107
outs 9 108 111 114 120 121 125 128 131 132
outs 9 135 136 138 141 142 144 147 149 151
outs 9 153 154 155 156 158 159 163 164 166
outs 9 167 170 172 174 177 180 185 186 187
outs 9 190 191 193 194 199 202 205 207 210
outs 9 213 217 219 223 227 233 236 238 241
outs 9 244 245 246 248 249 253 255 256 260
outs 9 261 262 263 265 268 269 272 274 275
outs 9 277 278 280 281 284 286 290 293 294
outs 6 299 300 301 303 304 305
(iinside/outside, number of residues declared per line, residue numbers)
```

7.5 Executing MPACK

Using MPACK is simple if proper inputs are created adhering to rules. The user simply interacts with questions. An interactive session with typical questions from MPACK is described below

Start the mpack script by calling "mpack" and follow the questions:

Type core name of the output file?

This will be used for creating different output files. Use simple names like model1 or mod1. Do not use extensions or other fancy characters.

Do you have a topology file (y/n) ?

Say yes if you would like to use a topology file to derive approximate constraints. You must adhere to the topology file format.

Do you have a template to extract constraints from ?

If you have a template to extract constraints from enter 'y'. Please note the first template used should correspond to your master alignment.

Name of the seq. align. file ?

Enter alignment file name, must have an extension *.aln and should be in clustalw format as explained in the alignment rules.

Enter the name of the pdb file ?

Enter the pdb file name, must have an extension *.pdb and should have exactly the residues contained in the alignment file. Refer to creating pdb file rules.

Enter interval (deg) for PSI/PHI (default 10):

Upper/lower intervals for dihedral angles.

Enter interval (deg) for OMEGA (default 5):

Upper/lower intervals for OMEGA dihedral angle. Note currently if the residues don't match in the alignment and if the template consists of CISPEP then manually one should edit the OMEGA angle (*ex.aco) appropriately and rerun *diarun.sh script.

Number of distance constraints per atom (default 15) :

Specify the number of constraints you would like to extract per atom. You can specify higher number if the length of the protein (alignment) is small. Higher number of the constraints will compromise the execution time of the program so do not exceed more than 30 unless you have 5-15 residue template fragment from which you want to extract constraints.

Enter threshold between upper and lower limits (default 0.5) :

Specify the threshold of upper and lower limits, increase threshold if you like to have more flexibility (generally the case if the %id between the template and target is low) and reduce it if the proteins are highly similar. Most cases the default value is fine.

Do you like to specify fragments ?

If you like to manually specify fragments to be used in the modeling then enter yes. But it is advised to use the automatic option for specifying the fragments so type 'n'.

Do you have another template to extract constraints from (y/n) ?

If you have multiple template specify yes. The start residue number corresponding to the master alignment should be noted down.

Enter the starting residue number of the target sequence in the alignment file ?

Enter the starting residue number. Next sequence of questions will be similar to the above.

If you answered no (by specifying 'n') for the question do you have another template to extract constraints from..... then

Do you have disulphide bonds in the target ?

If you have disulphide bonds (you must have already changed in the alignment 'C' to 'c', otherwise you must edit the alignment file and restart the program.

You will also be required to know the residue pairs involved in the CYSS bonds. If you answer 'y' then

No. of SS bonds ?

Specify number of disulphide bonds

Enter res. In 1 SSBND ? (RES1, RES2)

Enter residues involved in the 1st SSBOND with RES1 <RES2.

Do you want to run DIAMOD ?

If you say yes it will prepare input files for DIAMOD

How many structures do you want to generate ?

If you have used an alignment that were highly similar it is sufficient to generate 1 structure. If you have used approximate constraints or a template with very low similarity, multiple templates (you can generate just 1 structure, if it fails then generate more models) specify 20-50 structures. If you specify more than 1 structure then you may be asked to specify a random seed number (must be an odd number).

Enter how many cycles of SECODG do you want to run (1 if no SECODG)
?

This option will not have any meaning if you generate only one structure. If you generate multiple structure and have used approximate constraints you can run multiple cycles of DIAMOD.

7.6 Selecting crude model for energy minimization

If you have selected to generate 1 model then visualize it to see whether any mirror images, energetically impossible loops or crossings, complex meaningless topologies are present. If so generate more models by specifying suitable option or edit `*diarun.sh` file to generate more models and rerun it. If you have already generated more models, then check the file `*dia.ovw` to find the model with the least target function. Use this as the starting structure for FANTOM minimization. Currently FANTOM minimization is not included in the MPACK. One must create FANTOM minimization files or alternatively use perl script `fantomminimize.pl` that is supplied with the package. Finally to run energy minimization you must execute `fan_run.sh` that is created by `fantomminimize.pl`. You must monitor the output structure written at different stages of minimization. For a good model the conformational energy must be negative and the gradient should be close to '0'.

7.7 Output of MPACK

The output files will be changed in the higher version. So please refer to this section for any changes.

If the core file name given is 'XXXX' then the following output files will be produced

Let the following be the input files:

`align1.aln` (alignment file)

`align1.pdb` (template co-ordinate file)

Out put files:

`XXXXMPACK.LOG` (LOG file containing details of program execution, MOLMOL scripts)

`XXXX_Yex.aln` (Reformatted alignment file, for the 1st alignment ie master align Y=1)

`alignYre.pdb` (Renumbered pdb file to be used for MOLMOL, Y corresponds to the align)

XXXXexdisrun.sh (Script to run exdis, only for the master alignment)

XXXXex.aco (Angle constraint file)

XXXXex.upl (upper limit file)

XXXXex.lol (lower limit file)

XXXXex.ang (approximate starting structure for DIAMOD)

XXXXdia.min (minimization file for DIAMOD)

XXXXdiarun.sh (Script to execute DIAMOD)

XXXXdiaY.clol (Corrected lower limit file valid if you use SECODG > 1 option, Y is the round)

XXXXdiaY.cupl (Corrected upper limit file valid if you use SECODG > 1 option, Y is the round)

XXXXdiaY.lol (modified lower limit file in round Y)

XXXXdiaY.upl (modified upper limit file in round Y)

XXXXdiaY.ovw (Overview file containing target function for the round Y)

XXXXdiaY.res (result, minimization details of DIAMOD cycle run Y)

XXXXdiaYZZZ.cor (DG coordinate file for the model number ZZZ generated in cycle Y)

XXXXdiaYZZZ.ang (ANG file for the model number ZZZ generated in the cycle Y)

Mscr_targ.seq (FASTA format of target sequence)

7.8 Running MOLMOL to visualize model

The log file contains a MOLMOL script to visualize, overlay the models and the templates. It uses some default names to identify the model name and template name.

To extract the script from log file type this on the command line

```
$UNIX> grep "MOLMOLSCRIPT" *.LOG | awk '{print substr($0,23,length($0));}' > mol.mac
```

Some files has to be renamed or converted. If you are using a DIAMOD DG cor file then you may need to use `cofima` and convert it to PDB file `model.pdb`. If you want to view a specific model in the FANTOM output you must simply rename (better to copy) into a file name `model.pdb`. Start MOLMOL. Use `Edit` menu and select `InitAll` option. Then in the `File` menu click `Macro -> Execute User`. Please refer to MOMOL manual and the script to clearly understand what is going on.

7.9 Evaluating the model

Calculate RMSD using MOLMOL (you can use `Fit` command to do this), calculate surface area using `GETAREA` (can be used for ranking models), `PROCHECK` to perform RAMACHANDRAN Plots and other standard checks.

8 EXAMPLES

8.1 Example 1

This example demonstrates developing a model for C20 protein from Chicken. There are a total of 166 residues. The sequence corresponds to structural domain located at residues 179-344 in the full length protein (accession number Q91006 in TrEMBL). There are no SSBONDS. This sequence is T0164 in CASP5 competition (<http://predictioncenter.llnl.gov>).

Searching for suitable template:

Target sequence

```
>T0164 C20, chicken
NSTDVEETLKRIQNNDPDL EEVNLNNIMNIPVPTLKACAEALKTNTYVKKFSIVGTRSND
PVAFALAEMLKVNNTLKS LNVE SNFISGSGILALVEALQSNTSLIELRIDNQSQPLGNV
EMEIANMLEKNTTLLKFGYHFTQQGPRLRASNAMMNNNDLVRKRRL
```

Submit to different fold-recognition servers using METASERVER. All the servers gave (1DFJ:I) as the best hit. For our modeling we will use the output from SAM-T99 server. Alignment from SAM-T99 (reformatted to conform to the alignment formatting rules for MPACK and the file is in align1.aln)

```
CLUSTAL W (1.7) multiple sequence alignment
```

```
T0164          NSTDVEETLKRIQNNDPDL EEVNLNNIMNIPVPTLKACAEALKTNTY-VKKFSIVGTRSND
1DFJ          ITASGCRDLRCRVLQAKETLKELSLAG-NKLGDEGARLLCESLLQPGCQLES LWKSCSLT

T0164          DPVAFALAEMLKVNNTLKS LNVE SNFISGSGILALVEALQSNTS-LIELRIDNQSQPLGN
1DFJ          AACCQHVSLMLTQNKHLLLELQLSSNKLGD SGIQELCQALSQPGTTLRVLCLGDCEV--TN

T0164          NVEMEIANMLEKNTTLLKFGYHFTQQGPRLRASNAMMNNNDLVRKRRL
1DFJ          SGCSSLASLLL ANRSLRELDLSNNCVGDPGV LQLLGSLEQPGCALEQL
```

PDB file for 1DFJ was obtained from <http://www.rcsb.org>. The file was hand edited such that only chain I is present, and the residues in the alignment file correspond to the PDB format rules for MPACK. (The file is in align1.pdb)

```
Start MPACK:
```

```
$unix_born > mpack
MPACK v 0.5 3D MODELING PACKAGE FOR PROTEINS
*****
Computational Biology Group, SCSB, UTMB, GALV
READ INSTRUCTIONS BEFORE USE REPORT ERRORS TO
    venkat@bohr.utmb.edu
    http://129.109.73.53/MPACK/
*****
```

```
Type core name of the output files : modell
Do you have topology file (y/n) ?n
```


Do you have a template to extract constraints from ?y
 Name of the seq. align. file (This must have all target sequence) ? align1.aln
 Enter the name of the pdb file ? align1.pdb
 Enter interval (deg) for PSI/PHI (default 10) :
 Enter interval (deg) for OMEGA (default 5) :
 Number of distance constraints per atom (default 15) : 20
 Enter threshold between upper and lower limits (default 0.5):
 Do you like to specify fragments (y/n) ?n
 STOP 0 statement executed
 Do you have another template to extract constraints from (y/n) ? n

#####EXDIS COMPLETED#####

Do you have disulphide bonds in the target ? n
 Do you want to run DIAMOD ? y
 How many structures do you want to generate ? 1
 Enter how many cycles of SECODG do you want to run (1 if no SECODG) ?1
 EXECUTING DIAMOD

DIAMOD: Distance geometry Algorithm for MODelling
 (UNIX version 2.2, double precision)

.....

After the DIAMOD run MPACK produces a LOG file. Contents of the LOG file
 (modellMPACK.log)

```
#COMMENT MPACK V1.0 LOG FILE
#COMMENT EXDIS INPUT PARAM TO EXDIS
#COMMENT EXDIS_COREFILE model5
#COMMENT EXDIS_RUN1 PSI/PHI 10
#COMMENT EXDIS_RUN1 OMEGA 5
#COMMENT EXDIS_RUN1 DISTANCE_CONSTRAINT 20
#COMMENT EXDIS_RUN1 THRESHOLD 0.5
#COMMENT MOLMOLSCRIPT ReadPdb model.pdb
#COMMENT MOLMOLSCRIPT ReadPdb align1.pdb
#COMMENT MOLMOLSCRIPT DefPropAtom 'ab1' '#1:2-25,29-46,49-102,105-113,118-166 & bb'
#COMMENT MOLMOLSCRIPT DefPropAtom 'ab2' '#2:2-25,28-45,49-102,106-114,117-165 & bb'
#COMMENT MOLMOLSCRIPT Fit to first 'ab1 ab2'
#COMMENT MOLMOLSCRIPT SelectMol 'num = 1'
#COMMENT MOLMOLSCRIPT ColorBond 1.000 0.000 0.000
#COMMENT MOLMOLSCRIPT SelectMol 'num = 2'
#COMMENT MOLMOLSCRIPT ColorBond 0.000 0.000 1.000
#COMMENT EXECUTING EXDIS
#COMMENT EXDIS_SYS *****
#COMMENT EXDIS_SYS * EXDIS 1.01 *
#COMMENT EXDIS_SYS *****
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Type 'help' for commands
#COMMENT EXDIS_SYS exdis 1.01> Enter "f" for file mode or "c" for command mode:
#COMMENT EXDIS_SYS Sequence alignment file: Target : target
#COMMENT EXDIS_SYS Template: template
#COMMENT EXDIS_SYS Template sequence contains 165 residues
#COMMENT EXDIS_SYS PDB file of the template molecule: Warning, non-sequential
#COMMENT EXDIS_SYS Warning, non-sequential numbering of residues in PDB file
#COMMENT EXDIS_SYS 1211 atoms, 165 residues.
#COMMENT EXDIS_SYS ATOM 1 N ILE 1 5.956 49.302 27.100
#COMMENT EXDIS_SYS ATOM 1211 CD2 LEU 165 5.545 42.110 -5.780
#COMMENT EXDIS_SYS
```

```

#COMMENT EXDIS_SYS Target contains 166 residues
#COMMENT EXDIS_SYS Define fragments of model sequence that will be
#COMMENT EXDIS_SYS used to generate dihedral and distance constraints.
#COMMENT EXDIS_SYS Format: residue_number <space> residue_number
#COMMENT EXDIS_SYS Type "0 0" to take entire sequence into account,
#COMMENT EXDIS_SYS or to end fragment definition
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Enter pairs of numbers:
#COMMENT EXDIS_SYS Type core name of the output file.
#COMMENT EXDIS_SYS Default extensions will be provided by program: *Interval for #COMMENT
#COMMENT EXDIS_SYS Initial angle file model5ex.ang
#COMMENT EXDIS_SYS Initial angle file model5ex.seq
#COMMENT EXDIS_SYS Number of distance constraints per atom: Distance constraint
#COMMENT EXDIS_SYS Distance constraints file model5ex.lol
#COMMENT EXDIS_SYS 16029 distance constraints written
#COMMENT EXDIS_SYS exdis 1.01>
#COMMENT MPACK GENERATING MIN FILE FOR DIAMOD
#COMMENT MPACK GENERATING COM FILE FOR DIAMOD
#COMMENT MPACK EXECUTING DIAMOD

```

To visualize the diamod model copy the MOLMOLSCRIPT in the LOG file in a file mol.mac,

contents in the file should be:

```

ReadPdb model.pdb
ReadPdb align1.pdb
DefPropAtom 'ab1' '#1:2-25,29-46,49-102,105-113,118-166 & bb'
DefPropAtom 'ab2' '#2:2-25,28-45,49-102,106-114,117-165 & bb'
Fit to_first 'ab1 ab2'
SelectMol 'num = 1'
ColorBond 1.000 0.000 0.000
SelectMol 'num = 2'
ColorBond 0.000 0.000 1.000

```

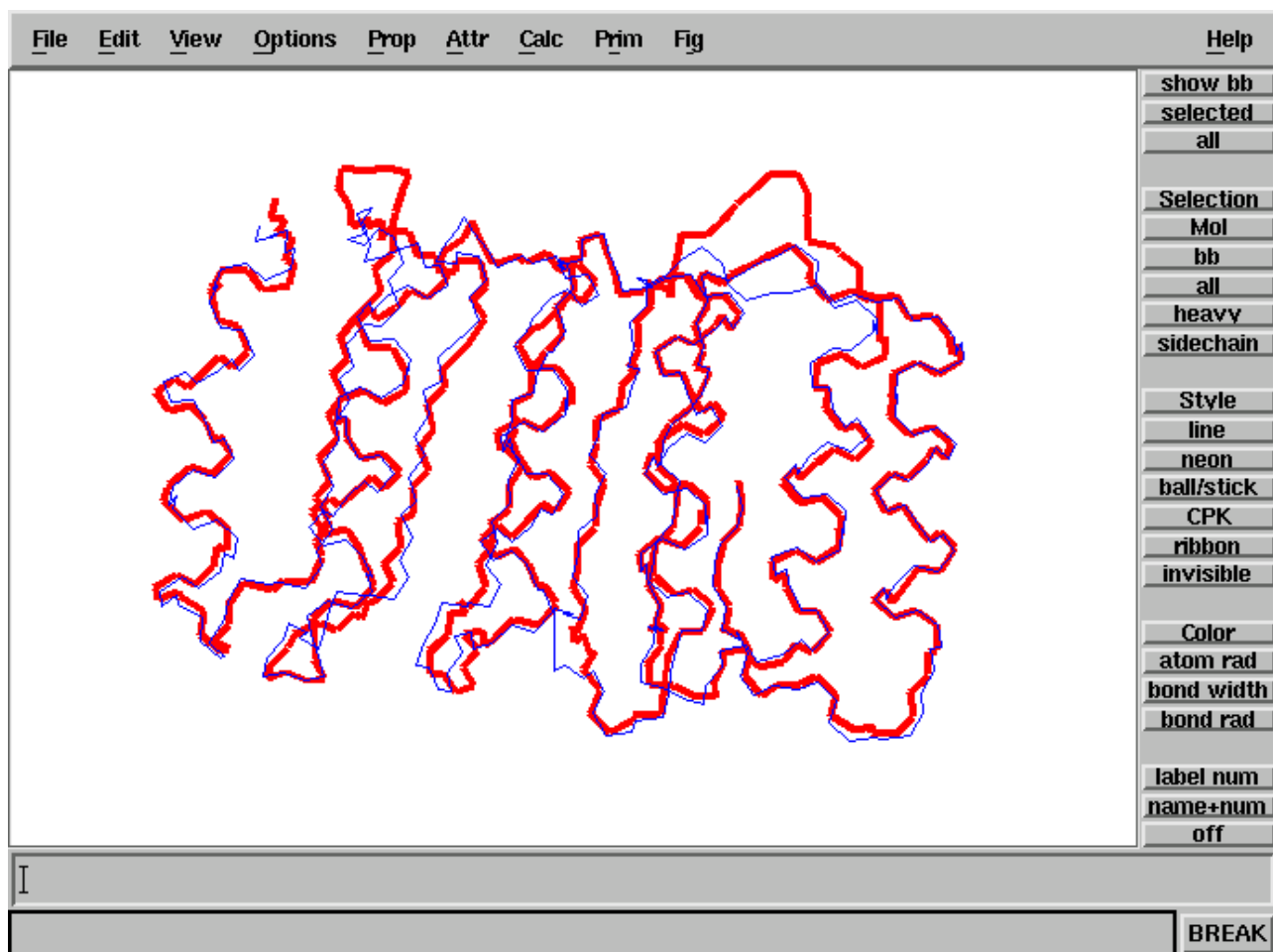
Use cofima to convert the file model1dia1001.cor into pdb. Follow the example

```

$unix>cofima
$unix>read model1dia1001.cor
$unix>writepdb model.pdb
$unix>quit

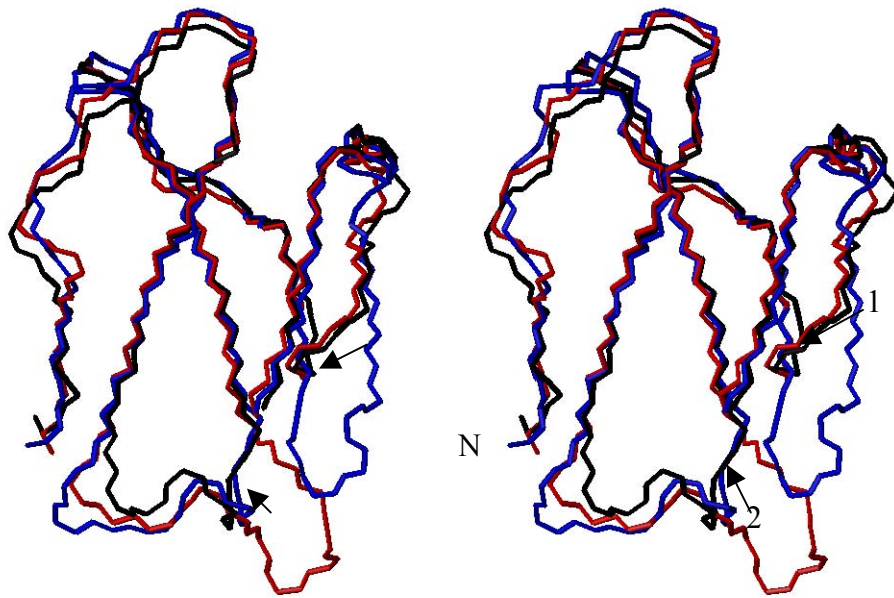
```

This will write a file called model.pdb. Now execute MOLMOL and execute the macro mol.mac. The model will be over layered with the template. The model will be in red color and the template in the blue color.

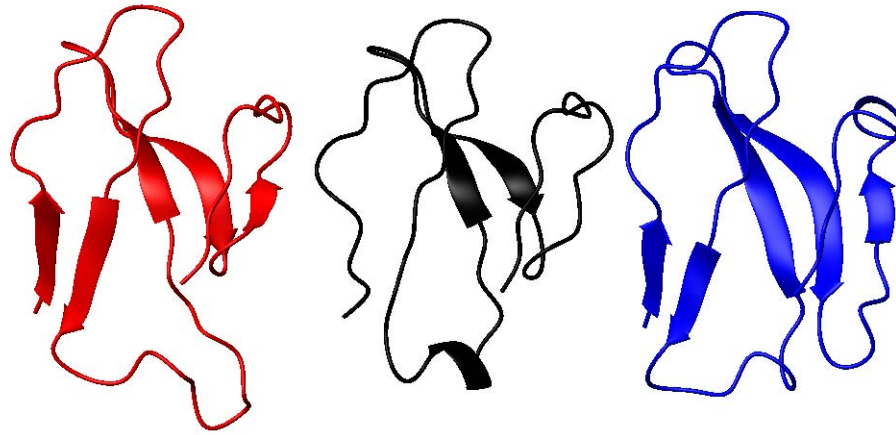


8.2 Example 2

In this example you want to use two templates to create a model. To have a feel for the modeling tools we will create a model for a known structure based on a closely related template. The model will be created for human myeloma Immunoglobulin IgG1 HIL (3-66) pdb file 8FAB, using templates Fab fragment of IgG for *Neisseria meningitidis* 1QKZ and anti-prion Fab 1CU4. Since the structure of all the protein is known a structure comparison is shown below (black is 1FAB (3-66), red is 1QKZL (3-68), blue is 1CU4H (3-71)).



Stereoview of overlay of protein structures. The black will be the model to be developed based on the blue and red as template. Difference in the structures are indicated by two arrows. In arrow 1 (blue looped out) region we will use 1QKZ as the template and in the arrow 2 (red looped out) region we will use 1CU4 as the template.



Ribbon diagram of the three structures. Shown in the black will be the model (actual structure) to be created using structures shown in the red and blue.

In order to model this we need to create alignments based on the two template. Simple logic includes create alignments in those regions where the template and target has good fit and also have a common region in each of the alignment so that two fragments can be oriented. There will be two alignment files in this case and the first alignment file is named as align1.aln and contains alignment for the region 1 indicated by arrow in the picture based on 1QKZ. The first alignment is the master alignment hence should contain the entire target residues.

CLUSTAL W (1.7) multiple sequence alignment

```

8FAB          ELTQPPSVSVSPGQTARITCSANALPN-----
1QKZ          -----VMTQTPLSLPVSLGDQASISCRSSQSLVHSNGN

8FAB          QYAYWYQQKPGRAPVMVIYKDTQRPSGIPQRFSSST
1QKZ          TYLHWYLQKPGQSPKLLIYTVSNRFGVDPDRFSGSG

```

Corresponding pdb file that consists of coordinates exactly as in the alignment for the template file is also created and named as align1.pdb. The second alignment is based on 1CU4 (indicated in the arrow region 2). This will contain a fragment of the template to be modeled starting from residue 1. This file is called as align2.aln and the contents are

```
CLUSTAL W (1.7) multiple sequence alignment
```

```
8FAB          ELTQPPSVSVSPGQTARITCSA--NALPNQYAYWYQQKPGRAPVMVIYKD-----
1CU4H          KLQQSGAELVRSGASVKLSCTASGFNIKDYIYIQWVKQRPEQGLEWIGWIDPENGNSEYAP
```

```
8FAB          -----
1CU4H          RFQ GKATMTA
```

corresponding pdb file for 1CU4H is named as align2.pdb. MPACK run is shown below.

```
$unix_born > mpack
MPACK v 0.5 3D MODELING PACKAGE FOR PROTEINS
*****
Computational Biology Group, SCSB, UTMB, GALV
READ INSTRUCTIONS BEFORE USE REPORT ERRORS TO
    venkat@bohr.utmb.edu
    http://129.109.73.53/MPACK/
*****

Type core name of the output files : modell
Do you have topology file (y/n) ?n
Do you have a template to extract constraints from ?y
Name of the seq. align. file (This must have all target sequence) ? align1.aln
Enter the name of the pdb file ? align1.pdb
Enter interval (deg) for PSI/PHI (default 10) :
Enter interval (deg) for OMEGA (default 5) :
Number of distance constraints per atom (default 15) : 30
Enter threshold between upper and lower limits (default 0.5):
Do you like to specify fragments (y/n) ?n
STOP 0 statement executed
Do you have another template to extract constraints from (y/n) ? y
Enter the starting residue number of the target sequence in the alignment file?1
Enter the name of the sequence alignment file ?align2.aln
Enter the name of the pdb file ? align2.pdb
    Enter interval (deg) for PSI/PHI (default 10) :
    Enter interval (deg) for OMEGA (default 5) :
Number of distance constraints per atom (default 15) : 30
Enter threshold between upper and lower limits (default 0.5):
Do you like to specify fragments (y/n) ?n
STOP 0 statement executed
Do you have another template to extract constraints from (y/n) ?n

#####EXDIS COMPLETED#####
```

Do you have disulphide bonds in the target ? n
 Do you want to run DIAMOD ? y
 How many structures do you want to generate ? 1
 Enter how many cycles of SECODG do you want to run (1 if no SECODG) ?1
 EXECUTING DIAMOD

DIAMOD: DIstance geometry Algorithm for MOdelling
 (UNIX version 2.2, double precision)

.....

After the DIAMOD run MPACK produces a LOG file. Contents of the LOG file
 (modellMPACK.log)

```
#COMMENT MPACK V1.0 LOG FILE
#COMMENT EXDIS INPUT PARAM TO EXDIS
#COMMENT EXDIS_COREFILE modell
#COMMENT EXDIS_RUN1 PSI/PHI 10
#COMMENT EXDIS_RUN1 OMEGA 5
#COMMENT EXDIS_RUN1 DISTANCE_CONSTRAINT 30
#COMMENT EXDIS_RUN1 THRESHOLD 0.5
#COMMENT MOLMOLSCRIPT ReadPdb model.pdb
#COMMENT MOLMOLSCRIPT ReadPdb align1.pdb
#COMMENT MOLMOLSCRIPT DefPropAtom 'ab1' '#1:29-63 & bb'
#COMMENT MOLMOLSCRIPT DefPropAtom 'ab2' '#2:35-69 & bb'
#COMMENT MOLMOLSCRIPT Fit to_first 'ab1 ab2'
#COMMENT MOLMOLSCRIPT SelectMol 'num = 1'
#COMMENT MOLMOLSCRIPT ColorBond 1.000 0.000 0.000
#COMMENT MOLMOLSCRIPT SelectMol 'num = 2'
#COMMENT MOLMOLSCRIPT ColorBond 0.000 0.000 1.000
#COMMENT EXECUTING EXDIS
#COMMENT EXDIS_SYS *****
#COMMENT EXDIS_SYS * EXDIS 1.01 *
#COMMENT EXDIS_SYS *****
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Type 'help' for commands
#COMMENT EXDIS_SYS exdis 1.01>Enter "f" for file mode or "c" for command mode:
#COMMENT EXDIS_SYS Sequence alignment file: Target : target
#COMMENT EXDIS_SYS Template: template
#COMMENT EXDIS_SYS Template sequence contains 69 residues
#COMMENT EXDIS_SYS PDB file of the template molecule: Warning, non-sequential
#COMMENT EXDIS_SYS 527 atoms, 69 residues.
#COMMENT EXDIS_SYS ATOM 1 N VAL 1 21.149 23.606 27.397
#COMMENT EXDIS_SYS ATOM 527 O GLY 69 18.843 25.822 43.908
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Target contains 63 residues
#COMMENT EXDIS_SYS Define fragments of model sequence that will be
#COMMENT EXDIS_SYS used to generate dihedral and distance constraints.
#COMMENT EXDIS_SYS Format: residue_number <space> residue_number
#COMMENT EXDIS_SYS Type "0 0" to take entire sequence into account,
#COMMENT EXDIS_SYS or to end fragment definition
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Enter pairs of numbers:
```

```

#COMMENT EXDIS_SYS Type core name of the output file.
#COMMENT EXDIS_SYS Default extensions will be provided by program: *Interval for
#COMMENT EXDIS_SYS Enter interval (deg) for PSI/PHI,OMEGA: Dihedral angle
#COMMENT EXDIS_SYS Initial angle file modellex.ang
#COMMENT EXDIS_SYS Initial angle file modellex.seq
#COMMENT EXDIS_SYS Number of distance constraints per atom: Distance constraint
#COMMENT EXDIS_SYS Distance constraints file modellex.lol
#COMMENT EXDIS_SYS 7184 distance constraints written
#COMMENT EXDIS_SYS exdis 1.01> #COMMENT SUBFRAG_MPACK V1.0 LOG FILE
#COMMENT SUBFRAG_INPUT PARAM TO EXDIS
#COMMENT COREFILESUBFRAG 6945mptemp
#COMMENT EXDIS_RUN2 PSI/PHI 10
#COMMENT EXDIS_RUN2 OMEGA 5
#COMMENT EXDIS_RUN2 DISTANCE_CONSTRAINT 30
#COMMENT EXDIS_RUN2 THRESHOLD 0.5
#COMMENT MOLMOLSCRIPT ReadPdb align2.pdb
#COMMENT MOLMOLSCRIPT DefPropAtom 'ab3' '#1:2-21,24-47 & bb'
#COMMENT MOLMOLSCRIPT DefPropAtom 'ab4' '#3:2-21,26-49 & bb'
#COMMENT MOLMOLSCRIPT SelectMol ''
#COMMENT MOLMOLSCRIPT Fit to_first 'ab3 ab4'
#COMMENT MOLMOLSCRIPT SelectMol 'num =1'
#COMMENT MOLMOLSCRIPT ColorBond 1.000 0.000 0.000
#COMMENT MOLMOLSCRIPT SelectMol 'num =3'
#COMMENT MOLMOLSCRIPT ColorBond 0.000 0.000 1.000
#COMMENTSUBFRAG EXECUTING EXDIS
#COMMENT EXDIS_SYS *****
#COMMENT EXDIS_SYS * EXDIS 1.01 *
#COMMENT EXDIS_SYS *****
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Type 'help' for commands
#COMMENT EXDIS_SYS exdis 1.01> Enter "f" for file mode or "c" for command mode:
#COMMENT EXDIS_SYS Sequence alignment file: Target : target
#COMMENT EXDIS_SYS Template: template
#COMMENT EXDIS_SYS Template sequence contains 70 residues
#COMMENT EXDIS_SYS PDB file of the template molecule: Warning, possible residue
#COMMENT EXDIS_SYS Delete extra residue entry from PDB file and rerun if
#COMMENT EXDIS_SYS Warning, non-sequential numbering of atoms in PDB file
#COMMENT EXDIS_SYS Warning, non-sequential numbering of residues in PDB file
#COMMENT EXDIS_SYS 552 atoms, 70 residues.
#COMMENT EXDIS_SYS ATOM 1 N LYS 1 9.663 -11.166 49.137
#COMMENT EXDIS_SYS ATOM 552 CB ALA 70 21.599 -10.629 37.678
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Target contains 48 residues
#COMMENT EXDIS_SYS Define fragments of model sequence that will be
#COMMENT EXDIS_SYS used to generate dihedral and distance constraints.
#COMMENT EXDIS_SYS Format: residue_number <space> residue_number
#COMMENT EXDIS_SYS Type "0 0" to take entire sequence into account,
#COMMENT EXDIS_SYS or to end fragment definition
#COMMENT EXDIS_SYS
#COMMENT EXDIS_SYS Enter pairs of numbers:
#COMMENT EXDIS_SYS Type core name of the output file.
#COMMENT EXDIS_SYS Default extensions will be provided by program: *Interval for
#COMMENT EXDIS_SYS Enter interval (deg) for PSI/PHI,OMEGA: Dihedral angle
#COMMENT EXDIS_SYS Initial angle file 6945mptempex.ang
#COMMENT EXDIS_SYS Initial angle file 6945mptempex.seq
#COMMENT EXDIS_SYS Number of distance constraints per atom: Distance constraint
#COMMENT EXDIS_SYS Distance constraints file 6945mptempex.lol

```

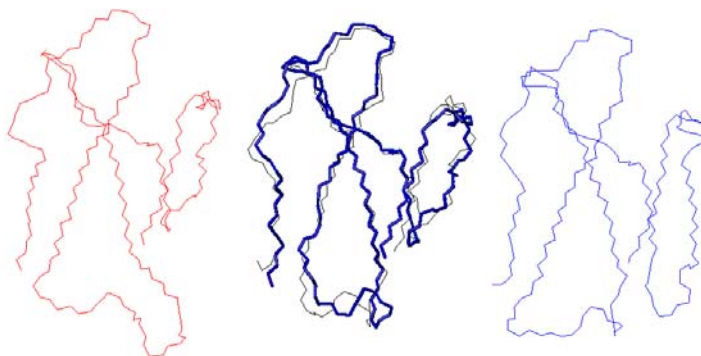


```
#COMMENT EXDIS_SYS 8868 distance constraints written
#COMMENT EXDIS_SYS exdis 1.01>
#COMMENT SUBFRAG NEW CONSTRAINTS ADDED FROM 0
#COMMENT MPACK GENERATING MIN FILE FOR DIAMOD
#COMMENT MPACK GENERATING COM FILE FOR DIAMOD
#COMMENT MPACK EXECUTING DIAMOD
```

In this case model produced after DIAMOD step is reformatted to PDB and renamed as model.pdb using COFIMA. The script to visualize the model and template is shown below.

```
ReadPdb model.pdb
ReadPdb align1.pdb
DefPropAtom 'ab1' '#1:29-63 & bb'
DefPropAtom 'ab2' '#2:35-69 & bb'
Fit to_first 'ab1 ab2'
SelectMol 'num = 1'
ColorBond 1.000 0.000 0.000
SelectMol 'num = 2'
ColorBond 0.000 0.000 1.000
ReadPdb align2.pdb
DefPropAtom 'ab3' '#1:2-21,24-47 & bb'
DefPropAtom 'ab4' '#3:2-21,26-49 & bb'
SelectMol ''
Fit to_first 'ab3 ab4'
SelectMol 'num =1'
ColorBond 1.000 0.000 0.000
SelectMol 'num =3'
ColorBond 0.000 0.000 1.000
```

The rmsd between the model and the original structure is 1.2A. This should be further energy minimized. To know how to energy minimize please refer to FANTOM manual.



Shown in red is 1QKZ on the left, in the middle is the model overlaid to the actual structure (model is in blue neon representation and the actual structure is shown in black wireframe). On the right is 1CU4 shown in light blue color. Current MPACK comes with a single example. The directions to use is present in the MPACK installation directory ./examples/

9 MPACK program details

The following files are present in the distribution.

MPACK specific:

INSTALL.README Installation directions

Install.sh	Installation script.
mpack0.7.pl	Perl file containing routines for MPACK
path.incl	A path include file created during installation
mpack	Executable created during installation
/doc	This file
/example	Example program with directions

COMPLETE LISTING (To obtain more details of each file refer to FANTOM, DYANA manuals):

```
mpack:
total 1380
-rw-r--r-- 1 venkat user 218 Feb 24 02:19 INSTALL.README
-rw-r--r-- 1 venkat user 1254 Feb 24 02:19 Install.sh
-rw----- 1 venkat user 4274 Feb 24 02:19 Makefile
-rw-r--r-- 1 venkat user 2250 Feb 24 02:19 aix.f
-rwxr-xr-x 1 venkat user 5025 Feb 24 02:19 alignrep.pl*
-rw----- 1 venkat user 25811 Feb 24 02:19 analyse.f
-rw----- 1 venkat user 1055 Feb 24 02:19 analyse.incl
-rw-r--r-- 1 venkat user 420 Feb 24 02:19 angle.incl
-rw-r--r-- 1 venkat user 6297 Feb 24 02:19 congrd.f
-rw-r--r-- 1 venkat user 53916 Feb 24 02:19 construct.f
-rw-r--r-- 1 venkat user 788 Feb 24 02:19 convert.f
-rw-r--r-- 1 venkat user 2067 Feb 24 02:19 cray.f
-rw-r--r-- 1 venkat user 20424 Feb 24 02:19 deriv.f
-rw-r--r-- 1 venkat user 231449 Feb 24 02:19 dia_01.f
-rw-r--r-- 1 venkat user 231152 Feb 24 02:19 diamod.f
drwxr-xr-x 2 venkat user 4096 Feb 24 02:19 doc/
-rw-r--r-- 1 venkat user 712 Feb 24 02:19 e_lib.incl
-rw-r--r-- 1 venkat user 416 Feb 24 02:19 e_pdb.incl
-rw-r--r-- 1 venkat user 10024 Feb 24 02:19 ecblk.f
-rw-r--r-- 1 venkat user 535 Feb 24 02:19 ecepp_par.incl
-rw-r--r-- 1 venkat user 18414 Feb 24 02:19 energy.f
```

```

-rw-r--r--      1 venkat  user      2963 Feb 24 02:19 erg.incl
drwxr-xr-x      2 venkat  user     4096 Feb 24 02:31 examples/
-rw-r--r--      1 venkat  user     7623 Feb 24 02:19 exdis.f
-rw-r--r--      1 venkat  user      942 Feb 24 02:19 exdis.incl
-rw-r--r--      1 venkat  user    88406 Feb 24 02:19 phantom.f
-rw-r--r--      1 venkat  user    16163 Feb 24 02:19 global.incl
-rw-r--r--      1 venkat  user    16166 Feb 24 02:19 globaldia.incl
-rw-r--r--      1 venkat  user     6534 Feb 24 02:19 grid.f
-rw-r--r--      1 venkat  user      298 Feb 24 02:19 grid.incl
-rw-r--r--      1 venkat  user    31013 Feb 24 02:19 homology.f
-rw-r--r--      1 venkat  user     1088 Feb 24 02:19 homology.incl
-rw-r--r--      1 venkat  user    55618 Feb 24 02:19 hydro.f
-rw-r--r--      1 venkat  user     2005 Feb 24 02:19 hydro.incl
-rw-r--r--      1 venkat  user    18066 Feb 24 02:19 in.f
-rw-r--r--      1 venkat  user     1221 Feb 24 02:19 io.f
-rw-r--r--      1 venkat  user      511 Feb 24 02:19 lib.incl
drwxr-xr-x      2 venkat  user     4096 Feb 24 02:19 library/
-rw-r--r--      1 venkat  user    13650 Feb 24 02:19 locon.f
-rw-r--r--      1 venkat  user      780 Feb 24 02:19 locon.incl
-rw-r--r--      1 venkat  user     3947 Feb 24 02:19 modch.f
-rw-r--r--      1 venkat  user    10397 Feb 24 02:19 mol.incl
-rw-r--r--      1 venkat  user    40294 Feb 24 02:19 monlo.f
-rw-r--r--      1 venkat  user     3705 Feb 24 02:19 monlo.incl
-rw-r--r--      1 venkat  user    34107 Feb 24 02:36 mpack0.7.pl
-rw-r--r--      1 venkat  user     1418 Feb 24 02:19 nec.f
-rw-r--r--      1 venkat  user     1967 Feb 24 02:19 newpar.incl
-rw-r--r--      1 venkat  user    11852 Feb 24 02:19 newton.f
-rw-r--r--      1 venkat  user    18599 Feb 24 02:19 numrec.f
-rw-r--r--      1 venkat  user    27734 Feb 24 02:19 out.f
-rw-r--r--      1 venkat  user      139 Feb 24 02:19 path1.incl
-rw-r--r--      1 venkat  user    12312 Feb 24 02:19 pdb.f
-rw-r--r--      1 venkat  user      410 Feb 24 02:19 pdb.incl
-rw-r--r--      1 venkat  user     1199 Feb 24 02:19 precis.double
-rw-r--r--      1 venkat  user     1225 Feb 24 02:19 precis.incl
-rw-r--r--      1 venkat  user     1158 Feb 24 02:19 precis.single
-rw-r--r--      1 venkat  user      354 Feb 24 02:19 precisdia.incl
-rw-r--r--      1 venkat  user     3534 Feb 24 02:19 qcksrt.f
-rw-r--r--      1 venkat  user      944 Feb 24 02:19 random.f
-rw-r--r--      1 venkat  user      337 Feb 24 02:19 random.incl
-rw-r--r--      1 venkat  user      303 Feb 24 02:19 rest_list.incl
-rw-r--r--      1 venkat  user     1052 Feb 24 02:19 rest_par.incl
-rw-r--r--      1 venkat  user    34607 Feb 24 02:19 restraint.f
-rw-r--r--      1 venkat  user     2320 Feb 24 02:19 routines.f
-rw-r--r--      1 venkat  user    10241 Feb 24 02:19 string.f
-rw-r--r--      1 venkat  user    19692 Feb 24 02:19 target.f
-rw-r--r--      1 venkat  user     7282 Feb 24 02:19 test.f
-rw-r--r--      1 venkat  user     4821 Feb 24 02:19 trl.par
-rw-r--r--      1 venkat  user    49312 Feb 24 02:19 trl_01.f
-rw-r--r--      1 venkat  user     2261 Feb 24 02:19 unix.f
-rw-r--r--      1 venkat  user     2211 Feb 24 02:19 update.f
-rw-r--r--      1 venkat  user      613 Feb 24 02:19 user.f
-rw-r--r--      1 venkat  user     1608 Feb 24 02:19 vax.f
-rw-r--r--      1 venkat  user      425 Feb 24 02:19 vec_list.incl
-rw-r--r--      1 venkat  user     2091 Feb 24 02:19 vector.f
-rw-r--r--      1 venkat  user     6783 Feb 24 02:19 vector3d.f

```

mpack/doc:

```
total 260
-rw----- 1 venkat user 260652 Feb 24 02:19 MPACKguide.pdf
```

mpack/examples:

```
total 276
-rw-r--r-- 1 venkat user 471 Feb 24 02:31 README.txt
-rw-r--r-- 1 venkat user 635 Feb 24 02:19 align1.aln
-rw-r--r-- 1 venkat user 229552 Feb 24 02:19 align1.pdb
-rw-r--r-- 1 venkat user 114 Feb 24 02:19 align2.aln
-rw-r--r-- 1 venkat user 10941 Feb 24 02:19 align2.pdb
-rw-r--r-- 1 venkat user 126 Feb 24 02:19 align3.aln
-rw-r--r-- 1 venkat user 13911 Feb 24 02:19 align3.pdb
```

mpack/library:

```
total 808
-rw-r--r-- 1 venkat user 335 Feb 24 02:19 chil.lib
-rw-r--r-- 1 venkat user 109288 Feb 24 02:19 dec.lib
-rw-r--r-- 1 venkat user 80337 Feb 24 02:19 diamod.lib
-rw-r--r-- 1 venkat user 94624 Feb 24 02:19 diana.lib
-rw-r--r-- 1 venkat user 147817 Feb 24 02:19 dpec.lib
-rw-r--r-- 1 venkat user 57221 Feb 24 02:19 ec.lib
-rw-r--r-- 1 venkat user 81317 Feb 24 02:19 pec.lib
-rw-r--r-- 1 venkat user 47 Feb 24 02:19 small.lib
-rw-r--r-- 1 venkat user 528 Feb 24 02:19 sym.lib
-rw-r--r-- 1 venkat user 180715 Feb 24 02:19 temp
-rw-r--r-- 1 venkat user 3376 Feb 24 02:19 temp1
-rw-r--r-- 1 venkat user 270 Feb 24 02:19 trl_a.lib
-rw-r--r-- 1 venkat user 270 Feb 24 02:19 trl_b.lib
-rw-r--r-- 1 venkat user 281 Feb 24 02:19 trl_e.lib
-rw-r--r-- 1 venkat user 273 Feb 24 02:19 trl_q.lib
```

10 VERSION

Current version is 1.0.

11 AUTHOR

MPACK is only a user interface for the programming suite (EXDIS/DIAMOD/FANTOM/TRANSLATE). MPACK was written by Venkatarajan Mathura at University of Texas Medical Branch, Galveston. Other programs including EXDIS, DIAMOD, FANTOM and TRANSLATE is copyrighted by Prof. Werner Braun. Each of these software may require separate request to WB.

12 DISCLAIMER

The author does not provide any guarantee for this package. Any loss due to bugs and errors in the program at any case cannot be blamed. The author does not oblige for any support or service to any one.