

LAFIRE manual

(Version 2.6)

(For LAFIRE version 2.6)
January 5, 2008

Min Yao

Index

1. Introduction	3
2. Limitation of the current version	4
3. Installation	4
3-1. Installation of Lafire	4
3-2. Installation of Qt	5
4. Using Lafire	6
4-1. Data for Lafire	6
4-2. Preparing for executing Lafire	6
4-3. Executing Lafire (executing, stopping, restarting)	10
4-4. Output files of Lafire	11
4-5. How to read results of Lafire	13
5. Lafire contact	14
6. Error message and fixing	15
7. Simplified manual of Lafire_molview	17

1. Introduction

Lafire (Local-correlation-coefficient-based automatic fitting for refinement) is an automatic refinement system for protein crystallography. The system is designed to start from an initial model (approximate or partial model) and finish with the final structure including water molecules (Fig.1). Lafire uses the refinement programs CNS or REFMAC5.

At first step, Lafire confirms amino acids of the initial model based on the sequence file that may be provided by the program user, and displaces the incorrect amino acids of the model (assignment).

In the refinement loop, the program first builds the parts that are missing in the model or seriously misfitted to the electron density map calculated from the observed phases or to the $2F_o-F_c$, F_o-F_c maps. Then, correlation coefficient is calculated for each residue, and if the correlation is not good enough, the corresponding residue is subjected to the refitting routine.

Lafire also writes out the files of Rfree/Rwork factors and Ramachandran plot for the current model in order to monitor the progress of refinement. The Rfree/Rwork factor plot and Ramachandran plot can be viewed easily using graphics program Lafire_moleview.

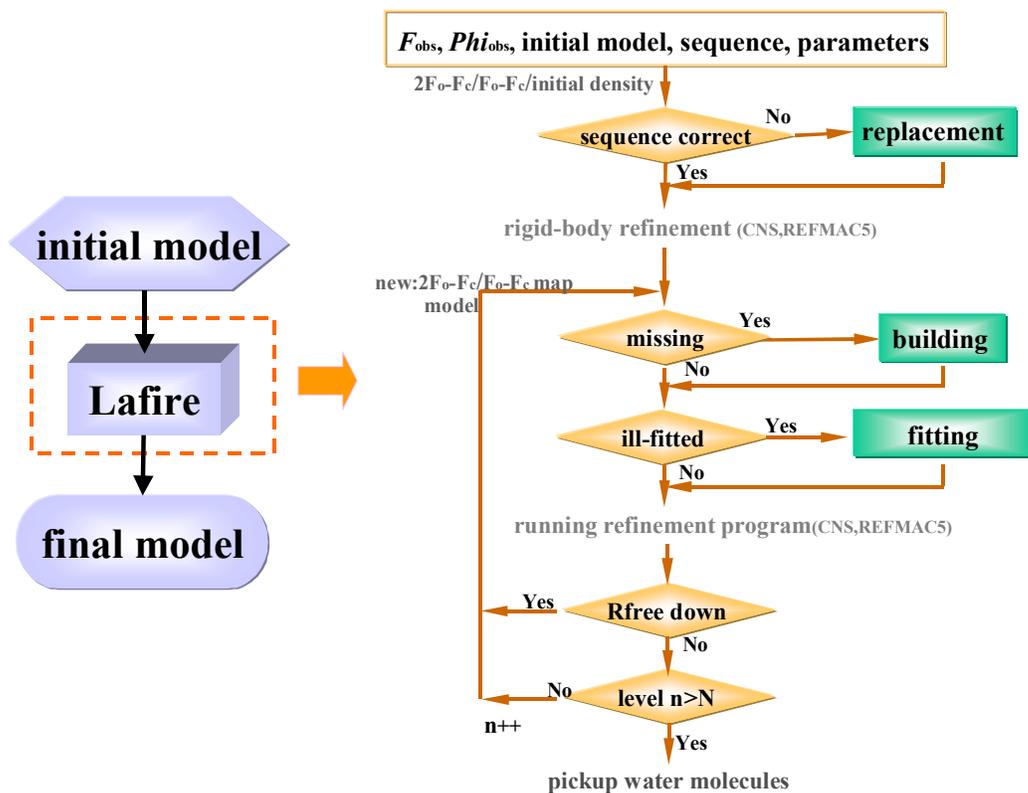


Figure 1. Automatic refinement using Lafire

2. Limitation of the current version

- 1). Since Lafire can not recognize amino acid residues at present, the residue number in the initial model must be consistent with that of the sequence file.
- 2). Lafire does not support for fitting or building ligands or nucleic acids at present.
- 3). Lafire does not support two-conformation.
- 4). Lafire does not support non-Pro cis-peptide.
- 5). Lafire can only use the refinement program CNS or REFMAC5.
- 6). Lafire do not generate the topology, parameter files, and patch of ligands automatically
- 7). Lafire only supports SGI (IRIS6.5 and after) and Linux (above RedHat 9.0 and SUSE10.0, Mandrake10.0).
- 8). The graphics program Lafire_moleview is used only for displaying the results of Lafire, and it was made using OpenGL and Qt. Lafire_moleview has been tested for SGI (IRIS6.5).

3. Installation

Lafire uses CCP4 program package and CNS program (If you want to use CNS program for refinement). Therefore, before installing Lafire, you should install CCP4 and CNS.

3-1. Installation of Lafire

3-1-1. Uncompressing Lafire

For example, if you want to install the Lafire at /home/xtal, please put Lafire.tar.gz into /home/xtal directory, then use `gunzip` and `tar` commands as below to uncompress the Lafire:

```
gunzip Lafire.tar.gz  
tar xf Lafire.tar
```

(or `gunzip Lafire.tar.gz | tar xf -`)

After uncompressing Lafire.tar.gz, the directory Lafire is made and four directories: **bin**, **client**, **lib**, **example** and one file of **configure** will be made at /home/xtal/Lafire directory.

3-1-2. Configuring the system for using Lafire

For checking environment required by Lafire, execute **configure** at first. The CCP4, CNS, Qt are

required by Lafire (Qt is only required for using graphic program Lafire_moleview).

./configure

3-1-3. Setting environment parameters for executing Lafire

Before executing Lafire, you have to set path and environment parameters and put them into .cshrc file. For example, if Lafire is installed under /home/xtal directory, please do as below:

```
setenv LAFIRE_DIR /home/xtal/Lafire (for Linux: export LAFIRE_DIR=/home/xtal/Lafire)  
set path=($path /home/xtal/Lafire/bin)  
alias cns_solve cnx_solve ( only for using CNX )
```

3-2. Installation of Qt library (only for using graphics program Lafire_moleview)

Qt is a graphical user interface library for C++. It is free software developed by Troll Tech of Norway. Normally, Qt is installed into /usr/local on the Linux RedHat9.0 (or newer than 9.0), while SGI machines don't have it in default situation. If you want to use graphic program Lafire_moleview, and there is not Qt library, please download and install Qt as following steps.

3-2-1. Download Qt

Download Qt from Web site <http://www.trolltech.com/download/qt/x11.html>, and uncompress it under /usr/local directory.

3-2-2. Setting environment parameters for compiling Qt

```
setenv QTDIR /usr/local/qt  
setenv PATH ${PATH}:${QTDIR}/bin  
setenv MANPATH ${MANPATH}:${QTDIR}/doc/man  
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${QTDIR}/lib  
setenv QMAKESPEC irix-g++  
(in the case of Linux, use ðexportð as ðsetenvð )
```

3-2-3. Compiling Qt

Configure system and compile Qt.

```
./configure -qt-gif -system-libpng -system-zlib -system-libjpeg  
./make
```

You may also see Web site for installing Qt
<http://paison.hp.infoseek.co.jp/paison/qt/qtinstall.html>

4. Using Lafire

Lafire requires a big size of memory and the size of memory is dependent on the size of the sample. Normally 0.5-1GB is required.

In the case of using ncs-restraint for refinement using CNS, Lafire modifies command-files in `~/Task_files` directory based on the file of `cns_configure` (see 4-2-5) to include ncs-restraint file. Preparation of ncs-restraint file is described on 4-2-8.

For using REFMAC5: `tls.inp` has to be defined and all files named as `refmac_*.com` in `~/Task_files` directory have to be modified by users.

If your model involves ligands such as ion, nucleic acids, small molecules, and you want to refine them with your parameters of topology and energy, you have to define and make the topology and energy parameter files (`*.top`, `*.param` files for CNS; `*.cif` file for REFMAC5), and put them into the directory of **Task_files** (see 4-2-8):

- (1). For using CNS, except for making topology and parameter files you need to define the name of topology and parameter files in the file `cns_configure` file.
- (2). For using REFMAC5, add the file name of `*.cif` file into all `refmac_*.inp` (in directory of **Task_files**).

4-1. Data for Lafire

- (1). Initial model (PDB file). (Residue number in the initial model file must be consistent with that of sequence file.)
- (2). `mtz` file with Fobs, Rfree-flag, and phases that was used for building initial model. (The phases are for using initial map in the cases of MAD/SAD/MIR, etc but not in MR method)
- (3). Sequence file (one character)
- (4). Second structure information (only for the case that you want to use the information of second structure)

4-2. Preparing for executing Lafire

4-2-1. Setting your sample directory for Lafire

First, use `öLafire_setupö` to setup your working directory. For example, when `/user/phoIDH/Lafire` is the working directory (Working-directory), then use command as

```
mkdir /users/phoIDH/Lafire
```

```
Lafire_setup /users/phoIDH/Lafire
```

After running `Lafire_setup`, the 5 directories (`Data`, `Pdb`, `Task_files`, `Log`, and `Result`) and 5 files (`lafire.inp`, `sequence.inp`, `helix.inp`, `sheet.inp`, and `cns_configure`) will be made (Figure2).

If you want to run example that is provided in Lafire program package, and the working directory is `öuser/phoIDH/Lafireö`, please use following command to setup

```
Lafire_setup /users/phoIDH/Lafire example
```

Then all data files of the example for running Lafire are setup under `/user/phoIDH/Lafire` directory. If you want to run Lafire using example data, you should modify the file of `phoIDH.inp` to give the name of working directory (see 4-2-7), then run Lafire (see 4-3-1).

/users/phoIDH/Lafire

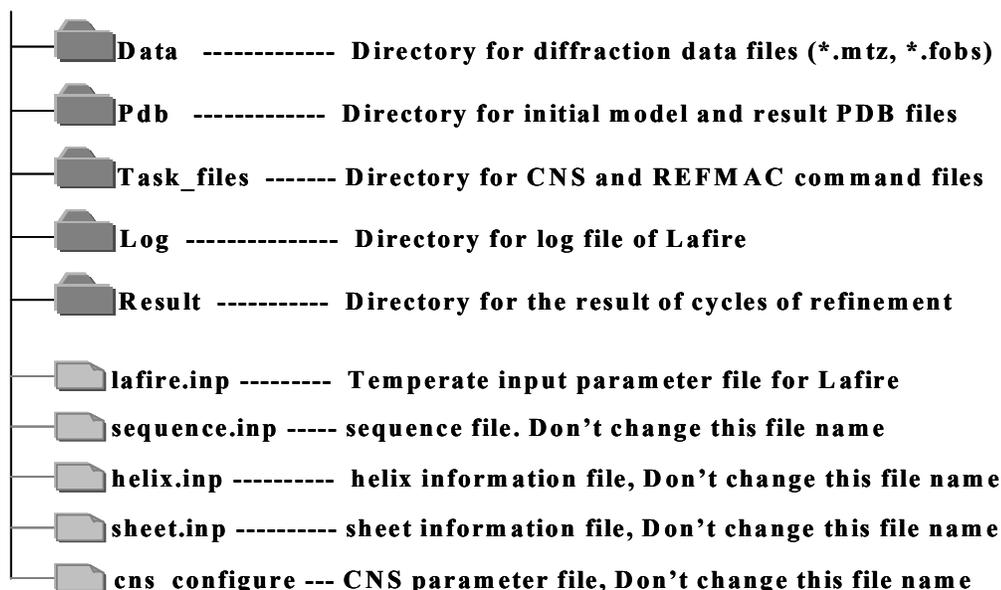


Figure 2. Lafire working directory

4-2-2. Preparing Diffraction data files

You should put diffraction data file (*.mtz) into directory of Working-directory /Data. You also have to set Rfree flag in your mtz file.

For using CNS, the diffraction data file of CNS-format will be created automatically under the directory of Working-directory /Data, and named `0lafire.fob0`. If user has already made diffraction data file of CNS-format, the file name of `0lafire.fob0` should be used and the file had to be move into the directory of Working-directory /Data. Also in this case, the mtz file has to be prepared from `0lafire.fob0`.

4-2-3. Preparing PDB file for initial model

You should make a PDB file for initial model under the directory that you have defined in Lafire parameter file (see 4-2-7). In the Lafire, the ligands such as DNA, RNA, or substrate, *et al*, could not be built or fitted, but it is possible to refine them by using refinement program CNS or REFMAC5.

- (1). Residue number of the initial model in PDB file must be same with that you have given in sequence file (sequence.inp) (see 4-2-4).
- (2). If there are multi molecules involved ligands such as nucleic acid, ion, substtate, inhibitor *et al*., you have to add the `0END0` or `0TER0` at the end of each molecule.

For using CNS, Lafire will name the segid of proteins as PROA, PROB, í ..., PROY, PROZ; the segid of DNA as DNAA, DNAB,í DNAY, DNAZ; the segid of RNA as RNAA, RNAB, í , RNAY, RNAZ; the segid of others as LIDA, LIDB,í , LIDY, LIDZ.

For using REMAC5, Lafire will name the chain-name as A, B, Cí .

- (3). If your protein binds peptides as substrate, it is best to process peptides as proteins in Lafire. You

should put the sequence of peptide into the sequence file (see 4-2-4), and set the parameter of `öcomplexö` to `ö1ö` (see 4-2-7).

- (4). If your sample is hetero-oligomer, you should make the PDB file that the order of chain must be same with that of sequence file. Moreover, there are multi hetero-oligomers in an asymmetric unit, you should put a complex together in PDB file. For example, there are 3 complexes in an asymmetric unit, and each complex contains two chains (chainA and chainB), the order molecules in PDB file must be chainA1, chainB1, chainA2, chainB2, chainA3, chainB3.

4-2-4. Preparing Sequence file

The following is the format of sequence file (sequence.inp). The maximum number of characters per a line is 120.

```
-----
2                               (the number of different chains in the asymmetric unit)a
>chain1                          (the name of the chain)
MSRKLAVPYESVRDSLVRADYIGSKAKIKFVERE          (sequence)
KHEVKVKGIGVFPNPNYIRVIMRAFAIDVNESVRDSL      (sequence)
LARLEAPFRARLRGTGYFPNEGA                    (sequence)

>chain2                               (second chain name, if it exists)
RDELAPYGVMRKSIPGWDKPFKPHITLARRKAPRVPVVL    (sequence)
FGLEWPVEGFALVRSELKPKGA                    (sequence)
-----
```

a. This parameter is 1 when there are homo-oligomers in the asymmetric unit,

4-2-5. Preparing CNS parameters (only for using CNS program)

The italic parts should be modified below in `cns_configure` file for using CNS. The `öPARAM:ö` is used default to define the path (`/Working-directory/Task_files`) of topology, parameter and ncs-restraint files that is made by users, while `öCNS_TOPPAR:ö` is used to define default path of topology and parameter files of CNS. `öCNS_TOPPAR:ö` and `öPARAM:ö` must be written.

```
-----
(== ==>) high_res=2.05;                               (high resolution limit used for refinement)
(== ==>) atom_select=(known and not (hydrogen));       (define selected atoms for refinement)
(== ==>) ncs_infile=öPARAM:ncs_main_chain.defö;        (NCS-restraint file name)a
(== ==>) restraint_infile=öPARAM:dna-rna.defö;         (DNA/RNA-restraint file name)b
(== ==>) parameter_infile_3=öCNS_TOPPAR:ion.paramö;    (ion parameter file name)c
(== ==>) parameter_infile_4=öCNS_TOPPAR:dna-rna.paramö; (DNA/RNA parameter file name)c
(== ==>) parameter_infile_5=öPARAM:atp.paramö;        (user-defined ligands parameter file name)d
(== ==>) porst_topology_infile=öPARAM:atp.topö;       (user-defined ligands topology file name)e
(== ==>) porst_parameter_infile=öPARAM:atp.topö;      (user-defined ligands parameter file name)e
-----
```

a. The ncs-restraint file should be made under **Task_files** directory.

b. If there are DNA/RNA in your sample, you should add the DNA/RNA-restraint, and the restraint file for DNA/RNA must be put in the directory of **Task_files**.

c. Those parameter files for ion and DNA/RNA are prepared by CNS program.

d. The parameter file of ligand. This file is defined and made by user

e. Those two lines are for generate.inp file to add user-defined parameter and topology files

4-2-6. Preparing second structure information files (only for you want to use it)

If the resolution of data is lower than 3.0Å, use of second structure information is suggested.

Presently, Lafire does not make second structure information automatically. Therefore you have to modify the file of helix.inp and sheet.inp by yourself. The helix.inp and sheet.inp are in working directory, and the formats of them are:

```
-----
2                                     (the number of different chains in the asymmetric unit)a
>chain1                               (the name of the chain)
12:26 and 51:63 and 102:120 and 210:222 and 263:282   (the first and the last residues number
300:316 and 333:345                               of each helix (or strand))b
>chain2                               (the name of the chain)
5:16 and 31:43 and 70:87 and 126:140 and 163:177 and 183:192   (the first and the last residues
                                                                number for each helix (or strand))b
-----
```

a. If there are homo-oligomers in the asymmetric unit, this parameter is 1, and the information of second structure is same for each monomer.

b. The &and is used to link each helix (or strand) The maximum number of characters per a line is 120.

4-2-7. Preparing parameter file of Lafire

Please modify the italic part below for your sample in Lafire.inp.

```
-----
work_dir = /users/proIDH/Lafire                                     (the name of working directory)
dir_prefix = cycle_                                               (prefix part of directory name for refinement cycle)
init_pdb = /users/proIDH/Lafire /Pdb/phoIDH_init.pdb               (the name of initial model file)
protein_prefix = phoIDH_                                         (the name of sample. It is used as a prefix of file name)
resolution = 2.05                                                 (high resolution limit)
MR = 0                                                            (the method used for solving structure. 0:not MR; 1:MR)
complex = 0                                                       (the flag of complex.0:monomer/homo-oligomer;1:hetero-oligomer)
nmole = 4                                                         (number of molecules/complex in asymmetry unit)a
jump = 0                                                         (job control flag for Lafire. 0: default
                                                                1:start from refinement step (1)b
                                                                2:start from refinement step (2)b
                                                                3:start from pickup water and then refinement without fitting)c
                                                                4:start from fitting, then pickup water molecules with refinement)c
MSPP = 1                                                         (the flag for building strategy. 1: use strategy, 0: don't use (fast processing))d
CNS = 1                                                           (the flag for using refinement program. 0:REFMAC5; 1: CNS)
annealing = 1                                                    (refinement control flag, only for CNS. 1:use SA, 0:don't use SA)
mtz_in = resolve_remote                                         (the name of mtz file without extension .mtz)
labo= remote                                                     (label of Fobs, sigma(Fobs) in mtz file for using refinement)e
labi = P                                                         (label of Fobs, sigma(Fobs) in mtz file. This Fobs is used for building initial model)e
phi = M                                                         (label of phases, only for using phases information)f
-----
```

a . In the case of complex, this parameter is number of complexes in an asymmetric unit

b. There are two cases for starting job from refinement. The case1 (jump=1) is that the program starts from refinement based on finished fitting and building by Lafire. In this case the cycle_0 (\$ {dir_prefix}_0) directory is necessary. The case2 (jump=2) is that the program starts from refinement without fitting and building by Lafire.

c. There are two cases for starting job from pickup water molecules. The case1 (jump=3) is that the program starts from pickup water molecules, and then refinement without fitting and building. The case2 (jump=4) is that the program starts from fitting, and then pickup water molecules and refinement.

d. The multi-level strategy for both of building and fitting has be included into Lafire from version2.0.

e. The name of label does not include F and SIGF. For example, if the label of F, and sigma(F) are öFPö, and öSIGFPö in mtz file, the name of label is P; the label of F, and sigma(F) are FPromote, and SIGFPromote, the name of label is Promote. Please don't use F and SIGF as the label name of F, and sigma(F) in mtz file.

f. The name of label does not include PHI. For example, if the label of phases is PHIM in mtz file, the name of label is M. If the label of phases is PHIdm in mtz file, the name of label is dm. Please don't

use PHI as the label name of phases in mtz file. The labels of HL(Hendrickson and Lattman)coefficient, should be same with PHI. In the above example, HL labels are HLAM, HLBM, HLCM, and HLDM. **This parameter is only available when MR=0.**

4-2-8. Preparing parameter file for NCS-restraint and ligands (if you want to use)

(a). NCS-restraint for using CNS

Lafire has two ncs-restraint files; ncs_all.inp (for all atoms) and ncs_main.inp (for main chain) in ~/Task_files. Those files are prepared for that the number of molecules is eight or less. If the number of molecules is more than eight, you should modify this two files and rigid.inp corresponding to your proteins. The names of segment name (segid) reference to 4-2-3. If ncs-restraint files were prepared by your-self, you should put them in **Task_files** directory, and add the name into cns_configure (see 4-2-5).

(b). NCS-restraint for using REFMAC5

The preparation of ncs-restraint files (tls.inp) refers to manual of CCP4. After prepare ncs-restraint file, this file has to be put in **Task_files** directory, and reffmac_*.com in **Task_files** directory have to be modified referring to manual of REFMAC in CCP4 suite.

(c). Topology and energy parameter files for using CNS

CNS program has the topology and energy parameter files of nucleic acid and ions, but there is not patch definition for binding. If you want to define the patch for binding, you should make topology and parameter files of the patch, then put them in **Task_files** directory and add the patch into generate.inp_org in **Task_files** directory.

The preparation of topology and energy parameters files for others refer a Web of HIC-up (<http://alpha2.bmc.uu.se/hicup/>).

(d). Topology and parameter files for using REFMAC5

The preparation of topology and energy parameters files (*.cif) for using REFMAC refers the manual of REFMAC in CCP4 suite. After preparation of these files (*.cif), you have to put them in **Task_files** directory and modify reffmac_*.com files in **Task_files** directory.

4-3. Executing Lafire (Executing, Stopping, Restarting)

4-3-1. Executing Lafire

Lafire [Parameter file name] &

For example, if the parameter file (see 4-2-8) name is phoIDH.inp, execute command as below:

Lafire phoIDH.inp &

4-3-2. Stopping Lafire

If you want to stop the job of Lafire, execute kill_Lafire as below

kill_Lafire [Parameter file name]

For example, if the parameter file name (see 4-2-8) is phoIDH.inp, execute command as below:

kill_Lafire phoIDH.inp

4-3-3. Restarting Lafire

If Lafire stops with error, or you force to stop Lafire, and restart it, execute **clear_Lafire** command first. Then execute Lafire again.

clear_Lafire [Parameter file name]

For example, if the parameter file (see 4-2-8) name is phoIDH.inp, execute command as below:

clear_Lafire phoIDH.inp

4-4. Output file of Lafire

There are three kinds of output files in Lafire, job running file, job monitor files (*.log) and result files (PDB, map files, etc).

4-4-1. Job running files

job.log is a job monitor file. They are used only for debugging and checking Lafire program. Normally, user does not need to read them. If problem occurred, and you could not resolve it, please send them to author.

lafire.fos

This is a diffraction data file of CNS-format, and is made under directory of Working- directory/Data by Lafire. If users want to use diffraction data file of CNS, which was made by yourself, the file name of *olafire.fobö* should be used and the file has to be move into the directory of Working-directory /Data. Also in this case, the mtz file has to be prepared from *olafire.fobö*.

'protein_prefix' *.seq

This file is sequence file for running Lafire, and made under working directory.

'protein_prefix' *.helix and 'protein_prefix' *.sheet

Those files are second structure information files for running Lafire, and made under working directory.

4-4-2. Log files (They are made under \$working-directory/Log directory.)

error.log

This is error messages file at the beginning of the job running. The error messages only show the problems of initial PDB or mtz files.

'protein_prefix' _mut.out

The list of amino acids that are replaced by Lafire based on sequence file

'protein_prefix' _built.out

The list of amino acids built by Lafire

'protein_prefix' _rfree.out

This is a list of Rfree factors during refinement, there are 4 columns in the file:

First column : Rfree factor

Second column: R factor

Third column: flag of refinement steps

0: auto-building and fitting step.

1: conjugate-gradient minimization refinement step.

- 2: B-factor refinement step (ML refinement step in REFMAC5)
- 3: simulated-annealing step
- 4: final step of small cycle before pickup water molecules
- 5: final step of pickup water molecules cycle

Fourth column: the file name in the refinement steps

4-4-3. Result files (They are made under \$working-directory/Pdb directory.)

‘protein_prefix’_Lafire.pdb

The best model (lowest free R factor) before pickup water molecules

‘protein_prefix’_Lafire_torsion.pdb

The best model (lowest free R factor) on the stage of torsion angle optimization (before pickup water molecules)

‘protein_prefix’_Lafire_water.pdb

The final model including water molecules

If the free R factor of `protein_prefix_Lafire_torsion.pdb` is less than 2% increased than that of `protein_prefix_Lafire.pdb`, `protein_prefix_Lafire_torsion.pdb` is used for pickup water molecules, otherwise `protein_prefix_Lafire.pdb` is used.

‘protein_prefix’_Lafire_remove_water.pdb

The final model without water molecules that occupied in large blob of fo-fc map

After final refinement, large blobs of fo-fc map are detected based on water molecules and fo-fc map made before picking up water molecules.

‘protein_prefix’_Lafire_removed_water.pdb

The water molecules that occupied in large blob of fo-fc map

After refinement with water molecules, large blobs of fo-fc map are detected based on water molecules and fo-fc map made before picking up water molecules. The water molecules occupied in the large blobs are removed from **‘protein_prefix’_Lafire_water.pdb**.

‘protein_prefix’_Lafire_01.ps

The ps file of Ramachandran plot of the best model (**‘protein_prefix’_Lafire.pdb**) in the stage of torsion angle optimization (before pickup water molecules)

‘protein_prefix’_Lafire_torsion_01.ps

The ps file of Ramachandran plot of the best model on the stage of torsion angle optimization (**‘protein_prefix’_Lafire_water.pdb**) before pickup water molecules

‘protein_prefix’_Lafire_water_01.ps

The ps file of Ramachandran plot of the final model (after pickup water molecules).

‘protein_prefix’_Lafire_2fo-fc.map (CCP4-format)

The 2fo-fc map calculated from the best model before pickup water molecules

‘protein_prefix’_Lafire_fo-fc.map (CCP4-format)

The fo-fc map calculated from the best model before pickup water molecules

‘protein_prefix’_Lafire_torsion_2fo-fc.map (CCP4-format)

The 2fo-fc map calculated from the best model on the stage of torsion angle optimization before pickup water molecules

‘protein_prefix’_Lafire_torsion_fo-fc.map (CCP4-format)

The fo-fc map calculated from the best model in the stage of torsion angle optimization before pickup water molecules

‘protein_prefix’_Lafire_water_2fo-fc.map (CCP4-format)

The 2fo-fc map calculated from the final model including water molecules, and it is saved under \$working-directory/Pdb directory.

‘protein_prefix’_Lafire_water_fo-fc.map (CCP4-format)

The fo-fc map calculated from the final model including water molecules

‘protein_prefix’_Lafire_remove_water_2fo-fc.map

The 2fo-fc map calculated from the final model without water molecules that occupied in large blob of fo-fc map

In the final refinement step of Lafire, large blobs of fo-fc map are detected based on water molecules and fo-fc map made before picking up water molecules. This 2fo-fc map is calculated after refinement without removed water molecules.

‘protein_prefix’_Lafire_remove_water_fo-fc.map

The fo-fc map calculated from the final model without water molecules that occupied in large blob of fo-fc map

In the final refinement step of Lafire, large blobs of fo-fc map are detected based on water molecules and fo-fc map made before picking up water molecules. This fo-fc map is calculated after refinement without removed water molecules.

4-5. How to read the result of Lafire

Since program Lafire consists of multi programs, the progress of refinement may not be affected by errors caused by fitting or building process (for example, Segmentation fault (core dumped) error). Therefore, you should check the output file of **‘protein_prefix’_rfree.out** to make sure that refinement is done well.

4-6-1. Checking Rfree/R factors

The changes of Rfree/R factor during refinement using Lafire, are listed in output file of **‘protein_prefix’_rfree.out** under \$working-directory/Log directory (see 4-4-2). If Rfree/R factors are set to 0 in the file, there is something wrong, and Lafire does not work well. If graphic program

Lafire_moleview is installed on your system, you can show Rfree/R plot easily (see 6), otherwise Microsoft Excel can be used to plot the Rfree/R factors.

4-6-2. Checking situation of building

The file of '**protein_prefix**'_built.out under \$working-directory/Log directory (see 4-4-2) shows the situation of building including replaced residues (based on sequence file), the number of residues built in every cycle, missing residues in final model.

4-6-3. Checking final models

The final models, the models before (`-protein_prefix_Lafire.pdb`) and after (`-protein_prefix_Lafire_water.pdb`) pickup water molecules, are given under \$working-directory/Pdb directory (see 4-4-3). The 2Fo-Fc and Fo-Fc maps for both final models are also given in the same directory (see 4-4-3). You should check them using graphic programs such as Coot, O, Turbo-Frodo, etc.

4-6-4. Checking Ramachandran plots of final models

If graphic program Lafire_moleview can be installed on your system, it is easy to show Ramachandran plot of final models (see 6). The PS files of Ramachandran plot for both final models (see 4-4-3) are also calculated using PROCHECK.

4-6-5. Checking ligands of final models

In the version2.6 of Lafire, it is possible to detect ligands.

After refinement, Lafire detects large blobs of fo-fc map as ligand regions using water molecules and fo-fc map calculated before picking up water molecules. If Lafire finds some large blobs occupied by water molecules, those water molecules will be removed and refinement again. Therefore, you can check whether there are ligands using `-protein_prefix_Lafire_remove_water_2fo-fc.map`, `-protein_prefix_Lafire_remove_water_fo-fc.map`, and `-protein_prefix_Lafire_removed_water.pdb`.

5. Lafire contact

If you have any problem, question, or suggestion, please contact to lafire@castor.sci.hokudai.ac.jp

6. Error message and fixing

The Lafire checks the data prepared by user for executing program. If the data do not satisfy conditions required by Lafire, error message and error cord will be given.

6-1. Error message (correctable by users)

6-1-1. Error code 1

The PDB file of initial model is not present under ~/Pdb directory, or you have given a wrong file name in parameter file (Lafire.inp) of Lafire (see 4-2-7).

6-1-2. Error code 2

The diffraction data (mtz file) is not present under ~/Data directory, or you have given a wrong file name in parameter file (Lafire.inp) of Lafire (see 4-2-2, 4-2-7).

6-1-3. Error code 3

The labels (Fobs or Phase) defined in *.inp file (Lafire.inp) are not present in mtz file, or the label σ_{Rfree} flag of R_{free} diffractions is not present in mtz file (see 4-2-7).

6-1-4. Error code 4 (only for jump=1)

The parameter of σ_{jump} in parameter file of Lafire was set to 1, Lafire will run from refinement step based on the model coordinates that have been built and fitted by Lafire. However, the first cycle directory (cycle_0) could not be found (see 4-2-7).

6-1-5. Error code 5

Residue number of the initial model in PDB file does not correspond with that you have given in sequence file. You must renumber residues of the initial corresponding with that in sequence file (see 4-2-3 (1)).

6-1-6. Error code 6

Residue number of the initial model in PDB file does not sort in small order (see 4-2-3) or there are not **END** or **TER** between molecules. Check the initial PDB file, and renumber residues correspond with that you have given in sequence file or add add **END** or **TER** between molecules.

6-1-7. Error code 7

There are some ligands in initial protein molecule in PDB file. Check the initial PDB file of initial model and remove ligands coordinates from protein part, or add **END** or **TER** between protein and ligands (see 4-2-4).

6-1-8. Error code 8 (only for nmol >1)

The number of protein molecules in asymmetric unit is not same with nmol that you have given in input parameter file (lafire.inp; see 4-2-7)

6-1-9 Error code 9 (only for nmol >1))

Numbering systems are difference between protein molecules in initial PDB file. Check the initial PDB file, and renumber residues in same numbering system correspond with that you have given in sequence file (see 4-2-3).

6-1-10 Error code 10

The label names of Fobs, $\sigma(\text{Fobs})$, or phases in the diffraction data file (mtz file, also in Lafire.inp file) may be FC (Fc), SIGFC (SIGFc), or PHIC (PHIc). Those label names have been used. You should change label names to others in your mtz file.

6-2. Error message (no correctable by users)

6-2-1. Available memory is below 512M

This error is happened in the case that there is not enough available memory for running Lafire or the size of available memory could not be read correctly. Check available memory. If available memory is not below 512M, please contact the author of Lafire (lafire@castor.sci.hokudai.ac.jp) with information of the name of computer, CPU, and OS.

6-2-2. Memory overflow in building

This error happened in the case that at least a bigger missing part need to build, and no enough memory can be used. In this case, the error does not affect the whole process. According to process of refinement, the size of missing part may be reduced, so this error may not happen in next cycle.

6-2-3. Memory overflow in fitting

This error happened in the case that at laest a bigger continued region (ill-match) need to fit, and no enough memory can be used. In this case, the error does not affect the whole process. According to process of refinement, the size of ill-match part may be reduced, so this error may not happen in next cycle

7. Simplified manual of Lafire_molview

Lafire_moleview is a graphic program for monitoring refinement. Unlike programs of O, Turbo-Frodo, or Quanta, this program can not be used for modifying the model, and it only displays 3/5/7 residues in main-windows with keeping a good view. The density maps are also be displayed by sphere of an R radius around an atom, not box.

Lafire_moleview supports only SGI (IRIS6.5).

7-1. Executing Lafire_moleview

Lafire_moleview &

7-2. Reading PDB file

There is a menu of [File] at the upper left of Lafire_moleview's window. You just choose [Open PDB file] in menu of [File] for reading PDB file. Lafire_moleview can read maximum 5 of PDB files.

7-3. Rfree/R plot

There is a menu of [Window] at the upper right of Lafire_moleview's window. You just choose [Rfactor] to open a sub-window of \bar{R} factor, and read '**protein_prefix**_rfree.out'. Then Rfree/R factor plots will be displayed on sub-window.

7-4. Ramachandran plot

First, you need to read target PDB file by choosing [Open PDB file] from the menu of [File] (see 6-2). Then choose [Ramachandran Plot] from the menu of [Window] (see 6-3) to open a sub-window of \bar{R} Ramachandran. After using button of [molecule] at the lower left in sub-window to choose target that you have read in, you just push button of [Draw Now] at the lower right in sub-window to display the Ramachandran plot. The blue marks are residues that are shown on main window.