LatFit - Manual

Martin Mann - University Freiburg

http://www.bioinf.uni-freiburg.de/

LatPack Tools Package
Version 1.8.2

1 Description

LatFit allows the conversion of a protein's full atom structure representation in Protein Data Bank (PDB) format into a coarse grained lattice model representation. This is done by

1. fitting the C_α-atoms to neighbored lattice positions for representation of the protein backbone.
2. If side chains are modelled, C_β or the center of mass of the residues are fitted to neighbored positions of the corresponding C_α atoms.

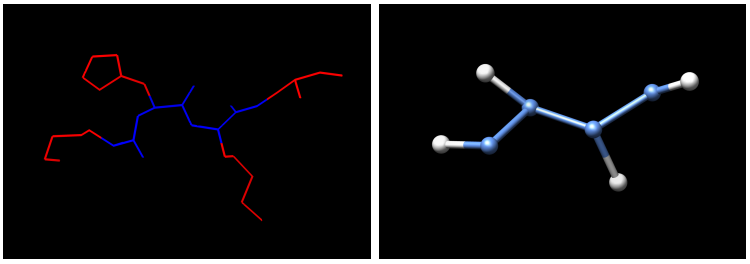

Figure 1: A full atom representation of amino acids and the corresponding side chain lattice model representation. The backbone is given in blue.

LatFit implements two greedy heuristics to find the best lattice model of a given protein utilizing the root mean square deviation (RMSD see 2.1) of a partial fit to the initial structure. The first strategy optimizes the symmetry independent distance RMSD (dRMSD) and is given in Sec. 2.3. The second approach given in 2.4 optimizes the coordinate RMSD (cRMSD), which was successfully applied in literature [1, 2, 3]. Both heuristics do not ensure to find

the optimal fit onto the lattice but a reasonable good one. For instance, using the Face Centered Cubic (FCC) lattice an approximation of the backbone with a dRMSD of 1.4 Angstroms is achieved.

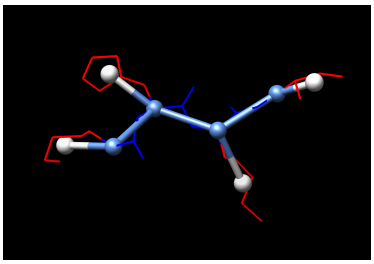

Figure 2: Mapping of the full atom and lattice model structure from Fig. 1.

2 Method

2.1 RMSD Definitions

Backbone Models: The used coordinate (Eqn. 1) and distance (Eqn. 2) RMSD:

$$cRMSD = \sqrt{\frac{\sum_{i=1}^{l}(|\hat{P}_i^b - L_i^b|)^2}{l}} \quad (1)$$

$$dRMSD = \sqrt{\frac{\sum_{i=1}^{l-1}\sum_{j=i+1}^{l}(|\hat{P}_i^b - \hat{P}_j^b| - |L_i^b - L_j^b|)^2}{(l\cdot(l-1))/2}} \quad (2)$$

where $\hat{P}_i^b$ and L_i^b denote the ith $b$ackbone coordinates of the rotated protein and the lattice fit of length l, respectively.

Side Chain Models: The used coordinate (Eqn. 3) and distance (Eqn. 4) RMSD:

$$cRMSD = \sqrt{\frac{\sum_{i=1}^{l}(|\hat{P}_i^b - L_i^b|)^2 + (|\hat{P}_i^s - L_i^s|)^2}{2\cdot l}} \quad (3)$$

$$dRMSD = \sqrt{\frac{\sum_{i=1}^{(2\cdot l)-1}\sum_{j=i+1}^{2\cdot l}(|\hat{P}_i - \hat{P}_j| - |L_i - L_j|)^2}{l\cdot((2\cdot l)-1)}} \quad (4)$$

where $\hat{P}_i^{b,s}$ and $L_i^{b,s}$ denote the ith $b$ackbone and $s$ide chain coordinates of the rotated protein and the lattice fit of length l, respectively, $\hat{P} = \hat{P}^s \cup \hat{P}^b$, and $L = L^s \cup L^b$.

2.2 Lattice scaling

To enable a reasonable RMSD evaluation of the structural distance it is important to scale the distances in the lattice to appropriate lengths. Since all lattice protein models represent the C_α-atom of each amino acid, a scaling based on the average C_α-distance is often applied. This distance is about 3.8 Angstroms on average in real protein structures. Thus, before applying our fitting procedure, all neighboring vectors of the lattice are scaled to this (user defined) length. This ensures, that the final lattice fit can be superpositioned to the real initial protein structure. Furthermore, all calculated coordinate and distance RMSD values are therefore in Angstroms as well.

2.3 Fitting : optimizing distance RMSD

Given:

$P = P_1, \ldots, P_n$	:	3D coordinates of the original atoms to approximate
N	:	the neighboring vectors of the lattice model to use
k	:	number of best substructures to store per iteration

Result:

$L = L_1, \ldots, L_n$	:	3D coordinates of the best fit onto the lattice

Method:

The approximation follows a greedy structure-elongating approach:

1: $B \leftarrow \{k$ best fits of $P_1\}$ ▷ best structure fits of last iteration
▷ initialized with the k best fits of first monomer
2: $C \leftarrow \emptyset$ ▷ structures generated in current iteration
3: **for** $i = 2 \ldots n$ **do**
4: **for all** $L \in B$ **do** ▷ L has length $(i-1)$
5: **for all** $\vec{v} \in N$ **do**
6: **if** $L_{(i-1)} + \vec{v} \notin L_1, \ldots, L_{(i-1)}$ **then** ▷ selfavoidingness
7: $C \leftarrow C \cup \{(L_1, \ldots, L_{(i-1)}, L_{(i-1)} + \vec{v})\}$ ▷ store elongation
8: **end if**
9: **end for**
10: **end for**
11: $B \leftarrow$ best k fits of C according to dRMSD to $P_1, \ldots, P_i$
12: $C \leftarrow \emptyset$ ▷ reset structure storage
13: **end for**
14: report best fit $L \in B$ according to cRMSD to P

Note: Since the dRMSD is based on a simple sum of distances (see 2.1), no full dRMSD computation has to be done in line 11. It is sufficient to update the dRMSD of the elongated fit $(L_1, \ldots, L_{(i-1)})$ with the sum of distance differences of the appended monomer L_i to $L_1, \ldots, L_{(i-1)}$ compared to the orginal chain.

This reduces the time complexity for each dRMSD evaluation from $O(n^2)$ to $O(n)$.

Note: In order to calculate the cRMSD of the final fit, we have to calculate a superpositioning of the lattice fit and the original data. We apply the algorithm introduced by Kabsch [4, 5] onto all symmetric structures of the best fit found. The best superpositioning gives the final cRMSD of the dRMSD optimized lattice fit.

2.4 Fitting : optimizing coordinate RMSD

Given:

$P = P_1, \ldots, P_n$	:	3D coordinates of the original atoms to approximate
N	:	the neighboring vectors of the lattice model to use
r^X, r^Y, r^Z	:	rotation of the lattice according to X,Y, and Z-axis
k	:	number of best substructures to store per iteration

Result:

$L = L_1, \ldots, L_n$	:	3D coordinates of the best fit onto the lattice

Method:

The approximation follows a greedy structure-elongating approach:

1: $N' \leftarrow N$ rotated by the angles r^X, r^Y, r^Z ▷ lattice rotation
2: $B \leftarrow \{k$ best fits of $P_1\}$ ▷ best structure fits of last iteration
▷ initialized with the k best fits of first monomer
3: $C \leftarrow \emptyset$ ▷ structures generated in current iteration
4: **for** $i = 2 \ldots n$ **do**
5: **for all** $L \in B$ **do** ▷ L has length $(i-1)$
6: **for all** $\vec{v} \in N'$ **do**
7: **if** $L_{(i-1)} + \vec{v} \notin L_1, \ldots, L_{(i-1)}$ **then** ▷ selfavoidingness
8: $C \leftarrow C \cup \{(L_1, \ldots, L_{(i-1)}, L_{(i-1)} + \vec{v})\}$ ▷ store elongation
9: **end if**
10: **end for**
11: **end for**
12: $B \leftarrow$ best k fits of C according to cRMSD to $P_1, \ldots, P_i$
13: $C \leftarrow \emptyset$ ▷ reset structure storage
14: **end for**
15: report best fit $L \in B$ according to cRMSD to P

Note: Since the cRMSD is based on a simple sum of distances (see 2.1), no full cRMSD computation has to be done in line 12. It is sufficient to update the cRMSD of the elongated fit $(L_1, \ldots, L_{(i-1)})$ with the distance of the appended monomer L_i to the orginal point P_i. This reduces the time complexity for each cRMSD evaluation from $O(n)$ to $O(1)$.

Note: Due to the greedy storing of the k best structures only, it may occur that none of the k best of the last iteration can be extended in a selfavoiding way (Line 6 gives 'false'). Therefore, B would get empty and the approximation stops without finding a selfavoiding fit of the whole structure P. This problem can usually be solved by increasing k but to the cost of additional computations and runtime.

2.4.1 Lattice Rotation

To find the best lattice approximation of a full atom protein structure P not only one lattice orientation has to be considered. Different rotations of the lattice along the X, Y, and Z-axis have to be generated during the search for an optimal fit. For each rotation tuple (r^X, r^Y, r^Z), the best possible fit can be obtained using the method introduced in Sec. 2.4.

A systematic search can be done that divides evenly a given rotation range $[0, m \cdot \pi]$ into s values for each of the rotation angles r^X, r^Y, and r^Z, with $m > 0$ as a user defined maximal rotation factor. Afterwards, all s^3 different rotation combinations are used to find the best structure approximation. This yields the best fit L of structure P onto a lattice with the corresponding best rotation angles r_b^X, r_b^Y, and r_b^Z.

The symmetry of the lattice gives directly a maximal rotation factor m necessary. For instance in the cubic lattice, a rotation of 90° is symmetric according to all axes. Therefore, m can be limited to 0.5 to avoid unnecessary calls of the fitting procedure for symmetric rotation angles. The same hold for the cubic and face centered cubic lattice.

2.4.2 Refinement

The best structure found via systematic search is usually not the best possible due to the discretized rotation steps and the large size of the interval searched. Here, a refinement of this structure can help.

Therefore, a small interval around each of the so far best rotation angles $r_b^i \in \{r_b^X, r_b^Y, r_b^Z\}$ is defined. For a user given refinement rotation factor $m_r > 0$ the intervals are $[r_b^i - (m_r \cdot \pi), r_b^i + (m_r \cdot \pi)]$. Once again, a systematic search is performed by an even division of the interval in s_r values. This results in additional s_r^3 calls of the fitting procedure.

2.5 Glycin Handling in Side Chain Models

Glycine (GLY or G) is the smallest and simplest amino acid found in proteins. Its chemical formula H_2N-CH_2-COOH reveals the missing side chain at the C_α atom. This is no problem when fitting backbone models but for side chain models a special handling has to be defined. Here, no distinguishing between different amino acid structures is done and each has to be represented with two monomers. Thus, we have to introduce a dummy side chain for Glycine as well for which a coordinate to fit has to be set.

We decided to fit both, the backbone and the side chain monomer, of a glycin lattice protein equivalent onto the C_α-position of the original Glycine. Thus no artificial 'original' side chain position has to be set and the RMSD deviation should be relatively small.

3 Available Lattices

Several lattice models can be used to fit a structure onto. For side chain models, a combination of two different lattices can be used (see parameter **-scLat**).

The currently supported lattice models and the corresponding neighboring vectors are:

ID	Name	Neighborhood vectors	#
SQR	Square	$\{\pm(1,0,0), \pm(0,1,0)\}$	4
CUB	Cubic	$\{\pm(1,0,0), \pm(0,1,0), \pm(0,0,1)\}$	6
FCC	Face Centered Cubic	$\left\{ \begin{array}{c} \pm(1,1,0),\pm(1,0,1),\pm(0,1,1), \\ \pm(1,-1,0),\pm(1,0,-1),\pm(0,1,-1) \end{array} \right\}$	12
210	Chess Knights Walk	$\left\{ \begin{array}{c} \pm(2,1,0), \pm(2,-1,0), \pm(2,0,1), \pm(2,0,-1), \\ \pm(1,2,0), \pm(-1,2,0), \pm(0,2,1), \pm(0,2,-1), \\ \pm(1,0,2), \pm(-1,0,2), \pm(0,1,2), \pm(0,-1,2) \end{array} \right\}$	24

4 Program Parameters

PDB Input

-pdbFile The full atom protein representation to fit onto a lattice in Protein Data Bank (PDB) format. If this parameter is not given or set to 'STDIN', the structure is read from the standard input.
Note: **-fitSideChain** is used, reading from 'STDIN' is not possible due to a necessary double parsing of the PDB input.

-pdbAtom The atom identifier that has to be fitted as the backbone monomer of the lattice structure. Usually, 'CA' for C_α-atoms is used. If 'CoM' is given, the center of mass of the amino acid side chain is fitted. If **-fitSideChain** is used, the given atom is fitted as the side chain monomer of the structure and C_α for the backbone.

-pdbAtomAlt If the PDB file contains alternatives for atoms to fit, an alternative identifier has to be given to allow a unique fit.

-pdbChain In case the PDB file contains several amino acid chains, one chain to process has to be specified.

-pdbChainGaps Some PDB files show imcomplete atom position information such that no information is given for some amino acids. Such *gaps* in the sequence are usually rejected by the program that tries to fit an entire chain. If such a non-consecutive chain with gaps should be fitted instead, the parameter **-pdbChainGaps** has to be given.

Lattice Settings

-lat The lattice onto that the backbone has to be fitted. The available list of lattice identifiers is given in Sec. 3.

-bondLength The distance in Angstroms between two neighbored C_α-atoms in the lattice. Used to scale all neighboring vectors of the lattice (Sec. 3) to this length. For default we suggest the common value '3.8' used in literature.

Side Chain Settings

-fitSideChain If present, a fit of two monomers per amino acid is done. One for backbone and one representing the side chain. The C_α-atom ('CA') is used to fit the backbone monomer. The atom specified with **-pdbAtom** is used for the side chain monomer fit.
Note: The fit of a Glycine amino acid includes a side chain monomer as well, even it has none in real proteins! Here, both monomers (backbone and side chain) approximate the C_α-atom position.

-scLat Per default the same neighboring vectors as for the backbone fit (**-lat**) are used for the neighboring of backbone (C_α) and side chain monomers. Using **-scLat**, a different set of allowed neighboring vectors (=lattice) can be specified. The length of these vectors are calculated in relation to the backbone vector lengths (see **-bondLength**). The available list of lattice identifiers is given in Sec. 3.

-scContrib Allows for a weight of the side chain fit according to the backbone approximation. This factor is multiplied to each side chain monomer RMSD that is added to the overall RMSD. Therefore, higher the value yield a better fit of the side chain monomer compared to the C_α atom.
Note: a value $\neq 1.0$ makes the reported RMSD meaningless due to the scaling of the side chain distances!

-fitDirVec If present, a fit of direction vectors instead of side chain atoms is performed. A direction vector is given by $\vec{d} = k \cdot (\mathbf{pdbAtom} - C_\alpha)$, whereby k is a calculated scaling factor to set the length of $\vec{d}$ to **dirVecLength**.

-dirVecLength The length of the direction vector to fit, if **-fitDirVec** is specified.

Fitting Parameters (see Method Sec. 2)

-optMode Defines what optimization strategy to use :

`D :` distance RMSD (see 2.3) [default]

`C :` coordinate RMSD (see 2.4)

The following parameters apply to the coordinate RMSD optimization only (see 2.4):

-rotMax Factor that limits the maximal rotation angles in radian measure. The rotations are done within $[0.0, \mathbf{rotMax} \cdot \pi]$ for each dimension X,Y, and Z.

-rotSteps Number of discrete rotation steps done to find a good fit. Therefore, the interval $[0.0, \mathbf{rotMax} \cdot \pi]$ is divided into **rotSteps** equal intervals.

-nKeep Number of best structures to store that are extended in the next iteration.

-refRotSteps Determines if a refinement of the best structure of the "global" screening should be done or not. If set to 0 no refinement is done. Otherwise, the best approximation should be improved. This is done via a fine grained rotation around the best angles r^X, r^Y, r^Z so far. The rotation is done in the intervals $[r^i - \Delta r, r^i + \Delta r]$ around each rotation angle with $i \in \{X, Y, Z\}$ and $\Delta r = \mathbf{refRotMax} \cdot \pi$.

-refRotMax Factor that defines the rotation intervals around the best rotation angles so far in which a refinement should be done (see **-refRotSteps**).

Output

-outMode The format in that the output should be written. Possible formats are:

ID	Format Description
CML	Chemical Markup Language (XML)
PDB	Protein Data Base format
XYZ	Coordinate output (plain text)

-outFile If not specified or set to 'STDOUT' the final structure output is done to standard output. Otherwise it is written to the specified file.

-outAllBest Every time a better fit than the last best found is achieved the corresponding structure is printed.

-outLatPoint Prints the non-rotated lattice structure with discrete lattice positions instead of the rotated.
Note: Also the neighbor vector scaling via **-bondLength** is ignored!

-outOrigData If present, adding the atom positions of the original protein structure (**-pdbFile**) to the output.

Miscellaneous

-v Give verbose output during computation.

-s Give only minimal output during computation.

-help Prints the available program parameters.

5 Contact

Martin Mann **http://www.bioinf.uni-freiburg.de/**
Bioinformatics Group
University Freiburg, Germany

References

[1] B. H. Park and M. Levitt: **The Complexity and Accuracy of Discrete State Models of Protein Structure**, *Journal of Molecular Biology* 1995, **294**:493-507

[2] J. Miao, J. Klein-Seetharaman and H. Meirovitch: **The Optimal Fraction of Hydrophobic Residues Required to Ensure Protein Collapse**, *Journal of Molecular Biology* 2003, **344**:797-811

[3] A. Godzik, A. Kolinski and J. Skolnick: **Lattice Representations of Globular Proteins: How Good Are They?**, *Journal of Computational Chemistry* 1993, **14(10)**:1194-1202

[4] W. Kabsch: **A solution for the best rotation to relate two sets of vectors**, *Acta Crystallographica* 1976, **A32**:922-923

[5] W. Kabsch: **A discussion of the solution for the best rotation to relate two sets of vectors**, *Acta Crystallographica* 1978, **A34**:827-828