Model building, refinement, and model validation

Alan Brown

EM course 2017
Then

Sun et al (2011) PNAS 108, 20473-20478
Now
In practice, these are not discrete steps
Useful packages for cryo-EM

http://www.ccpem.ac.uk/

http://www.phenix-online.org/

https://www.rosettacommons.org/software/
Maps for model building
EM maps for model building

• A single B-factor applied to the whole map may not be best for model building /refinement
• Local map sharpening can be extremely useful for model building
Map sharpening/blurring

Deposited map (EMD-2984)
Map sharpening/blurring

Blur 20 Å²
Map sharpening/blurring

Blur $40 \text{ Å}^2$
Map sharpening/blurring

Blur 60 Å²
Map sharpening/blurring

Blur 80 Å²
Map sharpening/blurring

Blur 100 Å²
Map sharpening/blurring in CCP-EM

CCP-EM > MRC to MTZ

- The plot shows the mean structure factor amplitude ($<|F|>$) vs resolution (1 / Å) plot
- It should reach 0 at high resolution

Deposited map
Blur by 40 Å²
Map sharpening/blurring aids model building

Deposited map (EMD-2984)
Map sharpening/blurring aids model building

Blur 40 Å²
Automated map sharpening

Developer: Tom Terwilliger

Aim:
• Maximize surface area
• As few as possible continuous regions

Adjusted surface area = surface area - weight*no. of regions
Other map manipulations

• strongly recommended to avoid standard crystallographic procedures for map modification with cryo-EM maps

• For example, 2Fo-Fc maps are necessarily model biased

• Any new method of “map improvement” must be rigorously tested

• The safest approach is always to use the observed maps; these maps are the last link between the data and the atomic models.

• For an in-depth discussion of the potential misuse of crystallographic maps and density modification in cryo-EM see:
Template generation and fold recognition
Generating starting models

Structure exists in PDB
If the model is from X-ray data, use PDB_redo to re-refine the model using the latest software

Similar structures exist in PDB
Generate homology models using template-based modeling

There’s nowt like it
Generate secondary structure predictions (e.g. JPRED)

Ab initio structure predictions

Some useful programs:

iTASSER (http://zhanglab.ccmb.med.umich.edu/I-TASSER/)
Phyre2 (http://www.sbg.bio.ic.ac.uk/phyre2/)
Evfold (http://evfold.org/evfold-web/evfold.do)
Fold recognition: the BALBES-MOLREXP pipeline

Brown et al (2017) [hopefully...]

Escherichia coli
Acyl Carrier Protein (ACP)

ACP fitted to map
Automated model building
Automated model building
Automated model building

• Less time consuming than building models manually
• Removing human judgment can minimize errors

• Current methods:
  • Buccaneer
  • Phenix.map_to_model
  • Rosetta
Buccaneer

• **Key developer:** Kevin Cowtan (University of York)
• **Basic premise:** trace protein structures in density maps by identifying connected alpha-carbon positions using a likelihood-based density target
• **Availability:** through CCPEM
• **References:** http://www.ccp4.ac.uk/newsletters/newsletter44/articles/buccaneer.html
Buccaneer: how it works

- Uses a simulated map from a known (reference) model to obtain likelihood target, and then search for this target in the unknown map

Developed by Kevin Cowtan at the University of York
1. Find candidate Cα positions
   • superimposed Cα positions from a known reference structure
   • uses a 4 Å sphere around Cα position
   • the likelihood function can therefore be described in terms of an expected density and a weighting

Mean density calculated over many Cα groups
Variance density calculated over many Cα groups
Buccaneer: 10 stages

2. Grow Cαs into chain fragments
   • Model grows sideways from existing chain fragments by looking for new Cαs at an appropriate distance from the existing chain

3. Join and merge the fragments, resolving branches

4. Link nearby N and C terminii (if possible)
5. Sequence the chains (i.e. dock sequence)
   • Looks for Cβ environment
   • Likelihood comparison between the density of each residue in the work structure and the residues of the reference structure allows sequence to be assigned to longer fragments

6. Correct insertions/deletions
7. Filter based on poor density
8. NCS Rebuild to complete NCS copies of chains [optional]
9. Prune any remaining clashing chains
10. Rebuild side chains
Phenix.map_to_model

• **Key developer**: Tom Terwilliger (Los Alamos National Laboratory)
• **Basic premise**: builds protein/RNA/DNA into EM maps.
• **Availability**: through Phenix. Currently command line only.
• **References**: https://www.phenix-online.org/version_docs/dev-2428/reference/map_to_model.html
Phenix.map_to_model

Can build both proteins and nucleic acids (type of chain to be built will be based on the supplied sequence file)

- Map_box: Reduces box to just around density
- Segment_and_split_map: Automatically segments map
- Create maps (variable sharpening)
- Trace chain & build model: RESOLVE model building / Trace-chain model-building algorithm
- Idealize secondary structure & refine
- Assemble and refine
- Apply symmetry & refine

Integrated with phenix.real_space_refine
Phenix.map_to_model: locating fragments

- Method based on RESOLVE
- Pattern matching algorithm
- Uses fragments larger than individual atoms – secondary structure
- Starts by FFT-based identification of helices and strands
  - Helical template: 6 amino acids (average density from ~200 6-residue helical segments)
  - β-strand template: 4 amino acid, average density
- Superimpose on this template each fragment in a library (helix, sheet)
  - Helix fragment library: 53 helices 6-24 amino acid long
  - Beta-strand fragment library: 24 strands 4-9 amino acid long
- Identify longest segment in good density
Phenix.map_to_model: growing fragments and assigning sequences

• Extends chains using a tri-peptide fragment library
  • N-terminal extension (3 full amino acids), 9232 members
  • C-terminal extension (CA C O + 2 full amino acids), 4869 members
• find fragment that can itself be optimally extended (look-ahead scoring)
• For each fragment:
  • superimpose CA C O on same atoms of last residue in chain (extending by 2 residues)
  • pick the 10 highest scoring fragments
    • For each of these extend again by 2 residues and pick best 1
• Test all overlapping fragments as possible extensions
• Choose one that maximizes score when put together with current fragment
• When current fragment cannot be extended: remove all overlapping fragments, choose best remaining one, and repeat
• The sequence is assigned to the mainchain by determining the relative probability of every amino acid at each position (based on density and sequence composition)
• Rotamers are chosen based on correlation coefficient
Rosetta (model building for cryo-EM)

- **Key developer**: Frank DiMaio (University of Washington)
- **Basic premise**: de novo structure determination from cryo-EM maps by combining conformational sampling with all-atom energy functions
- **Availability**: through the Rosetta software package
- **Tutorial**: https://faculty.washington.edu/dimaio/wordpress/software/
Rosetta: step 1

For overlapping 9-residue windows of sequence:

- identify “fragments” in the PDB with similar local sequences and predicted secondary structures
- Perform a 6-dimensional search (rotations and translations) to dock these into the map

4.8 Å
The resulting fragment placements are evaluated using a score function consisting of 4 terms:

1. A density correlation term assessing the agreement of fragment and map
2. An overlap term favoring fragment pairs assigning the same residue to the same location
3. A “closability” term favoring fragment pairs close in sequence that are close in space
4. A clash term preventing two residues from occupying the same place
• Monte Carlo sampling guided by this score function finds the maximally consistent subset of fragment placements from this larger set. Sometimes no fragment is selected.

• These fragments are assembled into a partial model.

• During iterations, density that has been assigned is masked out and only fragments that are unassigned are used.
Rosetta : step 3

• The partial model (70% complete) is completed using RosettaCM (comparative modeling) guided by the map
• For each partial model, 1,000 full-length models are generated.
• These are filtered by Rosetta energy
• And finally by best fit to the density
Rosetta: step 3 (Rosetta enumerative sampling)

• A recent update (Frenz et al. (2017) doi:10.1038/nmeth.4340)
• Starts with an incomplete model
• Grows one residue at a time, starting with the terminal residue adjacent to the missing segment
• The conformation up to the previous 9 residues is sampled
• Each generated solution is evaluated against the experimental data and added to the 'beam'—the pool of partial models
• Following each sampling step, the model pool is culled to contain a set number of solutions
  • (usually 64 or 128)
• This process is repeated until all missing residues have been assigned.
Manual model building
Know your density

Hryc et al (2017) PNAS
Coot

- **Key developer**: Paul Emsley (MRC-LMB)
- **Basic premise**: macromolecular model building, model completion and validation
- **Availability**: CCP4/Phenix
- **Tutorial**: https://www2.mrc-lmb.cam.ac.uk/personal/pemsley/coot/files/EM-Tutorial-Coot-PE.pdf
Coot : tools for building proteins

• Turn on restraints to ensure that help manual model building

Cis-nonPro peptides are very rare (~0.03%), but you can turn off this restraint to model real cis-peptides

Smaller means better geometry
Coot : Jiggle Fit

• Loop $n$ (say 1000) times:
  • Generate random angles and translations
  • Transform atom selection by these rotations and translation
  • Score and store the fit to density

• Rank density fit scores
  • Pick top 20 solution, for each of them
    • Rigid body fit and score solutions
    • Pick the highest scoring solution if it's better than the starting model

Radius of Convergence is larger when using a low-pass map
Coot : secondary structure elements

• Good starting point for *de novo* model building

• Coot has an option to automatically identify all secondary structure element in the map (*SSE identification*)

• Or add a helix/strand in a specific section of density (*Add helix/strand here*)

• In conjunction with jiggle fit finds correct orientation of α-helix every time at maps with resolution better than 4 Å
Add terminal residue

• Build one residue at a time starting at a previously positioned amino acid
• Remains the most popular way of model building
• either add as a alanine residue and mutate to correct residue afterwards
• Or assign a sequence to the model so that the identity of the next residue is known
Coot : tools for building proteins

Cα baton mode
trace the main chain of a protein by placing correctly spaced α-carbon atoms

Cα Zone -> main chain

(this is not a real EM map)
Coot: tools for building nucleic acids

**Ideal DNA/RNA** - build an ideal DNA or RNA fragment

**Add terminal residue** - extend a nucleic acid

**Rcrane** (http://pylelab.org/software) – allows for semi-automated building of RNA models within Coot through the identification of phosphate positions within the density map
Coot: tools for moving atoms around

**Real space refinement** - optimize the fit of the model to the density, while preserving stereochemistry

  - Sphere refine – real-space refinement for an environment

**Regularize** - optimize stereochemistry

  - Sphere Regularize - optimize stereochemistry for an environment

**Rigid body fit (local)** - optimize the fit of a rigid body to the density

**Rotate/translate zone** - manually position a rigid body

**Rotamer tools** (auto fit rotamer, manual rotamer, mutate and autofit, simple mutate)

**Torsion editing** (edit chi angles, edit main chain torsions, general torsions)

**Other** (flip peptide, flip sidechain, cis <-> trans)
Coot: live validation

Refinement in Coot gives immediate feedback
iSOLDE : interactive molecular dynamics

- **Key developer**: Tristan Croll (CIMR, University of Cambridge)
- **Basic premise**: Allows users to interact in real time with molecular dynamics simulations
- **Availability**: through VMD. In the future will be available through Chimera X.
- **Tutorial**: https://www.youtube.com/watch?v=kqJpYIHOiDY
iSOLDE : interactive molecular dynamics

- Select a region of interest to run molecular dynamics on
- This might be a small problem region (~10–20 contiguous residues and their immediate spatial neighbors)
- A further 8 Å shell of surrounding atoms is included in the simulation to maintain the physical context of the mobile atoms, but remains fixed in space
- Mask maps to within user-specified distances from the mobile atoms
- Map is converted to potential energy maps to which the simulation is then coupled
- Standard stereochemical restraints are included (bond, angle, torsions)
- Simulation also takes into consideration long-range interactions (electrostatics and van der Waals)
iSOLDE: interactive molecular dynamics

• User can interact in real time with the molecular dynamics simulation
• coupled to a haptic interface – this allows the user to “pull” on any atom within a running simulation, while it (and its surroundings) responds in a manner akin to a real molecule
• This makes model building feel more like working in a physical environment
Conundrums

• What to do about atoms without density?
  • Do I include loops with indistinct density?
  • Should I truncate sidechains?

• Is one model enough?
  • Cryo-EM is an averaging technique. Your final map will be the result of averaging an ensemble of states that the macromolecule exists in.
  • Is a single model therefore appropriate?

These same questions exist in X-ray crystallography and have similarly not been resolved.
Refinement
The purpose of refinement

1. Improve the fit of your model to density
2. Ensures your molecule agrees with prior knowledge

Refinement improves fit

Refinement reduces clashes
The purpose of refinement

Fitting to density is easy, fitting to density while maintaining good geometry is a delicate balance.
What changes during refinement…

1. Model coordinates \((x, y, \text{ and } z)\)
2. B-factors (these can be for each individual atom or group of atoms)
3. Occupancies (for multiple conformations)

…and what doesn’t

1. The map (currently)
Restraints

Restraints are a way of specifying prior knowledge

Standard restraints (used by default) include:

1. Bond lengths
2. Angles
3. Chirals
4. Planes
5. Some torsion angles
6. B-values

Atoms are bonded to each other in specific ways

Atoms close to one another cannot be dramatically different

Restraints stabilize refinement, reduce the chance of overfitting, and ensures that the final model is consistent with prior knowledge

<table>
<thead>
<tr>
<th>Bond</th>
<th>Bond Length (Å)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C−C</td>
<td>1.54</td>
</tr>
<tr>
<td>C= C</td>
<td>1.34</td>
</tr>
<tr>
<td>C≡ C</td>
<td>1.20</td>
</tr>
<tr>
<td>C− N</td>
<td>1.43</td>
</tr>
<tr>
<td>C≡ N</td>
<td>1.38</td>
</tr>
<tr>
<td>C≡ N</td>
<td>1.16</td>
</tr>
<tr>
<td>C− O</td>
<td>1.43</td>
</tr>
<tr>
<td>C≡ O</td>
<td>1.23</td>
</tr>
<tr>
<td>C≡ O</td>
<td>1.13</td>
</tr>
</tbody>
</table>
1. **NCS constraints**
   - If symmetry was applied during map reconstruction, the molecules will be exactly the same.

2. **Global NCS restraints**
   - Molecules are similar. Differences are minimized.

3. **Local NCS restraints**
   - Domains may be similar, but the orientation of the domains relative to one another may differ.
External restraint generation

For proteins and nucleic acids:

- **ProSMART**: Rob Nicholls (CCP-EM)
- **LIBG**: Fei Long (CCP-EM)
- **Phenix.secondary_structure_restraints**: Oleg Sobolev / Pavel Afonine (Phenix)

For ligands:

- **ACEDRG**: Fei Long (CCP4 / CCP-EM)
- **Phenix.elbow**: Nigel Moriarty (Phenix)
External restraints from homologous structures

Target + Reference = Refined
External restraints from homologous structures

All restraints can be visualised and applied in Coot:
Secondary structure restraints

Visualization of helix restraints in Coot
Base-pair restraints

Purple = without restraints
Yellow = with base pair restraints
Parallel-plane restraints

- Identifies and maintains sets of atoms that should be in parallel planes
- Nucleic acid bases
- Also amino acid sidechains (Trp, Tyr, His, Arg, Lys, Asn, Gln)
- And base:amino acid sidechain interactions

Green = before refinement; blue = refinement without stacking restraints; Yellow = after refinement with stacking restraints
Options

1. REFMAC (Fourier/reciprocal-space refinement)

2. Phenix.real_space_refine

3. Rosetta
REFMAC

- **Key developer**: Garib Murshudov (MRC-LMB)
- **Basic premise**: Macromolecular refinement using maximum likelihood and elements of Bayesian statistics
- **Availability**: CCPEM / CCP4

\[ f_{\text{tot}} = w f_{\text{data}} + f_{\text{geom}} \]

- \( f_{\text{data}} = -\log[P(\text{obs}; \text{model})] \)
- \( f_{\text{geom}} = -\log[P(\text{model})] \)
- \( w \) : relative weighting

- likelihood of the data
- probability of the model
Phenix.real_space_refine

- **Key developer**: Paul Adams / Pavel Afonine (Lawrence Berkeley National Laboratory)
- **Basic premise**: Refines a model into a map in real space.
- **Availability**: Phenix
- **Tutorial**: https://www.youtube.com/watch?v=shmBHtyUdCc
# Features available in REFMAC and Phenix

<table>
<thead>
<tr>
<th>Options</th>
<th>REFMAC</th>
<th>Phenix</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUI / command line</td>
<td>both</td>
<td>both</td>
</tr>
<tr>
<td>Morphing (available in Coot)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Rigid-body refinement</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Simulated annealing</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Jelly-body refinement</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>B-factor refinement</td>
<td>✓</td>
<td>✓ (reciprocal space)</td>
</tr>
<tr>
<td>Composite map refinement</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>Reference structure restraints</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Secondary structure restraints</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Nucleic acid restraints</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Symmetry restraints / constraints</td>
<td>both</td>
<td>both</td>
</tr>
<tr>
<td>Ramachandran restraints</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Rotamer restraints</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Ligand restraint handling</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Rosetta (model refinement for cryo-EM)

- **Key developer**: Frank DiMaio (University of Washington)
- **Basic premise**: rebuilds models using iterative fragment-based sampling
- **Availability**: through the Rosetta software package
- **Tutorial**: https://faculty.washington.edu/dimaio/wordpress/software/
Rosetta refinement: stage 1

- Hand-built models usually fit the density well
- but are incorrect geometrically (the model is strained)
- Identify problematic residues by assessing local model-strain & local agreement to density
- Rebuild problematic regions using iterative fragment-based rebuilding followed by all-atom refinement
- This rebuilding happens in just one of the half maps
Rosetta refinement: fragment-based backbone rebuilding

- Backbone fragments are collected from the PDB
- Superposed onto the current model
  1. First, use 25 x 17-residue fragments
  2. And then use 25 x 9-residue fragments
- Each fragment is optimized to fit the density
- At each position, the fragment with best fit to the density that has an r.m.s. of less than 0.5 Å over the terminal residues is selected
- Backbone atomic positions from the selected fragment then replace the corresponding backbone in the current model
- The backbone geometry at the stitching site is regularized
Select models

- Identify the top 200 stereo-chemically correct models with best agreement to an independent half-map (this prevents overfitting)
- Select top 50 models with good geometry
- Picks top 10 models with best agreement to the final full map
Rosetta refinement: stage 3

- Models are further optimized to the full-reconstruction
- This refinement uses a weight optimally scaled between experimental data and the forcefield using the 'validation' half map
- Can perform a magnification refinement – if the magnification on your microscope is poorly calibrated
- Performs a B-factor refinement
Refinement : B-factors

B-factors are usually interpreted as a measurement of the amount of motion that an atom experiences.

Should correlate with local resolution
Validation

“with great resolution comes great responsibility”
Model validation

1. Does the model agree with the map?
   • Global measure of how well the model fits the map
   • Local measures of fit
   • B-factor calculations

2. Have we overfitted the data?
   • Cross-validation

2. Does the model look like other macromolecules and what we know of chemistry?
   • Consistency of 3D structure with 1D sequence
   • Deviations from ideal values (bonds, angles, etc)
   • Non-bonded clashing atoms
   • Stereochemistry (Ramachandran plot)
   • Rotamers
Global measure of fit: FSC_{average}

- FSC_{average} (as used in REFMAC) is largely independent of Bfactor.
- FSC is calculated over resolution shells. If the shells are sufficiently narrow the weights are roughly the same within each shell.

\[
FSC_{\text{average}} = \frac{\sum_{i=1}^{N_{\text{shell}}} N_i FSC_i}{\sum_{i=1}^{N_{\text{shell}}} N_i}
\]

Graph: After 10 cycles vs. Before refinement.
The crystallographic R-factor is inappropriate for monitoring fit-to-density as it can be artificially lowered by changing the B-factor.
Local measure of fit

- Per-residue correlation coefficient
- Can be calculated using many different programs:
  - phenix.real_space_correlation
  - Rosetta (-denstools::perres)
  - score_smoc.py (CCP-EM, overlapping residue windows)
Overfitting

What leads to overfitting?

1. Insufficient data (low resolution, partial occupancy)
2. Ignoring data (cutting by resolution)
3. Sub-optimal parameterization
4. Bad weights
5. Excess of imagination
Overfitting (half map validation)

Particles → 3D map refinement → Half Map 1, Half Map 2 → Final map → Model refinement → Model-to-map FSC

Refinement parameters → Scrambled model → Model refinement → Refined model

FSC\text{work} → Half Map 1, Half Map 2 → FSC\text{test}
Overfitting (half map validation)

Resolution limit must be defined in refinement

- Both FSC curves should overlap
- Sharp fall beyond resolution used for refinement (loss of predictive power)
- Better fitting to the map the model was refined against (fitting to noise)
Verify3D

• **Key developers**: David Eisenberg (UCLA)

• **Basic premise**: Determines the compatibility of an atomic model (3D) with its own amino acid sequence (1D)

• **Availability**: [http://services.mbi.ucla.edu/Verify_3D/](http://services.mbi.ucla.edu/Verify_3D/)

MolProbity

- **Key developers**: David and Jane Richardson (Duke University)
- **Basic premise**: Evaluates model quality.
- **Availability**: online ([http://molprobity.biochem.duke.edu/](http://molprobity.biochem.duke.edu/)) and through the Phenix distribution (phenix.molprobity)
MolProbity

• Don’t wait until the end of model building before running MolProbity
• By analyzing hydrogen-bonding networks, can automatically detect and correct flipped N/Q/H residues
MolProbity

What information does MolProbity provide?

PDB : 5K12, EMDB: EMD-8194, resolution = 1.8 Å

<table>
<thead>
<tr>
<th>Protein Geometry</th>
<th>All-Atom Contacts</th>
<th>Clashscore, all atoms: 19.75</th>
<th>26th percentile* (N=837, 1.80Å ± 0.25Å)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poor rotamers</td>
<td>12</td>
<td>0.82%</td>
<td>Goal: &lt;0.3%</td>
</tr>
<tr>
<td>Favored rotamers</td>
<td>1440</td>
<td>98.36%</td>
<td>Goal: &gt;98%</td>
</tr>
<tr>
<td>Ramachandran outliers</td>
<td>0</td>
<td>0.00%</td>
<td>Goal: &lt;0.05%</td>
</tr>
<tr>
<td>Ramachandran favored</td>
<td>1674</td>
<td>96.21%</td>
<td>Goal: &gt;98%</td>
</tr>
<tr>
<td>MolProbity score^</td>
<td>2.05</td>
<td>59th percentile* (N=11444, 1.80Å ± 0.25Å)</td>
<td></td>
</tr>
<tr>
<td>CPβ deviations &gt;0.25Å</td>
<td>0</td>
<td>0.00%</td>
<td>Goal: 0</td>
</tr>
<tr>
<td>Bad bonds:</td>
<td>0 / 14070</td>
<td>0.00%</td>
<td>Goal: 0%</td>
</tr>
<tr>
<td>Bad angles:</td>
<td>0 / 18996</td>
<td>0.00%</td>
<td>Goal: &lt;0.1%</td>
</tr>
<tr>
<td>Cis Prolines:</td>
<td>6 / 66</td>
<td>9.09%</td>
<td>Expected: ≤1 per chain, or ≤5%</td>
</tr>
</tbody>
</table>

In the two column results, the left column gives the raw count, right column gives the percentage.

* 100th percentile is the best among structures of comparable resolution; 0th percentile is the worst. For clashscore the comparative set of structures was selected in 2004, for MolProbity score in 2006.

^ MolProbity score combines the clashscore, rotamer, and Ramachandran evaluations into a single score, normalized to be on the same scale as X-ray resolution.
MolProbity + Coot

- Validation and model building are not separate entities
Ramachandran plot

NOTE: not everything flagged as an outlier is wrong - check
Ramachandran plots

General

Glycine

Isoleucine/Valine

Pre-Proline

Trans-Proline

Cis-Proline
EMRinger

- **Key developer**: Ben Barad/James Fraser (UCSF)
- **Basic premise**: Rotates C-gamma atom around the $\chi_1$ angle of a side chain, interpolating the density value in the map as it rotates
- **Availability**: phenix.emringer
EMRinger

- EMRinger score is correlated with resolution
- Gives an idea of what a good score should be
PDB Validation Reports

Generate a validation report: https://validate-rcsb-1.wwpdb.org/validservice/

Help: https://www.wwpdb.org/validation/2016/EMValidationReportHelp
Presenting validation statistics

"Table 1"

<table>
<thead>
<tr>
<th></th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Collection</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pixel size (Å)</td>
<td>1.34</td>
<td>1.06</td>
</tr>
<tr>
<td>Defocus range (μm)</td>
<td>-1.5 to -3.5</td>
<td>-0.5 to -3.5</td>
</tr>
<tr>
<td>Voltage (kV)</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>Electron dose (e- Å(^2))</td>
<td>25</td>
<td>39</td>
</tr>
<tr>
<td><strong>39S intermediate with folded rRNA</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>39S intermediate with unfolded rRNA</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Map reconstruction</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Particles</td>
<td>134,685</td>
<td>379,869</td>
</tr>
<tr>
<td>Resolution (Å)</td>
<td>3.06</td>
<td>3.03</td>
</tr>
<tr>
<td>Map sharpening B-factor (Å(^2))</td>
<td>-85.0</td>
<td>-95.0</td>
</tr>
<tr>
<td><strong>Model composition</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-hydrogen atoms</td>
<td>99,025</td>
<td>96,747</td>
</tr>
<tr>
<td>Protein residues</td>
<td>8,230</td>
<td>8,135</td>
</tr>
<tr>
<td>RNA bases</td>
<td>1,497</td>
<td>1,148</td>
</tr>
<tr>
<td>Ligands (Zn(^{2+} / Mg(^{2+}))</td>
<td>3/93</td>
<td>3/49</td>
</tr>
<tr>
<td><strong>Fit to map</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Correlation coefficient (entire box)</td>
<td>0.76</td>
<td>0.77</td>
</tr>
<tr>
<td>Correlation coefficient (around atoms)</td>
<td>0.77</td>
<td>0.79</td>
</tr>
<tr>
<td>Fouriers shell correlation (entire box)</td>
<td>0.76</td>
<td>0.77</td>
</tr>
<tr>
<td>Fourier shell correlation (around atoms)</td>
<td>0.62</td>
<td>0.83</td>
</tr>
<tr>
<td><strong>Protein geometry</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Molprobity score</td>
<td>1.66</td>
<td>1.70</td>
</tr>
<tr>
<td>All-atom clashscore</td>
<td>5.73</td>
<td>5.94</td>
</tr>
<tr>
<td>EMRinger score</td>
<td>3.69</td>
<td>3.85</td>
</tr>
<tr>
<td>RMSD deviation bonds (Å)</td>
<td>0.010</td>
<td>0.016</td>
</tr>
<tr>
<td>RMSD deviation angles (°)</td>
<td>1.02</td>
<td>1.27</td>
</tr>
<tr>
<td>Favored rotamers (%)</td>
<td>95.7</td>
<td>95.0</td>
</tr>
<tr>
<td>Poor rotamers (%)</td>
<td>0.48</td>
<td>0.90</td>
</tr>
<tr>
<td>Ramachandran favored (%)</td>
<td>94.9</td>
<td>94.4</td>
</tr>
<tr>
<td>Ramachandran outliers (%)</td>
<td>0.02</td>
<td>0.06</td>
</tr>
<tr>
<td><strong>Nucleic acid geometry</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poor sugar puckers (%)</td>
<td>1.14</td>
<td>1.39</td>
</tr>
<tr>
<td>Bad backbone conformations (%)</td>
<td>26.3</td>
<td>27.7</td>
</tr>
</tbody>
</table>
Deposition

https://deposit-pdbe.wwpdb.org/deposition

- Deposit at the same time as EM maps.
- Recommended depositions:
  - Postprocessed map
  - Both half maps
  - Any masks applied during processing
  - Any map that has been modified in any way (excluding blurring/sharpening)
  - Model