Automated phase improvement (density modification) and model building with Parrot and Buccaneer

Kevin Cowtan/Bernhard Lohkamp
kevin.cowtan@york.ac.uk
Bernhard.Lohkamp@ki.se
X-ray structure solution pipeline...
X-ray structure solution pipeline…

in CCP4i2 context
• Traditional density modification: e.g. 'dm', 'solomon', 'parrot', CNS

• Statistical density modification: e.g. 'resolve', 'pirate'
Density modification

- Density modification is a problem in combining information:

![Diagram showing Reciprocal space and Real space with phases and solvent envelope.](image)
Density modification

1. Rudimentary calculation:

\[ |F|, \varphi \] \quad \xrightarrow{\text{FFT}} \quad \rho(x) \]

\[ \varphi = \varphi_{\text{mod}} \]

\[ |F_{\text{mod}}|, \varphi_{\text{mod}} \] \quad \xrightarrow{\text{FFT}^{-1}} \quad \rho_{\text{mod}}(x) \]

Real space \quad \text{Reciprocal space}
Density modification

2. Phase weighting:

\[ |F|, \varphi \]

\[ \varphi = f(\varphi_{\text{exp}}, \varphi_{\text{mod}}) \]

\[ \text{FFT} \]

\[ \rho(x) \]

\[ \text{Modify } \rho \]

\[ \rho_{\text{mod}}(x) \]

\[ \text{FFT}^{-1} \]

\[ |F_{\text{mod}}|, \varphi_{\text{mod}} \]

Reciprocal space

Real space
Density modification

3. Phase probability distributions:

\[ |F|, P(\varphi) \xrightarrow{\text{centroïd}} |F_{\text{best}}|, \varphi_{\text{best}} \xrightarrow{\text{FFT}} \rho(x) \]

\[ P(\varphi) = P_{\text{exp}}(\varphi) \times P_{\text{mod}}(\varphi) \]

Reciprocal space

\[ |F_{\text{mod}}|, \varphi_{\text{mod}} \xrightarrow{\text{FFT}^{-1}} \rho_{\text{mod}}(x) \]

Real space
Density modification

4. Bias reduction (gamma-correction):

\[ |F|, P(\varphi) \rightarrow |F_{\text{best}}|, \varphi_{\text{best}} \rightarrow \rho(x) \]

\[ P(\varphi) = P_{\text{exp}}(\varphi) \times P_{\text{mod}}(\varphi) \]

\[ P_{\text{mod}}(\varphi) \rightarrow |F_{\text{mod}}|, \varphi_{\text{mod}} \rightarrow \rho_{\gamma}(x) \]

\[ \text{FFT} \rightarrow \text{FFT}^{-1} \]

DM, SOLOMON, (CNS)

J.P. Abrahams
Density modification

5. Maximum Likelihood H-L:

\[ |F|, P(\phi) \rightarrow |F_{best}|, \phi_{best} \rightarrow \rho(x) \rightarrow \text{Modify } \rho \rightarrow \rho_{mod}(x) \rightarrow \rho_{\gamma}(x) \]

\[ |F_{mod}|, \phi_{mod} \rightarrow \rho_{\gamma}(x) \rightarrow \text{FFT}^{-1} \]
Density modification

6. Statistical density modification:

\[ |F|, P(\phi) \xrightarrow{\text{centroid}} |F_{\text{best}}|, \phi_{\text{best}} \xrightarrow{\text{FFT}} \rho(x) \]

\[ P(\phi) = P_{\text{exp}}(\phi) \times P_{\text{mod}}(\phi) \]

\[ P_{\text{mod}}(\phi) \xrightarrow{\text{Transform distribution}} P(\rho(x)) \]

\[ \rho(x) \xrightarrow{\text{Infer}} P(\rho(x)) \]

Reciprocal space

Real space

RESOLVE, PIRATE
Density modification

Traditional density modification techniques:
- Solvent flattening
- Histogram matching
- Non-crystallographic symmetry (NCS) averaging
Solvent flattening
Solvent flattening
Histogram matching

A technique from image processing for modifying the protein region.

- Noise maps have Gaussian histogram.
- Well phased maps have a skewed distribution: sharper peaks and bigger gaps.

Sharpen the protein density by a transform which matches the histogram of a well phased map.

Useful at better than 4Å.
Non-crystallographic symmetry

- If the molecule has internal symmetry, we can average together related regions.

- In the averaged map, the signal-noise level is improved.

- If a full density modification calculation is performed, powerful phase relationships are formed.

- With 4-fold NCS, can phase from random!
Non-crystallographic symmetry

Crystallographic

Aligned 2-fold

Non-crystallographic

Unaligned 2-fold

Aligned 6-fold

Aligned 5-fold
Non-crystallographic symmetry

Useful terms:

- Proper and improper NCS: (closed and open)

- Multi-domain averaging:

- Multi-crystal averaging:
Non-crystallographic symmetry

- How do you know if you have NCS?
  - Cell content analysis – how many monomers in ASU?
  - Self-rotation function.
  - Native Patterson map (pseudo-translation only).

- How do you determine the NCS?
  - From heavy atoms.
  - From initial model building.
  - From molecular replacement.
  - From density MR (hard).

- Mask determined automatically.
Density modification

How do we represent phase probabilities?

Henrickson-Lattman coeffs: 4 numbers - A,B,C,D representing a bimodal distribution in phase angle:

A,B represent a unimodal distribution (equivalent to $\phi$, FOM)
C,D represent the superimposed bimodality.

$P(\phi) = F(ABCD)$
Estimating phase probabilities

Problem: How do we go from a single phase estimate to a full phase probability distribution?

- We need to make an estimate of the error in the estimated phase.
- The errors in the phases are a parameter of the model itself, and may be estimated by likelihood methods.

\[ |F|, P(\phi) \xrightarrow{\text{centroid}} |F_{\text{best}}|, \phi_{\text{best}} \xrightarrow{\text{FFT}} \rho(x) \]

\[ P(\phi) = P_{\text{exp}}(\phi), P_{\text{mod}}(\phi) \]

\[ P_{\text{mod}}(\phi) \xrightarrow{\text{likelihood}} |F_{\text{mod}}|, \phi_{\text{mod}} \xrightarrow{\text{FFT}^{-1}} \rho_{\text{mod}}(x) \]
Estimating phase probabilities

Sim/σ_A weighting:

\[ F_{\text{true}} = F_{\text{part}} + F_{\text{miss}} \]

We know |F_{\text{true}}|, |F_{\text{part}}|, \phi_{\text{part}}

Assuming \phi_{\text{part}}, \phi_{\text{miss}} are independent, then we expect the difference in magnitudes between |F_{\text{true}}| and |F_{\text{part}}|, averaged over reflections, to give an indication of the phase error.
Estimating phase probabilities
Combining phase probabilities

Once we have an estimate for the error in $\phi_{\text{mod}}$, we can construct a probability distribution $P_{\text{mod}}(\phi)$. The next cycle can be started with

$$P_{\text{new}}(\phi) = P_{\text{exp}}(\phi)P_{\text{mod}}(\phi)$$

**Problem:** $P_{\text{exp}}(\phi)$ and $P_{\text{mod}}(\phi)$ are not independent. The result is bias, increasing with cycle.
Estimating phase probabilities

Traditional approach: Rice likelihood function

1. Estimate the accuracy of the modified F/phase
2. Turn this into a phase probability distribution
3. Combine with the experimental phase probability

The estimate for the accuracy of the modified F/phase come from the agreement between the modified F and the observed F. **Source of bias.**
Estimating phase probabilities

Problem:

Error estimation does not take into account experimental phase information.

The experimental data tells us that the probable error is different in the two cases.

Using the additional information from the phases improves the error model and reduces bias.
Estimating phase probabilities

Solution:
MLHL-type likelihood target function.

Perform the error estimation and phase combination in a single step, using a likelihood function which incorporates the experimental phase information as a prior.
This is the same MLHL-type like likelihood refinement target used in modern refinement software such as refmac or phenix.refine.
Bias reduction

Solution:
Make each reflection only dependent on the other reflections in the diffraction pattern, and not on its own initial value.
Omit one reflection at a time, and use only the modified value of the omitted reflection. (Very slow.)
But can be implemented efficiently:
• Solvent flipping
• The $\gamma$-correction

$$\rho_{\gamma}(x) = \rho_{\text{mod}}(x) - \gamma \rho(x)$$
Density modification in Parrot

Builds on existing ideas:

• **DM:**
  - Solvent flattening
  - Histogram matching
  - NCS averaging
  - Perturbation gamma

• **Solomon:**
  - Gamma correction
  - Local variance solvent mask
  - Weighted averaging mask
Density modification in Parrot

New developments:

• MLHL phase combination
  – (as used in refinement: refmac, phenix.refine)

• Anisotropy correction

• Problem-specific density histograms
  – (rather than a standard library)

• Pairwise-weighted NCS averaging...
Other Features:

Pairwise-weighted NCS averaging:
• Average each pair of NCS related molecules separately with its own mask.
• Generalisation and automation of multi-domain averaging.
Parrot

A screenshot of the CCP4-7.0.014 Project Viewer demonstrating the 'Density modification - PARROT' job interface. The job is pending, and the options for selecting experimental data, sequence for solvent content estimation, and NCS information are highlighted. The title of the job is 'Density modification - PARROT'. The text 'Reflections: Data has undefined value' is also visible.
Parrot

Summary:
A new classical density modification program, employing the latest techniques.

- Fully automated
- Fast
- Better results than DM
Density Modification
Kevin Cowtan, York.

Statistical density modification:
e.g. Resolve, Pirate
Density modification

- **Traditional density modification:**
  *Take the phases to the mask.*
  Use them to calculate a map.
  But how do we get back to:
  - reciprocal space?
  - probabilities?

- **Statistical density modification:**
  *Take the mask to the phases.*
  - First convert mask to probability.
  - Then transform that probability.
Statistical density modification

- Form a statistical description of expected map features.

  - e.g.
    - Protein has higher mean, and is more peaky (higher variance)
    - Solvent has lower mean, and is flatter (lower variance)
Statistical density modification

- Probability of a map is determined by how well it fits these distributions:
Statistical density modification

- Probability of a map is determined by how well it fits these distributions:
Statistical density modification

- Probability of each structure factor is given by the probability of the corresponding map.
Statistical density modification

- Obtain per-grid density probability distributions.
- Transform to reciprocal space.
- Combine with experimental phases.
  - Map probability becomes phase probability distribution.

Bricogne (1992) Proc. CCP4 Study Weekend

Improved phases and maps.
Statistical density modification

Advantages:

- Reduced bias.
- Better phases.

Disadvantages:

- Slow.
- PIRATE in particular works well for some cases and badly for others.
Density Modification
Kevin Cowtan, York.

Some results...
DM vs Parrot

Map correlations

Parrot: No new features enabled.
Parrot: Rice vs MLHL

Comparing old and new likelihood functions.

Map correlations
Parrot: Isotropic vs Anisotropic

Comparing with and without anisotropy correction.

Map correlations.
Parrot: simple vs NCS averaged

Comparing with and without NCS averaging.

Map correlations
DM vs PARROT vs PIRATE

% residues autobuilt and sequenced
50 JCSG structures, 1.8-3.2Å resolution

74.2%  78.4%  79.1%
DM      PARROT    PIRATE
DM vs PARROT vs PIRATE

Mean time taken
50 JCSG structures, 1.8-3.2Å resolution

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DM</td>
<td>6</td>
</tr>
<tr>
<td>PARROT</td>
<td>10</td>
</tr>
<tr>
<td>PIRATE</td>
<td>887</td>
</tr>
</tbody>
</table>
Buccaneer

The buccaneer software for automated model building of protein structures across a broad range of resolutions.

Kevin Cowtan
YSBL, University of York
kevin.cowtan@york.ac.uk
X-ray structure solution pipeline...

- Data collection
- Data processing
- Experimental phasing
- Molecular Replacement
- Density Modification
- Model building
- Refinement
- Rebuilding Validation
X-ray structure solution pipeline…
in CCP4i2 context
Model Building

Model building software:
• Buccaneer
Buccaneer

Statistical model building software based on the use of a reference structure to construct likelihood targets for protein features.

- 1.0 release in 2008
  - Tracing, sequencing, ncs and refinement
- 1.2 release (1.1.9) in CCP4 6.1.2
  - Faster, better sequencing
- 1.4 release in CCP4 6.1.13
  - MR support, known structure, command line
- 1.5 release in CCP4 6.2
  - Model tidying
  - Minor sequencing improvements
- 1.6 release from CCP4 6.3 (current 1.6.3)
  - Different MR modes
Buccaneer: Method

- Compare simulated map and known model to obtain likelihood target, then search for this target in the unknown map.
Buccaneer: Method

- Compile statistics for reference map in 4Å sphere about \( C_\alpha \) => LLK target.

- Use mean/variance.

4Å sphere about \( C_\alpha \) also used by 'CAPRA' Ioeger et al. (but different target function).
Buccaneer

Use a likelihood function based on conserved density features.

The same likelihood function is used several times. This makes the program very simple (<3000 lines), and the whole calculation works over a range of resolutions.

Finding, growing: Look for C-alpha environment

(4.0A sphere about Cα)

Sequencing: Look for C-beta environment

(5.5A sphere about Cβ)

ALA  CYS  HIS  MET  THR  ...

x20
Buccaneer

10 stages:
- **Find** candidate C-alpha positions
- **Grow** them into chain fragments
- **Join** and merge the fragments, resolving branches
- **Link** nearby N and C termini (if possible)
- **Sequence** the chains (i.e. dock sequence)
- **Correct** insertions/deletions
- **Filter** based on poor density
- **NCS Rebuild** to complete NCS copies of chains
- **Prune** any remaining clashing chains
- **Rebuild** side chains
Case Study:

A difficult loop in a 2.9Å map, calculated using real data from the JCSG.
Find candidate C-alpha positions
Grow into chain fragments
Join and merge chain fragments
Sequence the chains
Correct insertions/deletions
Prune any remaining clashing chains
Rebuild side chains
Comparison to the final model
Buccaneer

Model completion uses “Lateral growing”:
Grow sideways from existing chain fragments by looking for new C-alphas at an appropriate distance “sideways” from the existing chain:
Unmodeled density
Lateral growing likelihood function
New C-alpha candidates
Resulting model
Buccaneer: Results

Model completeness not very dependent on resolution:
Buccaneer: Results

Model completeness dependent on initial phases:
Buccaneer
Buccaneer (MR)

- different modes to balance trade-off between model bias and stabilizing of calculation with poor data
Buccaneer

What you need to do afterwards:

- Tidy up with Coot.
  - Or ARP/wARP when resolution is good.
  - Buccaneer+ARP/wARP better+faster than ARP/wARP.

- Typical Coot steps:
  - Connect up any broken chains.
  - Use density fit and rotamer analysis to check rotamers.
  - Check Ramachandran, MolProbity, etc.
  - Add waters, ligands, check unmodeled blobs...
  - Re-refine, examine difference maps.
Buccaneer: Future

Buccaneer 1.7
- Loop building (sloop)

Buccaneer 1.8
- Reworked joining and correction code

Buccaneer 1.9
- CIS peptides? GLY conformations? NCS?
Buccaneer: Summary

A simple, (i.e. MTZ and sequence), very fast method of model building which is robust against resolution. User reports for structures down to 3.7Å when phasing is good. Results can be further improved by iterating with refinement in refmac (and in future, density modification). Proven on real world problems.

Use it when resolution is poor or you are in a hurry. If resolution is good and phases are poor, then ARP/wARP may do better. Best approach: Run both!
Nautilus (poly-nucleotide building)
Nautilus (available in CCP4i)
Coot integration

- Cootaneer: sequencing part of Buccaneer
- Fast secondary structure finding
- Cootilus: automated nucleic acid finding/building ([https://www.youtube.com/watch?v=QGN6tF-zKOE](https://www.youtube.com/watch?v=QGN6tF-zKOE))
- DB loop/Fitting from Database: loop/fragment building based on SLoop
Acknowledgments

Help:

- JCSG data archive: www.jcsg.org
- Garib Murshudov, Raj Pannu, Pavol Skubak
- Eleanor Dodson, Paul Emsley, Randy Read, Clemens Vonrhein

Funding:

- The Royal Society, BBSRC, CCP4
Buccaneer

Cycle BUCCANEER and REFMAC for most complete model

Loop building tool

Nucleotide building tool