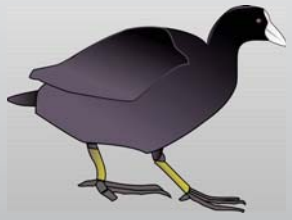
Model Building

An Introduction to *Coot*



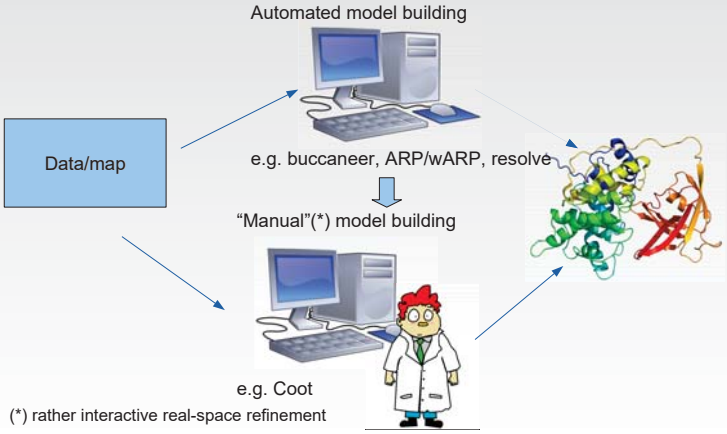
Bernhard Lohkamp

Karolinska Institutet, Sweden

Lund June 20221

Model-Building

- Place protein atoms in (best) electron density



Model-Building

Kendrew (1957)

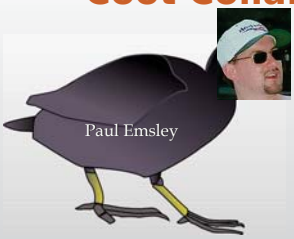
Rubin

... but why bother?

- In the days of **Automation**
  - why build something **Interactive**?
- Automated model-building for complete models is still impossible
  - It takes a brain to validate
- Concerted correction/improvement of a model is difficult on the larger scale
- Coot is built with Novice users in mind
  - (but not exclusively)

It's fun!!!

Coot Collaborators



Bernhard Lohkamp

Kevin Cowtan

Eugene Krissinel

Stuart McNicholas

Martin Noble

Alexei Vagin

Coot

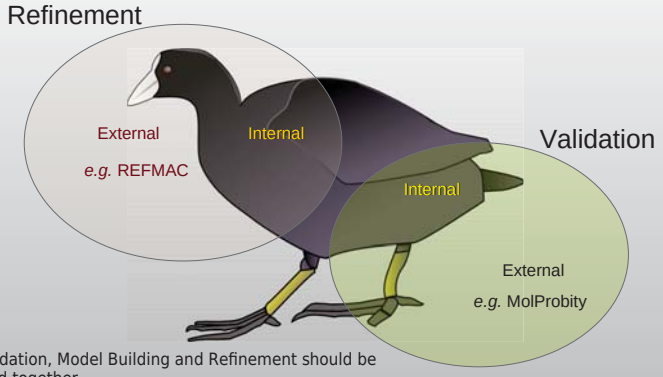
- Molecular Graphics application
  - Protein crystallographic(*) model-building tools (Crystallographic Object-Oriented Toolkit)
  - Aims:
    - Model building, completion, validation
    - "Slick and powerful" interface to CCP4
- Interface to other programs: SHELXL, Refmac, AceDrg, Probe&Reduce (MolProbity), EBI, EDS, Povray, Raster3D, PHENIX, ...
- Several model-building and validation tools

(*) EM tools now too

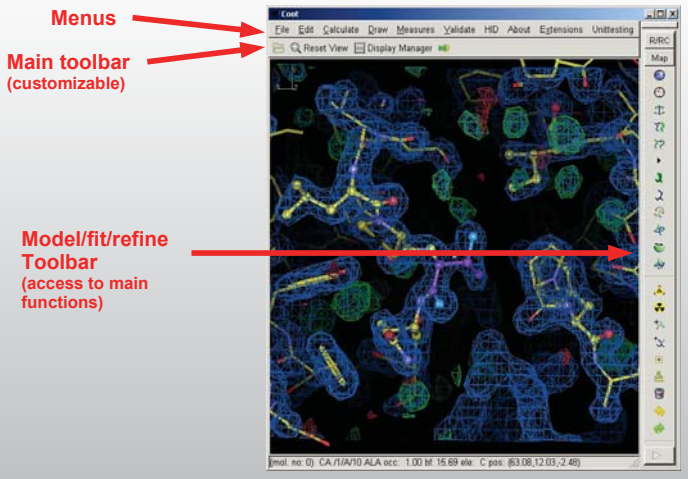
(some) Coot features

- displays maps and atomic macromolecular models
- allows model manipulation, e.g.
  - structure idealization & real space refinement
  - manual rotation/translation, rigid body fitting & rotamer selection
  - ligand search, solvation, mutations
  - Ramachandran plots, non-crystallographic symmetry (NCS)...
- interfaced to other programs
  - Refinement
  - Phasing
  - Model validation and analysis

Feature Integration



Validation, Model Building and Refinement should be used together



What is "Refinement"?

- The adjustment of model parameters (coordinates) so that the calculated structure factors match the observations (map) as nearly as possible
  - In "one-shot" real-space refinement, such as in Coot, this translates to:
    - move the atoms into as high density as possible while minimizing geometrical distortions

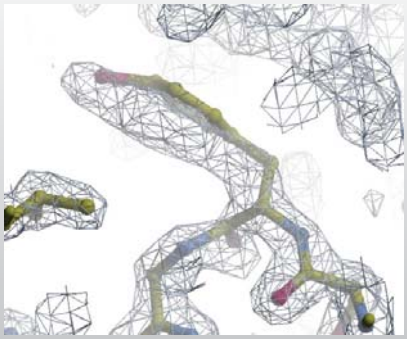
Refinement
real vs reciprocal space

- Real Space
  - Refinement against electron density (map; E_{map})
- Reciprocal Space
  - Refinement against structure factors (E_{xray})

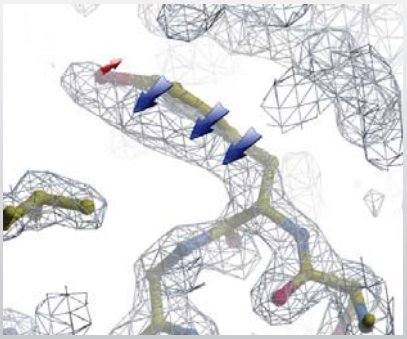
Minimize:

$$E_{total} = E_{map/xray} + w \cdot E_{geometry}$$

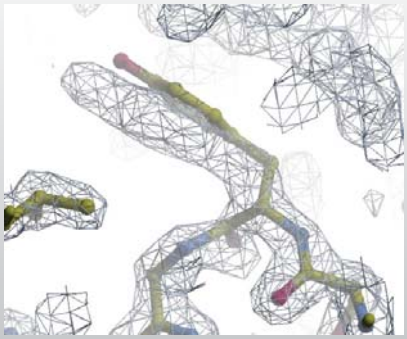
Distorted Geometry Pre-Refinement



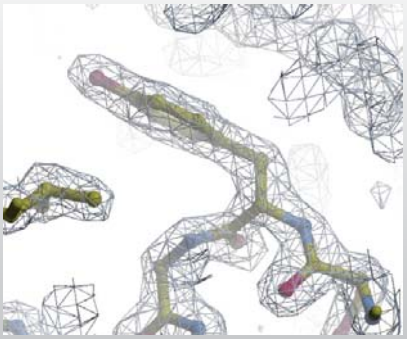
Refinement Gradients



Refinement: Cycle 3



Refinement Cycle 200: Minimized

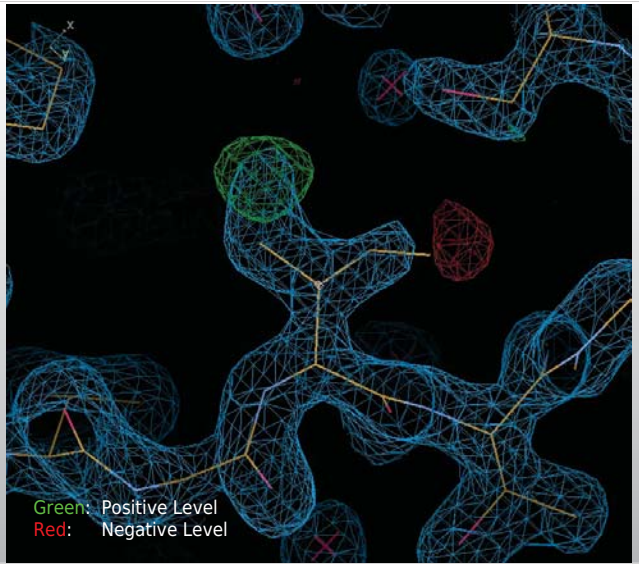


Real Space Refinement

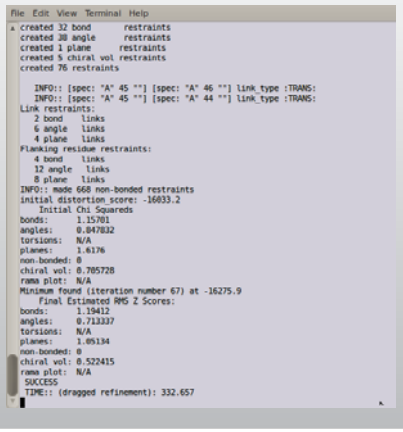
- Major feature of Coot — Diamond, R. (1971). *Acta Cryst*. A 27, 436-452.
  - Gradient minimizer (BFGS derivative)
  - Based on mmCIF standard dictionary
  - Minimizing bonds, angles, planes, non-bonded contacts, torsions, [chiral volumes, Ramachandran]
- Provides "interactive refinement"
  - Atom positions can be moved after refinement (white)
- Chi squares (easy evaluation of result)
- Threaded (0.9 onwards): — Fast & Animated
  - Target function and derivative evaluation, model and map all happen simultaneously now
  - => more atoms, smoother updates and/or closer to the minimum

Which map to use?

- Direct maps
  - Calculated from experimental amplitudes and phases inferred from diffraction or model
  - 2Fo-Fc, density modified map (experimentally phased, MR or refinement), usually blue
  - Covers the model (if Fo≈Fc => Fo)
- Difference map
  - Highlight errors in model
  - Fo-Fc, usually red (negative, i.e. too much model, Fc>Fo) and green (positive, i.e. not enough model, Fo>Fc)
- Composite map
  - Combination of above
  - e.g. 3Fo-2Fc ≈ 2Fo-Fc + Fo-Fc



Representation of Refinement Results:

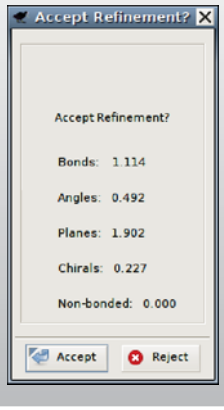


The first attempt

Student Reaction:

"Oh, I don't look at that window..."

Representation of Refinement Results:



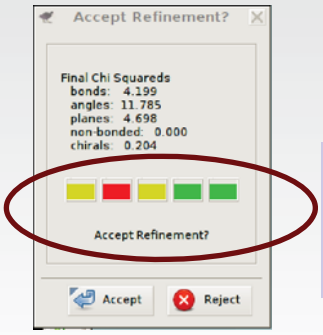
Second attempt...

Student Reaction:

"Oh, box of meaningless numbers.

Go away"

Refinement "Traffic Lights"



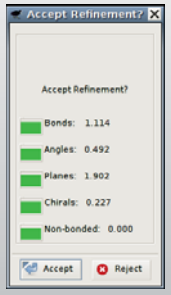
"Traffic Lights" represent the chi-squared values for each of the refined geometry types

e.g. bonds length (b):

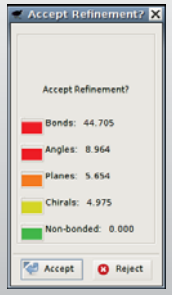
$$\chi^2_{bonds} = \frac{\sum (b_i - b_0)^2}{\sum \sigma_i^2}$$

Representation of Results: "Traffic Lights"

"Traffic Lights" represent the chi-squared values for each of the refined geometry types



Good refinement

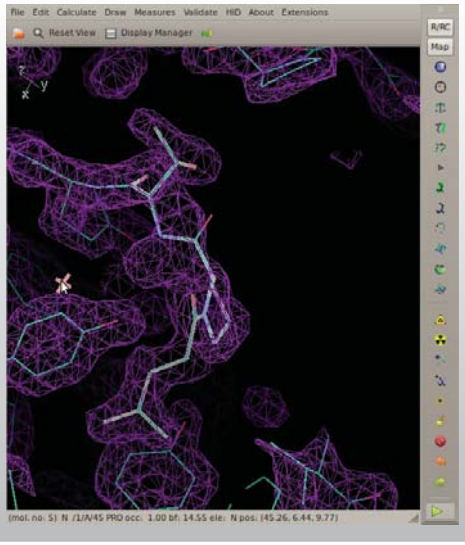


Bad refinement

Refinement Techniques

- Single-Atom Drag
- Key-bindings:
  - Triple Refine
  - Single Residue Refine with Auto-accept
- Ramachandran Refinement
- Sphere refinement
- Crankshaft Peptide Optimisation
- Coming at some point..?
  - VR (Hamish Todd)
  - AR

Wii/VR Refinement?



Single-drag

(Ctrl-left mouse)

(over dragging – out of density: not required any more usually)

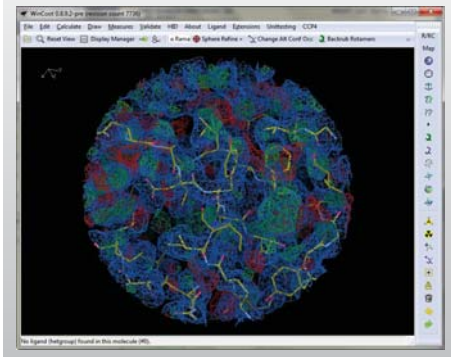
Coot Key-binding Crib-Sheet

"Sphere" Refinement

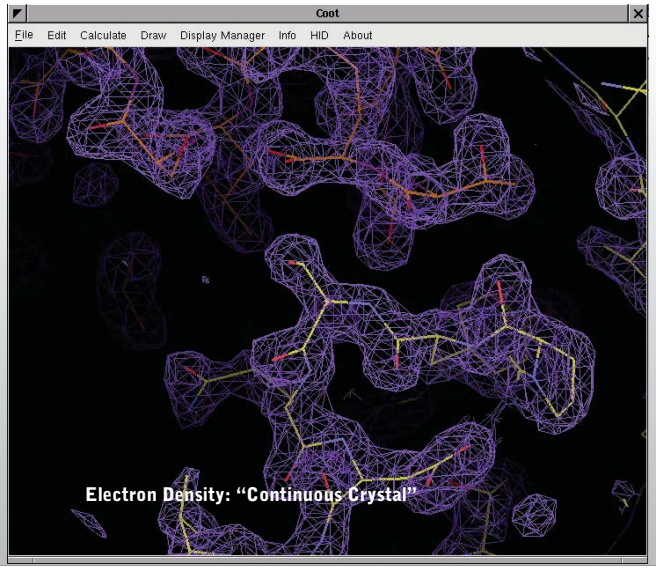
- Given an "Active"* Residues*
  - Define a sphere of residues around it and use them all for refinement
    - NOT just a linear selection
  - Residues from different chains (or different parts of the same chain) interact
  - Make CYS-CYS or glycosylation links as you find them
    - Use the group and link_list chem_link in the dictionary

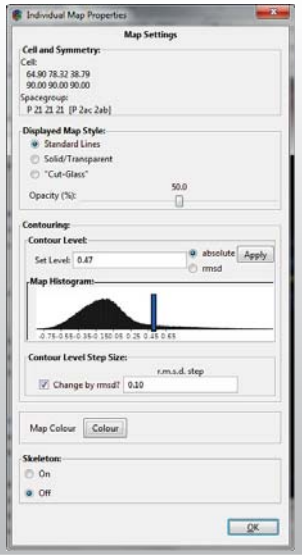
* Active residue = residue closest to the screen centre

Limit model and map display



- Map
  - Sphere
  - Use Preferences to set radius
- Coordinates
  - All
  - Within radius
  - Symmetry...

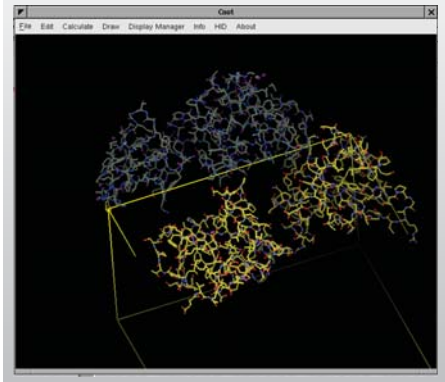




Map properties

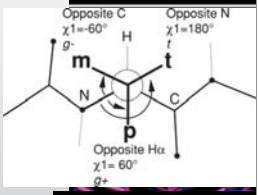
- Different display styles
  - Standard
  - Solid
  - Cut-glass
- Histogram (interactive)

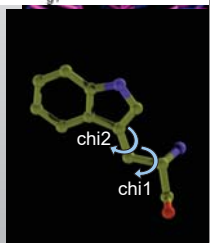
Symmetry



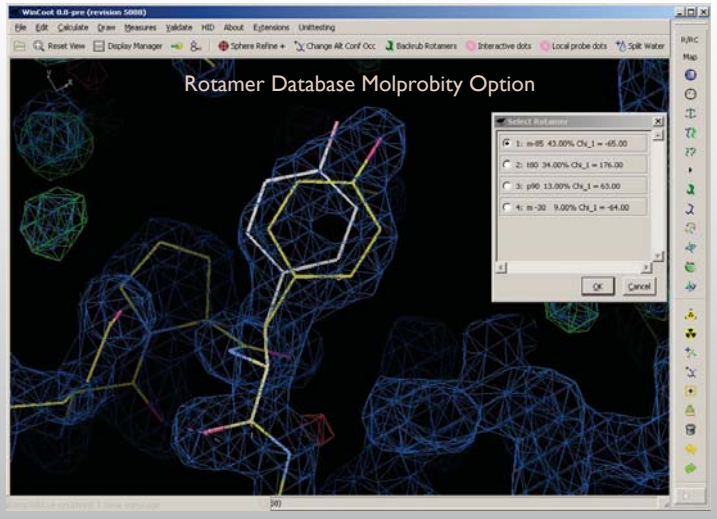
- Display
  - crystallographic symmetry copies
  - Unit cell
- Investigate crystal packing (voids?)
- Helps tracing/model building

Side-chain Rotamers



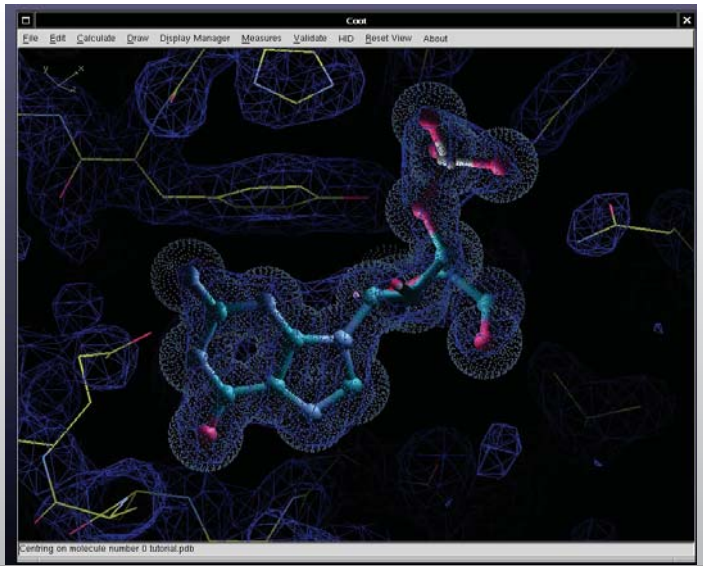


- Rotamers are preferred configurations of a side-chains rotatable bonds
- "preferred" means these configurations occur more frequently in a set of reference protein structures
- "preferred" because they are low-energy conformations (staggered rather than eclipsed)
- Several Rotamer "databases" exist



Some more Coot Tools…

- Add Terminal Residue
  - ϕ,ψ hypothesis scoring
- Alternate Conformations
- Ligand fitting/search
- Rigid-body Fitting
  - Steepest Descent
  - Simplex (slower but better)
- "Move Molecule Here"
- Water Search
- Fill-partial-residues (after MR)
- Dots, ball&stick representation



Low Resolution/Extra Tools

Extra Restraints....

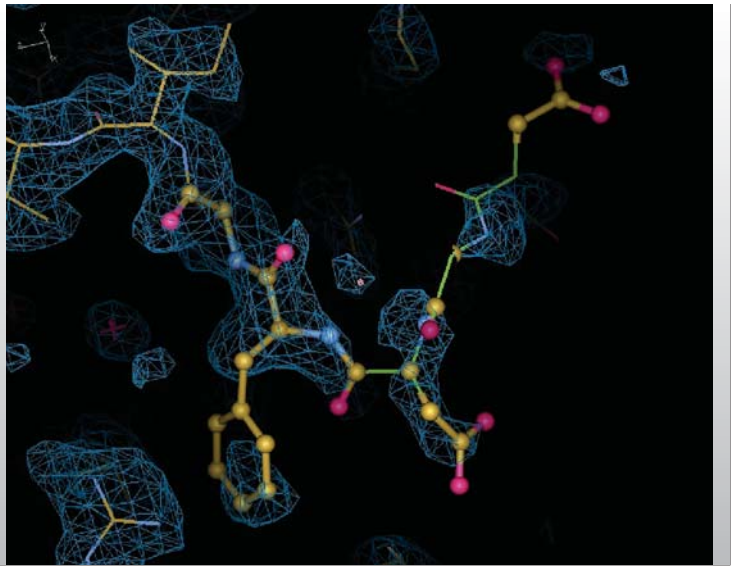
Extra Restraints....

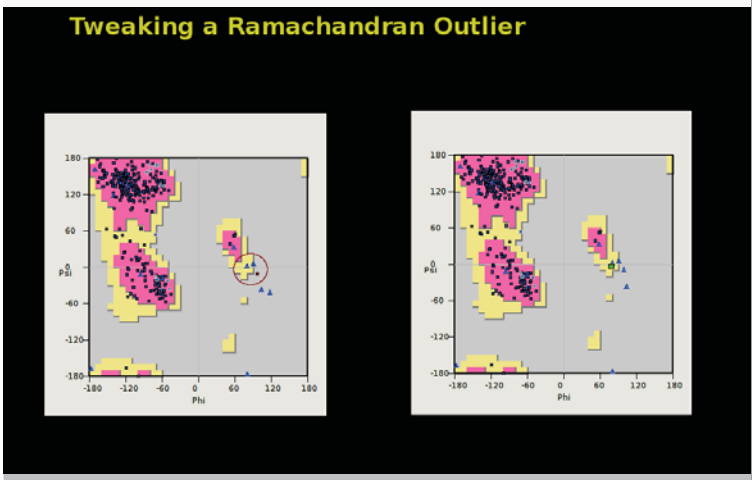
- Ramachandran restraints
- Secondary structure restraints
- Remove degree of freedom
  - Torsion angle restraints
  - Backrub rotamers
- Manually add restraints

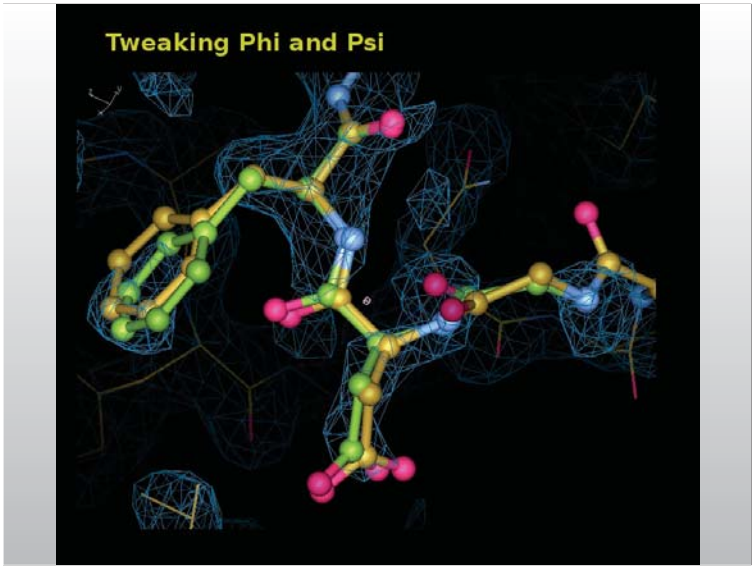
- Map sharpening
- [Jiggle fit]
- [Morphing]

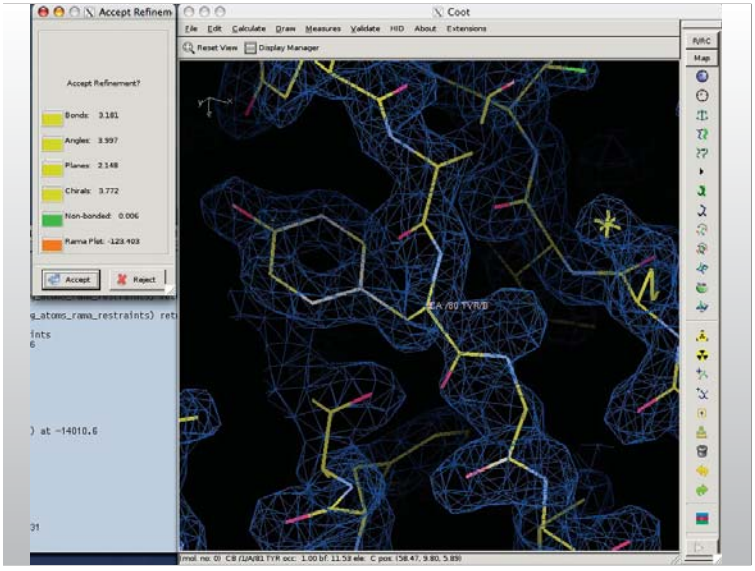
Ramachandran Restraints

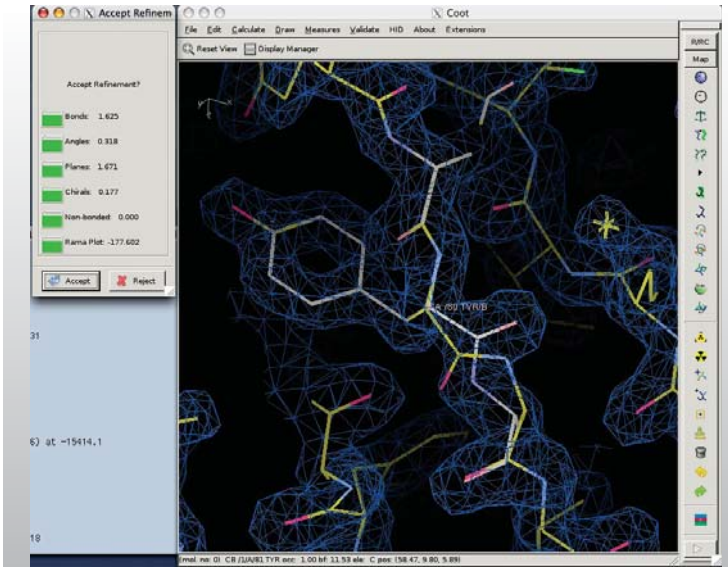
- Scenario:
  - I have a loop, with poor density, I know the atoms are there somewhere and I want to provide a “reasonable” model
- Controversial Feature?
  - Ramachandran Plots have been used for “validation” - but here we are deliberately optimizing them
- Ramachandran Plots can be added to the geometry target function











Ramachandran Restraints

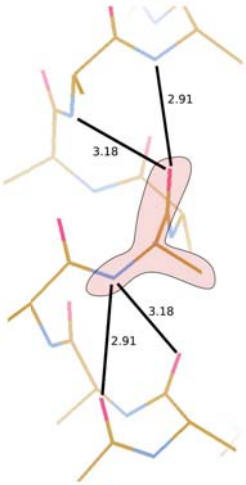
- Controversial?
  - "… the Ramachandran Plot is one of the simplest and most sensitive means for assessing the quality of a protein model…"
    - Gerard Kleywegt & Alwyn Jones (1996)
- But to quote Jane Richardson:
  - Do you want a better structure – or a better idea of the quality of your structure?

Adding Torsion Angle Restraints

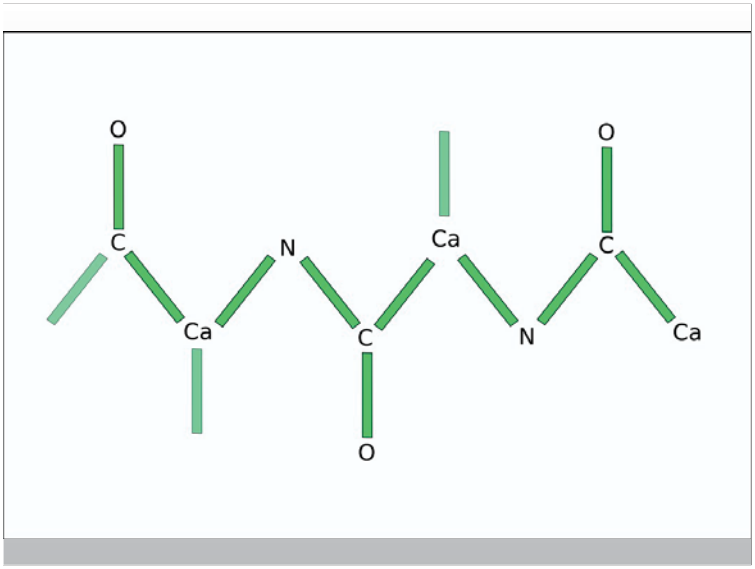
- Torsion angle refinement is slow (relatively)
  - Simple addition of these restraints to the geometry target function
    - often makes the region "stuck and unsatisfied"
    - i.e. trapped in local minimum

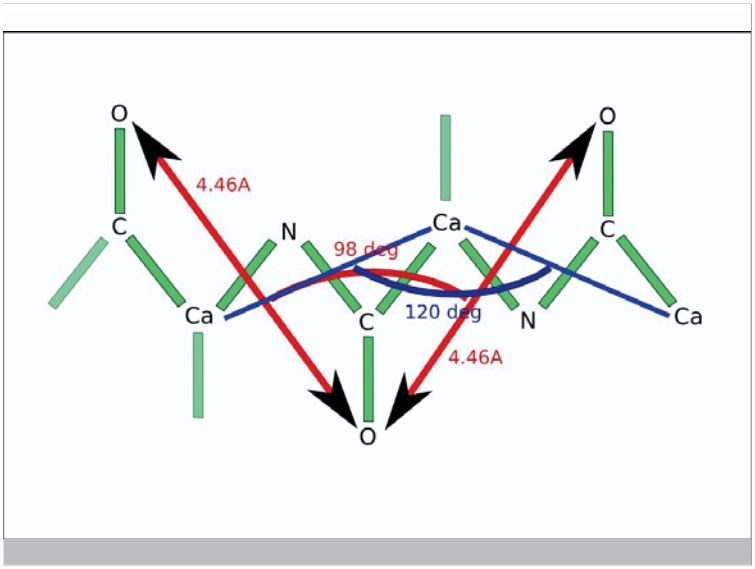
- Add Pseudo-bonds

Alpha Helix pseudo-bond restraints

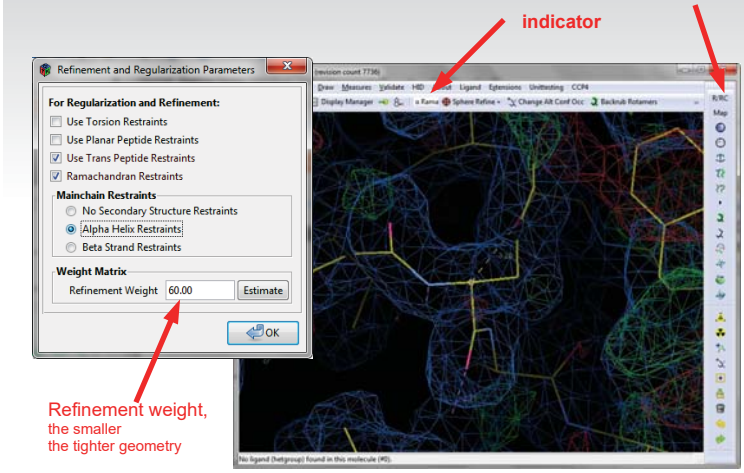


Restrain the Hydrogen-bonding atom distances



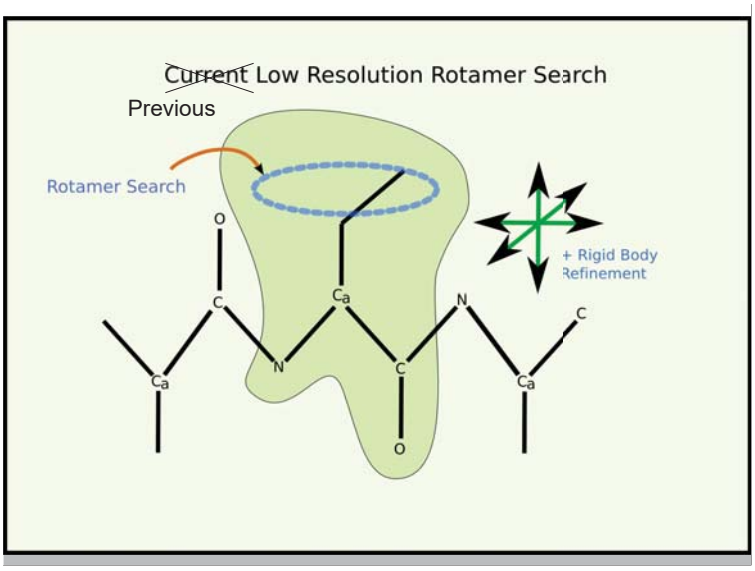


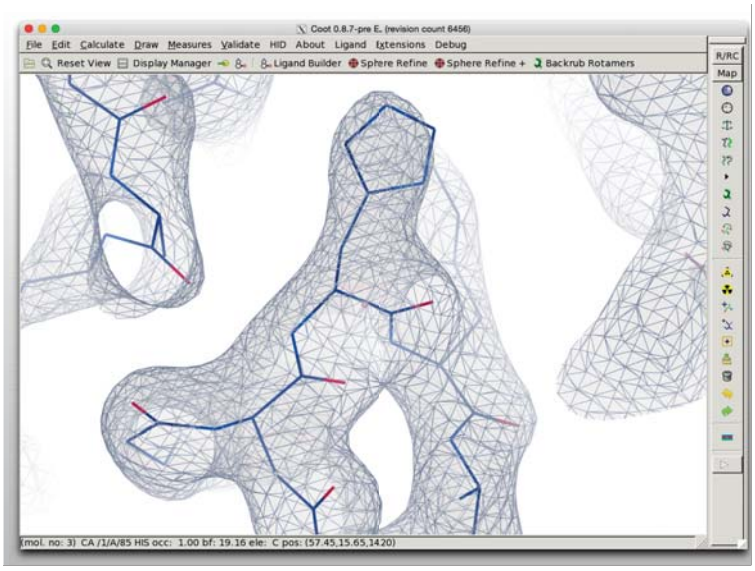
Activate restraints

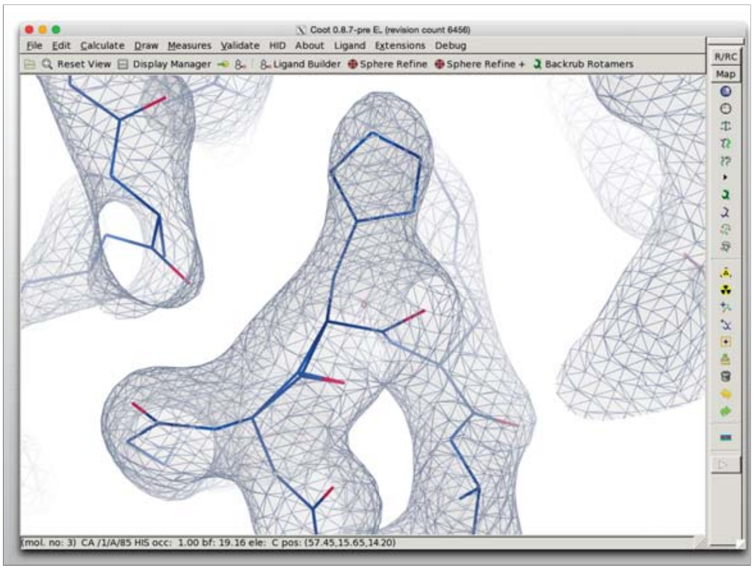


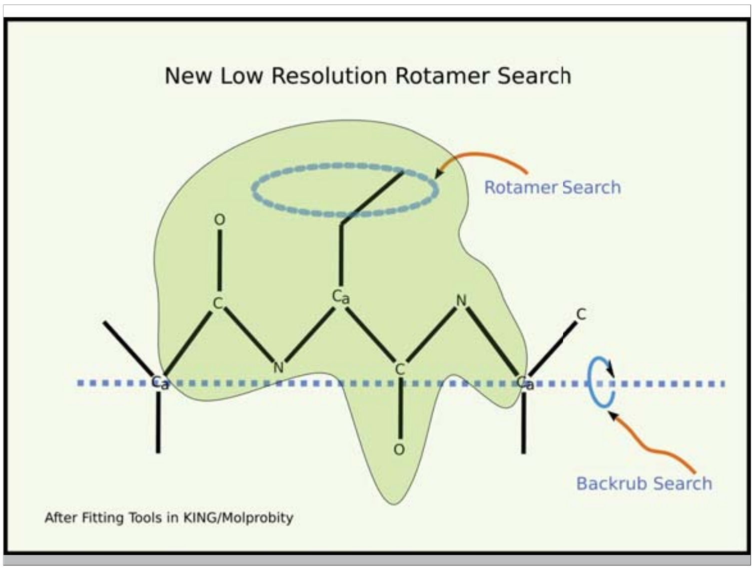
”Backrub Rotamers”

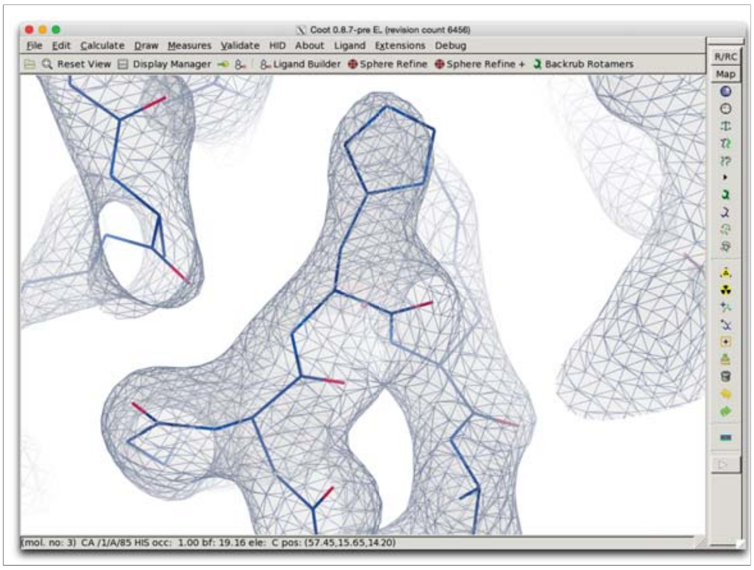
- High probability models with low resolution data

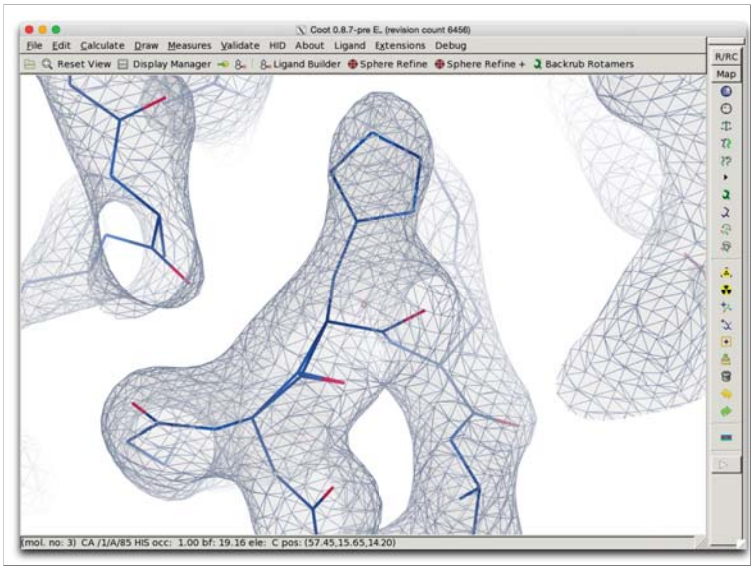


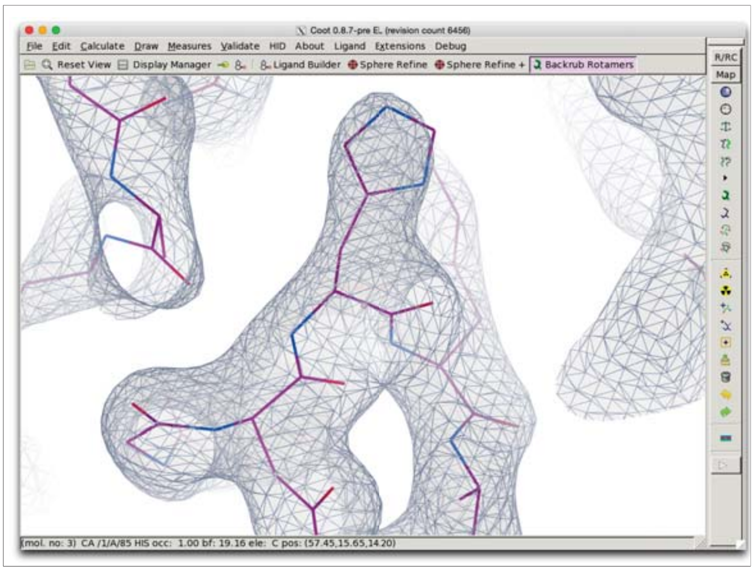


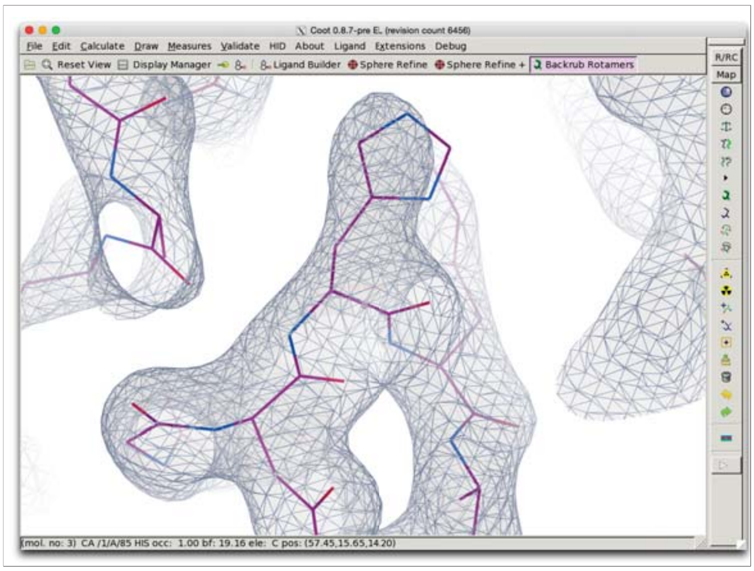

Coot 0.8.7-pre EL (revision count 6456)
File Edit Calculate Draw Measures Validate HID About Ligand Extensions Debug
Reset View Display Manager Ligand Builder Sphere Refine Sphere Refine + Backrub Rotamers
R/RC
Map
(mol. no: 3) CA /1/A/85 HIS occ: 1.00 bf: 19.16 ele: C pos: (57.45,15.65,14.20)

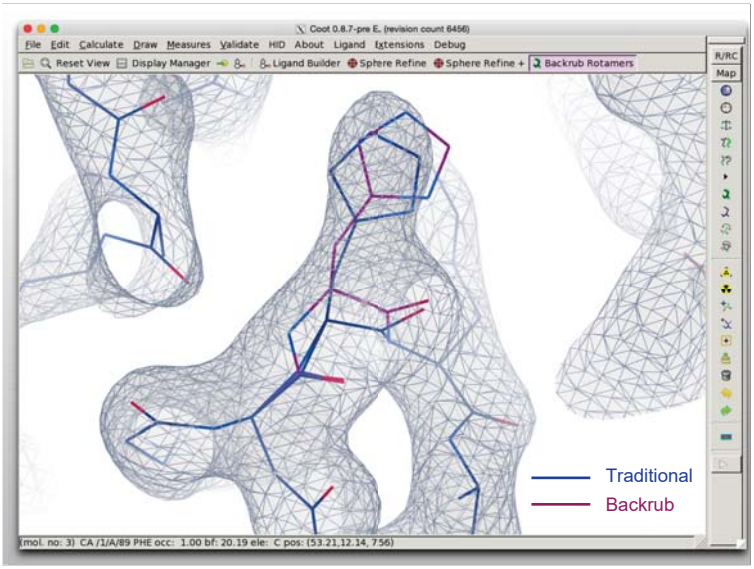

New Low Resolution Rotamer Search
Rotamer Search
O
C
Ca
N
C
N
Ca
C
O
Backrub Search
After Fitting Tools in KING/Molprobity


Coot 0.8.7-pre EL (revision count 6456)
File Edit Calculate Draw Measures Validate HID About Ligand Extensions Debug
Reset View Display Manager Ligand Builder Sphere Refine Sphere Refine + Backrub Rotamers
R/RC
Map
(mol. no: 3) CA /1/A/85 HIS occ: 1.00 bf: 19.16 ele: C pos: (57.45,15.65,14.20)


Coot 0.8.7-pre EL (revision count 6456)
File Edit Calculate Draw Measures Validate HID About Ligand Extensions Debug
Reset View Display Manager Ligand Builder Sphere Refine Sphere Refine + Backrub Rotamers
R/RC
Map
(mol. no: 3) CA /1/A/85 HIS occ: 1.00 bf: 19.16 ele: C pos: (57.45,15.65,14.20)


Coot 0.8.7-pre EL (revision count 6456)
File Edit Calculate Draw Measures Validate HID About Ligand Extensions Debug
Reset View Display Manager Ligand Builder Sphere Refine Sphere Refine + Backrub Rotamers
R/RC
Map
(mol. no: 3) CA /1/A/85 HIS occ: 1.00 bf: 19.16 ele: C pos: (57.45,15.65,14.20)


Coot 0.8.7-pre EL (revision count 6456)
File Edit Calculate Draw Measures Validate HID About Ligand Extensions Debug
Reset View Display Manager Ligand Builder Sphere Refine Sphere Refine + Backrub Rotamers
R/RC
Map
(mol. no: 3) CA /1/A/85 HIS occ: 1.00 bf: 19.16 ele: C pos: (57.45,15.65,14.20)



To turn it on...

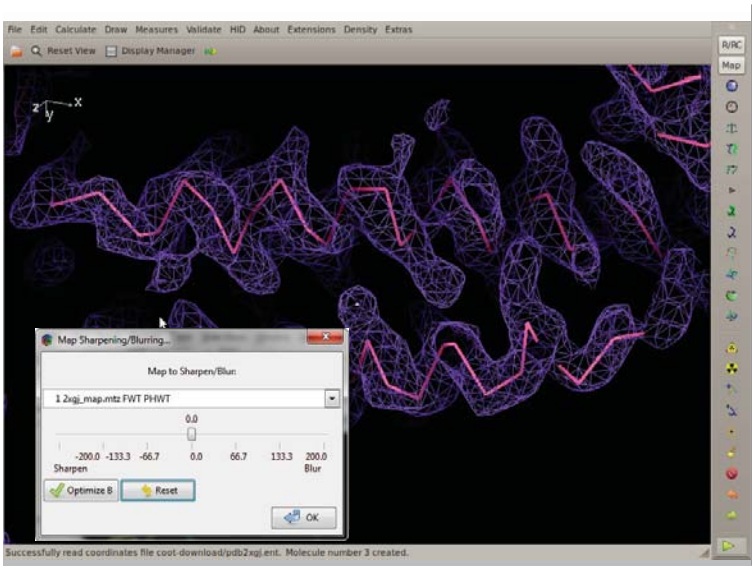
- automatically for resolution > 2.9Å
- via
  - Extensions → Modelling → Rotamer Search
  - scripting:
    - guile: `(ROTAMERSEARCHLOWRES)`
    - python: `set_rotamer_search_mode(ROTAMERSEARCHLOWRES)`
  - toolbutton

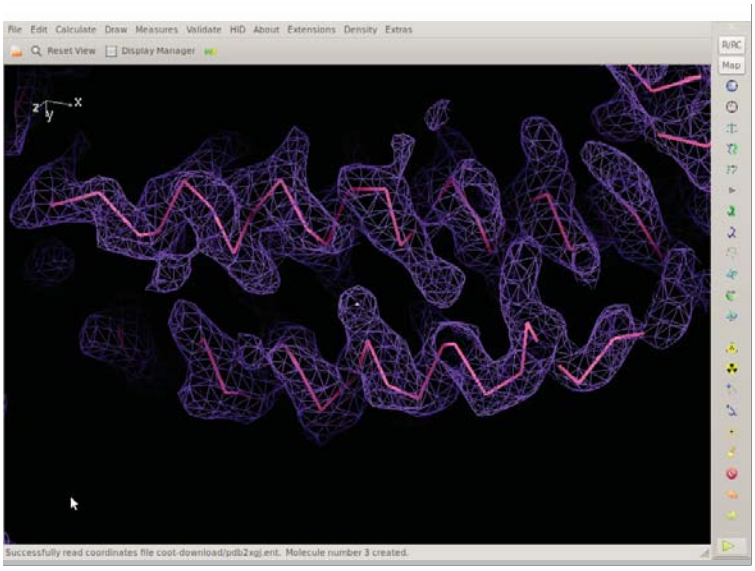
Map Sharpening

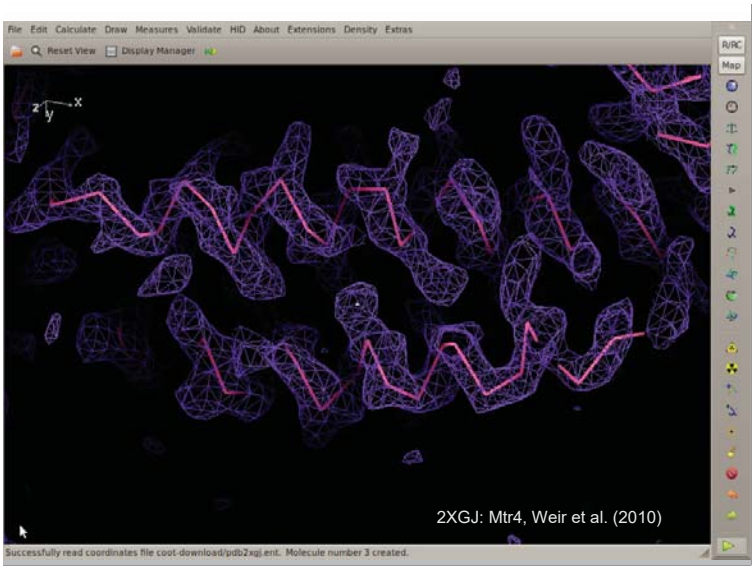
Which B-factor shall I use to get the most interpretable map?

Interactively adjust the structure factor amplitudes and re-generate the map with FFT and recontouring...

Try to optimise using map kurtosis



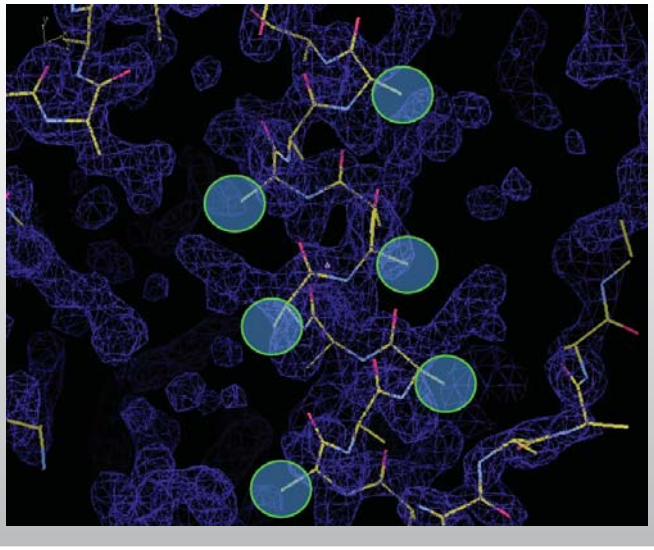


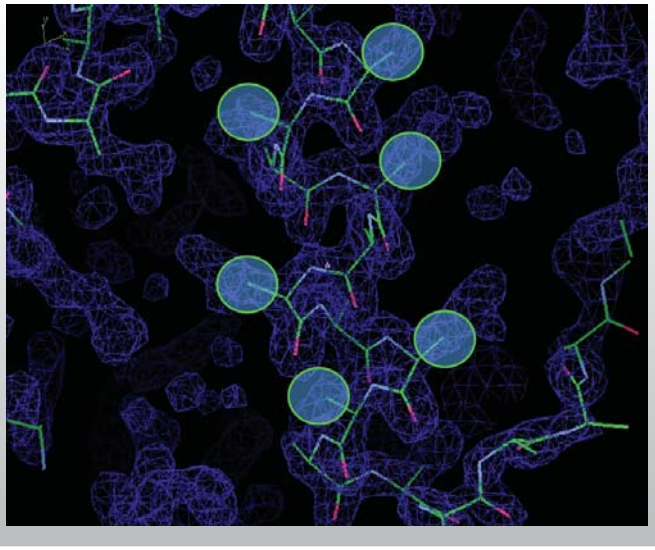


Secondary structure building

Alpha Helix Placement

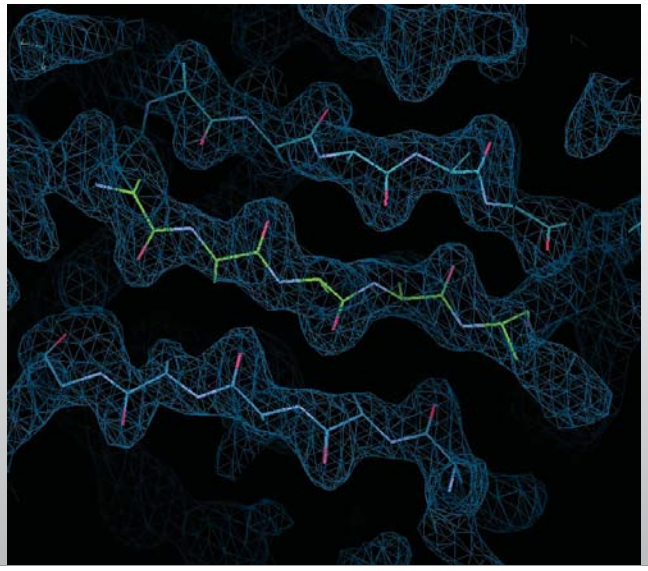
- Scenario: Looking at a new map, not built with automatic tools:
  - “I can see that there’s a helix here - build it for me!”
- From a given point:
  - Move to local averaged maximum
  - Do a 2D MR-style orientation search on a cylinder of electron density
  - Build a helix (both directions)
  - 1D Rotation search to find best fit
  - Score based on density at CB positions
  - Trim ‘n Grow



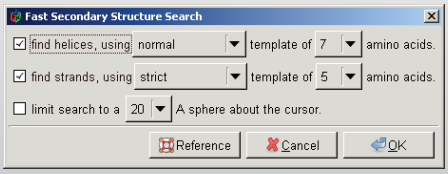


Strand Placement

- Similar but unlike Helices, Strands have to be treated as non-idealized
  - Repeating a single phi/psi value doesn't make a structure that fits “real-world” density
- Curvature of strands should be taken into account
  - Use selections from a “database” of good structures

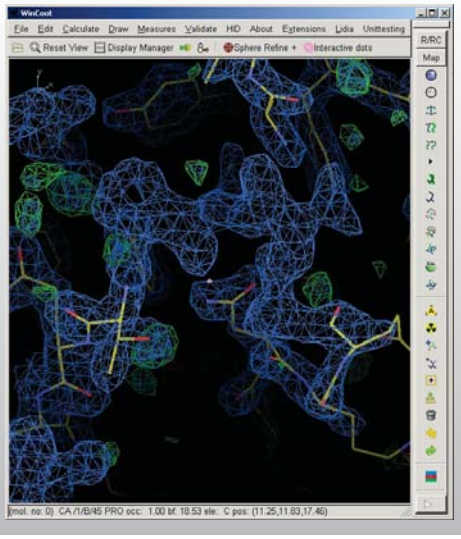


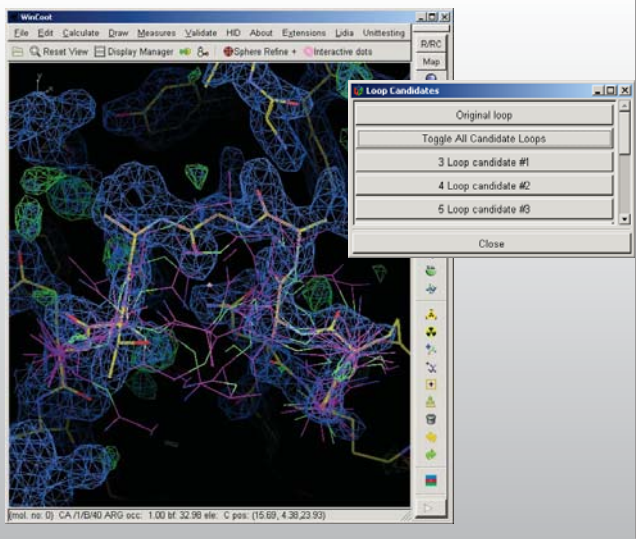
Automated Fast Secondary Structure Search



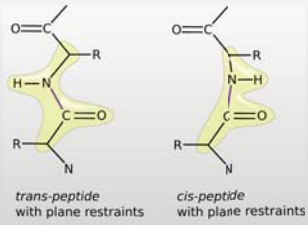
Loop fitting

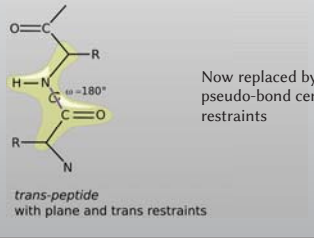
- Simple loop fitting
  - Add residue by residue (from both termini)
- DB loop
  - Fitting fragments from database





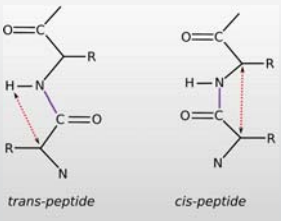
cis-Peptides

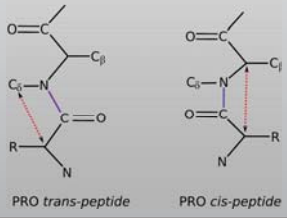




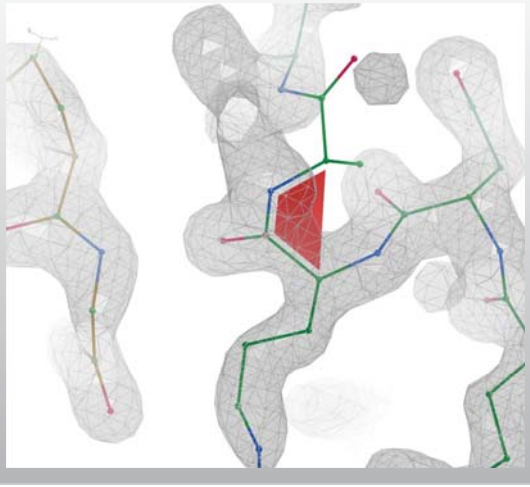
Now replaced by trans-peptide pseudo-bond centre distance restraints

cis-Peptides





cis-peptide Representation



Pre-PRO

Twisted-trans

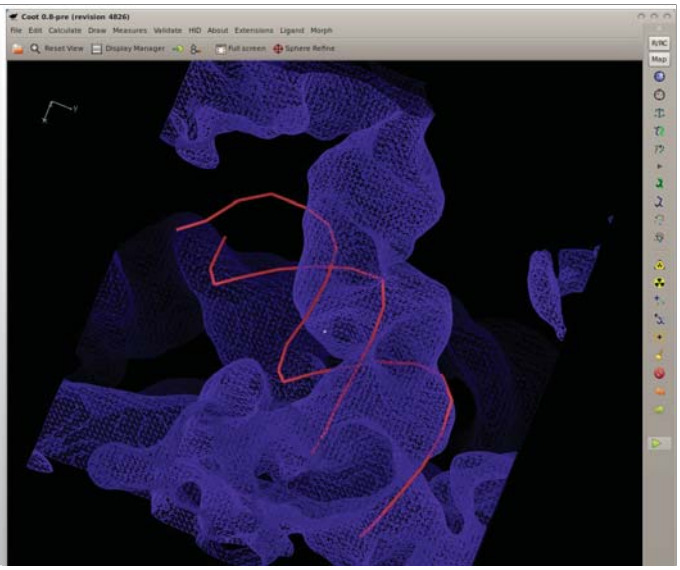
Non-pre-PRO

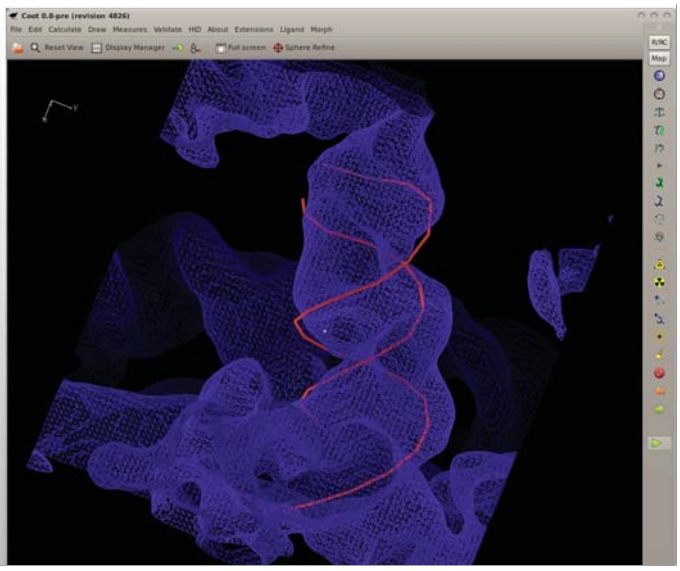
Jiggle Fit

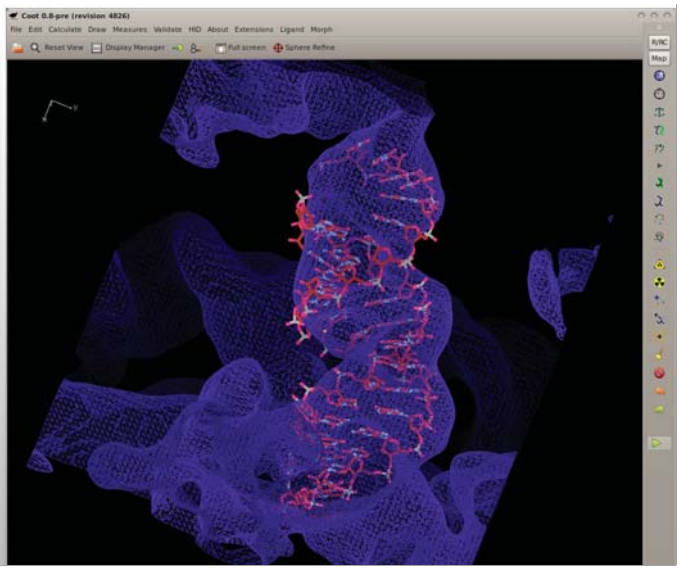
- How do I rotate and translate these atoms to fit the density?
  - 6-dimensional problem
- Originally used to fit simple ligands/solvent molecules to blobs of density
- Now extended to fit arbitrary atom selections
  - *e.g.* by Chain

Jiggle Fit: How it Works

- Loop 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



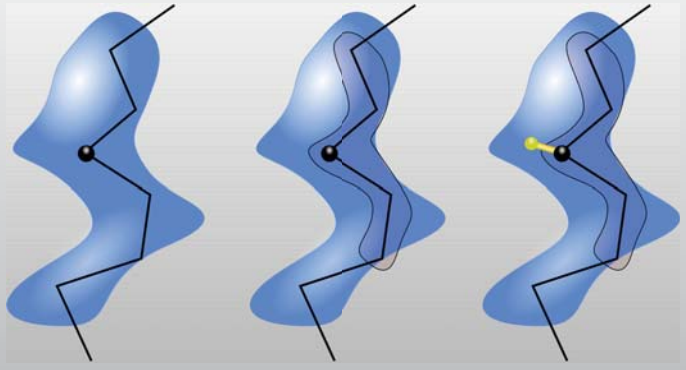




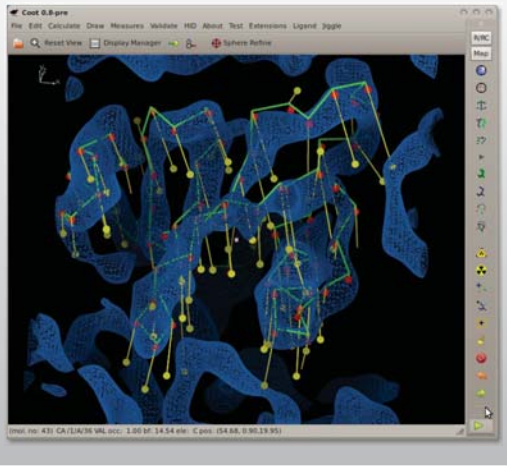
Model Morphing: How it Works

- For each residue in a chain, we ask:
  - where does a small fragment centred on this residue want to go?
  - (Robust) average the transformations and apply them on a per-residue basis
- Repeat

Model Morphing: Generating the Raw RTs

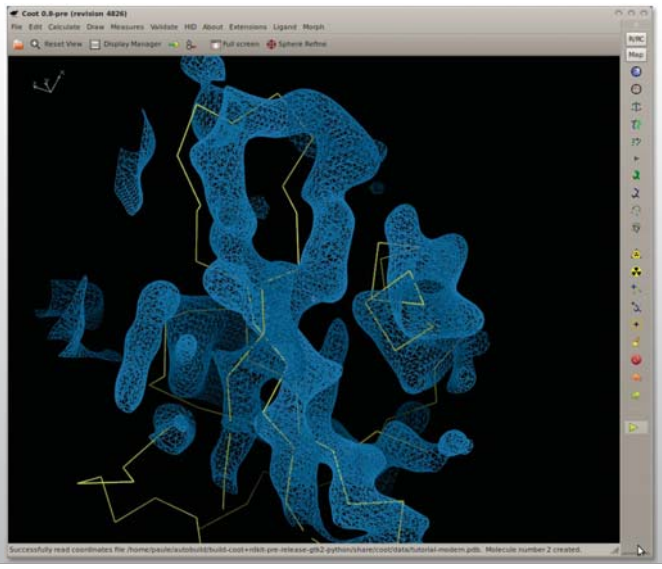


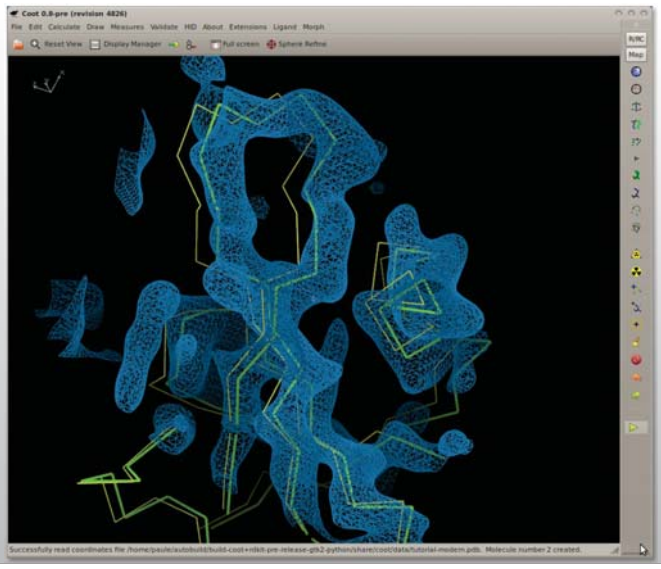
Model Morphing: Example

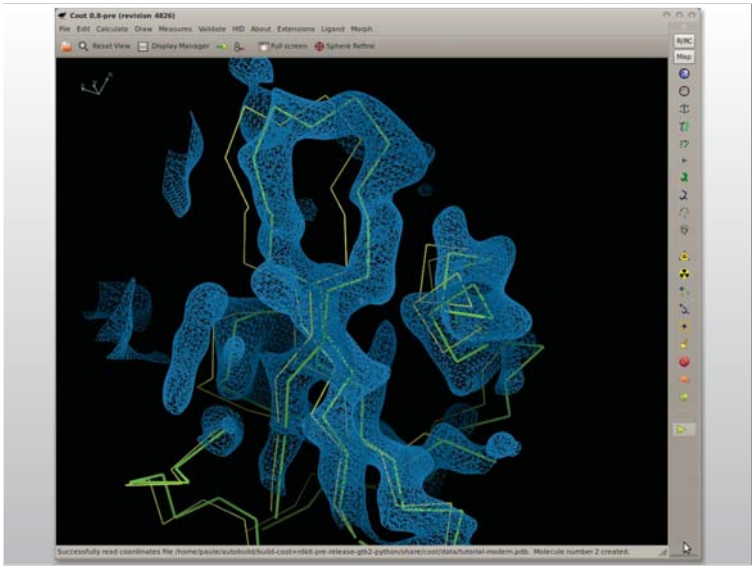


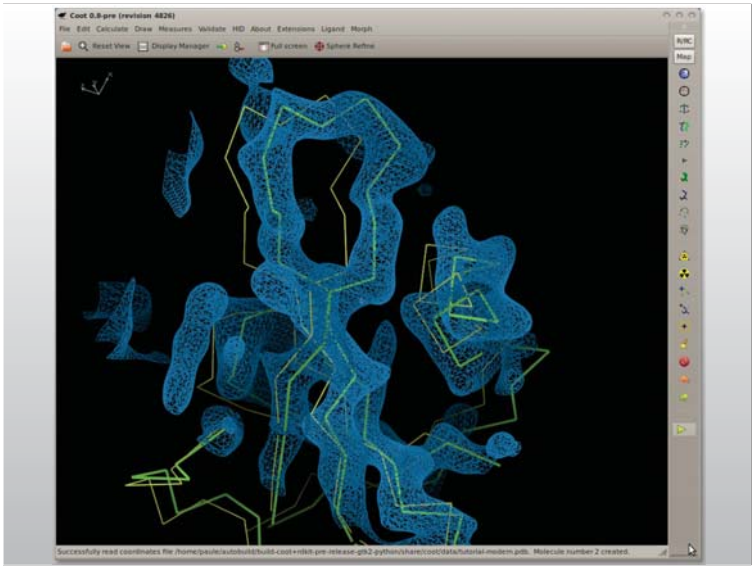
Model Morphing: Robust Averaging

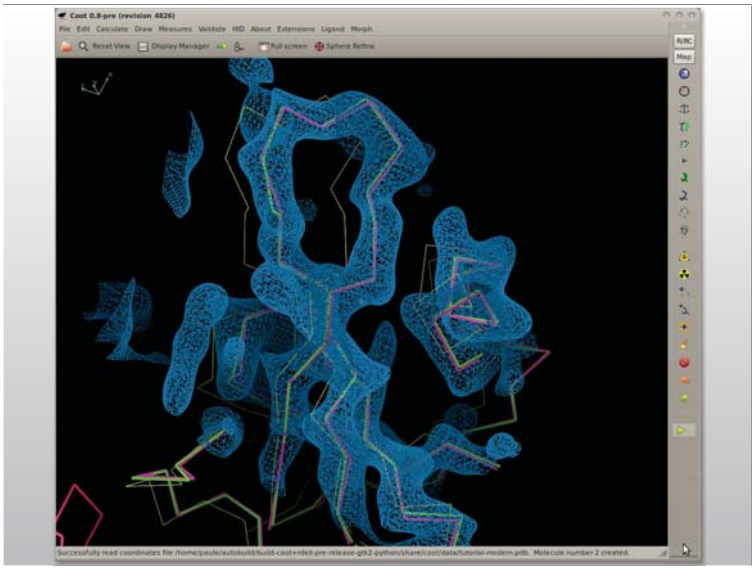
- What are the residues in the environment of a residue?
  - What are their RTs?
  - Create a metric 'distance', sort on that
  - Discard the top and bottom 20%
  - Use remaining RTs to generate average
  - ...which is then applied to central residue
- Repeat for all residues
- Larger environment radii make the shifts smaller/more conservative
  - More cycles needed



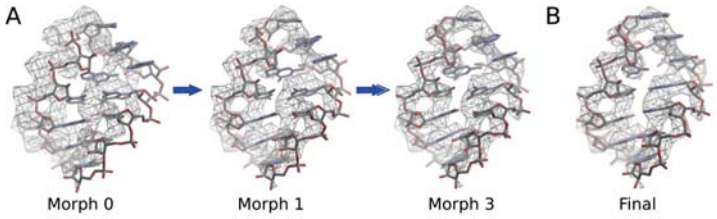








Model Morphing



CURLEW:
Coot Utilities and Refinement Library Extention Wranger

- Easy access to ”interesting” Coot scripts

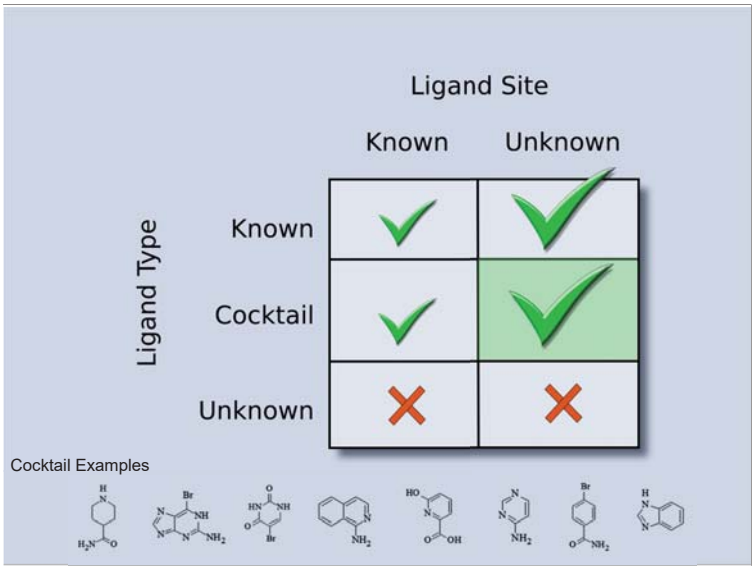
A Few Tools More...

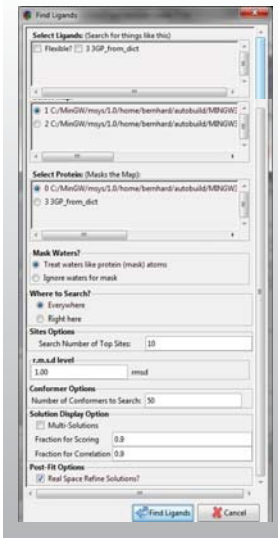
- More restraints
  - ProSMART
  - User defined
  - Planes, DNA (libg) etc.
- Carbohydrate-fitting
  - N-linked glycosylation
- Use of NCS
  - Copy chains
- Scripting

Ligands in Coot

- Ligand fitting
- Importing and building ligand from scratch
  - AceDRG, PRODRG, pyrogen, LIBCHECK
- Validation
- Representation
  - Surfaces
- Analysis
  - Molprobity, LIDIA

Ligand building



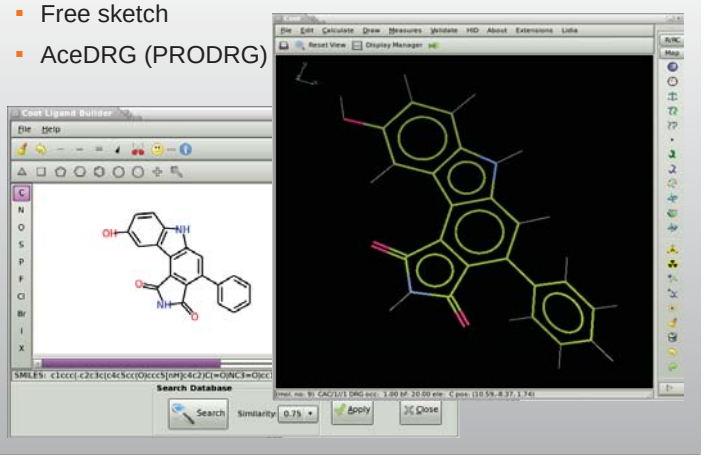


Ligand search in Coot

- Which ligand (flexibility?)
- Which map
- What to do with the protein (mask)
- Where to search
- How many sites to find
  - Acceptance levels
- Map level
- No of conformers
- Real space refine

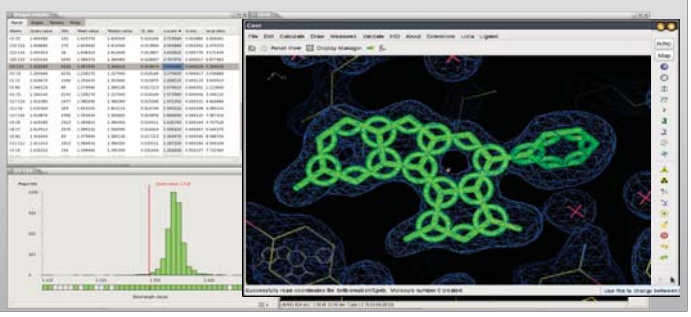
2D Ligand Builder

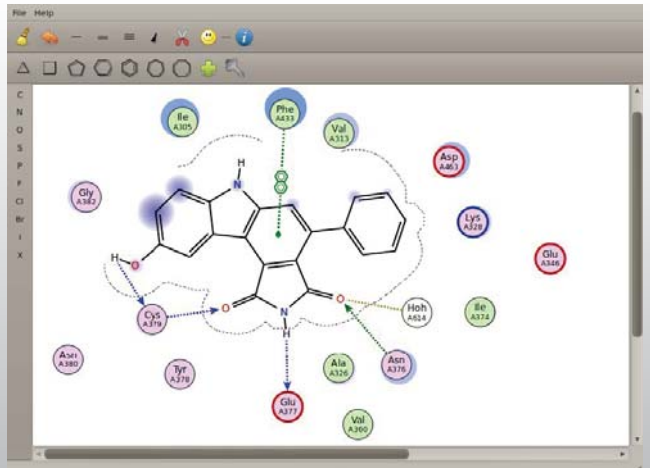
- Free sketch
- AceDRG (PRODRG)

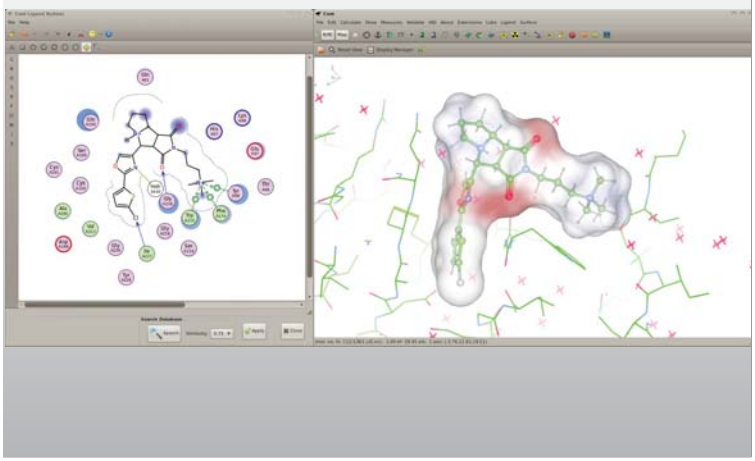


Ligand Validation

- Mogul plugin in Coot (distortions can be shown without)
  - Run mogul, graphical display of results
  - Update restraints (target and esds for bonds and angles)
  - Ligand distortion can be analysed without Mogul







A bit of validation and blobology

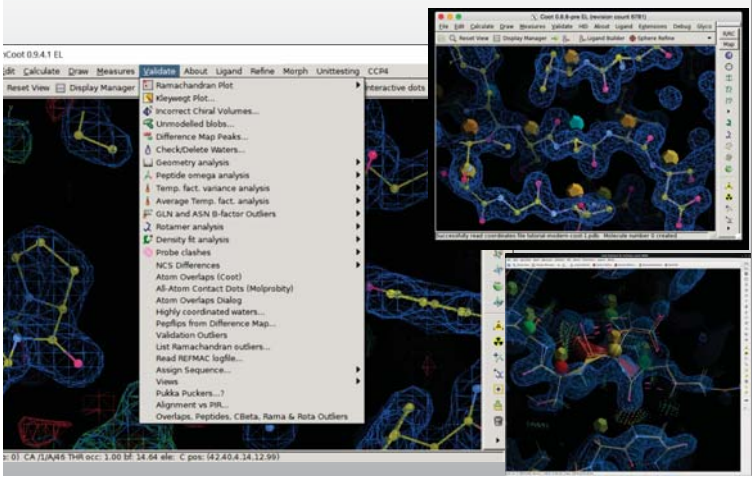
Validation of model only

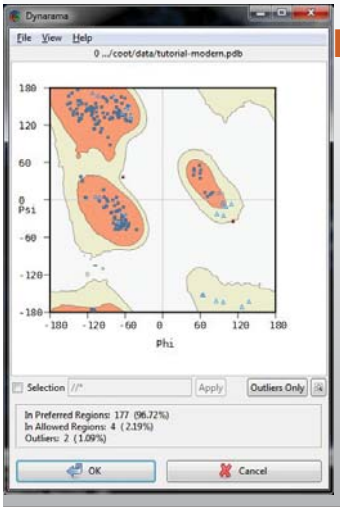
- Ramachandran Plot
  - Kleywegt Plot (NCS differences)
- Geometry Analysis
- Peptide ω Analysis
- Temperature Factor Analysis
- Rotamer Analysis
- Clashes

Validation of model fit to density

- Density Fit Analysis
- Difference Map Peaks
  - Variance analysis at water positions
- Unmodelled blobs

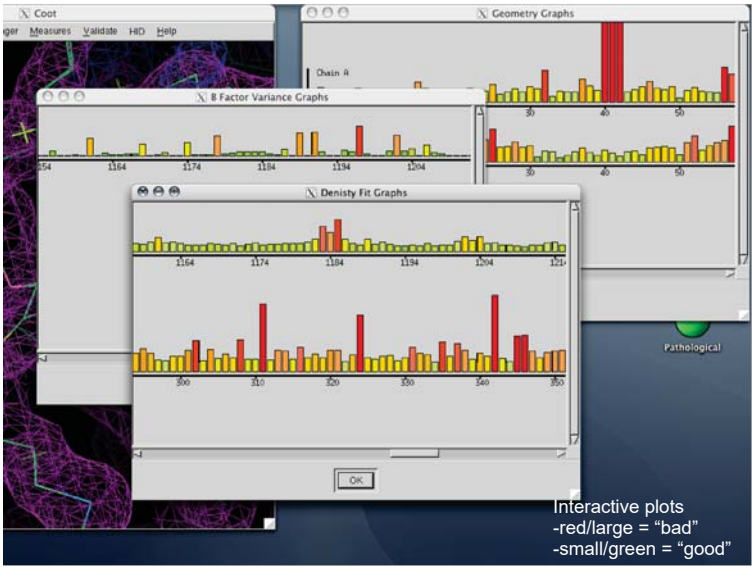
Validation tools in Coot





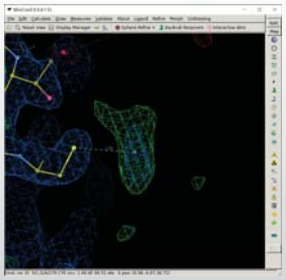
New Ramachandran plot

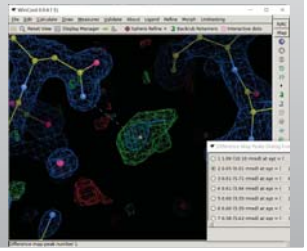
- Smooth outline
- Outliers only
- Selection only
- Save pdf
- Stand-alone
- Quick change to Kleywegt plot (incl. chains)



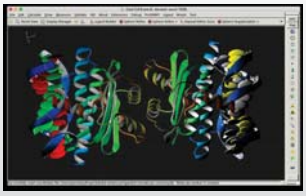
Blobology

- What can be there?
  - Crystallisation solution
  - Protein buffer
  - Purification buffers
  - Medium (and metabolites)
- Can it be there? i.e. do we have sensible interactions
  - Distance? (use pointer or environment distance
  - Correct chemistry?
  - Check symmetry
- Accept that not everything can be modelled but provide the best possible model



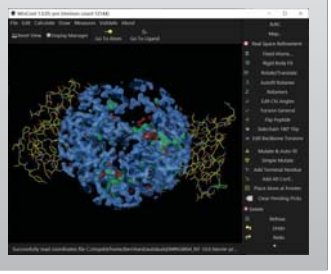


Coot Present, and Futures…



- Aim:
  - Slick, easy to use
  - Powerful
  - Smooth interface to external applications
- Under Development
  - Interesting things move quickly
  - There may be bugs

  Python 3, GTK3 – Coot 1.0



Further information

- Coot WIKI
  - http://strucbio.biologie.uni-konstanz.de/ccp4wiki/index.php/Coot
- Coot BB (mailing list)
  - http://www.jiscmail.ac.uk/lists/coot.html
- Coot documentation
  - http://www2.mrc-lmb.cam.ac.uk/personal/pemsley/coot/web/docs/
- YouTube
  - Various tutorial
  - https://www.youtube.com/c/PaulEmsley

Acknowledgements

- Paul Emsley
- Kevin Cowtan
- Eleanor Dodson
- Keith Wilson

http://www2.mrc-lmb.cam.ac.uk/personal/pemsley/coot/

or

Google: Coot

or for WinCoot

https://github.com/bernhardcl/coot

- Libraries, dictionaries
  - Alexei Vagin, Eugene Krissinel, Stuart McNicholas
  - Dunbrack, Richardsons
- Coot Builders and Testers
  - William Scott, Ezra Peisach
  - York YSBL, Dundee, Glasgow (early adopters)
  - Coot Mailing List subscribers
- Funding
  - BBSRC, CCP4, MRC, RÅC

Which map to use? Which contour level?

- Coot defaults:
  - Direct maps: 1.5 rmsd
  - Difference maps: 3 rmsd
- Adjust contour level with mouse wheel so that there is no noise!?
  - Use solvent region for decision making
  - Often 1-1.2 rmsd is more appropriate
  - May be locally different

How do we get the atoms into the density?

- Which atoms (groups) are placed?
  - Single atoms
  - Amino acids/nucleic acid
  - Secondary structure elements
  - Backbone (baton)
- How to place?
  - Manual move
  - Computational fit (refinement)
  - Both, interactive
  - All atoms are “blue”, no distinction between the “heavy” atoms (unless atomic resolution)

Side chain density what to do if there isn't any?

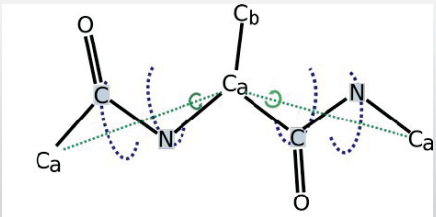
- Lower the contour level, maybe something shows up
- Possible "solutions"

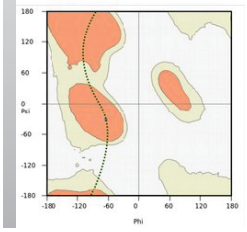
1) Mutate to Ala:
   - Doesnt reflect the truth
   - PDB will complain that there is a sequence mismatch
2) Stub the residue (i.e. remove all atoms beyond Cβ)
   - Again, not reality
   - PDB will complain about missing atoms
3) Set occupancies to 0 (or low value)
   - May be deceiving (0 occ not always clear)
   - Possible "distorted" side chains

Side chain density what to do if there isn't any?

4) Keep all atoms (and let refinement inflate B-factors)
   - Use a "fitting" (not clashing) high probable rotamer
   - Suggests a fixed position (unless you check B-factors)
5) Multiple conformations
   - Data may not justify this (resolution)
   - Confusing (?), which rotamers and how many

Crankshaft Peptide Optimisation





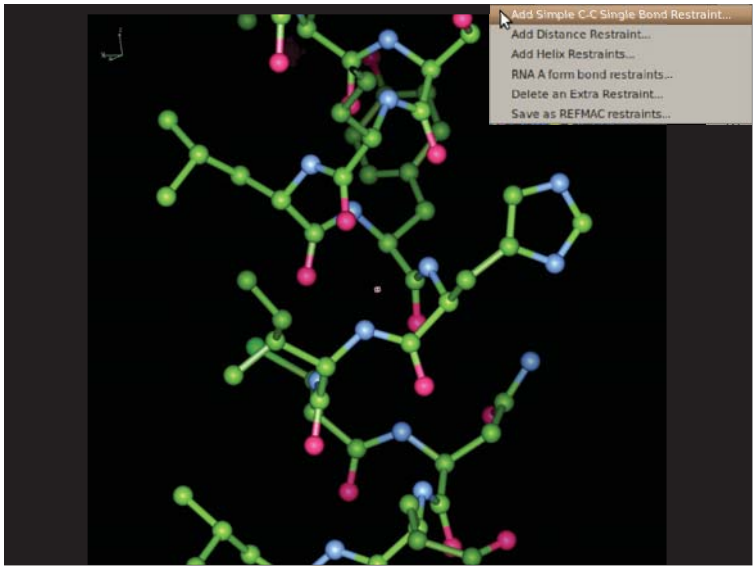
- Rotation around Cα-Cα vectors creates new positions for C and N atoms, leading to new {φi,ψi}i=1-3 angles and positions in the Ramachandran Plot
  - 1 neighbour each side → 3 residue
  - 2 neighbours each side → 5 residue
- Pertsemlidis et al. (2007) Statistical Applications in Genetics and Molecular Biology, 4(1), 35
- Useful discussions: Z. Otwinowski

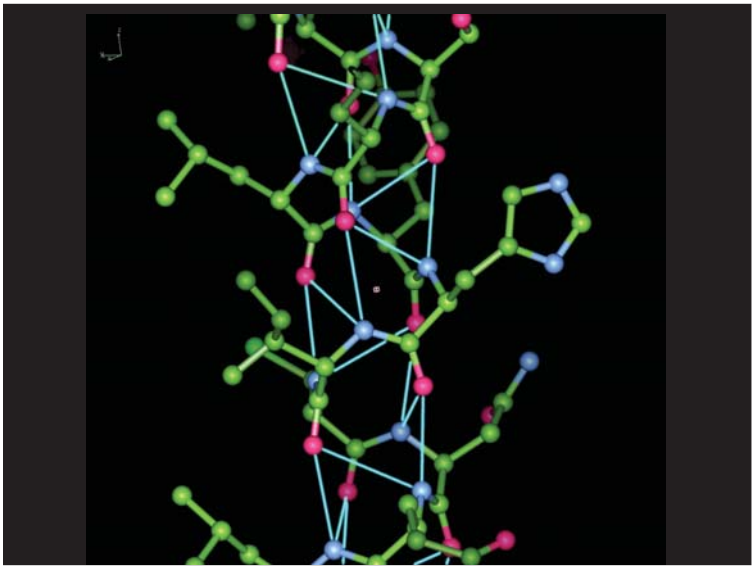
Crankshaft Peptide Optimisation

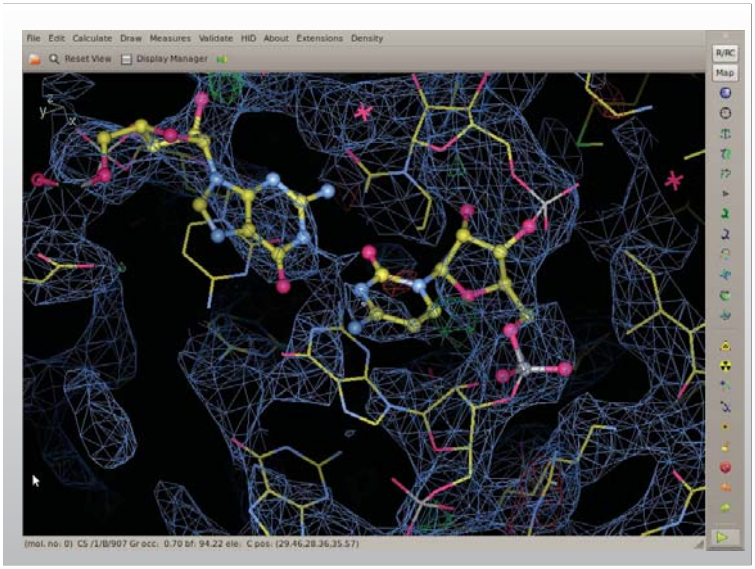
- By rotation of the peptide atoms around a Cα-Cα vector for a number of residue pairs, choose solutions for which ϕ,ψ most probable
  - cis→trans conversion (if needed) is the first step
  - a number of local-minima solutions are generated
  - each of which are (simultaneously) evaluated by real-space refinement
  - and assessed by posterior model distortion (model probability)
  - fit to map (likelihood) is used but has little discriminatory power for cryo-EM maps

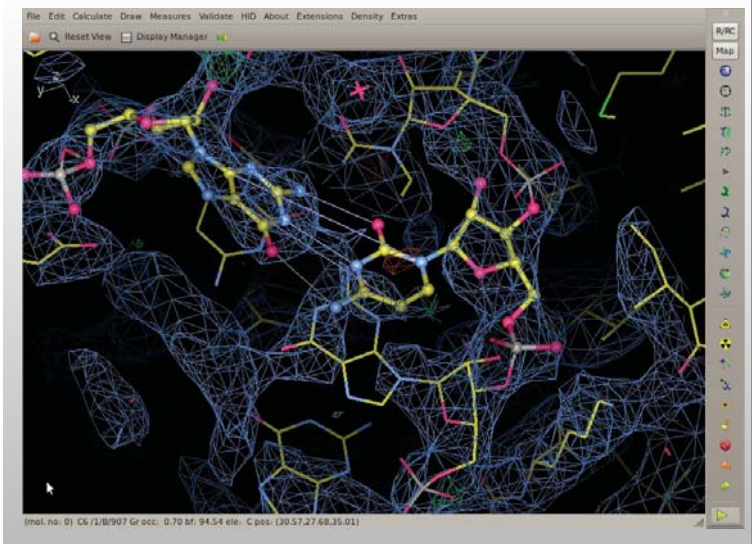
Restraints Editing in *Coot*

- Distance Restraints:
  - Alpha helices, A-form RNA, B-form DNA
- Add and delete individual restraints
  - User-selectable sigma
- Select 2 residues for range
- User-defined torsion restraints
- Input from ProSMART (ProSMART interface)
- Output to Refmac
- [planar restraints]



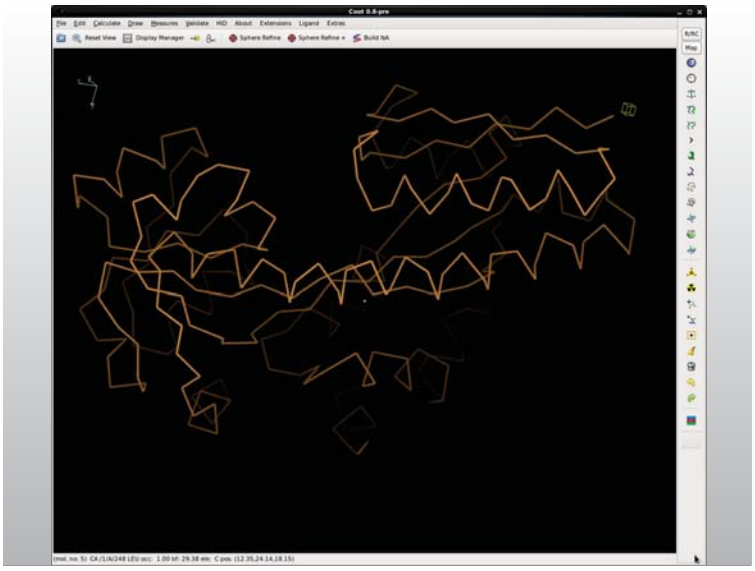


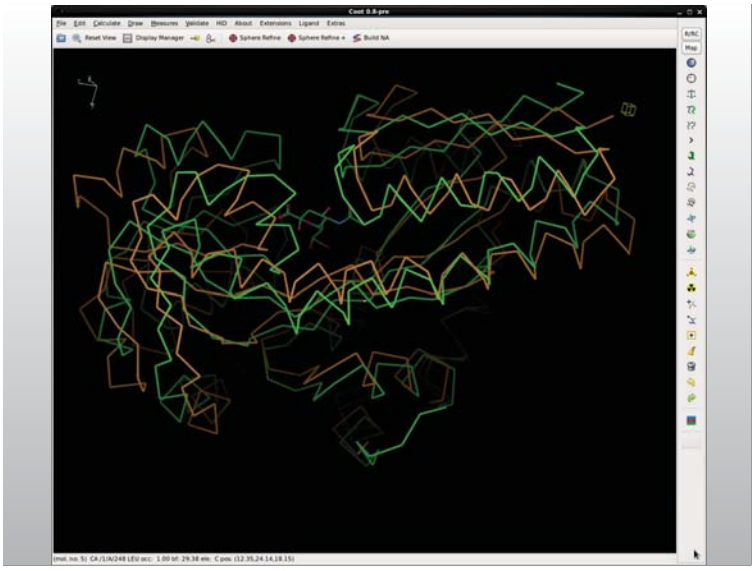


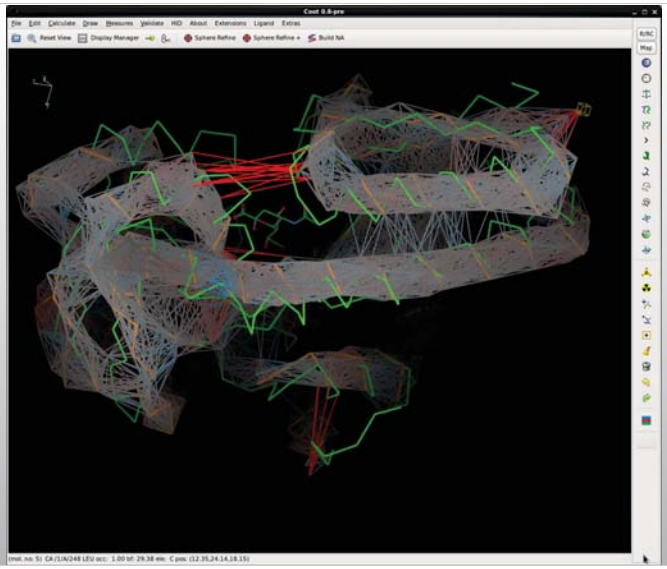


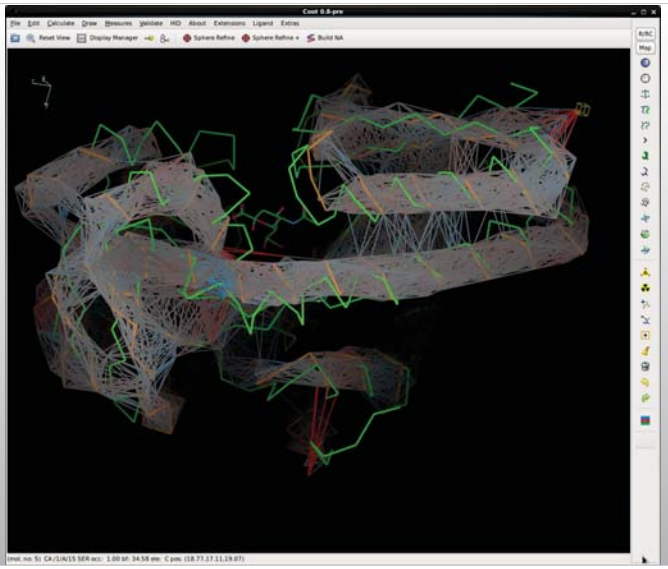
ProSMART Interface

- Use previous-solved “template” structures to inform the refinement of the (low resolution) target protein
- Conformation-independent structural comparison/superposition
- and restraint generation









Scripting

- Python or scheme
- 100s of functions are scriptable
- Accessed via:
  - the command line: **--script**
  - the GUI: Calculate -> Run Script...
  - Interactive: Calculate -> Scripting
- Use **--no-graphics** for "batch mode"

SSM Overlay by Scripting

- (superpose-with-atom-selection *imol1 mol2 atom-selection-string-1 atom-selection-string-2 move-copy-flag*)
- e.g. in scheme (superpose-with-atom-selection 0 1 "//A/20-120" "//B/30-130" 0)
- e.g. in Python superpose_with_atom_selection(0, 1, "//A/20-120", "//B/30-130", 0)
- General command:
  - Scheme: `(scheme-command arg1 arg2 ...)`
  - Python: `python_command(arg1, arg2, ..)`

More on Scripting

- If something is boring, stop it
  - Write a script
  - Or get someone to do it for you
    - me?
- Scripting available in Python or Scheme (lisp)
- Scripting example available on the mailing list
  - and the Coot Wiki

Some key bindings

- Any function can be bound to a key
  - Allows for personalization/customization
- Here's how you do it:

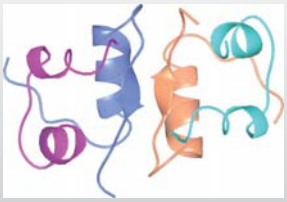
```
(add-key-binding "x" (lambda () (refine-active-residue)))
```

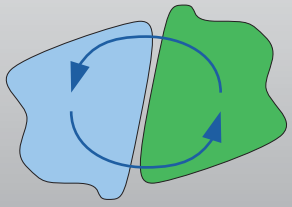
- Makes Coot easy to use
  - (but harder to learn)
- http://www2.mrc-lmb.cam.ac.uk/personal/pemsley/coot/web/docs/coot-keys-and-buttons.pdf

Handling NCS...

What is Non-Crystallographic Symmetry (NCS)?

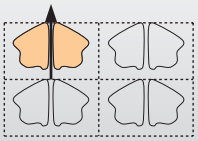
- 2 or more copies of a molecule in the unit cell not related by crystallographic symmetry
- NCS related molecules provide different representations of the same molecule





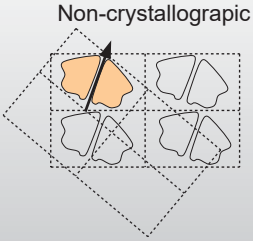
Non-crystallographic symmetry

Crystallographic



Aligned 2-fold

Non-crystallograpic



Unaligned 2-fold

- What are the problems?
  - Molecules are different
  - How to allow for differences, but minimize unnecessary rebuilding?

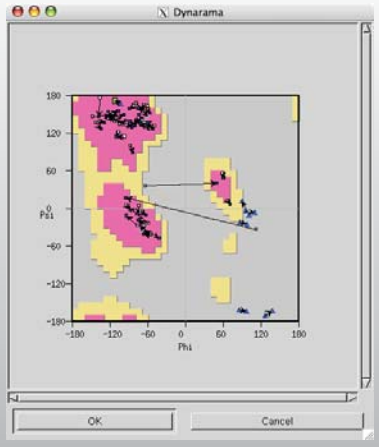
Handling NCS

Typical Scenario:

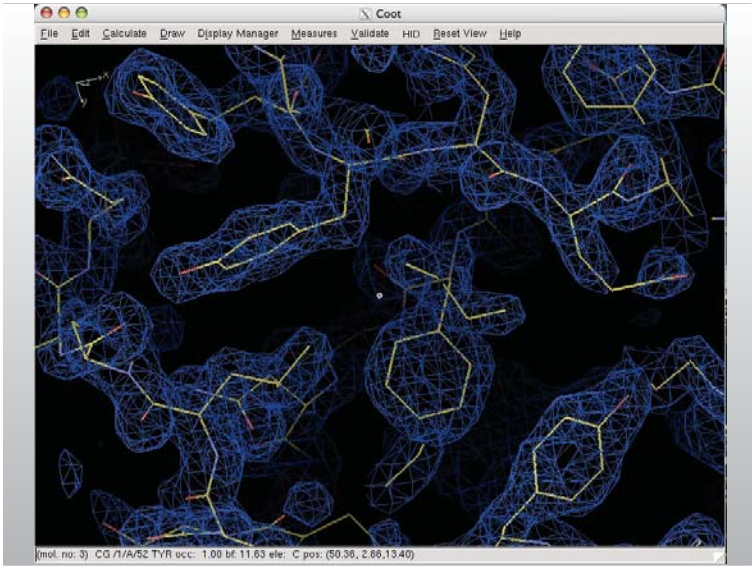
- I have done an LSQ overlap of my NCS-related molecules and from the graph, have seen significant deviations in the positions of some side-chains.
- Why are they different?

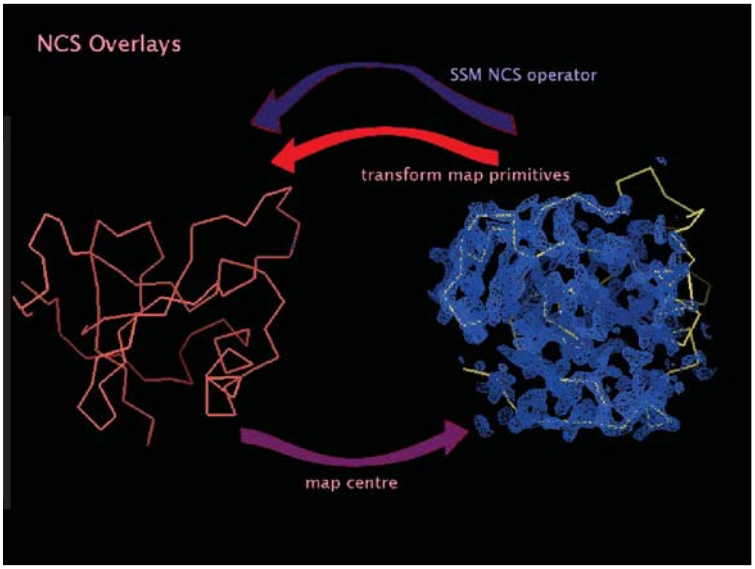
...or NCS Differences graph

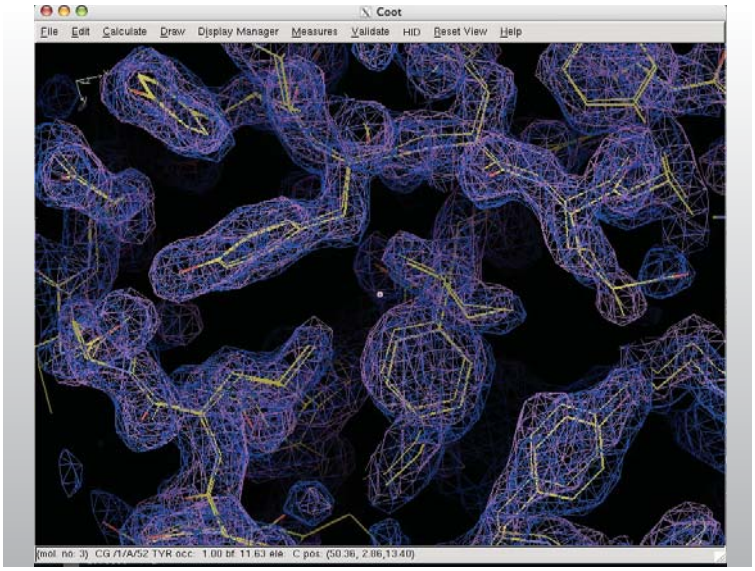
...or Kleywegt Plots[*]



[*] Named by George Sheldrick







NCS Model-modification Tools

- Automatic detection of NCS
  - And their operators
- Copy Master NCS molecule to others
  - Applies NCS transformation
- Copy NCS Master residue-range
- Change NCS Master chain
- NCS Skipping ('o' key)