Coot & ChimeraX Practical

Description of initial map and model

For this tutorial, Oli has generated a 3.7 Å cryo-EM reconstruction of human met-hemoglobin in cryoSPARC, starting from 50 randomly chosen micrographs from the EMPIAR-10250 dataset.

The original pixel size was 0.559 Å. 5857 particles (2x binned to 1.118 Å per pixel) were used in the final reconstruction, with C2 symmetry imposed.

The map is deliberately rather poor, to represent what you might encounter in the initial phases of a structure determination project. It is still very buildable, however.

The initial model was generated from PDB 6NBD, with the following modifications:

- The molecule has been deliberately misoriented in the map.
- The alpha chains (A & C) have had the C-terminal helix removed.
- The beta chains (B & D) have been replaced by alpha chains.
- The heme molecules bound to the B & D chains have been removed.

For reference, this is the sequence alignment of human HbA and HbB:

```
HBB_HUMAN/1-147    1 MVHLTPEEKSAVTALWGKV--NVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVM  56
HBA_HUMAN/1-141    1 --VLSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHF-DLS-----H  50

HBB_HUMAN/1-147   57 GNPKVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRLLGNVLVCV 114
HBA_HUMAN/1-141   51 GSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVT 108

HBB_HUMAN/1-147  115 LAHHFGKEFTPPVQAAYQKVVAGVANALAHKYH                          147
HBA_HUMAN/1-141  109 LAAHLPAEFTPAVHASLDKFLASVSTVLTSKYR                          141
```

✅ The aim of this tutorial is to show you how to first fit, and then fix this model.

Required Software

Coot 0.9-pre or later (earlier versions will work, but lack some bells and whistles useful for cryoEM).

Install with **CCP-EM** nightly package from here (not yet available for Windows):

CCP-EM software suite v1: https://www.ccpem.ac.uk/download.php

ChimeraX: https://www.cgl.ucsf.edu/chimerax/download.html

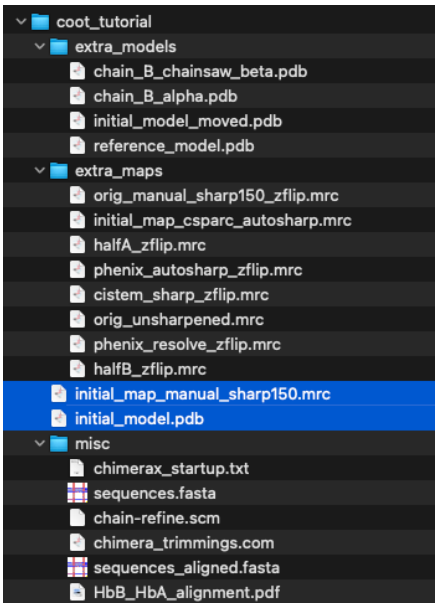
1. Getting Started

Paste the following link into your web browser:

https://www.dropbox.com/scl/fi/2r8v71k2xtzz8y63tvie1/coot_tutorial.zip?rlkey=lkrfziuure8qgl88w6uqdm6vl&dl=0

An archive of the files you'll need for this tutorial should download automatically. Unpack it and move it to your preferred working directory.

The contents are as follows (the initial model and map are highlighted):



Install coot-trimmings

These are Oli's custom scripts that add useful functions for building into cryoEM maps. To install coot-trimmings, copy the `coot_trimmings.py` script from the `misc` subdirectory to the hidden coot-preferences directory in your home directory:

```
cp misc/coot_trimmings.py ~/.coot-preferences/
```

When you restart Coot, you should see a new "*Custom*" menu, and have a set of extra keybindings and custom functions.

Add the Refine menu

This adds useful options for flexibly fitting entire chains or models.

Copy the `chain-refine.scm` script from the `misc` directory to the hidden coot-preferences in your home directory: `cp misc/chain-refine.scm ~/.coot-preferences/`

Add ChimeraX startup commands

Open *Preferences* > *Startup*

Paste the following commands (`misc/chimerax_startup.txt`) in “Execute these commands at startup” field:

```
camera ortho
alias cofron cofr centerofview showpivot 7,0.25
alias cofroff cofr centerofview showpivot false
alias symclip cofr centerofview; clip near -$1 far $1 position cofr

alias cootmode set bgColor black; surface cap false; surface style solid;
    lighting flat; graphics silhouettes false; style stick; ~rib;
    color ##num_residues gold; color byhet ; disp; ~disp @H*;
    style ions ball; style solvent ball; size ballscale 0.2;
    size stickradius 0.07; transparency 70; cofr centerofview;
    clip near -10 far 10 position cofr; color ~##num_residues cornflower blue

alias cootmode_mesh surface cap false; surface style mesh;
    lighting flat; graphics silhouettes false; style stick;
        ~rib; color ##num_residues gold; color byhet ; disp;
        ~disp @H*; style solvent ball; style ions ball;
        size ballscale 0.2;  size stickradius 0.07; cofr centerofview;
        clip near -10 far 10 position cofr; color ~##num_residues #3d60ffff;
        transparency 50
```

Adding `cofron` , `cofroff` , `symclip` , `cootmode` , and `cootmode_mesh` . More aliases can be found at https://github.com/olibclarke/chimerax-trimmings/

2. Flipping the map and rigid body fitting of the model

Flipping an inverted map

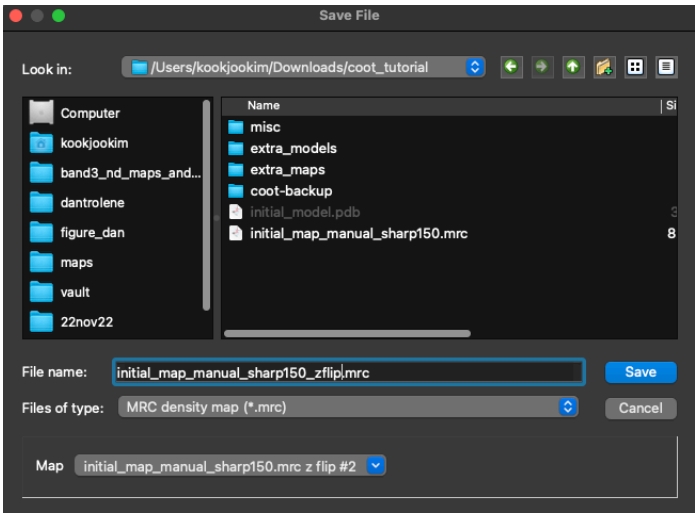
Open `initial_map_manual_sharp150.mrc` in ChimeraX. Set the map to an appropriate threshold using the Volume Viewer (~0.12 is good in this case).

🔎 Take a close look at the helices - you should notice that they are left handed. In this case, *ab initio* model generation converged on the model with inverted hand, so we need to flip the map.

Open the command line if it is not already present at *Tools > General > Command Line Interface*, then use:

```
vol flip #1
```

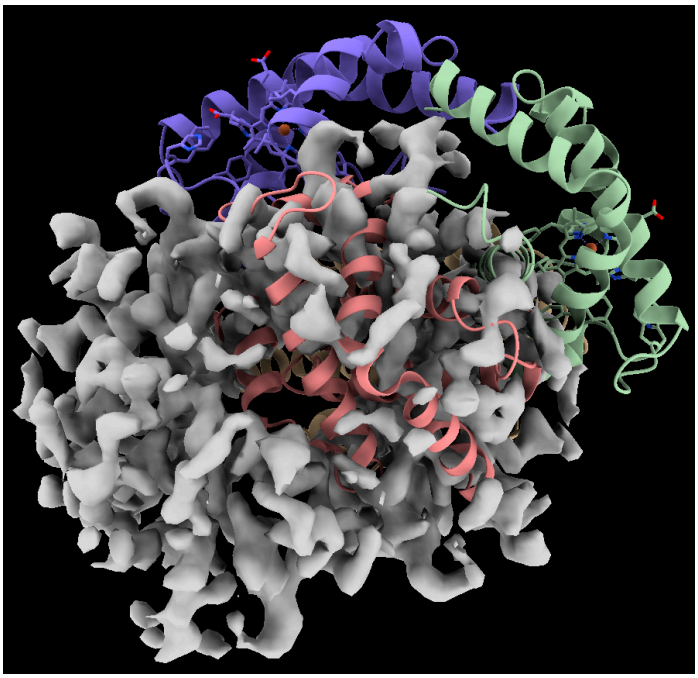
You should now have a z-flipped map of the correct hand, with ID #2. Close the original, inverted map (`close #1`), and save the new map (*File > Save*).



Select MRC density map (*.mrc) and the z-flipped map (#2).

Fitting a model

Open the initial model `initial_model.pdb`. You will notice that it does not fit the map very well:



Here, I have colored the molecule by chain and heteroatom:

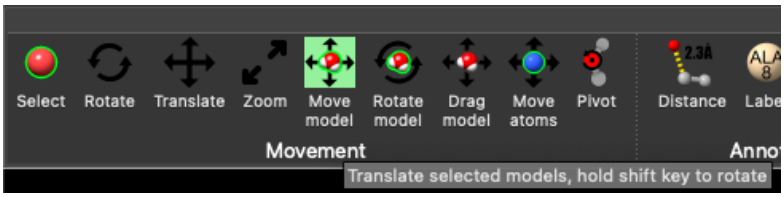
```
color #1 byc; color byh
```

You can lock the center of rotation to the center of the view using the alias `cofron`, which makes precise navigation in the map more straightforward, especially combined with `symclip 5`, which sets the clip planes symmetrically in a **5 Å** slab about the center of rotation. To turn off clipping, use `~clip` .

The molecule is displaced from the density and in the wrong orientation. To fix the first issue, **select the model**, by opening the “Model Panel” and selecting the selection checkbox.

Then, translate the model to the center of the map using the right mouse button.

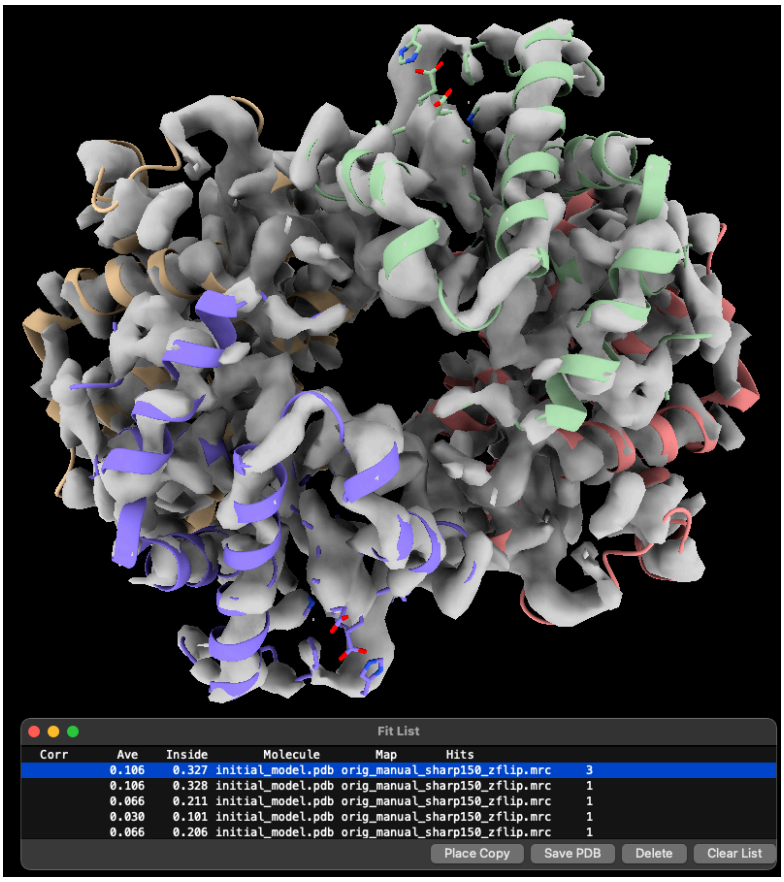
First you need to change the right mouse mode to “Move model”. You can now move the selected model by right-click dragging. Check that the model is well centered by rotating the view (do not worry about fit at this stage), and adjust as needed.



Now, to fit the molecule to the map, we will use `fitmap` to perform a global search:

```
fitmap #1 inmap #2 search 100 radius 5
```

This will test 100 random orientations, with a random displacement of up to 5 Å from the current position, and generate a table of scored candidate poses:



In this case, the top two correct solution are separated in score from the incorrect solutions. If this were not the case, I would suggest first fitting the model to a low pass filtered, or Gaussian blurred, version of the map (e.g. generated from the initial map using `vol gaussian #2 sdev 2.5`), as fitting to a smoother map will generally have a larger radius of convergence and is less prone to get stuck in false minima.

Save the resulting PDB file. A pre-generated z-flipped map is included in `extra_maps`, and a correctly oriented model is included in `extra_models`.

Fetch AlphaFold predictions

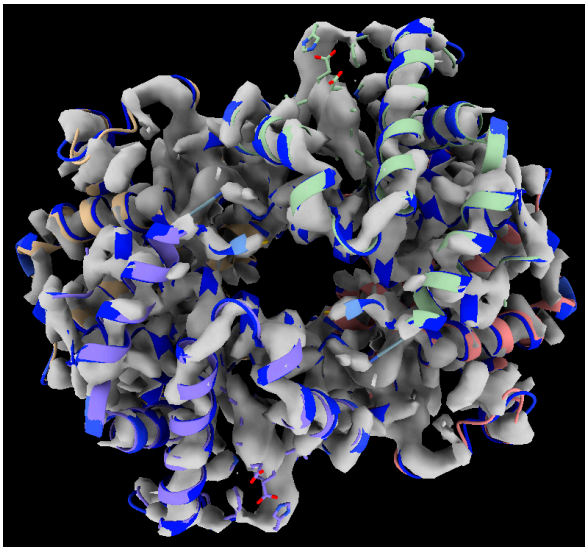
You can find and retrieve existing models from the AlphaFold database of models for UniProt protein sequences:

`alphafold fetch P69905` (alpha subunit) and `alphafold fetch P68871` (beta subunit)

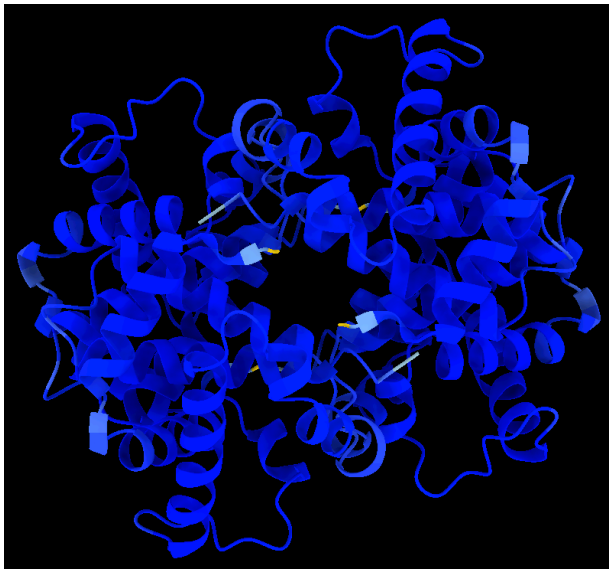
In our case, we already have an initial model fit in the map. ChimeraX can search the database for the protein sequence and align the structures:

`alphafold match #1 trim false`

ChimeraX automatically colors the predicted structures by the pLDDT scores in the B-factor field.
To manually color the predicted structures, use
`color bfactor #5 palette alphafold`



4 chains (2x HbA & 2x HbB) were matched



High (>90) pLDDT scores all around Hb

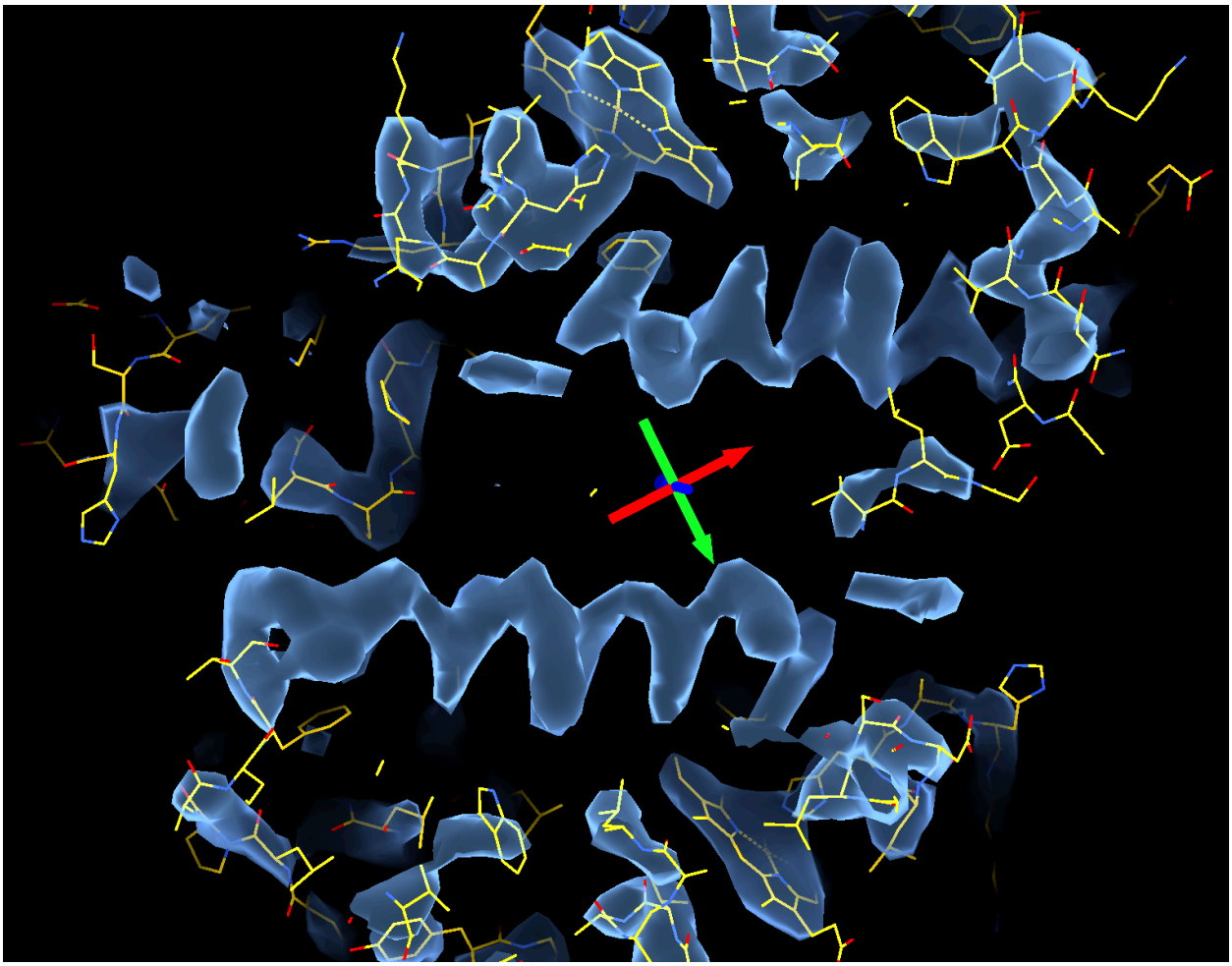
Inspect the overall fit of the model to the map

Close the AlphaFold predictions and let's inspect the overall map model fit.

- Do you notice any unmodeled density?
- Do some of the chains fit better than others?
- Look at the C-termini of chains A & C. Can you see where the missing C-terminal alpha helix should fit?

Set the model to wireframe (alias `cootmode_mesh`), and have a closer look at the fit of the model to the map.

Identify any obvious problem areas - misfit loops, bulky sidechains with no density, small sidechains with too much density - and note them down for later.



`cootmode_mesh` view of the map and the model. Unmodeled helical density, misfit residues, and the heme molecules in chains A and C are displayed.

3. Coot basics - loading, navigating, and adjusting display of models and maps.

Open Coot and load the map and model we have just saved

The display status and appearance can be adjusted in *Display Manager*, or try the following shortcuts:

`]` & `[` : cycle representation mode of the active model forward/back

`/` : Toggle display of all models on/off

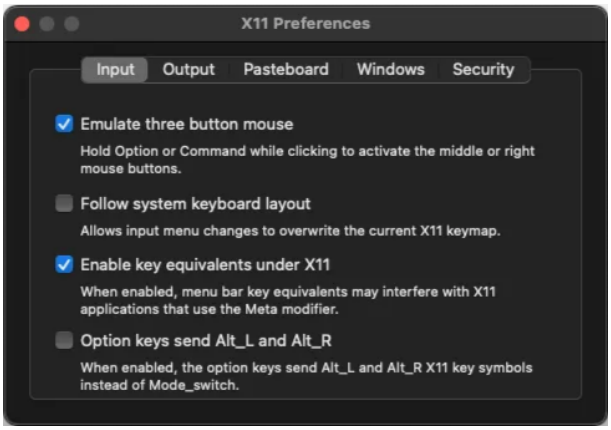
`?` : Cycle display of each model (if multiple are loaded)

`` ` `` : Toggle all maps on/off

`~` : Cycle display of each map (if multiple are loaded)

`Q`: to quick-save active PDB (it will automatically make a backup of the existing file)

If you are using a Mac with a trackpad, make sure to set “Emulate three button mouse” in XQuartz input preferences:



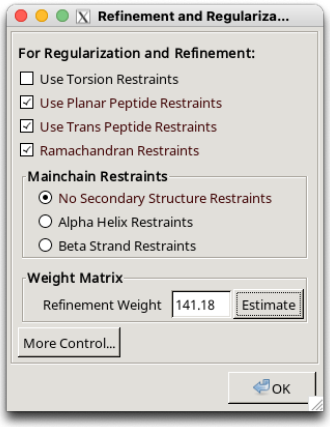
Navigation:

- Ctrl-click-drag to translate view
- Click-drag to rotate view
- Right-click drag to zoom in/out
- Middle-click on atom to center
- Ctrl-G to quick-go to atom (box appears; type chain ID and residue number and hit enter)
- Shift-click to label atom/residue

View:

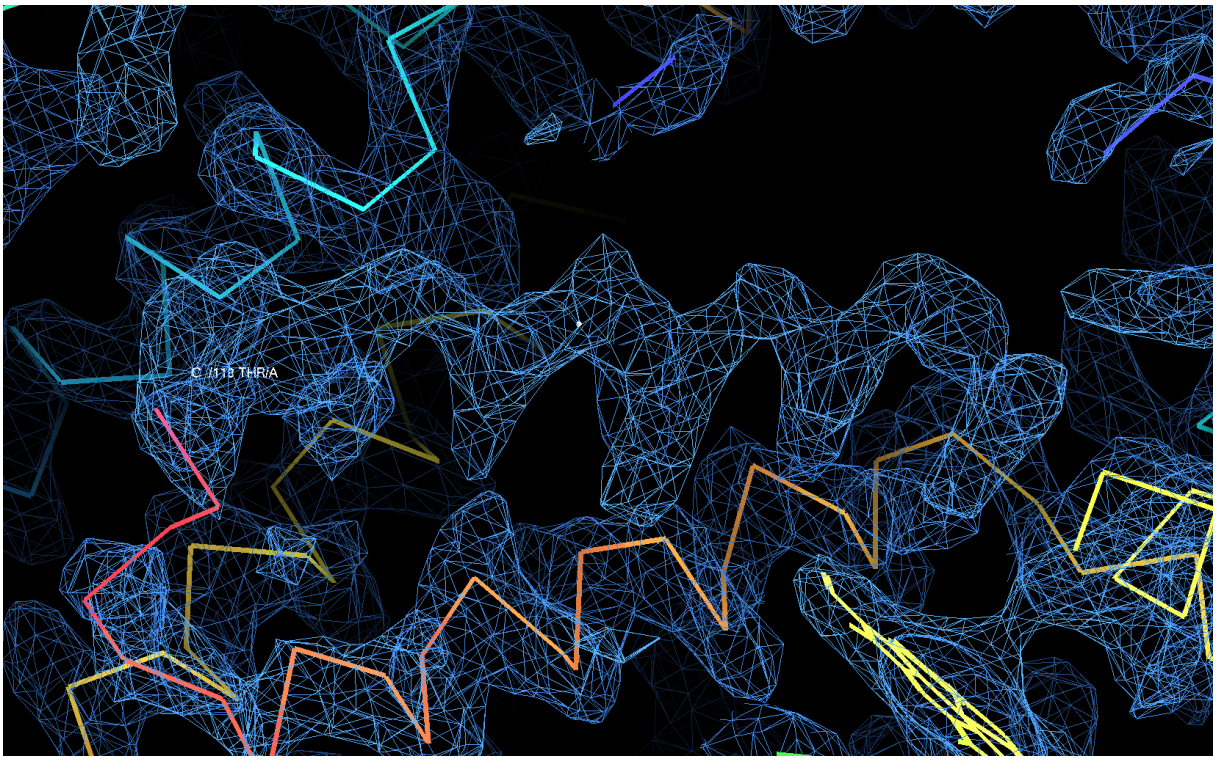
- Ctrl-right-click-drag up/down to translate slab in/out of plane
- Ctrl-right-click-drag left/right to change thickness of slab.

Estimate and set the refinement weight in Refinement/Regularization Control:



4. Placing, sequencing and refining the C-terminal helix of HbA, and merging it into the original model

Navigate to the (current) C-terminus of one of the alpha chains (e.g. Ctrl-G A118). You should notice that a tube of density protrudes from the present terminus:



This is the missing C-terminal helix of the alpha subunit - let's build it in.

Placing helices

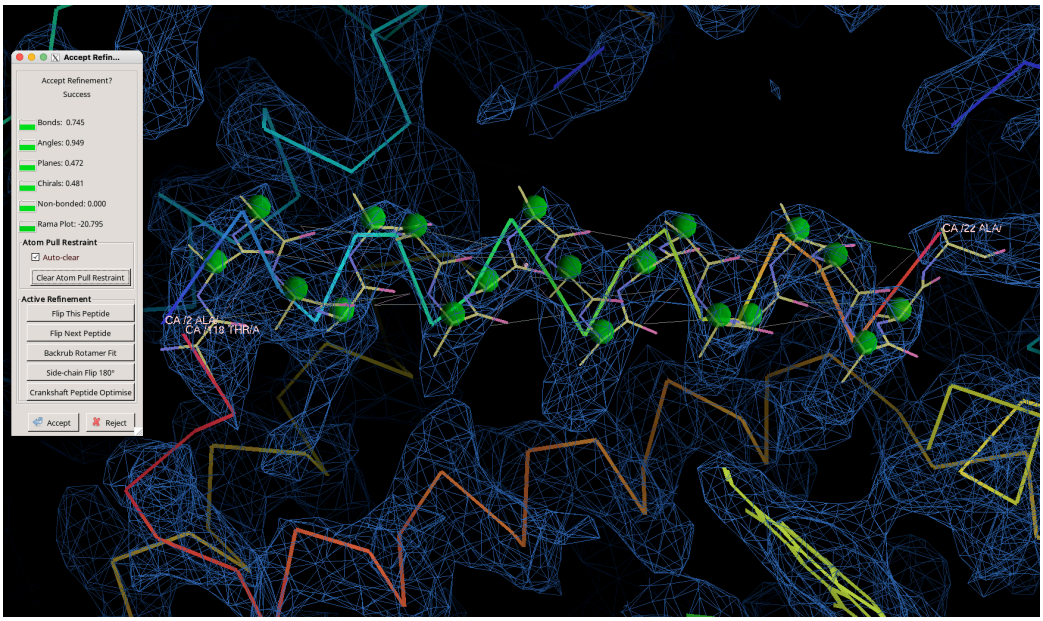
Navigate to the center of the missing helix, about midway along. Now, place a helix, either with the keyboard shortcut `h`, or the menus (*Calculate > Other Modelling Tools... > Place*

Helix Here).

Hopefully, a single helix will be placed, in the correct direction. If Coot is unsure of the fit, a candidate helix will be generated in both directions (in which case inspect the sidechains

> 💡 Helices are like Christmas trees🎄 with the sidechain branches pointing down towards the N-terminus at the base of the tree.

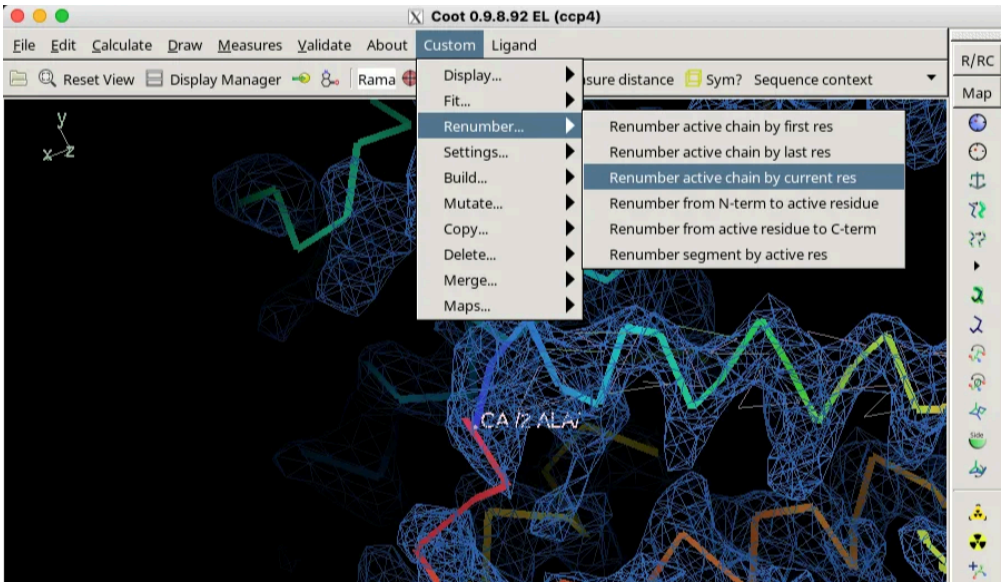
Trim the helix to fit the density (*Delete* > *Delete Zone*), and refine it into the density using "Real Space Refine Zone":



Notice the spheres? These are on the fly validation indicators indicating favored (green), allowed (orange) and disallowed (red) Ramachandran values for the marked residue.

Merge the models and mutate according to the raw sequence

Renumber the N-terminal residue of the helix such that it starts one residue after the end of the initial model:

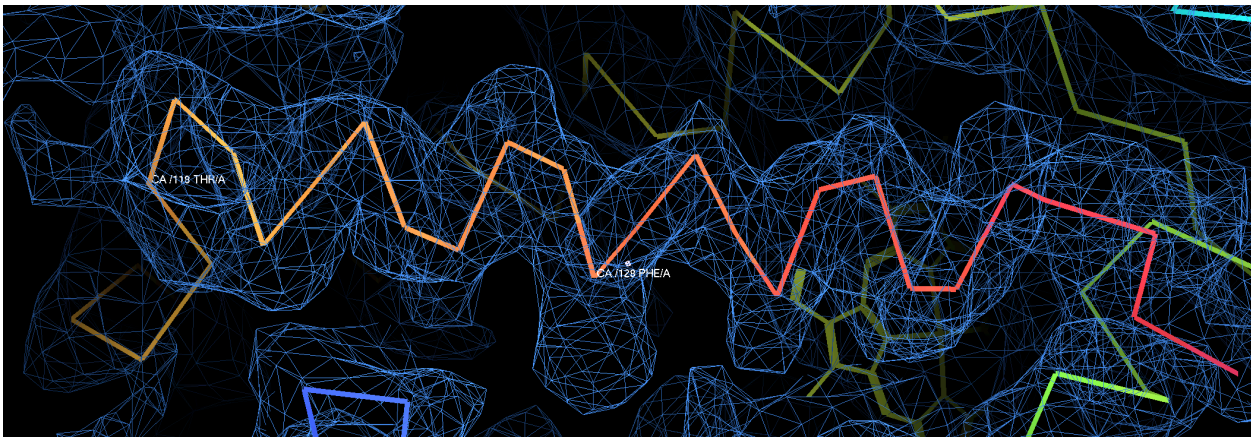


Merge the helix into the initial model (*Custom > Merge > Merge two mols*), and then merge it into chain A (*Custom > Merge > Merge chains*). Refine a zone including the helix, and then join with the initial model.

Now, let's assign sequence. Mutate the active chain to the target sequence ("Custom > Mutate > Mutate active chain to template sequence"), pasting in the raw sequence of HbA from `misc/sequences.fasta`). Inspect the sidechain assignments, and fill them in (`k`, fill partial residues).

If they do not look correct, or you are unsure, try adjusting the sequence register using the renumbering tools, and reassigning the sequence as above.

> 👉 Can you build any extra residues at the C-terminus of your new helix? Try using "Add terminal residue" (`y` shortcut) to extend it.



5. Getting a threaded model for the beta subunit and building it in

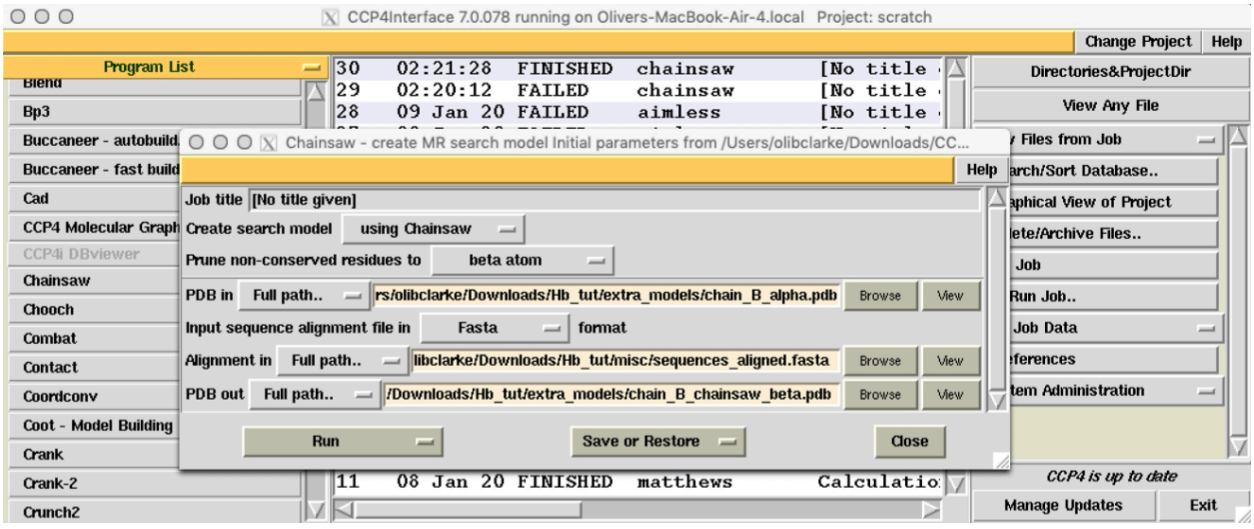
Go to an atom in chain B (e.g. “Ctrl-G B63”). This is one of the two **beta chains** that has been replaced with **alpha chain** coordinates. We would like to generate a simple, threaded homology model which we can then refine into the density and fix errors.

Copy the chain (*Custom > Copy > Copy current chain*).

Save the resulting molecule as a new PDB. A pre-generated one can be found in `extra_models/chain_B_alpha.pdb` .

Running CCP4 Chainsaw

If you have CCP4 available, open the GUI and use *chainsaw* to generate a simple homology model, using the isolated chain F saved in the previous step, as well as a FASTA format sequence alignment which can be found in the `misc` directory, as input:

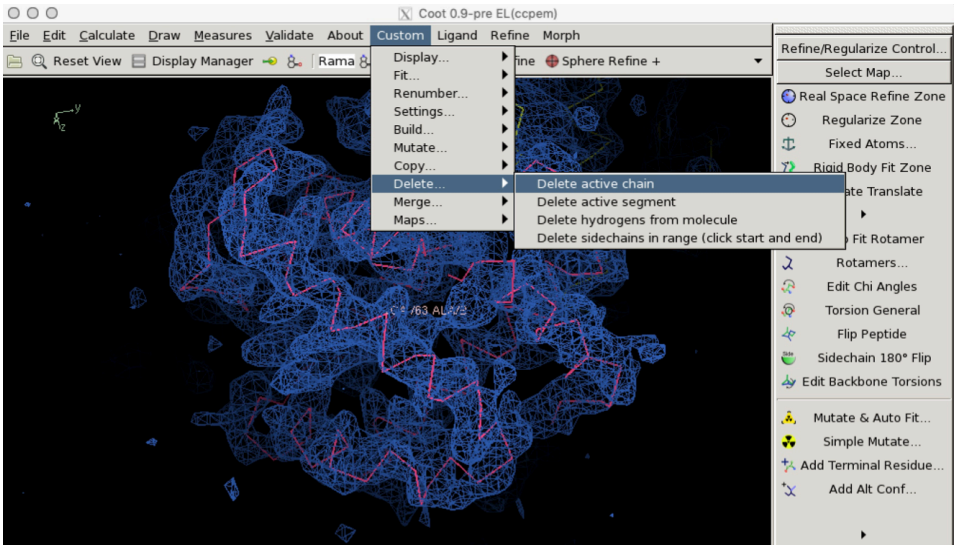


This will mutate your “alpha” model to the “beta” sequence, adjust sequence numbering, and truncate non conserved residues. It will not change the position of atoms, however - this is a simple “threading” approach. (If you don’t have CCP4 installed, don’t worry, a pre-prepared version can be found in `extra_models/chain_B_chainsaw_beta.pdb`).

Combining the threaded model with the starting model

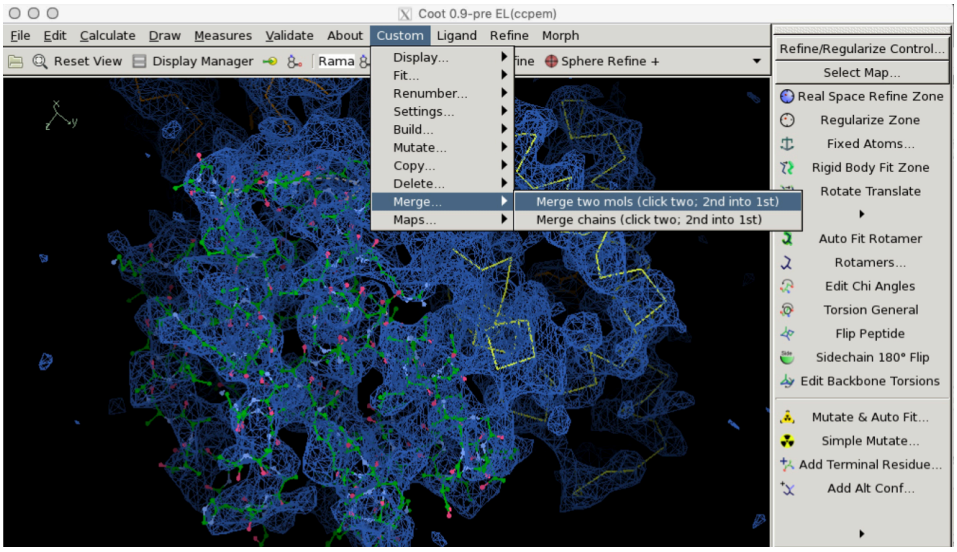
Open the threaded model you just generated in Coot (along with the map and model you already have open).

Delete the original chain B - center on it, and delete it using the *Custom > Delete > Delete active chain* menu item:



Merge the threaded model into the main molecule.

Display both molecules, and then use the *Custom > Merge > Merge two mols* menu item:



Click the original molecule, followed by the threaded model that we wish to merge in. You should now have a single molecule containing both components.

Flexibile fitting of the threaded beta chain

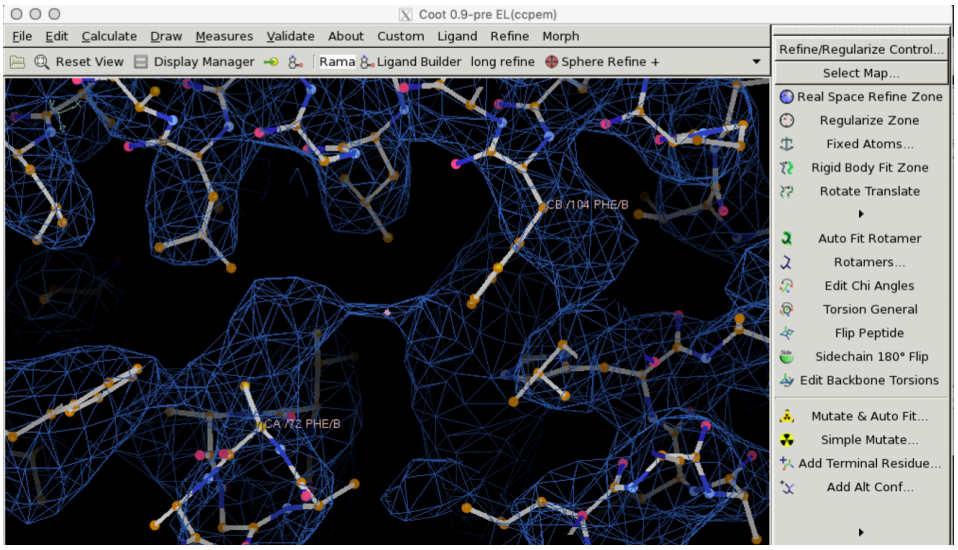
Use *Refine > Chain Refine* to refine the entire chain. It may take some time to settle.

You can interact with and guide the refinement by click-dragging on individual residues, or Ctrl-click-dragging on individual atoms. Once you are satisfied with the fit, hit return to accept the result.

Filling truncated sidechains and adjusting misfits

After globally fitting the chain, it is now time to go residue-by residue and inspect the fit to the map, filling partial residues (`k` , or `M` for mutate), adjusting rotamers (`R` to cycle through rotamers for the active residue), and locally refining regions using “Real space refine zone” and “Sphere Refine”.

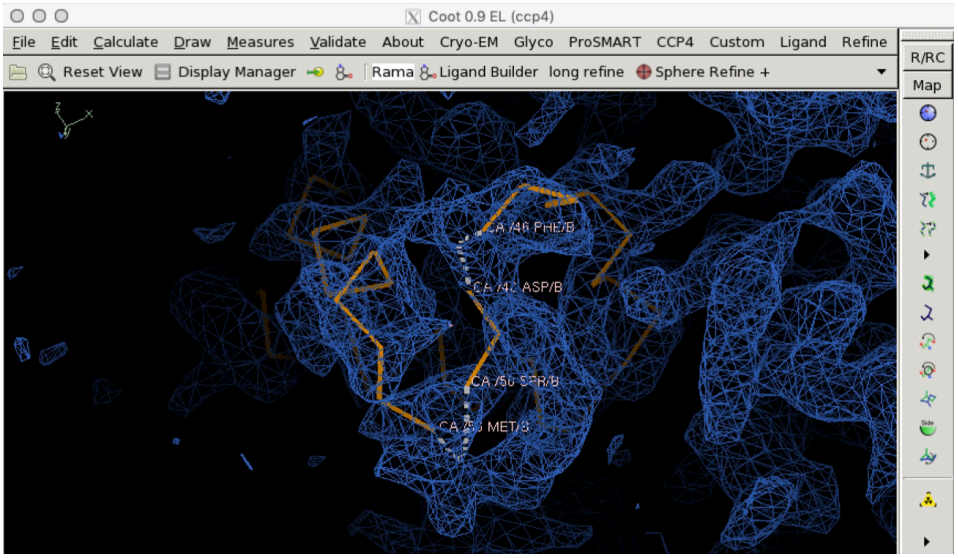
One such example site (where Phe72 needs a sidechain, and the sidechain of Phe104 needs adjustment) is illustrated below.



👉 Find and fix as many as you can! Consult the “Rotamer Analysis” and “Ramachandran Plot” sections of the Validation menu to identify problem areas.

Fitting a missing loop

There is a missing loop between residues 46 and 56 in the beta subunit (after Chainsaw):

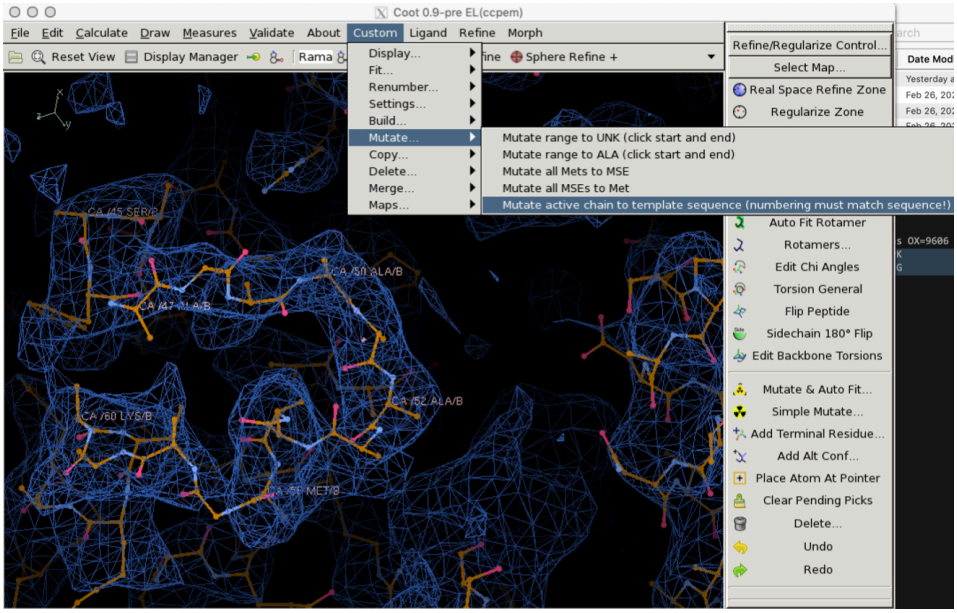


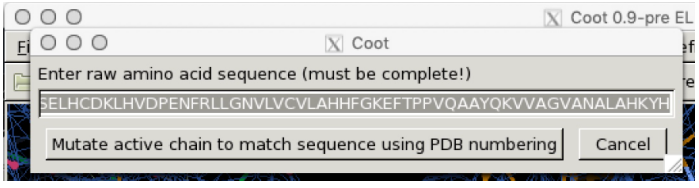
First, remove the disconnected peptide floating between the loop termini (*Delete Item > Delete Zone*). Refine both loop termini into the density using “Real Space Refine Zone”. Then, fill the loop with polyalanine using add terminal residue (key binding `y`), real space refining every several residues.

As you add residues, ensure at each step that the C-beta and carbonyl (or amide nitrogen) are oriented appropriately for addition of the next residue into the density. If not, center on the terminal residue and use the key binding `r` to refine that residue, and drag the C-beta to a more appropriate orientation.

Near loop completion, you might find that it becomes difficult to add residues in sensible places. Don't worry about this too much - add them anyway, until you complete the loop. Then, run a "Real Space Refine Zone" to drag-refine the loop into the density. The constraint of loop connectivity will help fix up any misbehaving residues near the join point.

The loop we have filled is currently polyalanine, but the real sequence is not. Let's fix that by mutating the chain to match the Hb-beta sequence:





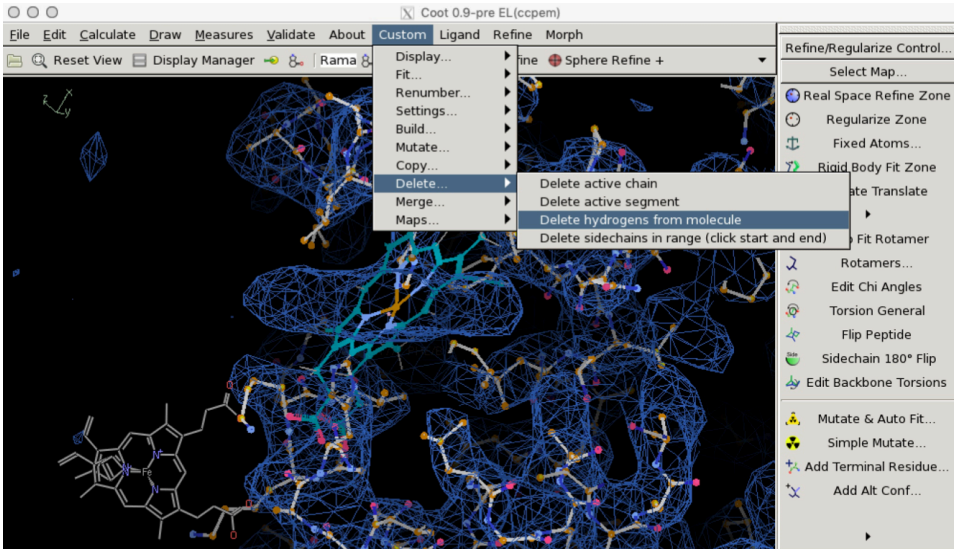
Copy the raw amino acid sequence from the HBB entry in `misc/sequences.fasta` and paste it into the dialog box. This function should change the sequence to match. It will leave all conserved sidechains alone, but will not add sidechains - you can do that yourself using the key binding `k` (fill partial residue).

Placing the missing heme

Navigate to and center on the density for one of the missing hemes (e.g. the density near F104 in the beta chain).

Retrieve the ligand from the CCP4 monomer library using *File > Get Monomer*, and enter the “HEM” code for heme.

A heme will appear. Delete the hydrogens on the heme:

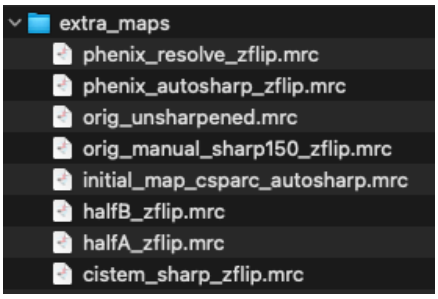


Approximately fit the heme to the density using Jiggle Fit (shortcut `J` while centered on one of the atoms of the ligand).

- Rotate the heme, if needed, using “Rotate Translate Zone”, such that the two carboxylate moieties face out of the cleft; also check the consistency of the two other substituents on the porphyrin ring with the density.
- Use “Rigid Body Fit Zone” to improve the fit
- Merge the ligand into the molecule using *Custom > Merge > Merge two molecules*
- Perform a sphere refine while centered on the heme and adjust surrounding sidechains to taste.
  (If you don’t have a sphere refine button in your Coot, you can add one by right-clicking on the upper toolbar and clicking “Manage buttons”).

6. Comparing multiple maps & on-the-fly sharpening/blurring/resampling

Several additional maps are provided in `extra_maps` directory:



These include maps sharpened using CisTEM, `phenix.auto_sharpen`, `phenix.resolve_cryo_em`, CryoSPARC, and manual application of a negative B-factor (which is what we have been using so far). Compare them in Coot and/or Chimera and see which you like best.

Inspect the auto-sharpened map out of cryoSPARC - this is a good lesson in not always trusting automatic routines.

> ⚠️ Despite being nominally 3.7 Å, the auto-sharpened map has almost no sidechain densities - it is only after manual tweaking of the sharpening B-factor that we get an interpretable map.

Also, try the on-the-fly tools for map sharpening, blurring and resampling in Coot:

- Load the CryoEM module (*Calculate > Modules > Cryo-EM*).
- In the new “Cryo-EM” menu, try “Sharpen/Blur”.

Apart from the sharpening tools, the resampling tool is very handy when one is getting close to Nyquist at high resolution - resampling the map on a finer grid can really help bring out details (waters, carbonyls, etc).