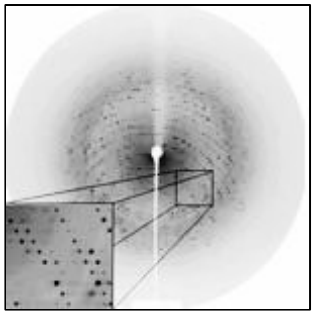
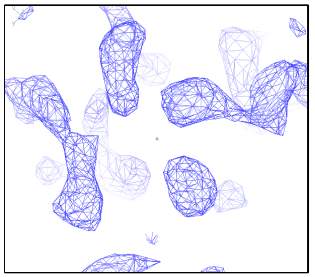
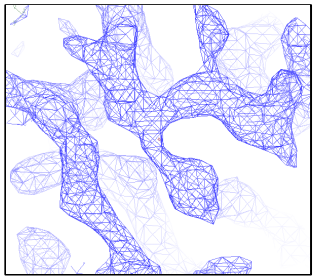
Automated phase improvement and model building

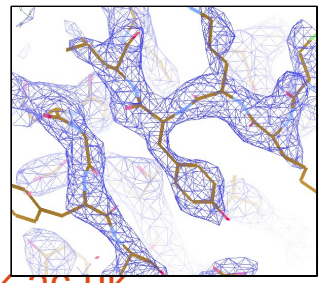
Kevin Cowtan
cowtan@ysbl.york.ac.uk

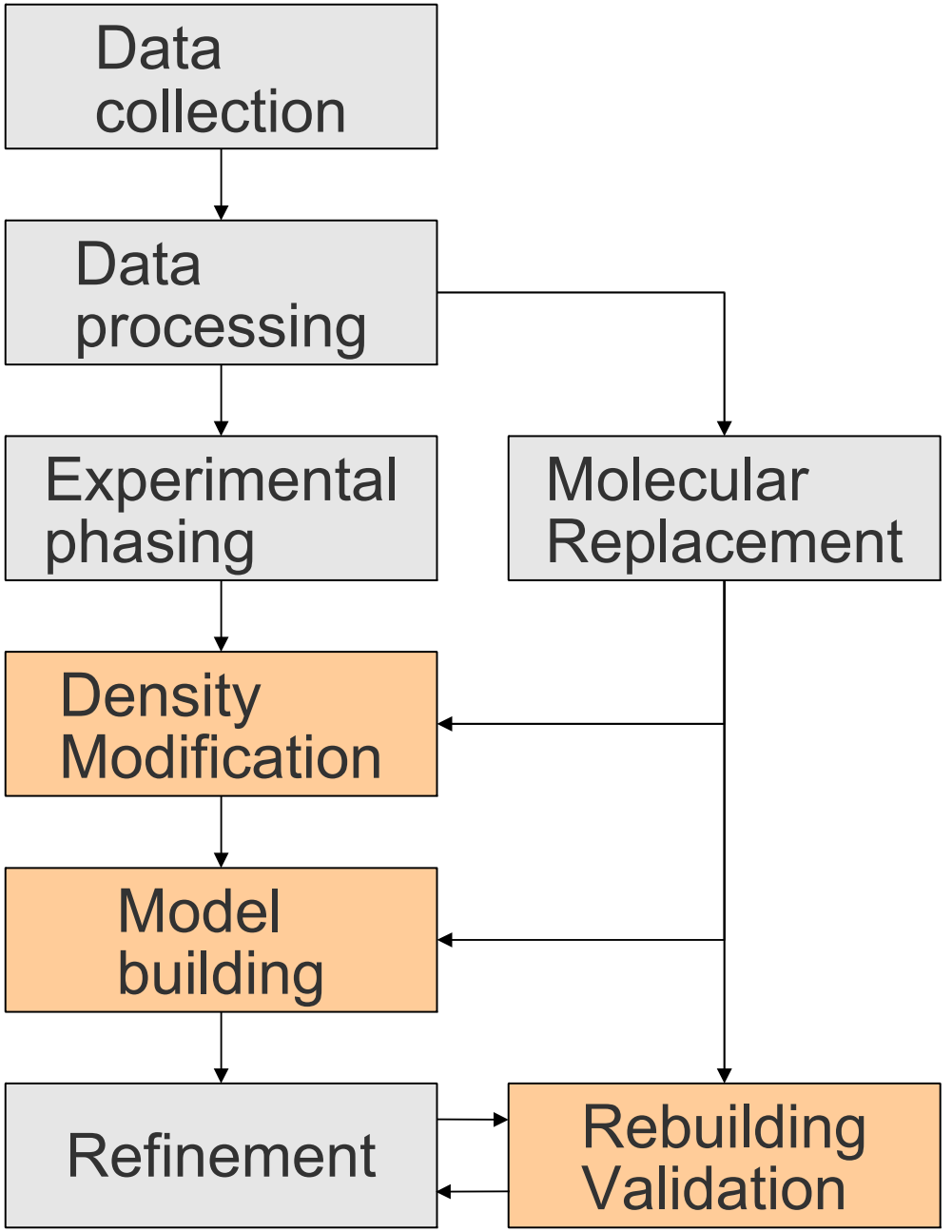
X-ray structure solution pipeline...







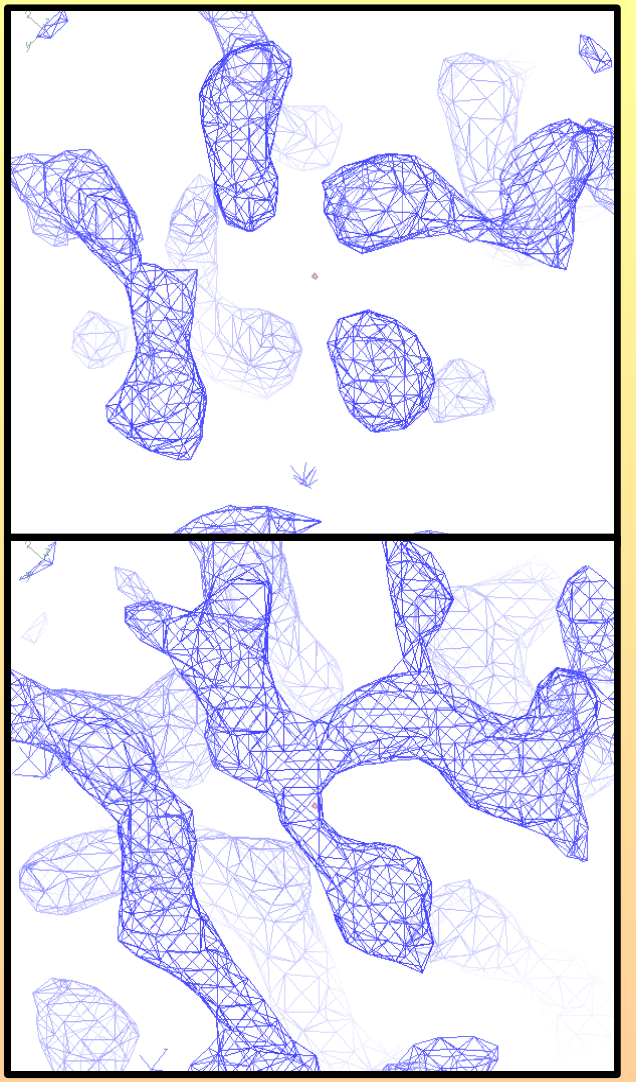




Kevin Cowtan, cowtan@ysbl.york.ac.uk

Density Modification

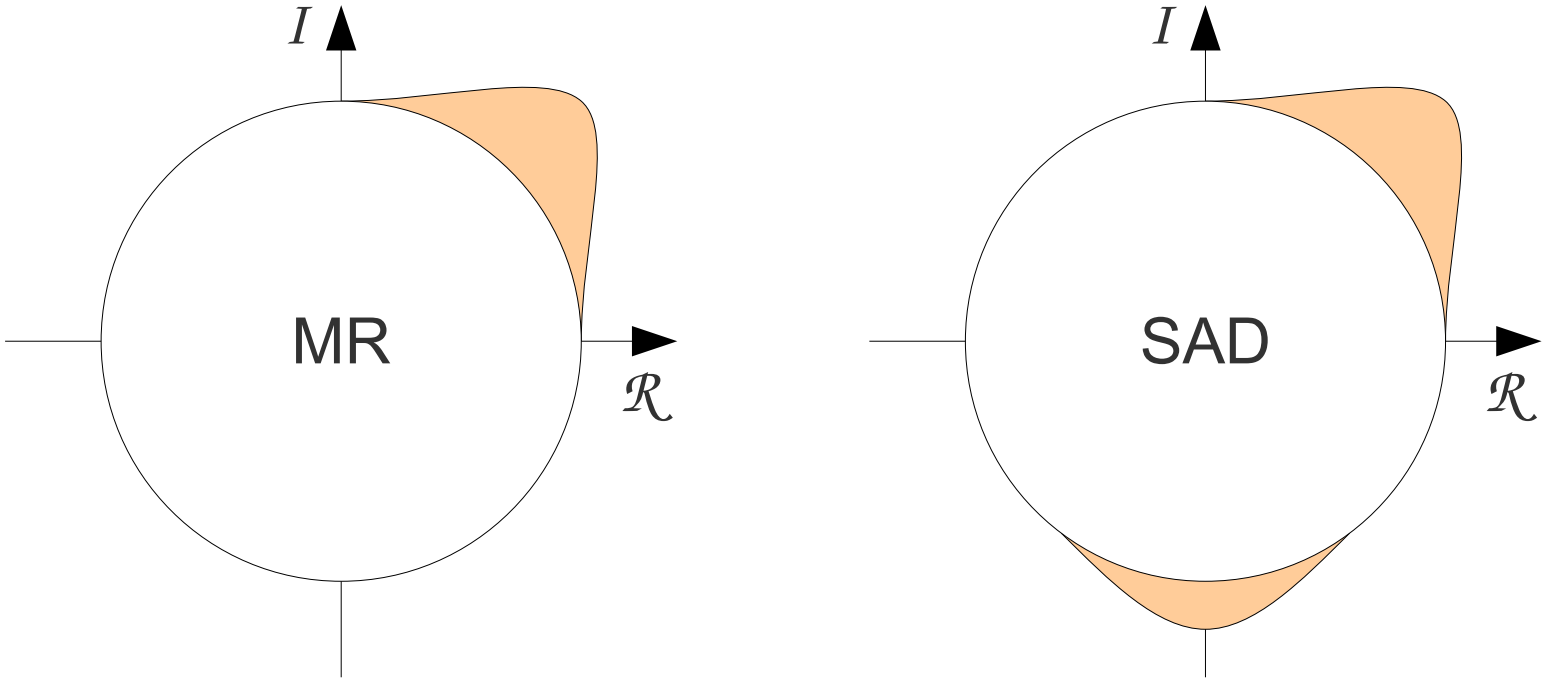
- Traditional density modification: e.g. 'dm', 'solomon', 'parrot', CNS
- Statistical density modification: e.g. 'resolve', 'pirate'



Density modification

Starting point:

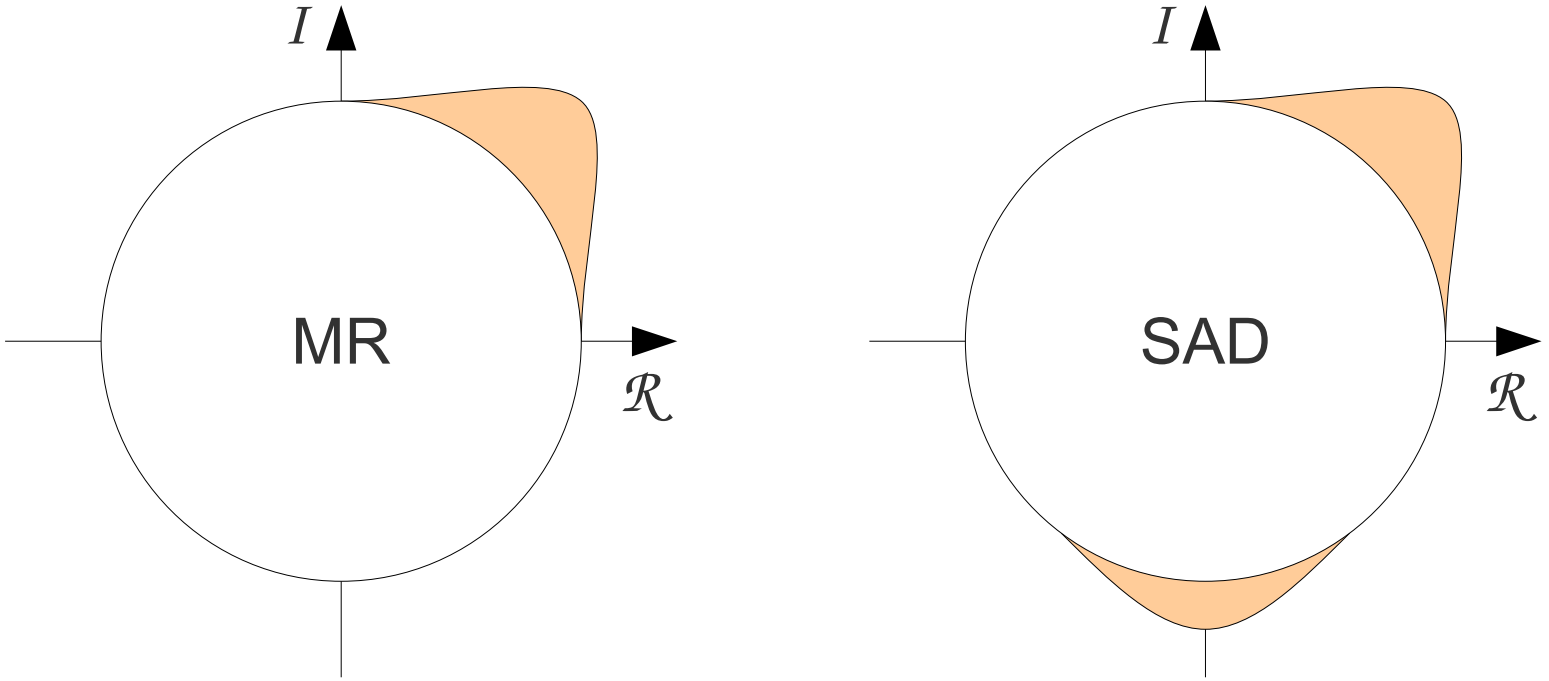
- Structure factor amplitudes
- Phase estimates:
  - MR: Unimodal distribution
  - SAD: Biomodal distribution



Density modification

How do we represent phase probability distributions?

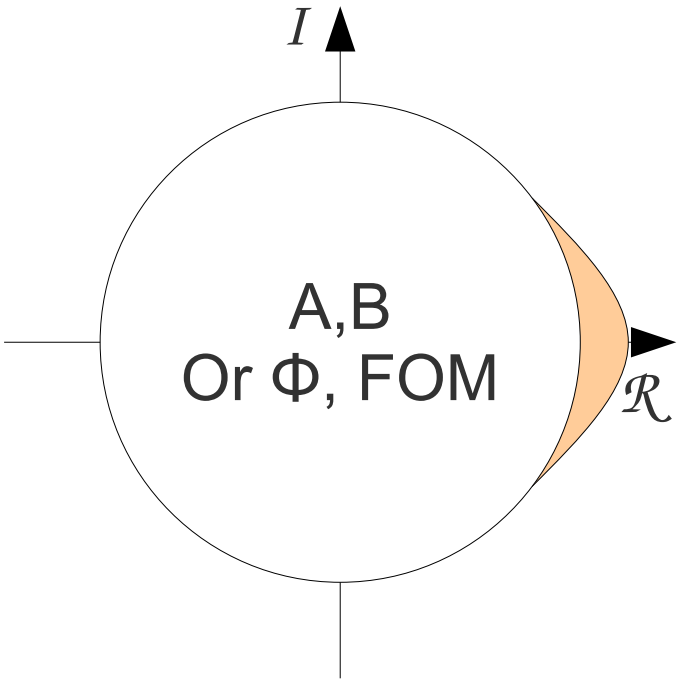
- Phase/figure of merit - Φ, FOM
  - (unimodal, MR only)
- Henrickson-Lattman coeffs – ABCD
  - (bimodal or unimodal, general)

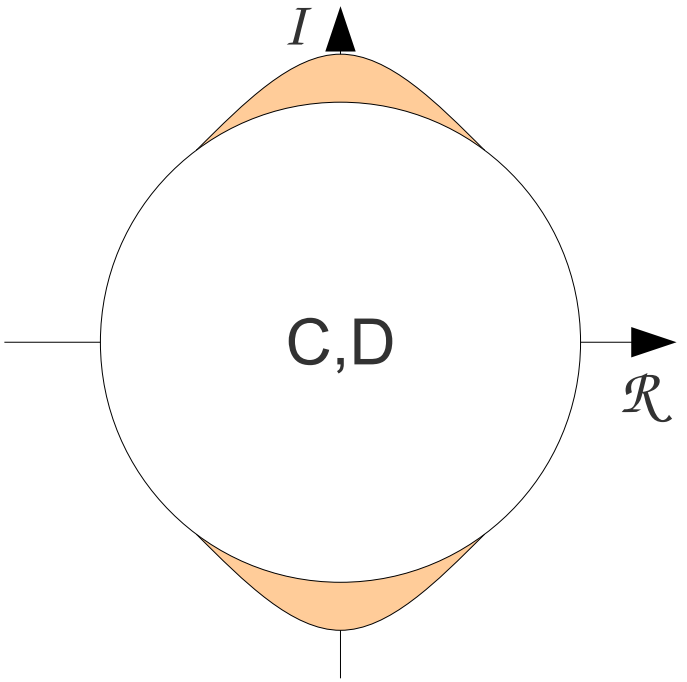


Density modification

A,B represent a unimodal distribution (equivalent to Φ, FOM)
C,D represent the superimposed biomodality.

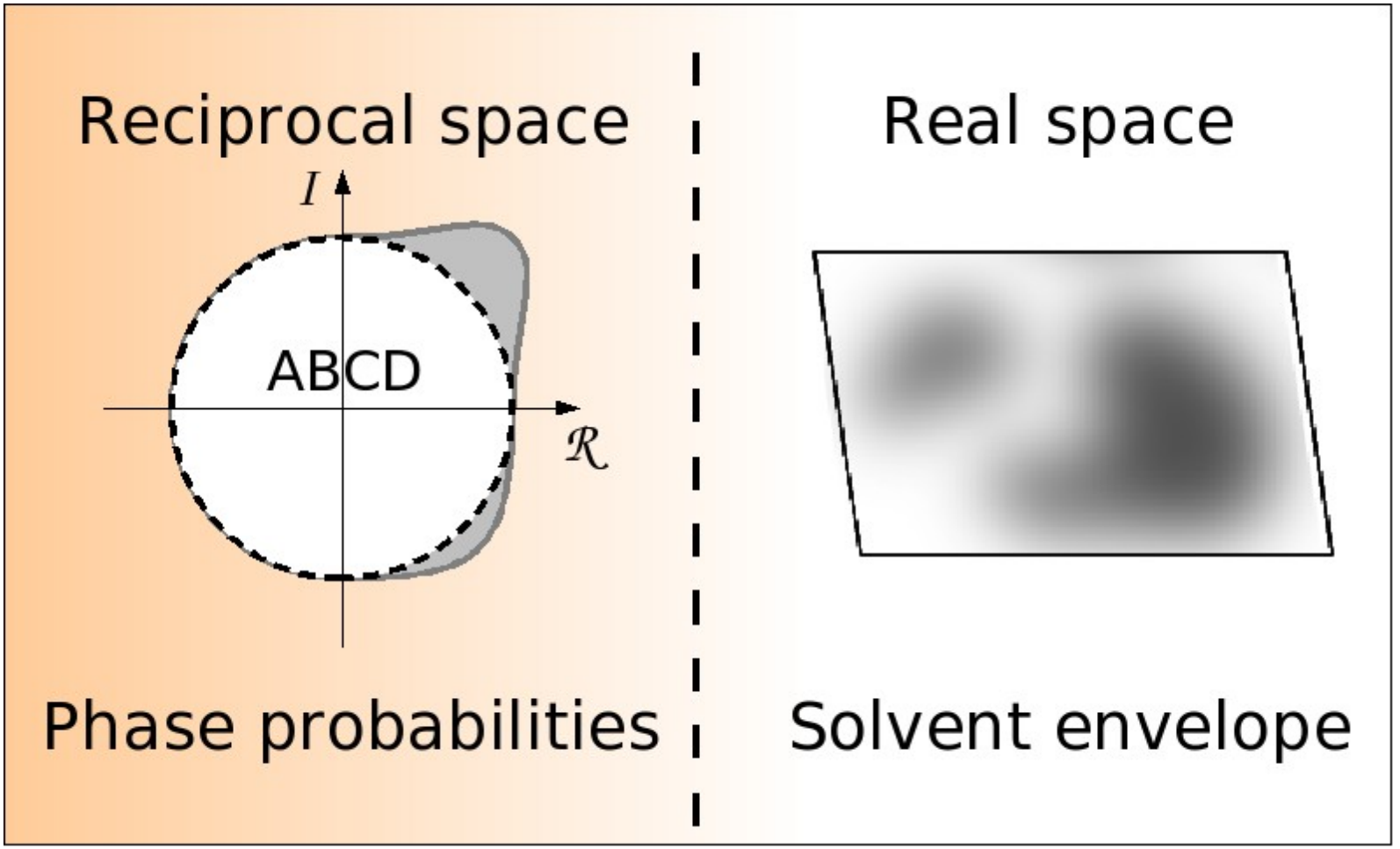
- Relative size and sign of A,B or C,D control the direction.
- Absolute size $(A^2+B^2)^{½}$ controls the sharpness.
- For MR, we get A,B (or Φ, FOM) i.e. C=D=0.
- Together A,B,C,D can describe a bimodal distribution with any combination of peak height and direction.





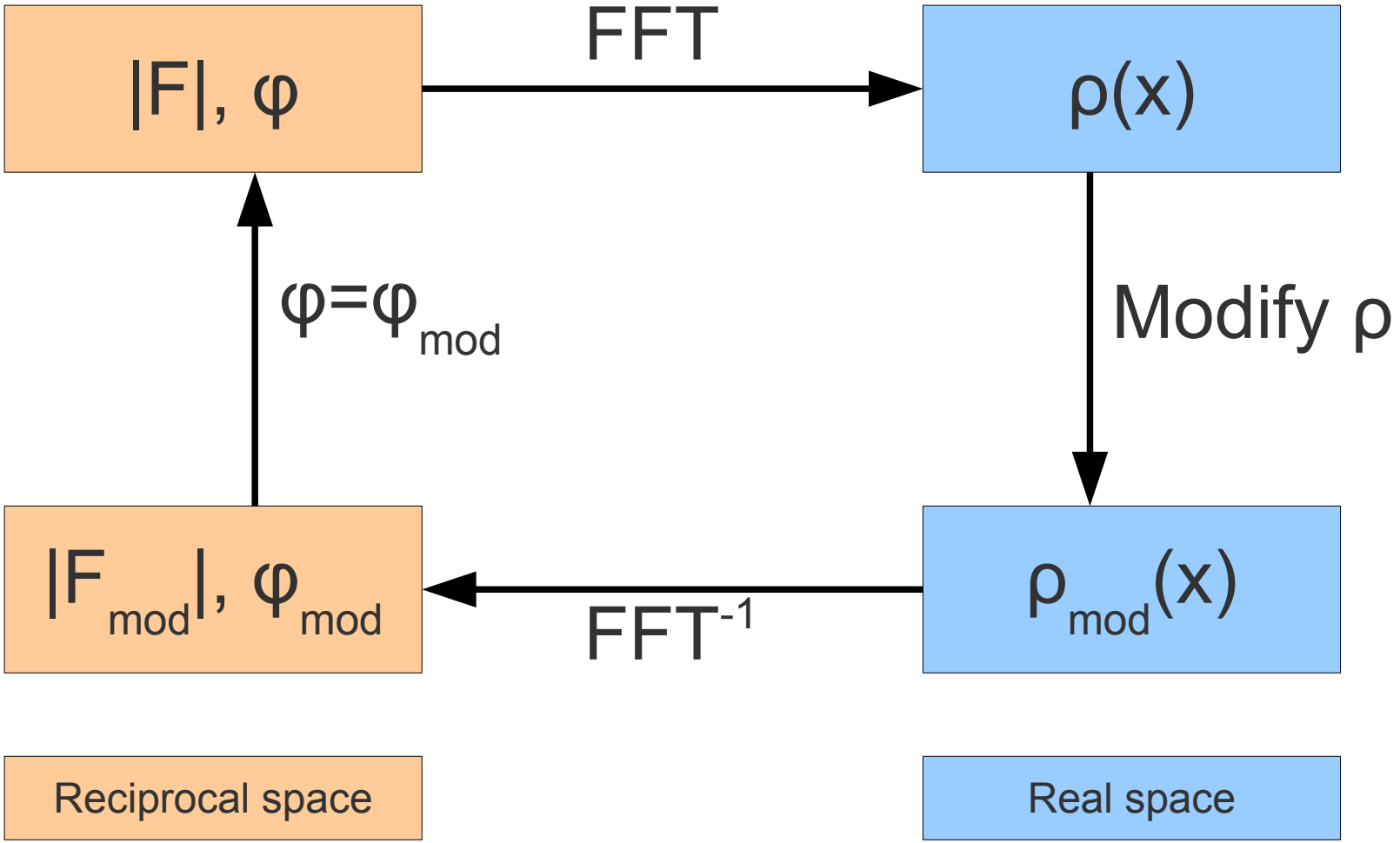
Density modification

- Density modification is a problem in combining information:



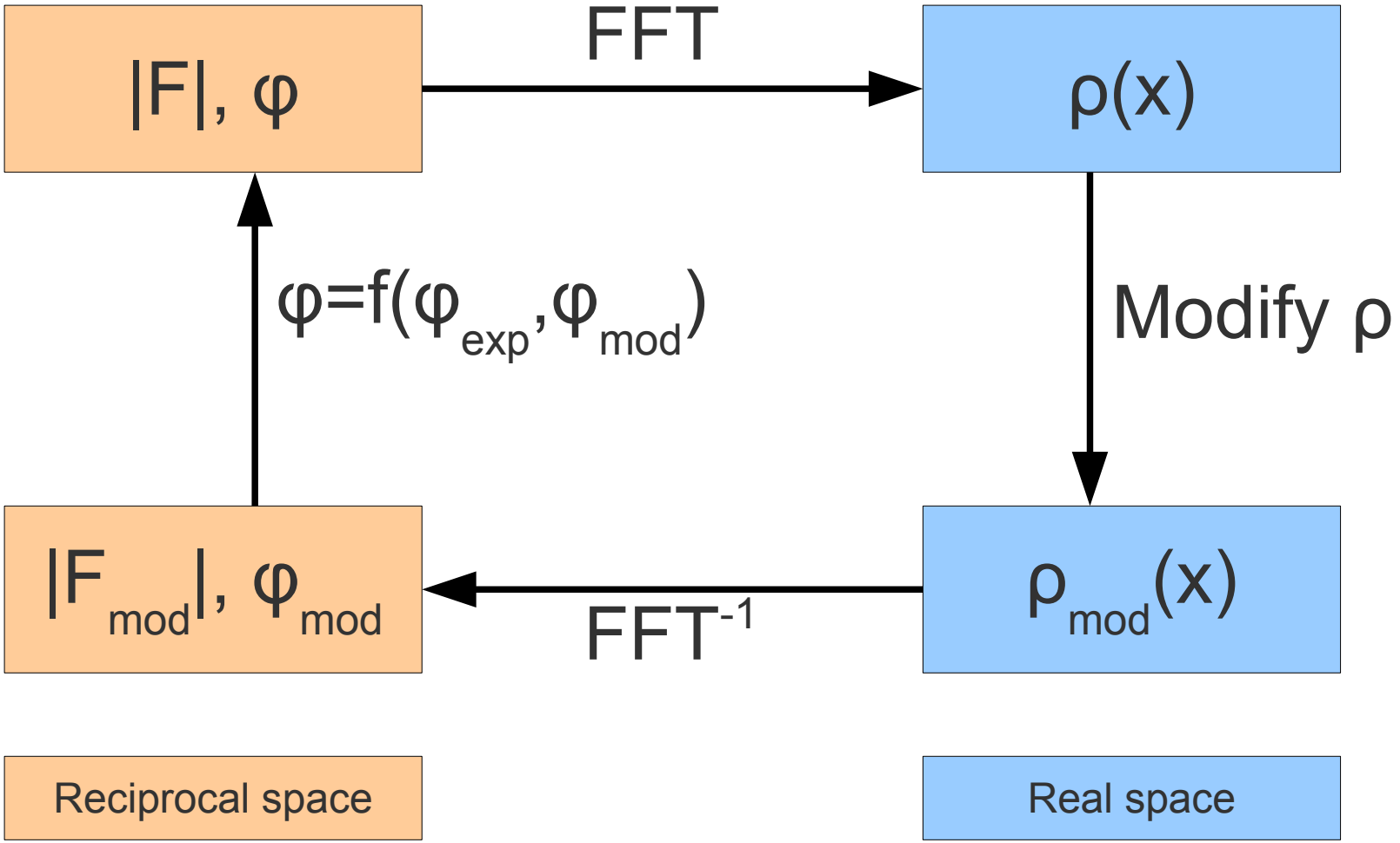
Density modification

1. Rudimentary calculation:



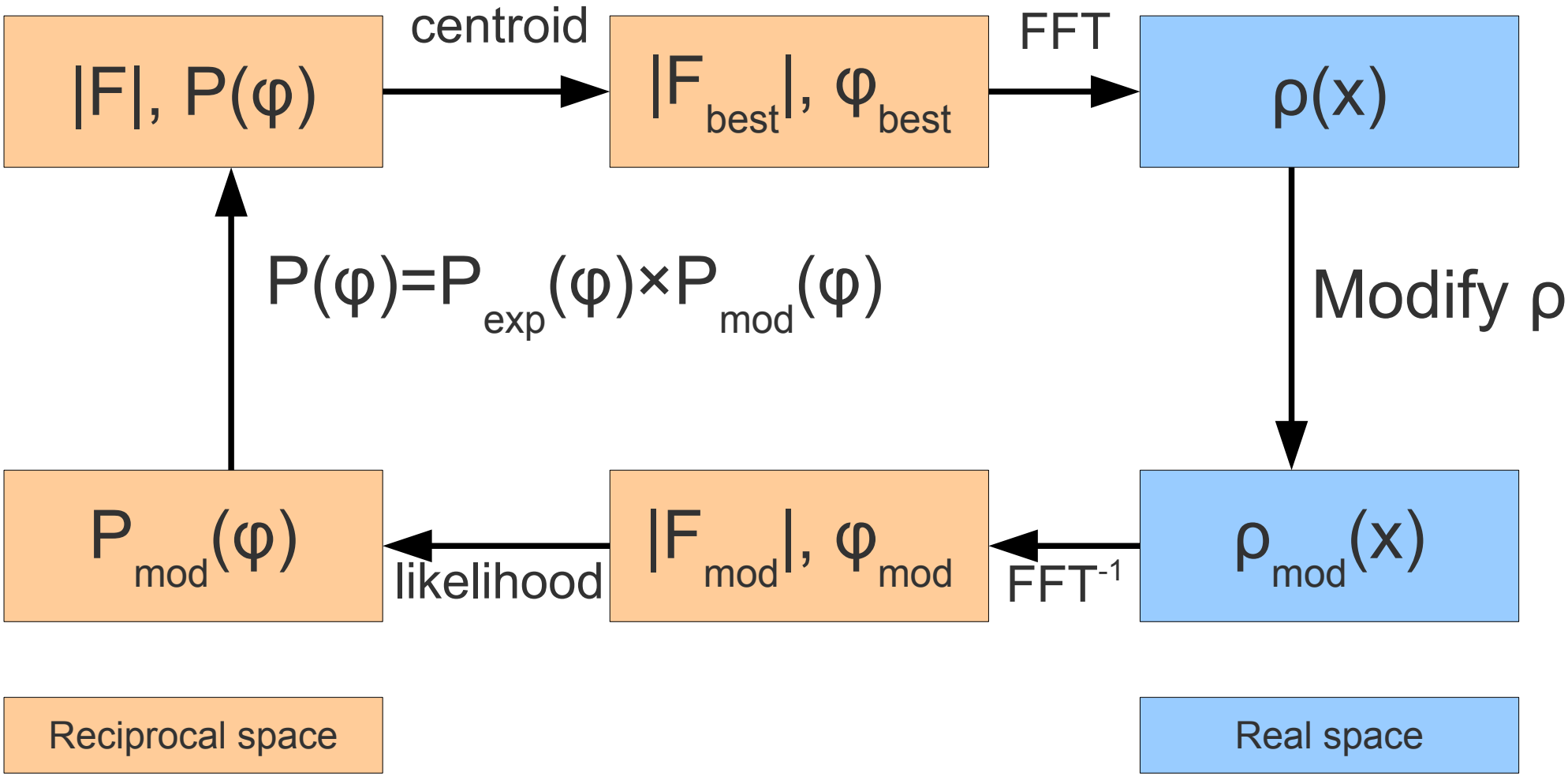
Density modification

2. Phase weighting:



Density modification

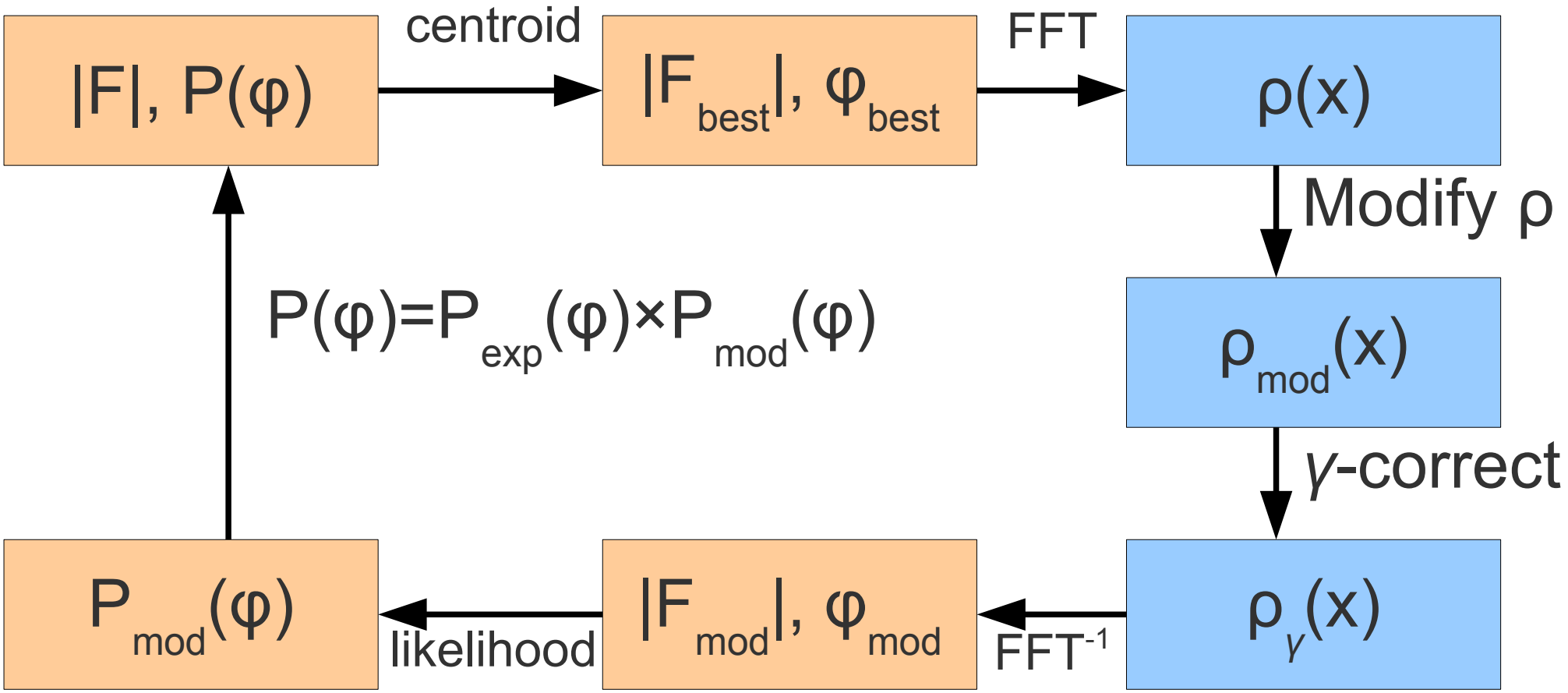
3. Phase probability distributions:



Density modification

DM, SOLOMON, (CNS)

4. Bias reduction (gamma-correction):

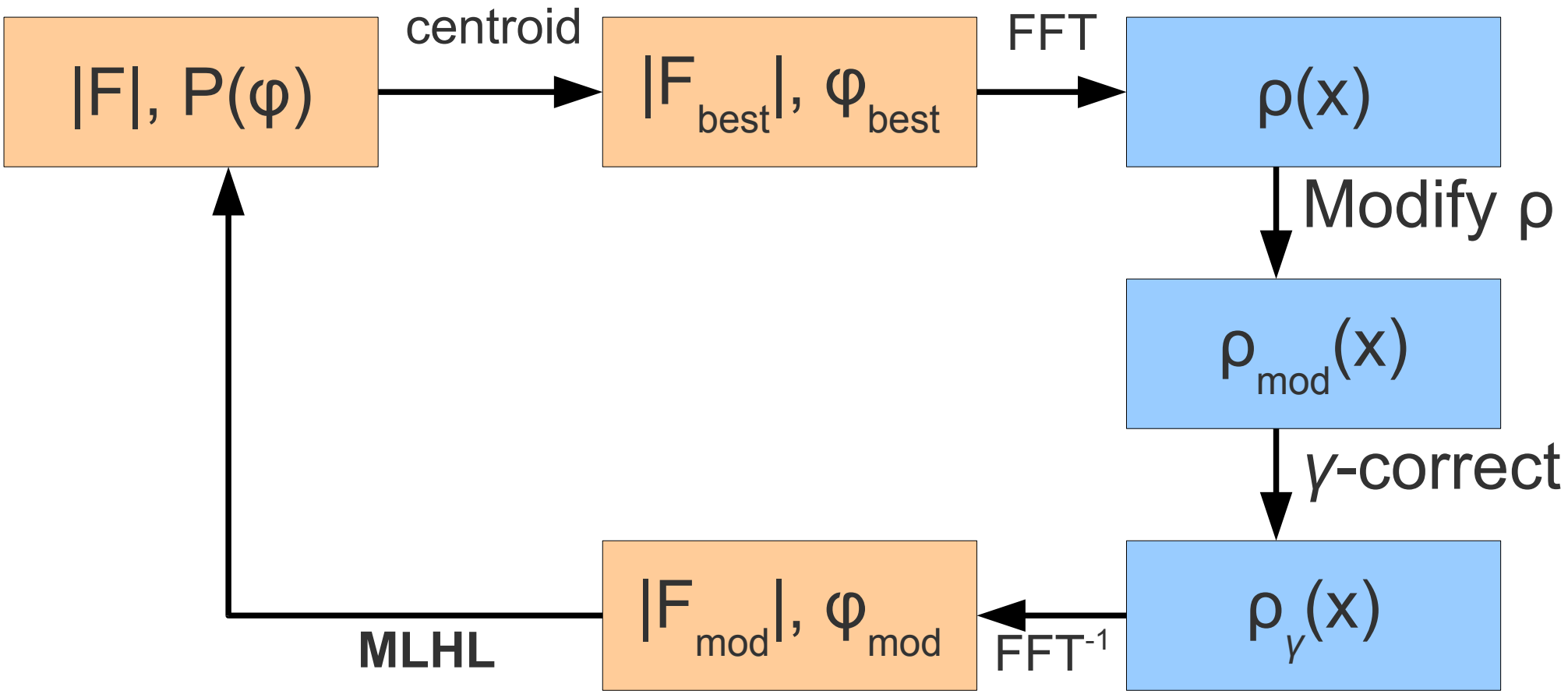


J.P.Abrahams

Density modification

PARROT

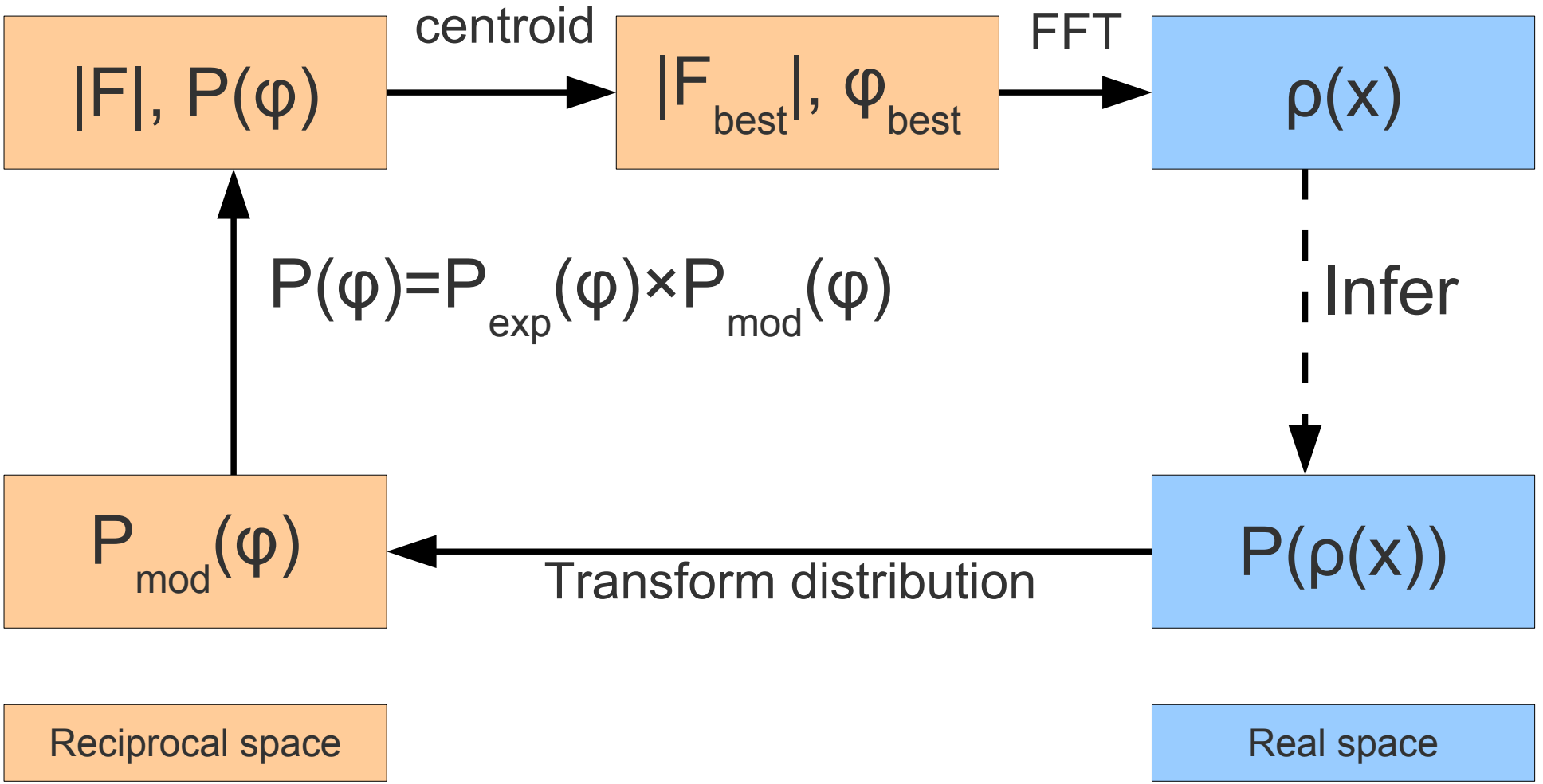
5. Maximum Likelihood H-L:



Density modification

RESOLVE, PIRATE

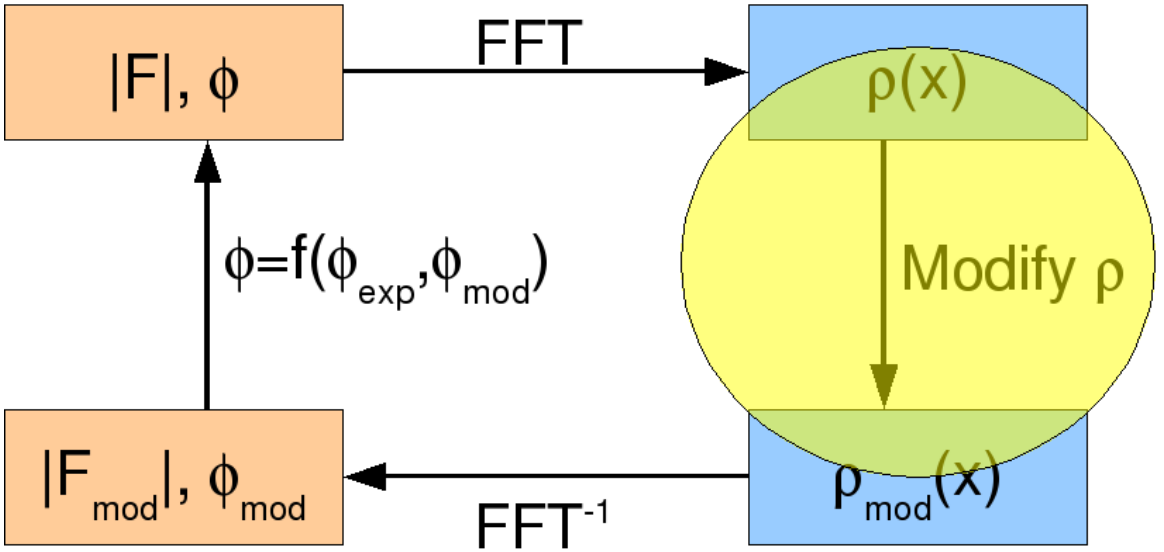
6. Statistical density modification:



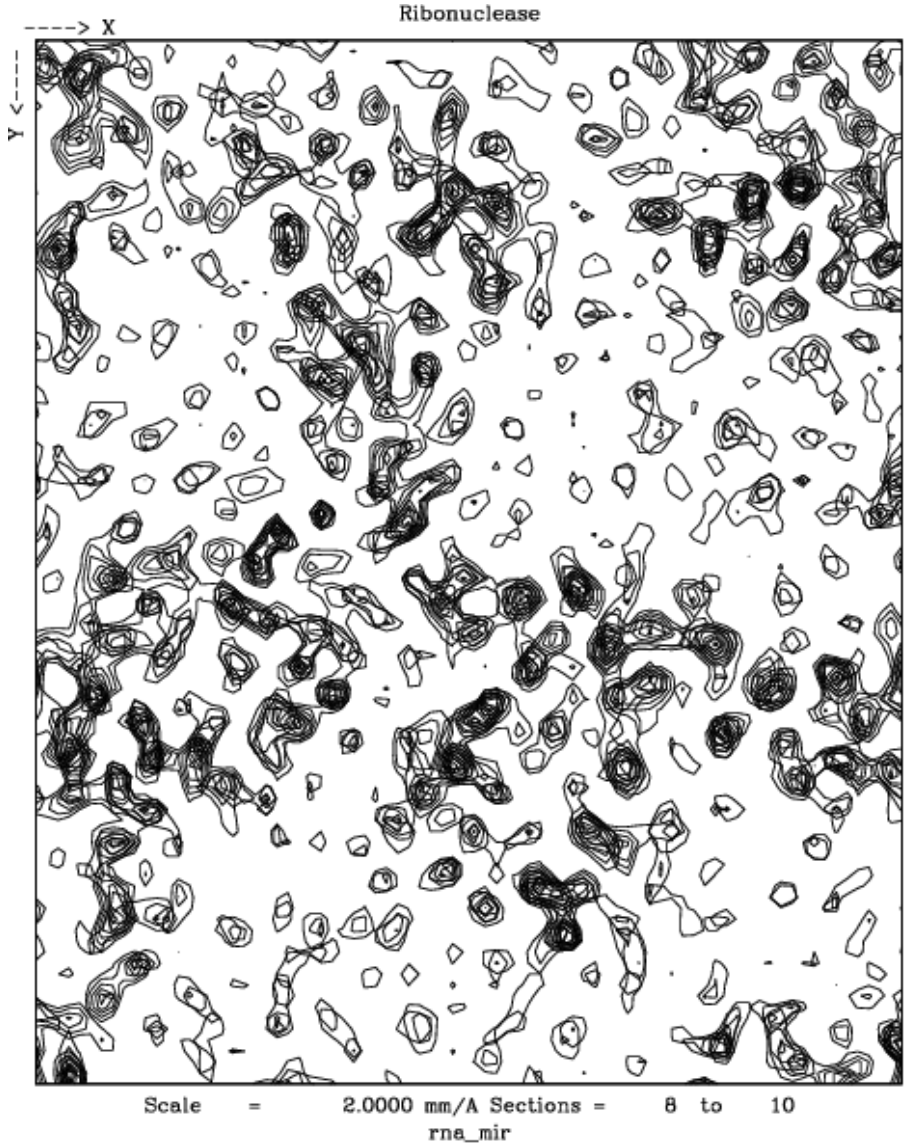
Density modification

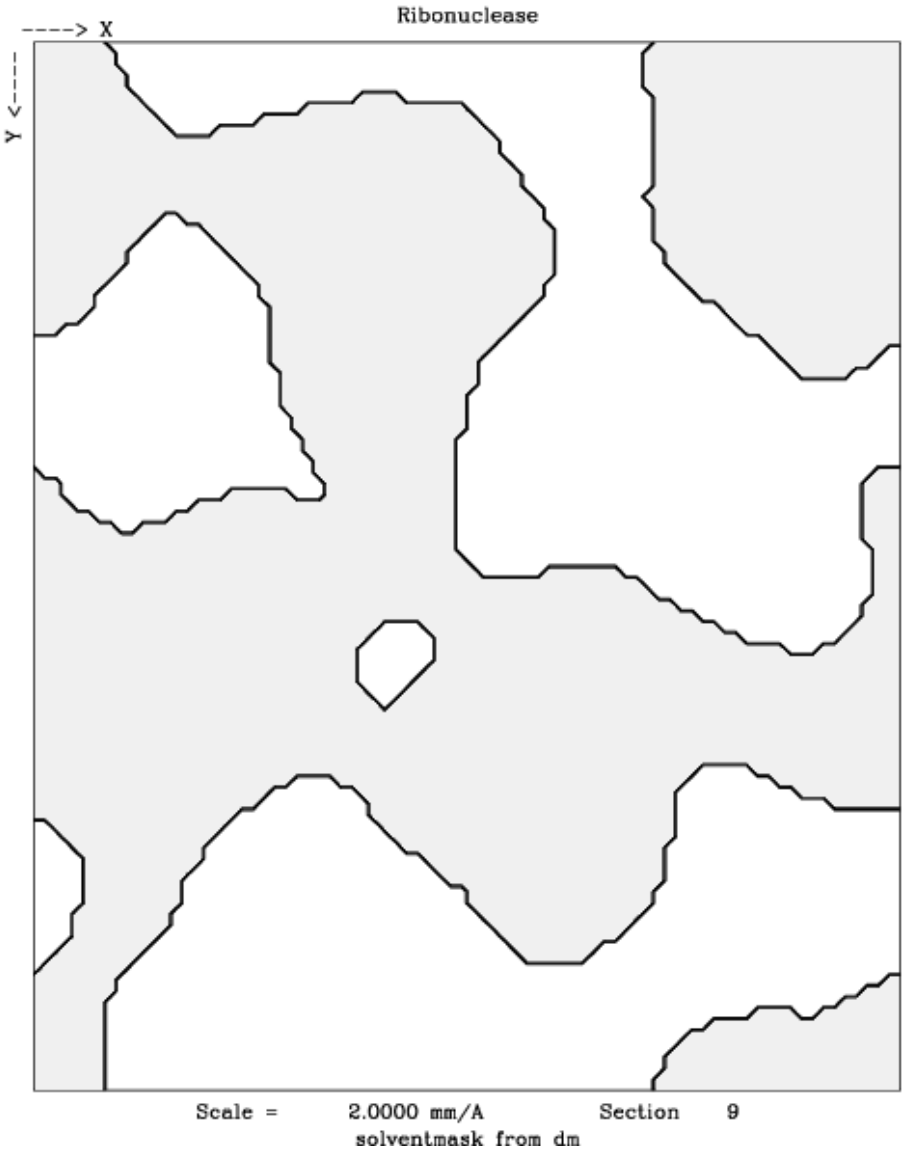
Traditional density modification techniques:

- Solvent flattening
- Histogram matching
- Non-crystallographic symmetry (NCS) averaging



Solvent flattening





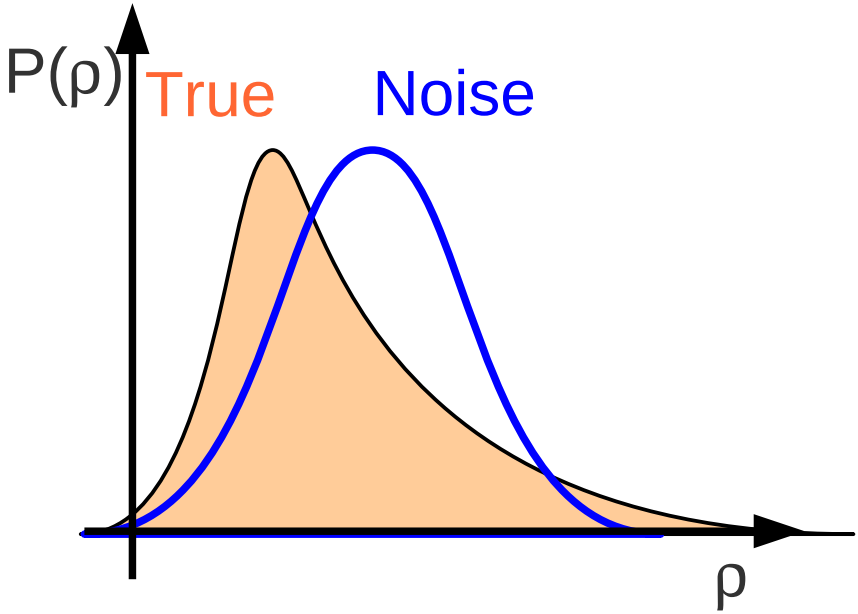
Histogram matching

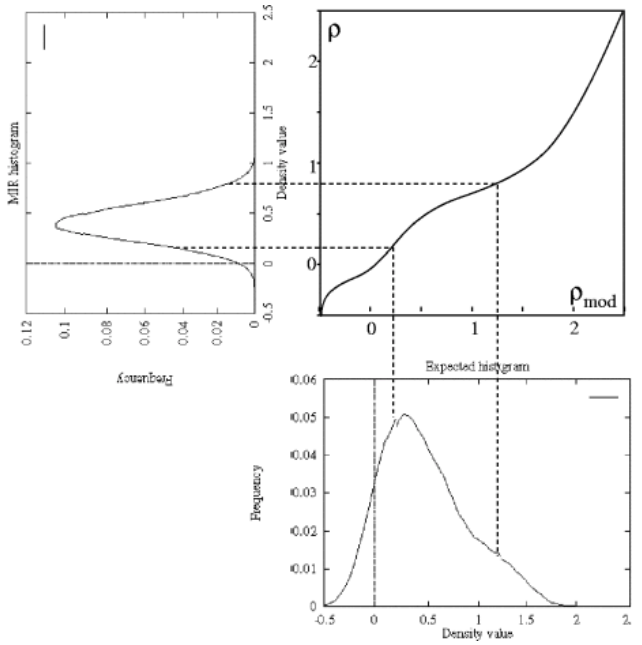
A technique from image processing for modifying the protein region.

- 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 4A.



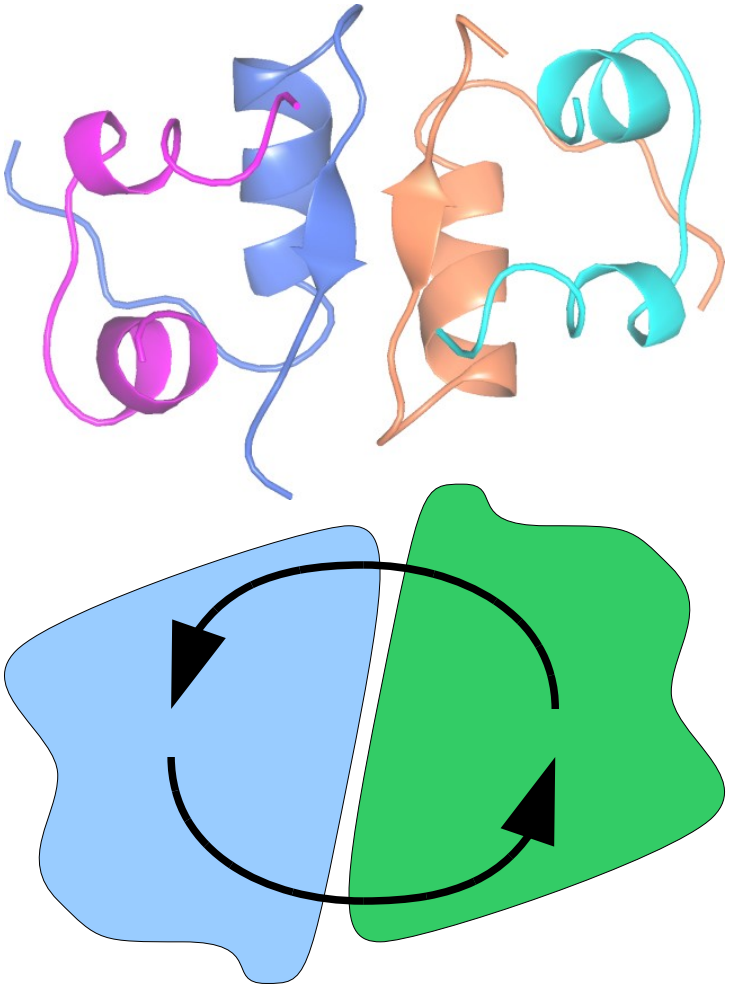


Kevin Cowtan, cowtan@ysbl.york.ac.uk

2012

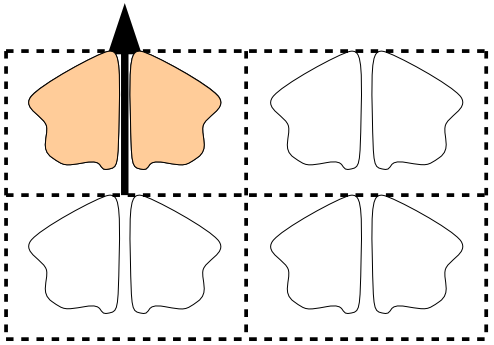
Non-crystallographic symmetry

- 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!

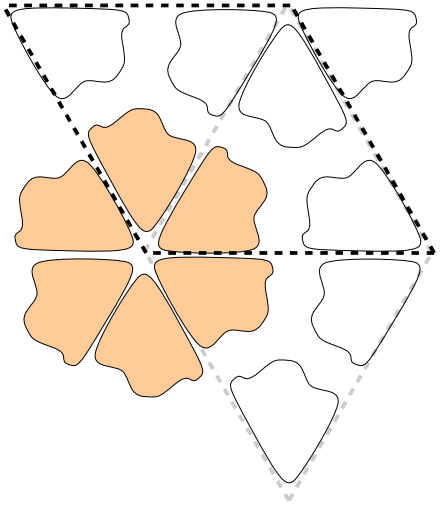


Non-crystallographic symmetry

Crystallographic

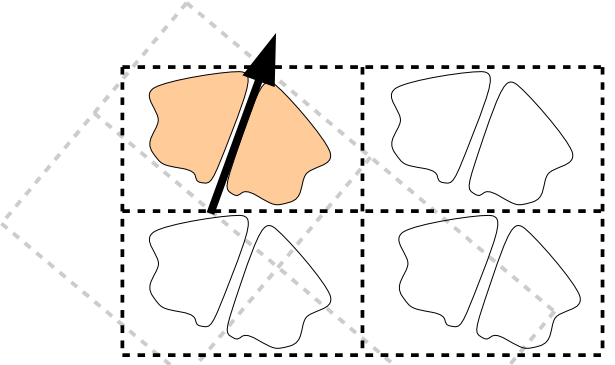


Aligned
2-fold

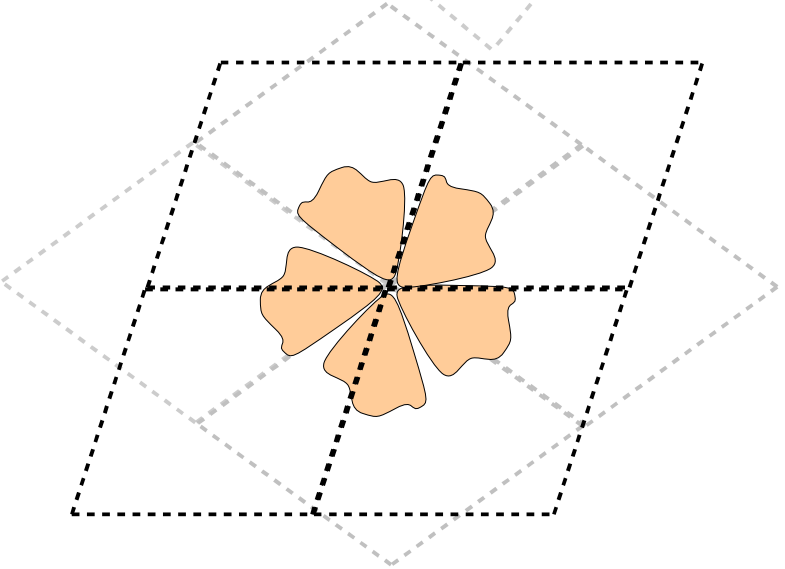


Aligned
6-fold

Non-crystallograpic



Unaligned
2-fold



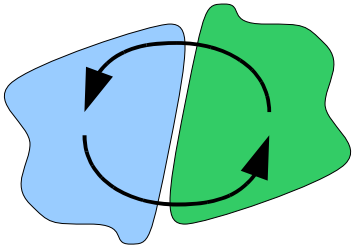
Aligned
5-fold

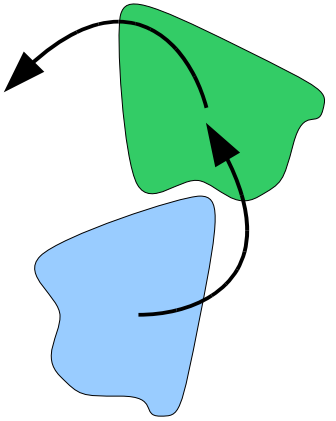
Kevin Cowtan, cowtan@ysbl.york.ac.uk

Non-crystallographic symmetry

Useful terms:

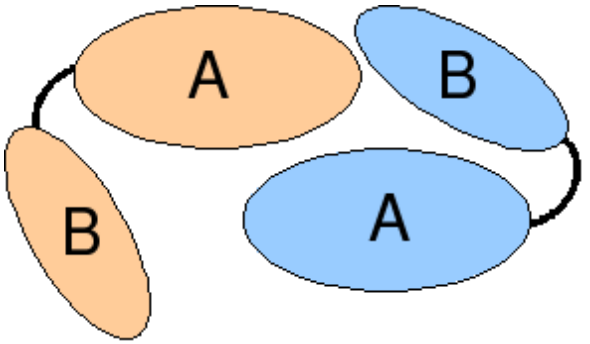
- Proper and improper NCS: (closed and open)

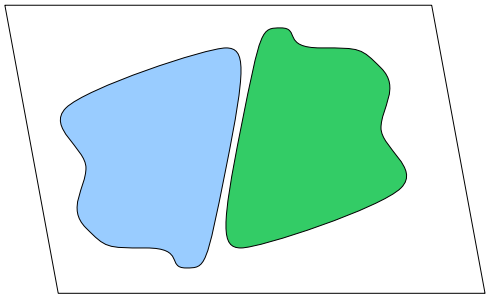


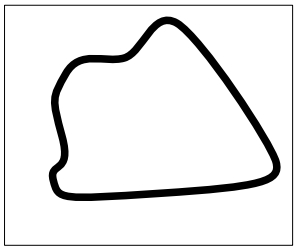


- Multi-domain averaging:

- Multi-crystal averaging:



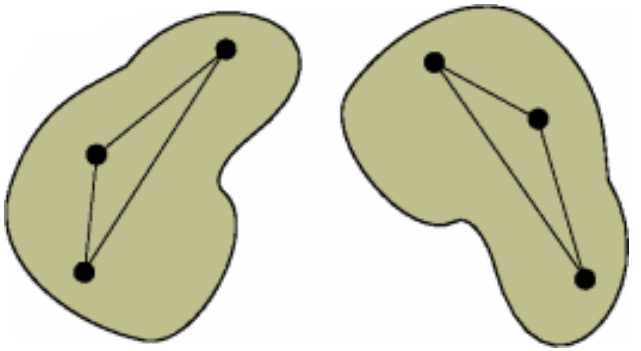




Kevin Cowtan, cowtan@ysbl.york.ac.uk

Non-crystallographic symmetry

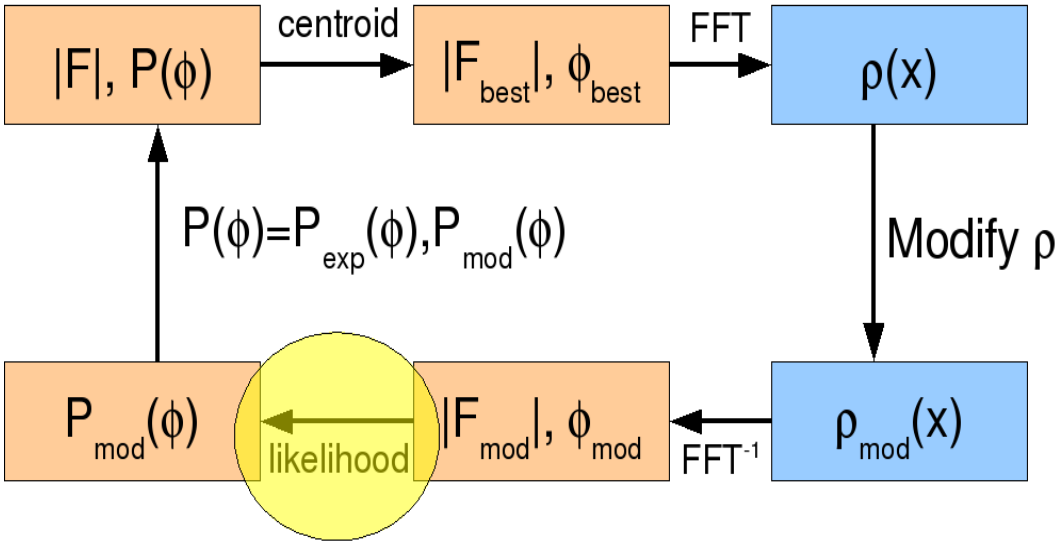
- How do you know if you have NCS?
  - Cell content analysis – how many monomers in ASU?
  - Self-rotation function.
  - Difference Pattersons (pseudo-translation only).
- How do you determine the NCS?
  - **From heavy atoms.**
  - From initial model building.
  - From molecular replacement.
  - *From density MR (hard).*
- Mask determined automatically.



Estimating phase probabilities

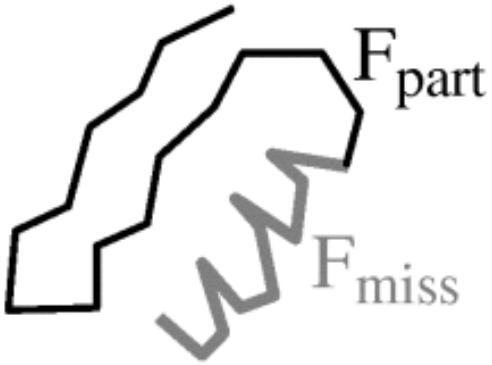
Problem: How do we go from a single phase estimate to a full phase probability distribution?

- 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.

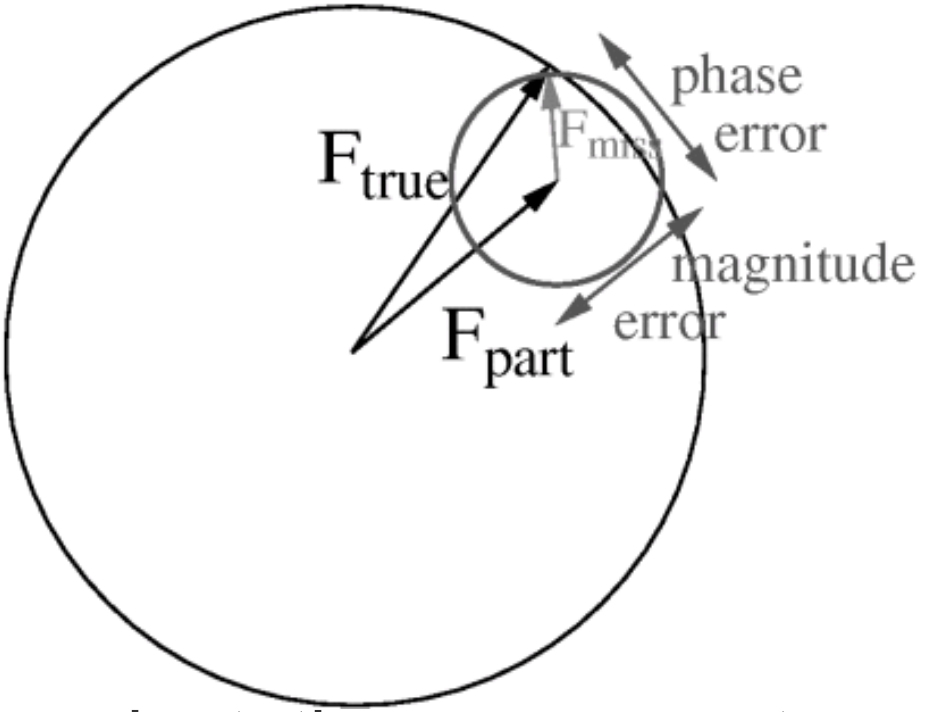


Estimating phase probabilities

Sim/σ_A weighting:



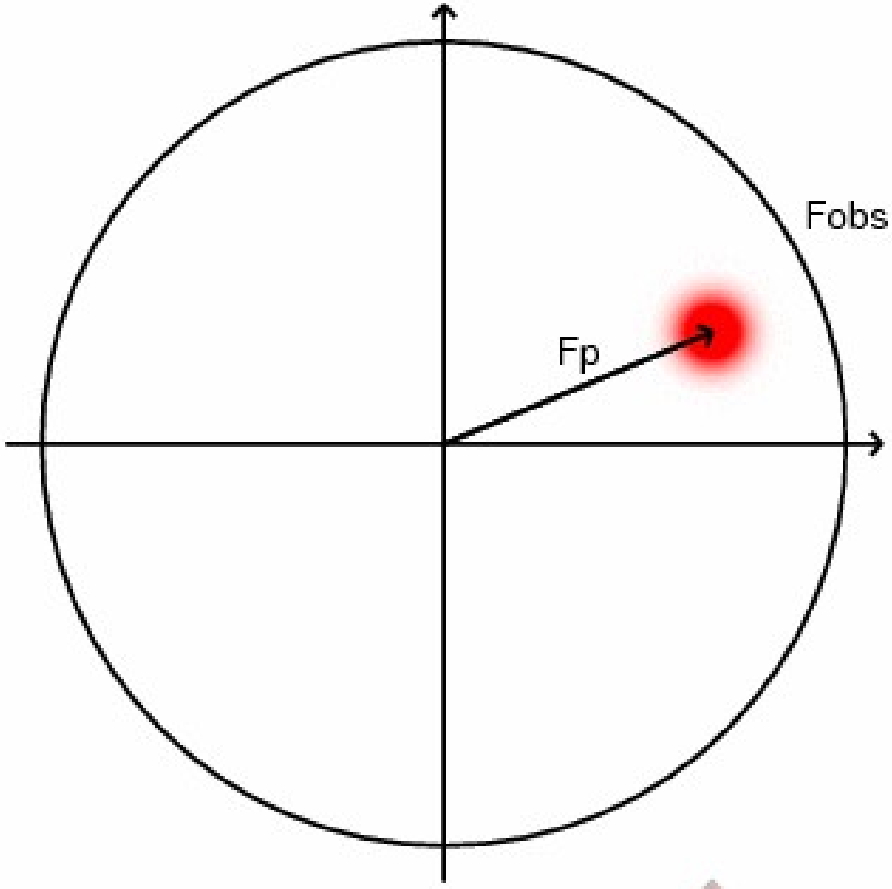
$F_{true} = F_{part} + F_{miss}$

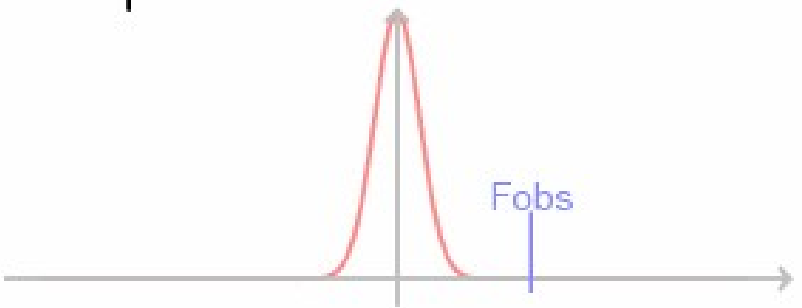


We know $|F_{true}|$, $|F_{part}|$, ϕ_{part}

Assuming ϕ_{part}, ϕ_{miss} are independent, then we expect the difference in magnitudes between $|F_{true}|$ and $|F_{part}|$, averaged over reflections, to give an indication of the phase error.

Estimating phase probabilities





Combining phase probabilities

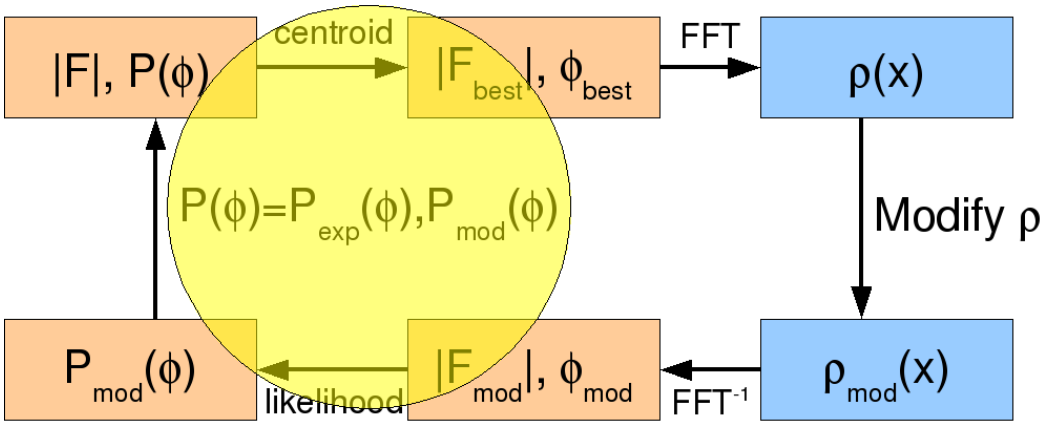
Once we have an estimate for the error in ϕ_{mod}, we can construct a probability distribution $P_{mod}(\phi)$.

The the next cycle can be started with

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

Problem: $P_{exp}(\phi)$ and $P_{mod}(\phi)$ are not independent.

The result is bias, increasing with cycle.



Bias reduction

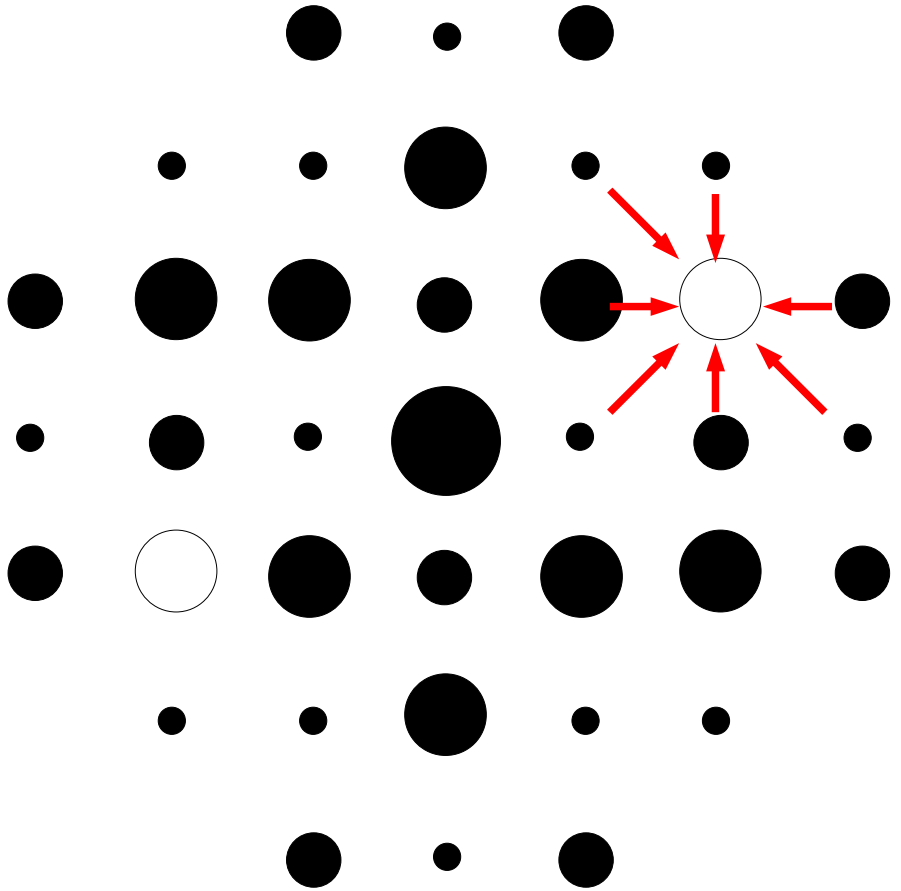
Solution:

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 γ-correction



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

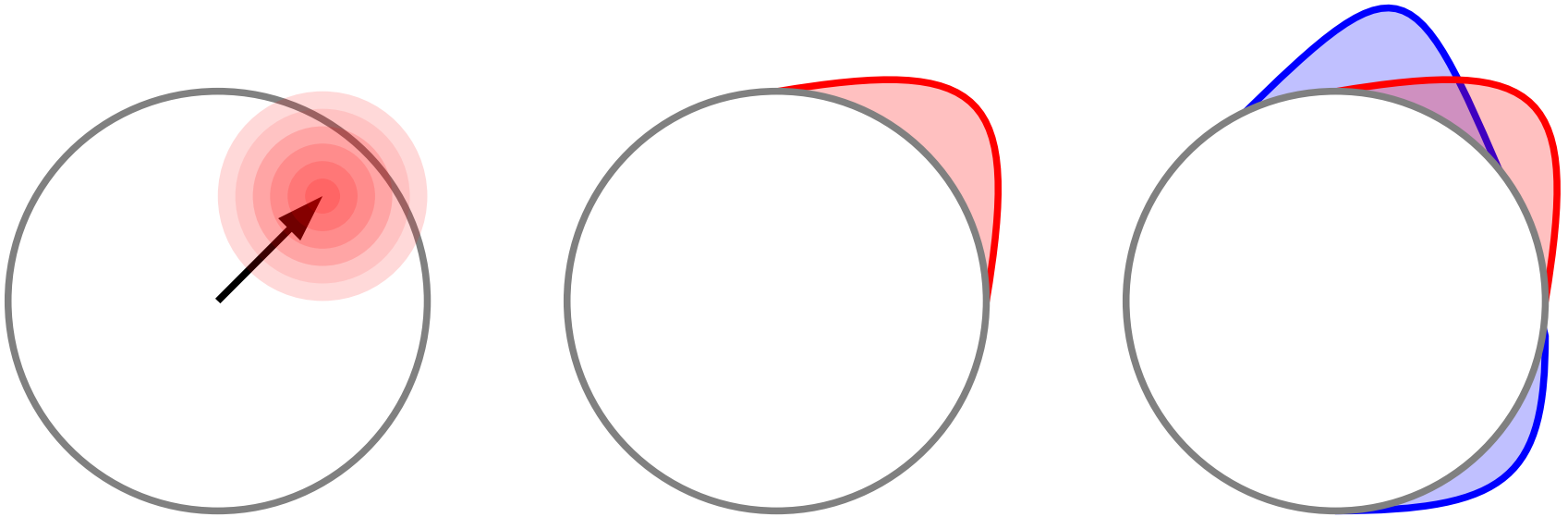
Density modification in Parrot

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...

Estimating phase probabilities

Traditional approach: Rice likelihood function



Estimate the accuracy of the modified F/phase

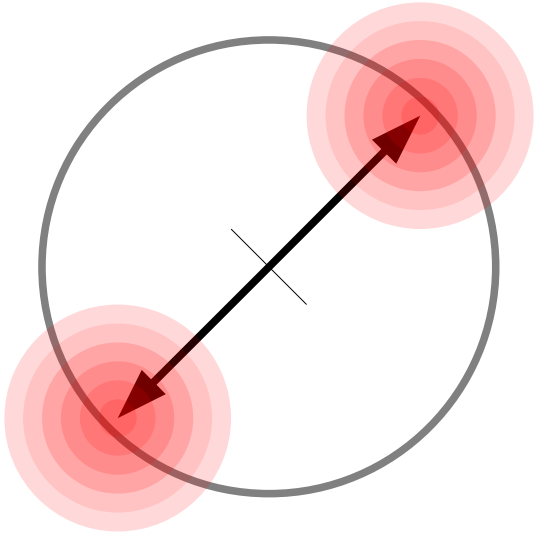
Turn this into a phase probability distribution

Combine with the experimental phase probability

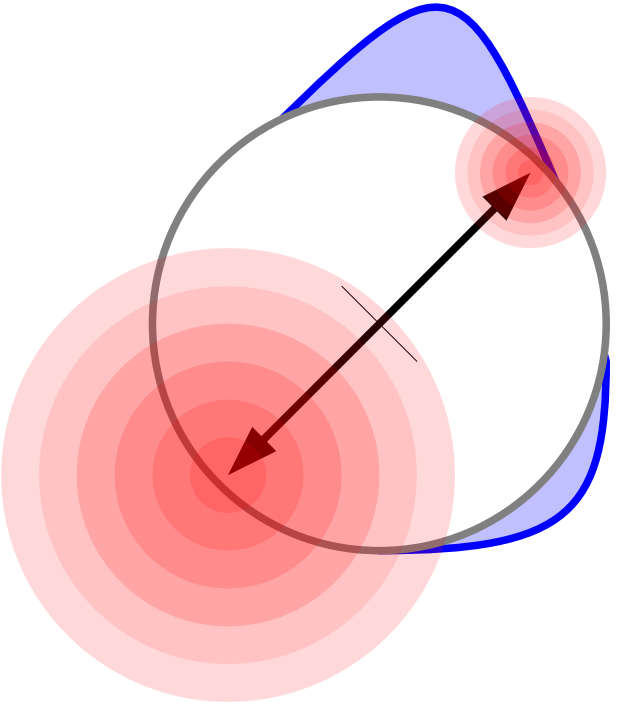
The estimate for the accuracy of the modified F/phase come from the agreement between the modified F and the observed F. **Source of bias.**

Estimating phase probabilities

Problem:



Error estimation does not take into account experimental phase information

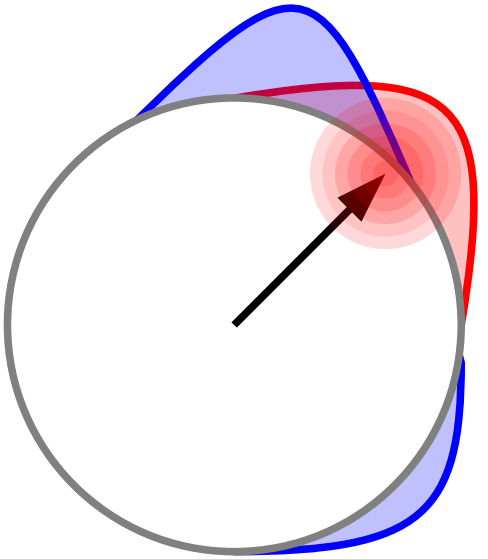


The experimental data tells us that the probable error is different in the two cases

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

Estimating phase probabilities

Solution:
MLHL-type likelihood target function.

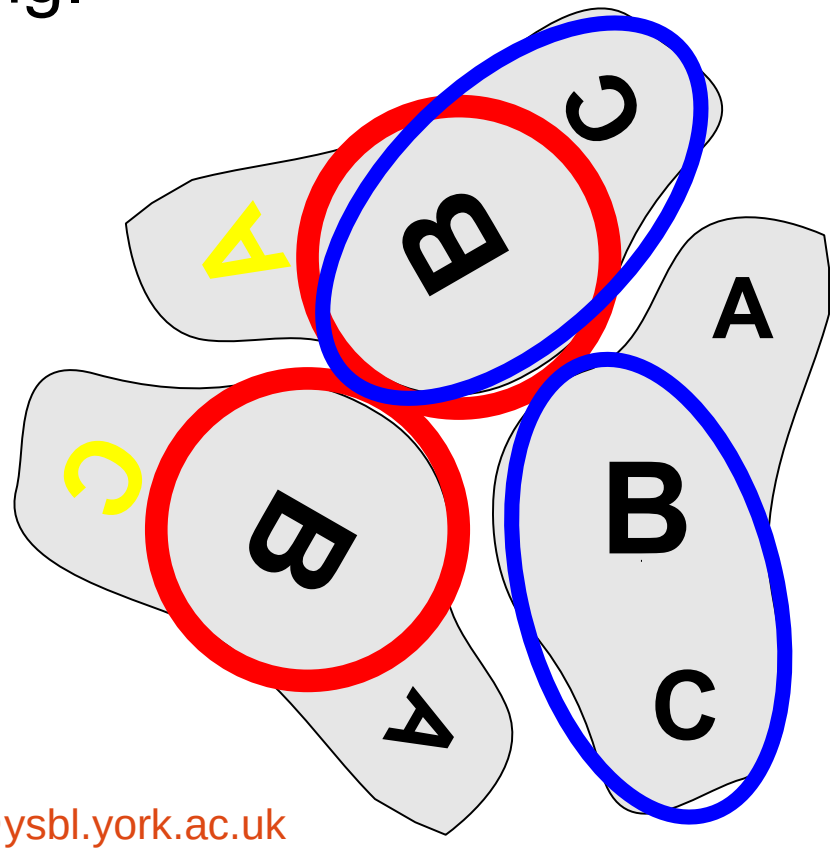


Perform the error estimation and phase combination in a single step, using a likelihood function which incorporates the experimental phase information as a prior.
This is the same MLHL-type like likelihood refinement target used in modern refinement software such as *refmac* or *phenix.refine*.

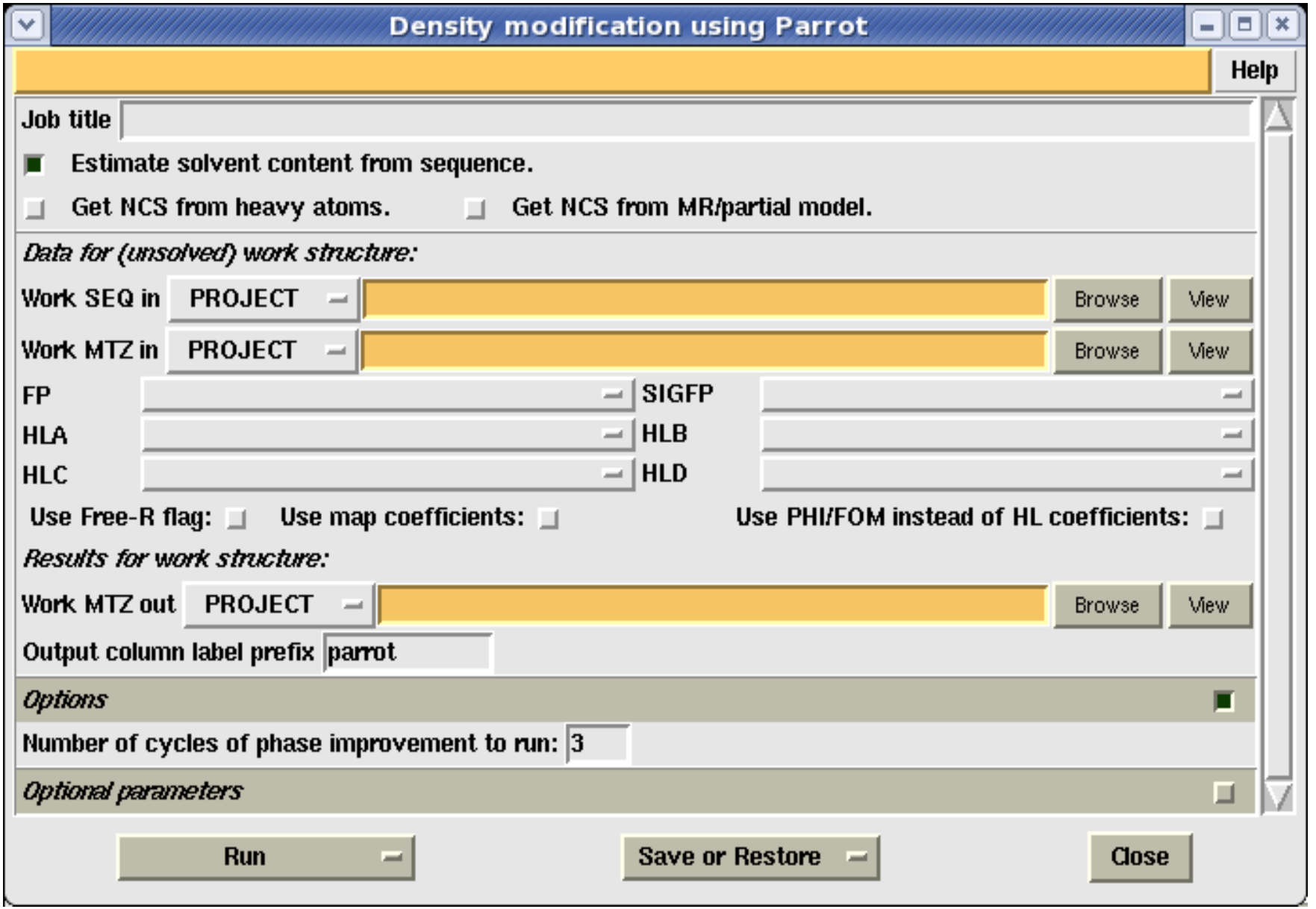
Recent Developments:

Pairwise-weighted NCS averaging:

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



Parrot


Density modification using Parrot
Help
Job title
Estimate solvent content from sequence.
Get NCS from heavy atoms.
Get NCS from MR/partial model.
Data for (unsolved) work structure:
Work SEQ in
PROJECT
Browse
View
Work MTZ in
PROJECT
Browse
View
FP
SIGFP
HLA
HLB
HLC
HLD
Use Free-R flag:
Use map coefficients:
Use PHI/FOM instead of HL coefficients:
Results for work structure:
Work MTZ out
PROJECT
Browse
View
Output column label prefix
parrot
Options
Number of cycles of phase improvement to run:
3
Optional parameters
Run
Save or Restore
Close

Parrot

Summary:

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

- Fully automated
- Fast
- Better results than DM

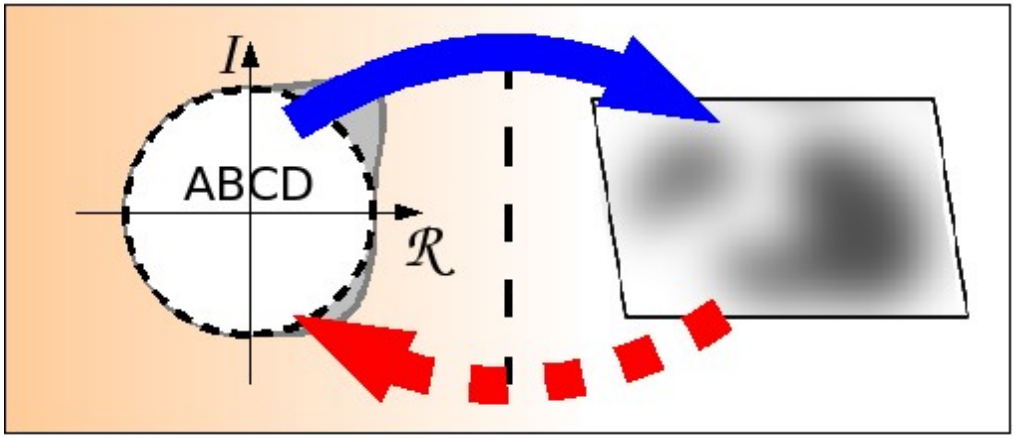
Density Modification

Kevin Cowtan, York.

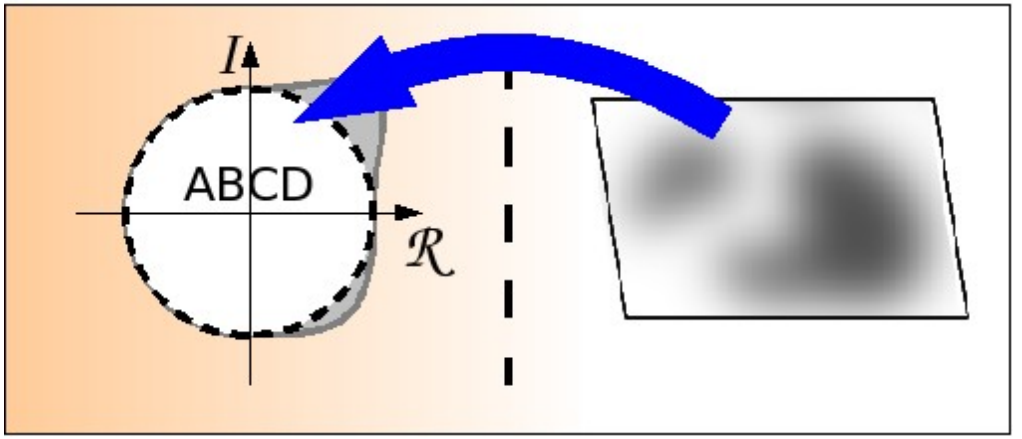
Statistical density modification:
e.g. Resolve, Pirate

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?

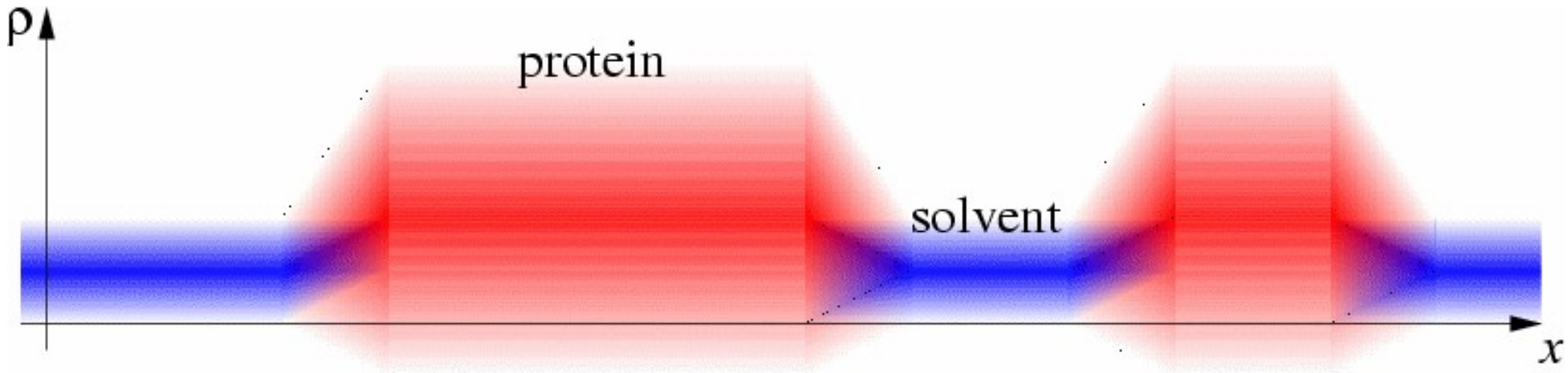


- Statistical density modification:
  *Take the **mask** to the **phases**.*
  - First convert mask to probability.
  - Then transform that probability.



Statistical density modification

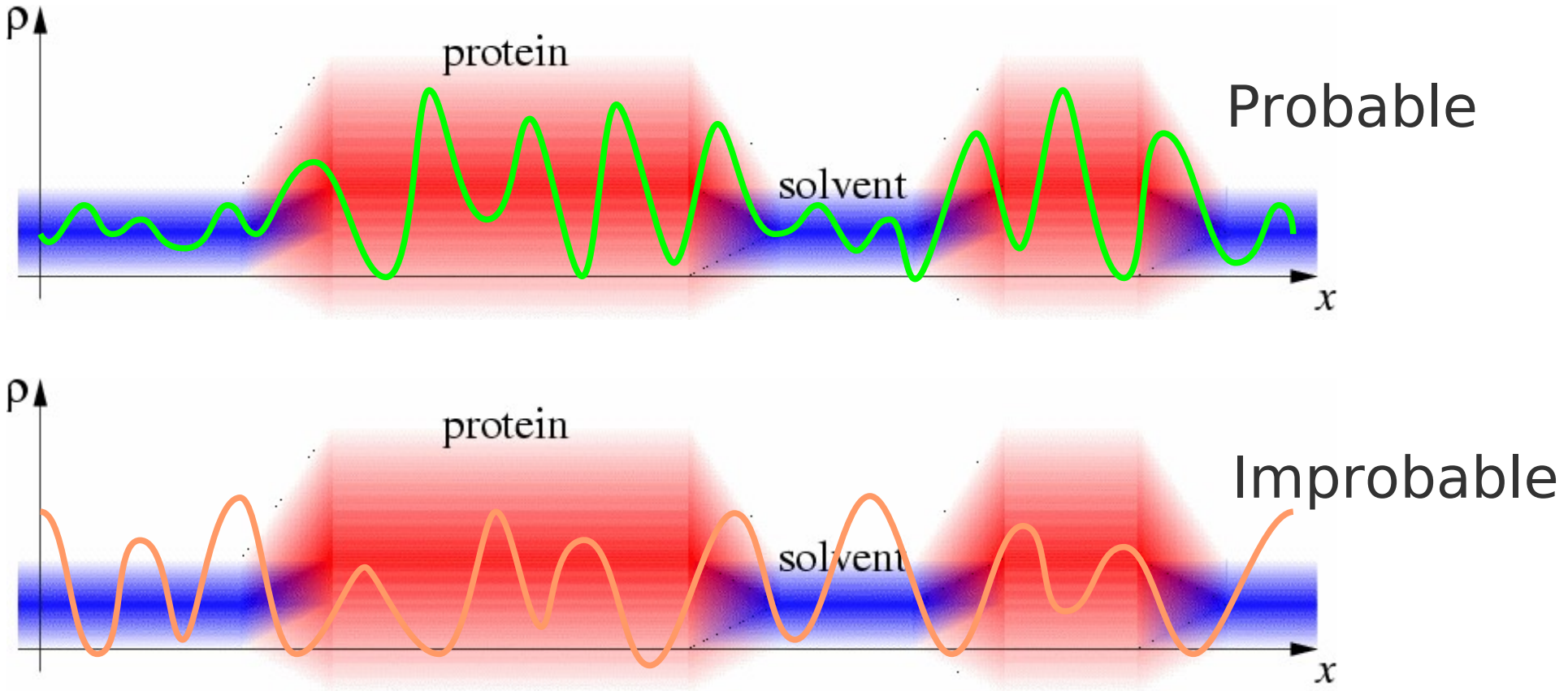
- Form a statistical description of expected map features.



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

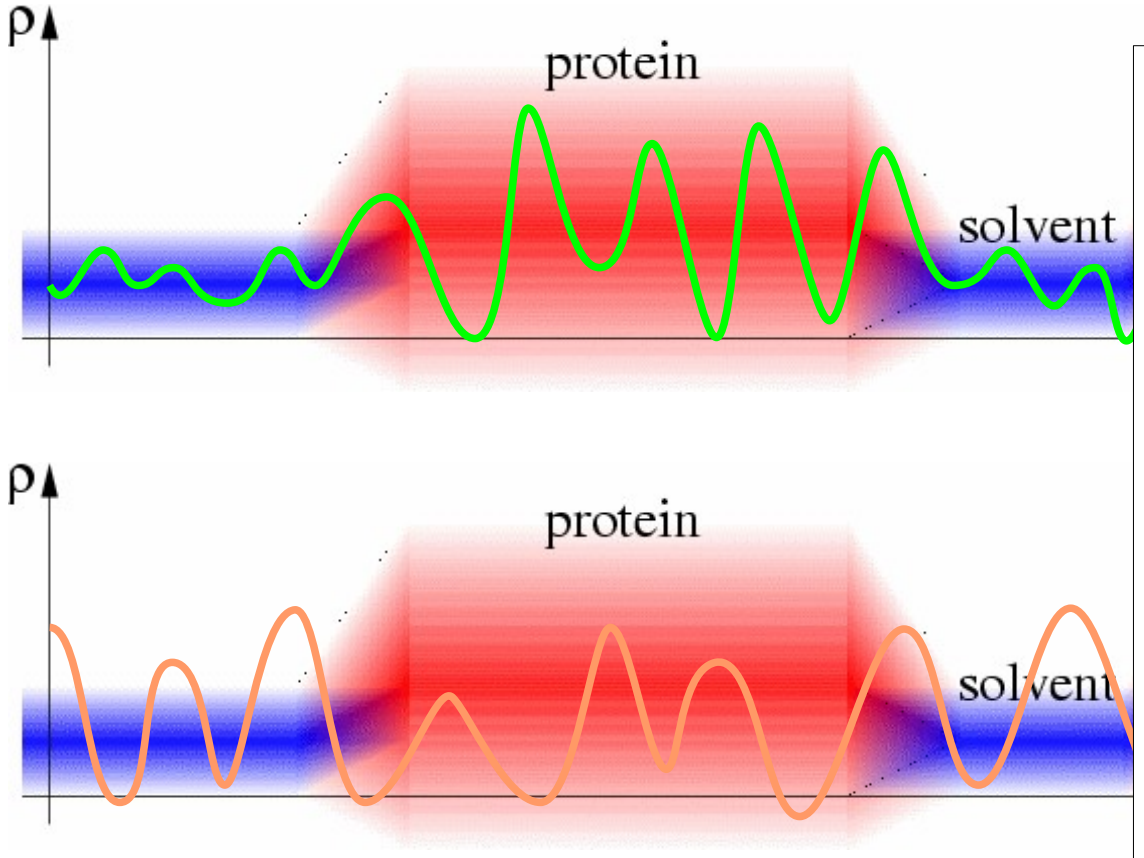
Statistical density modification

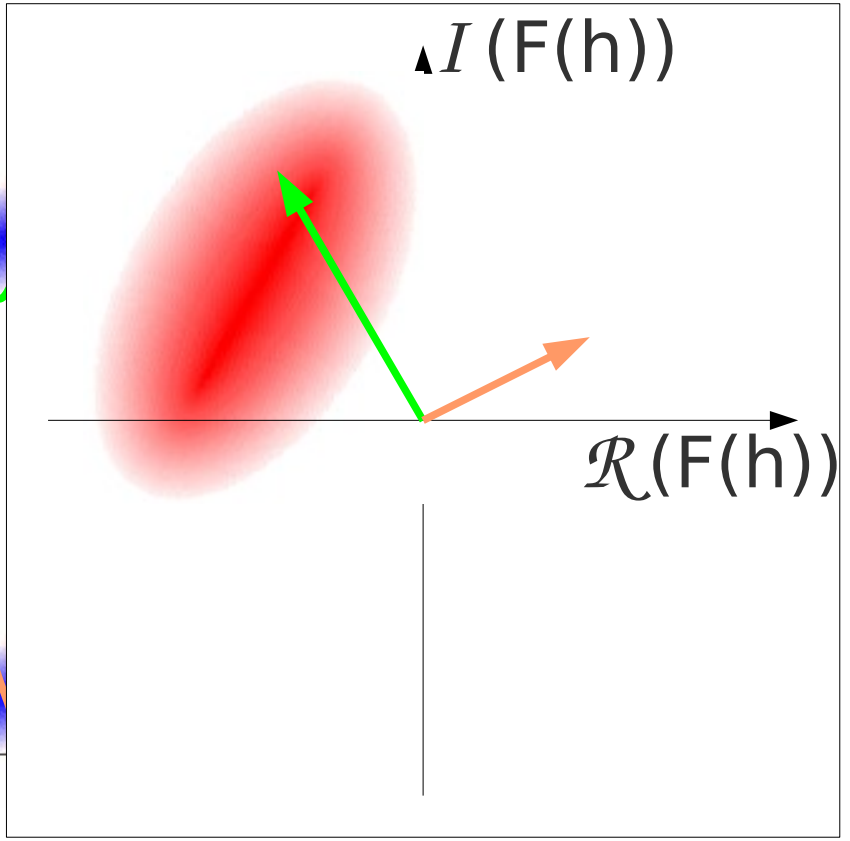
- Probability of a map is determined by how well it fits these distributions:



Statistical density modification

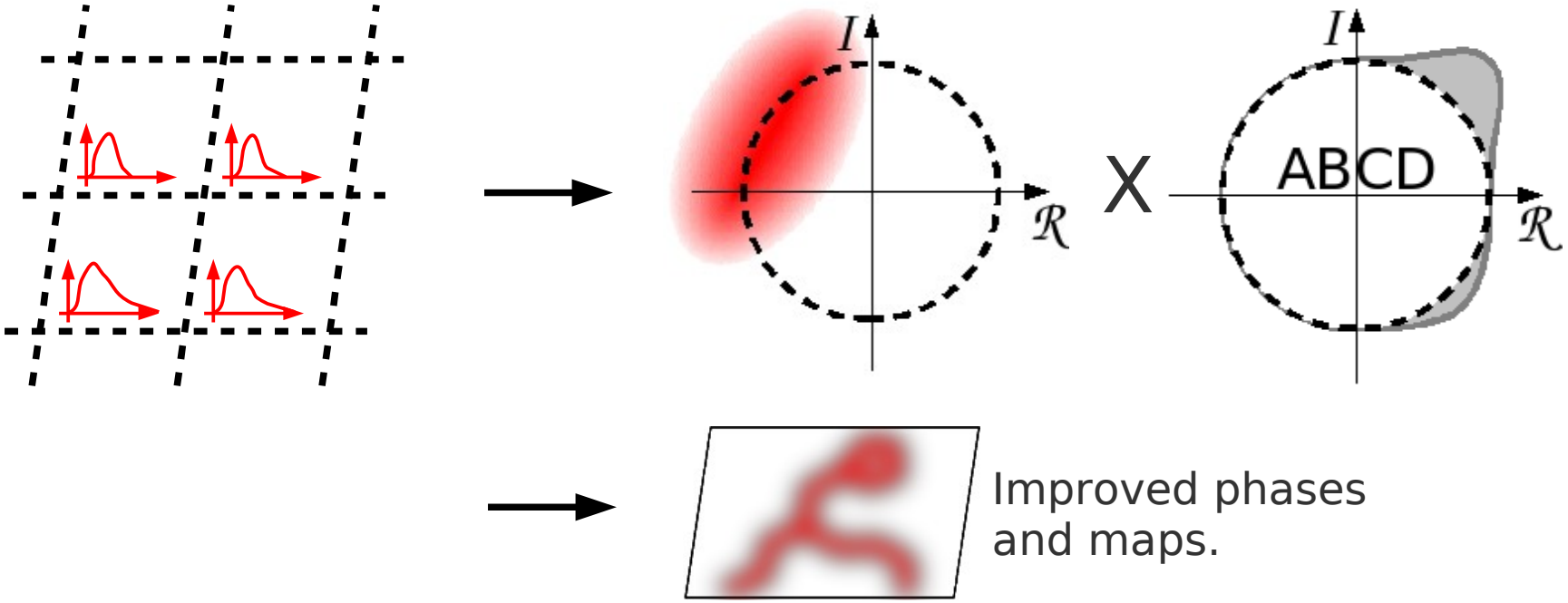
- Probability of each structure factor is given by the probability of the corresponding map.





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
Bricogne (1997) Methods in Enzymology

Statistical density modification

Advantages:

- Reduced bias.
- Better phases.

Disadvantages:

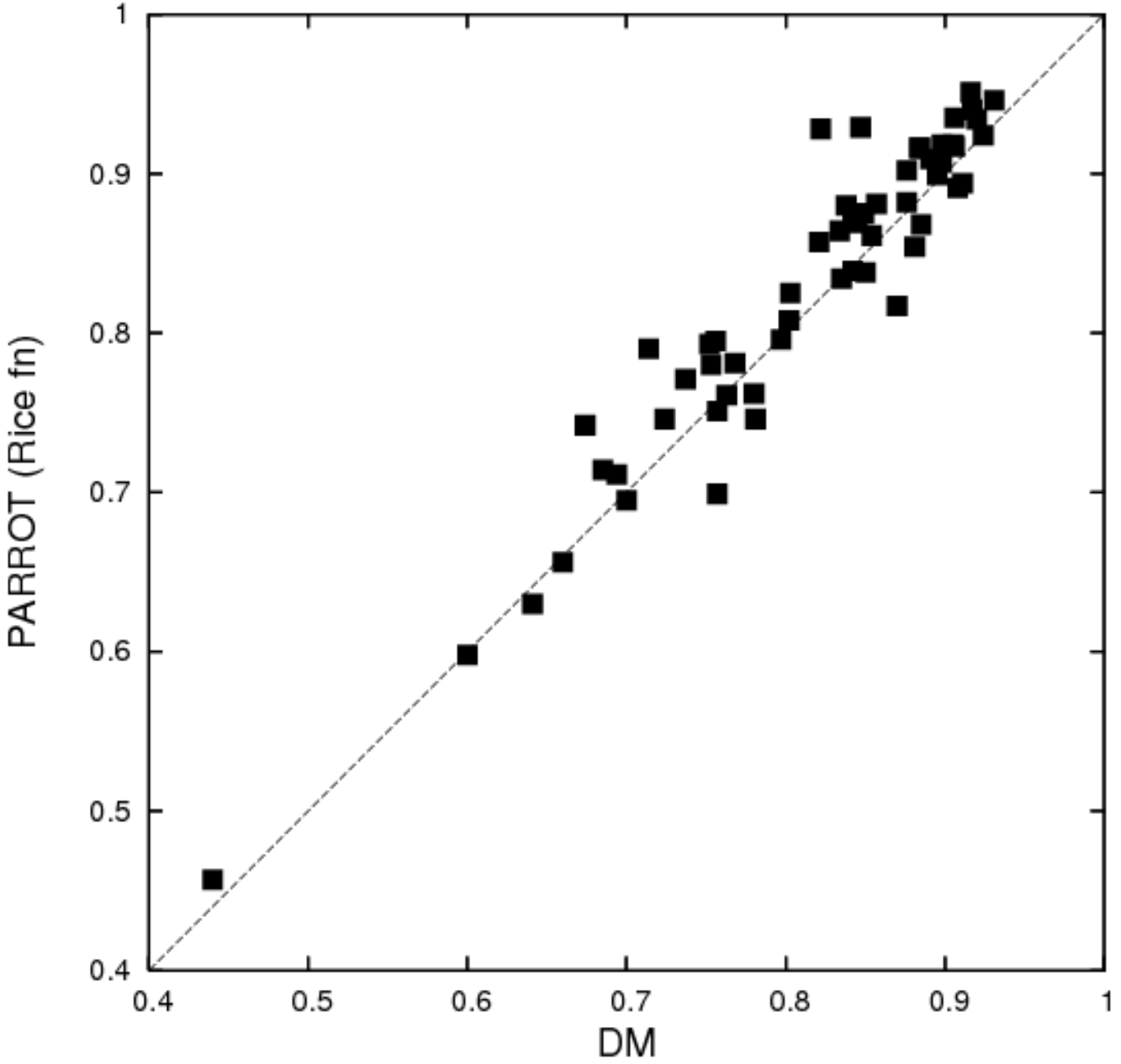
- Slow.
- PIRATE in particular works well for some cases and badly for others.

Density Modification

Kevin Cowtan, York.

Some results...

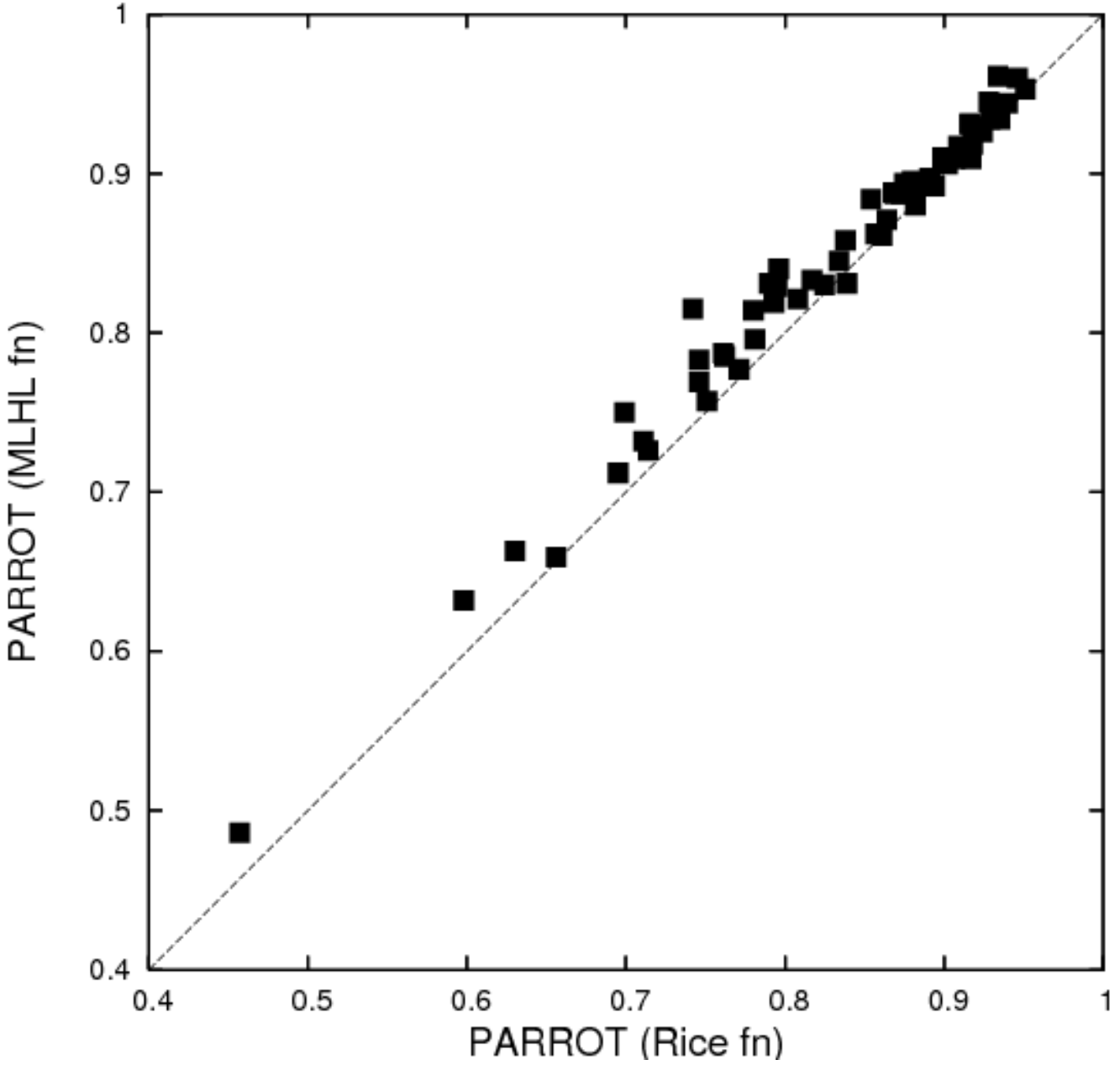
DM vs Parrot



Map correlations

Parrot: No new features enabled.

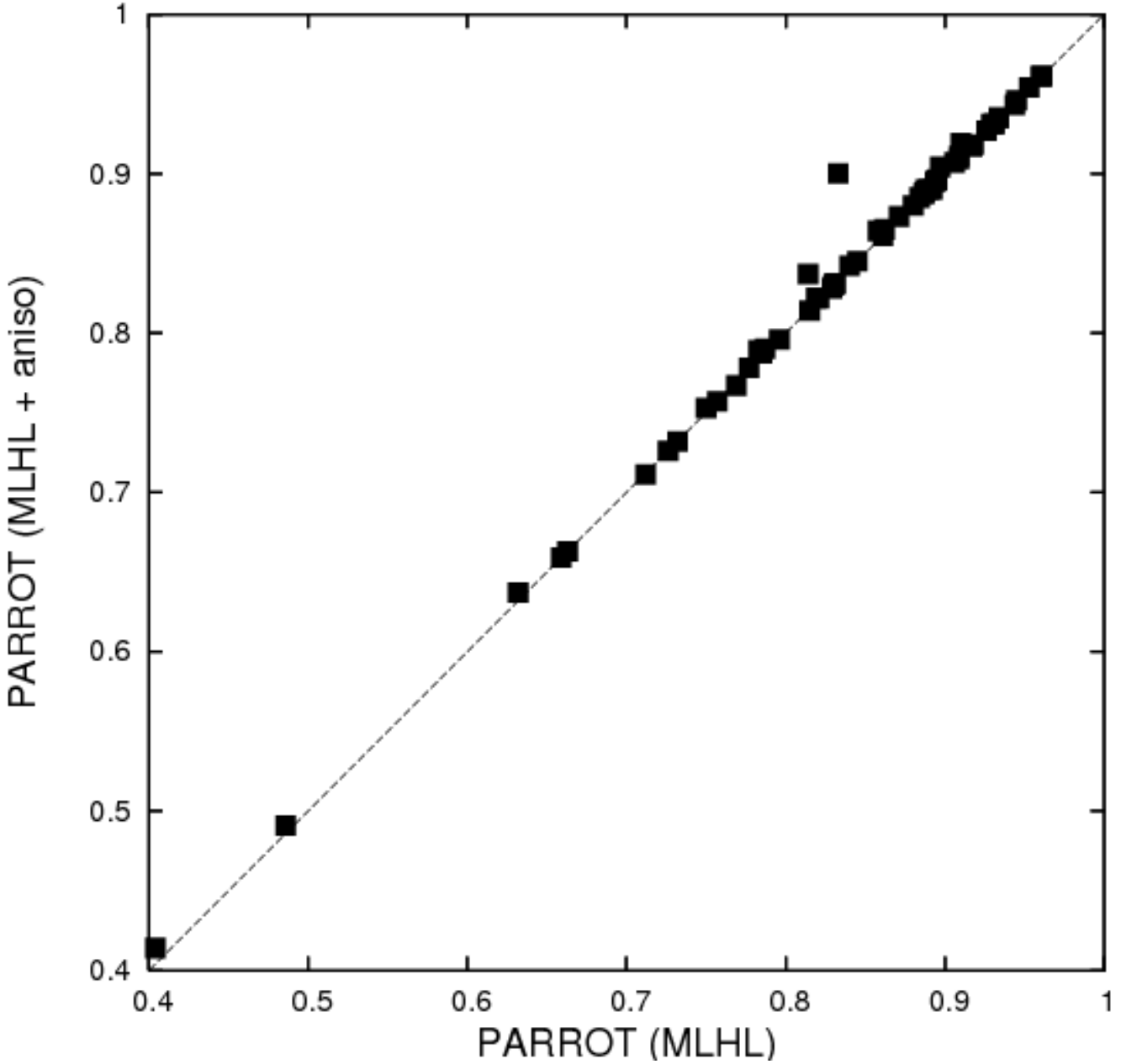
Parrot: Rice vs MLHL



Map correlations

Comparing old and new likelihood functions.

Parrot: Isotropic vs Anisotropic

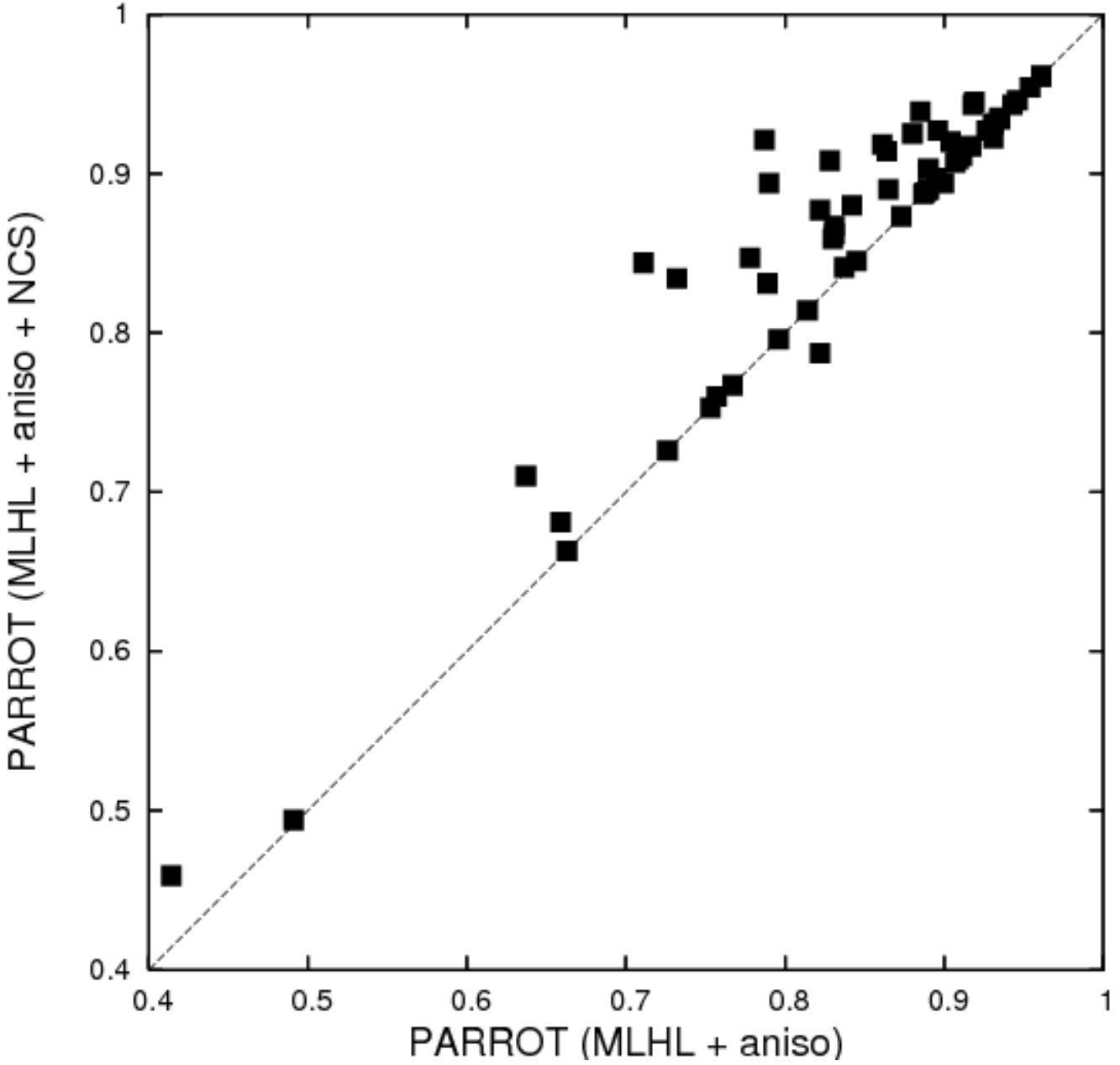


Map correlations

Comparing with and without anisotropy correction.

Kevin Cowtan, cowtan@ysbl.york.ac.uk

Parrot: simple vs NCS averaged



Map correlations

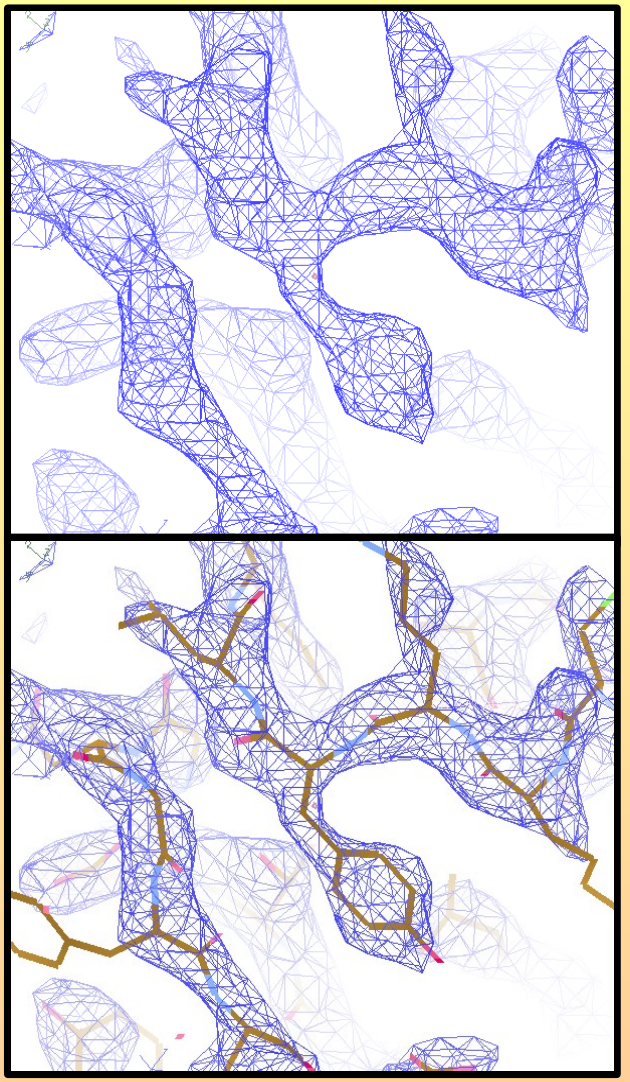
Comparing with and without NCS averaging.

Kevin Cowtan, cowtan@ysbl.york.ac.uk

Model Building

Model building software:

- Proteins:
  - Buccaneer
  - ARP/wARP
  - Phenix autobuild
- Nucleic acids:
  - Nautilus/Coot
  - ARP/wARP
  - Phenix autobuild



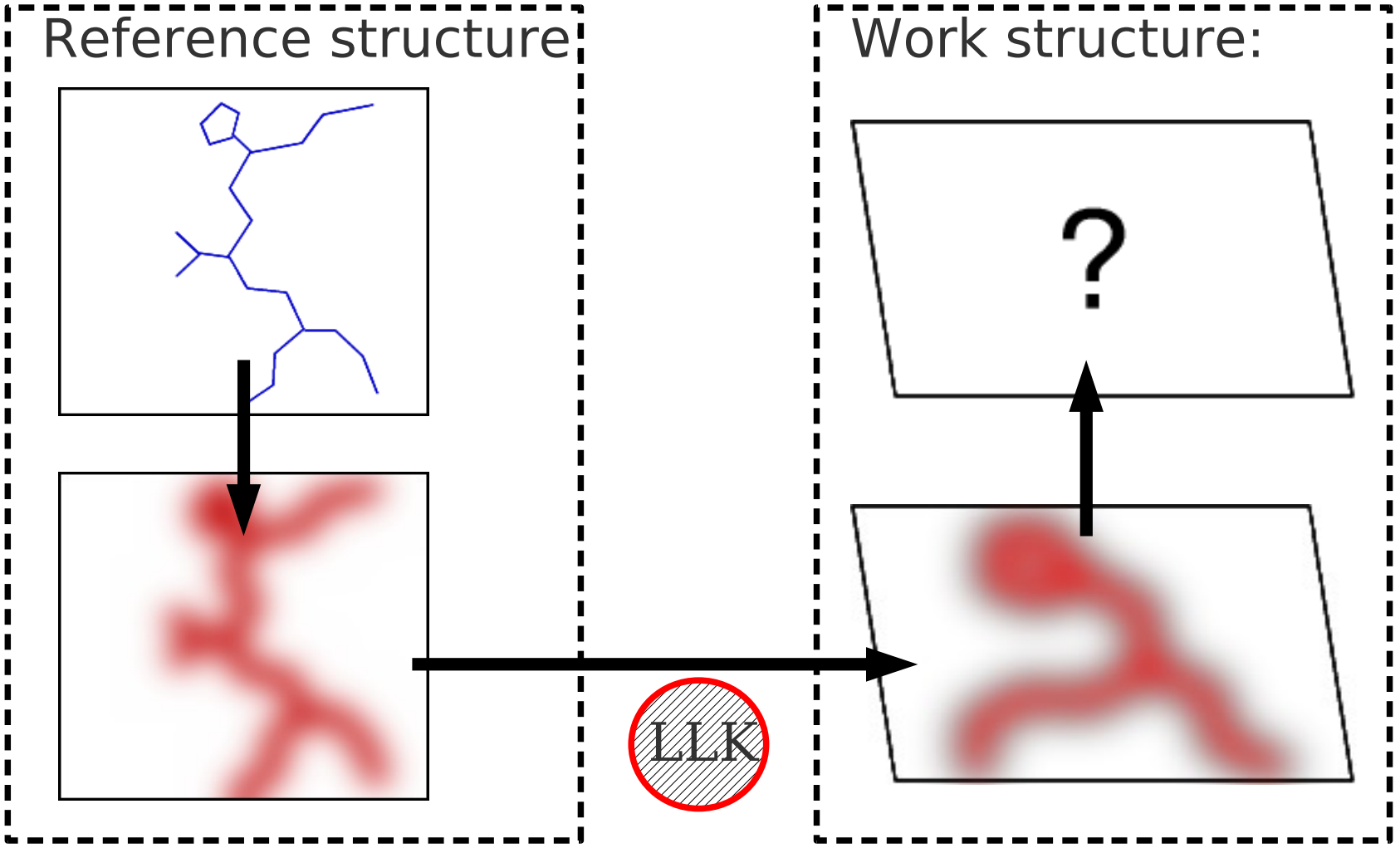
Buccaneer

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

Kevin Cowtan
YSBL, University of York
cowtan@ysbl.york.ac.uk

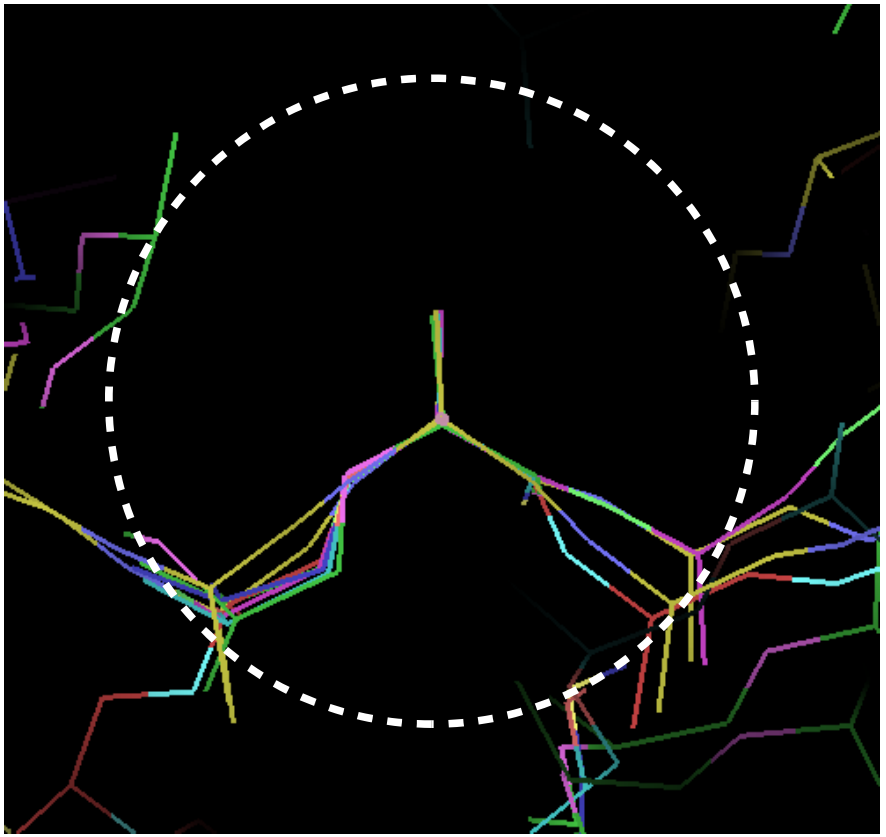
Buccaneer: Method

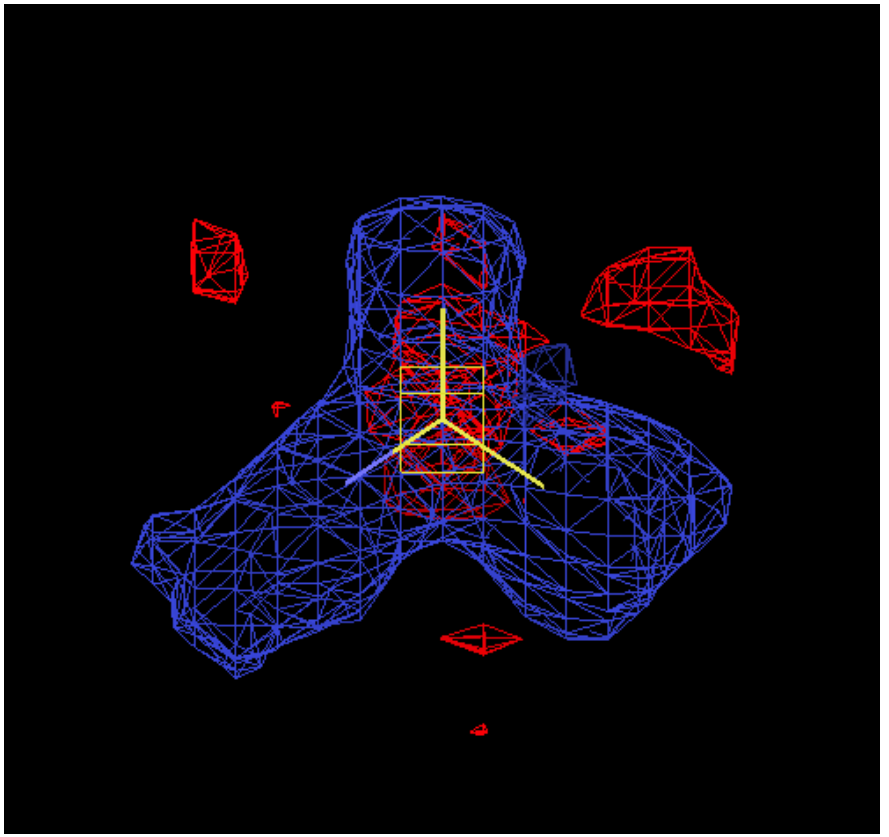
- Compare simulated map and known model to obtain likelihood target, then search for this target in the unknown map.



Buccaneer: Method

- Compile statistics for reference map in 4A sphere about Cα => LLK target.





- Use mean/variance.

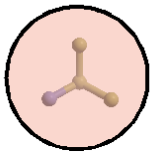
4A sphere about Ca also used by 'CAPRA' loeger et al. (but different target function).

Buccaneer

Use a likelihood function based on conserved density features.

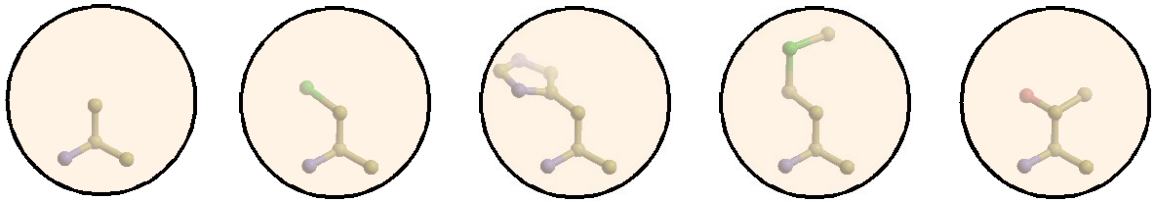
The same likelihood function is used several times. This makes the program very simple (<3000 lines), and the whole calculation works over a range of resolutions.

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

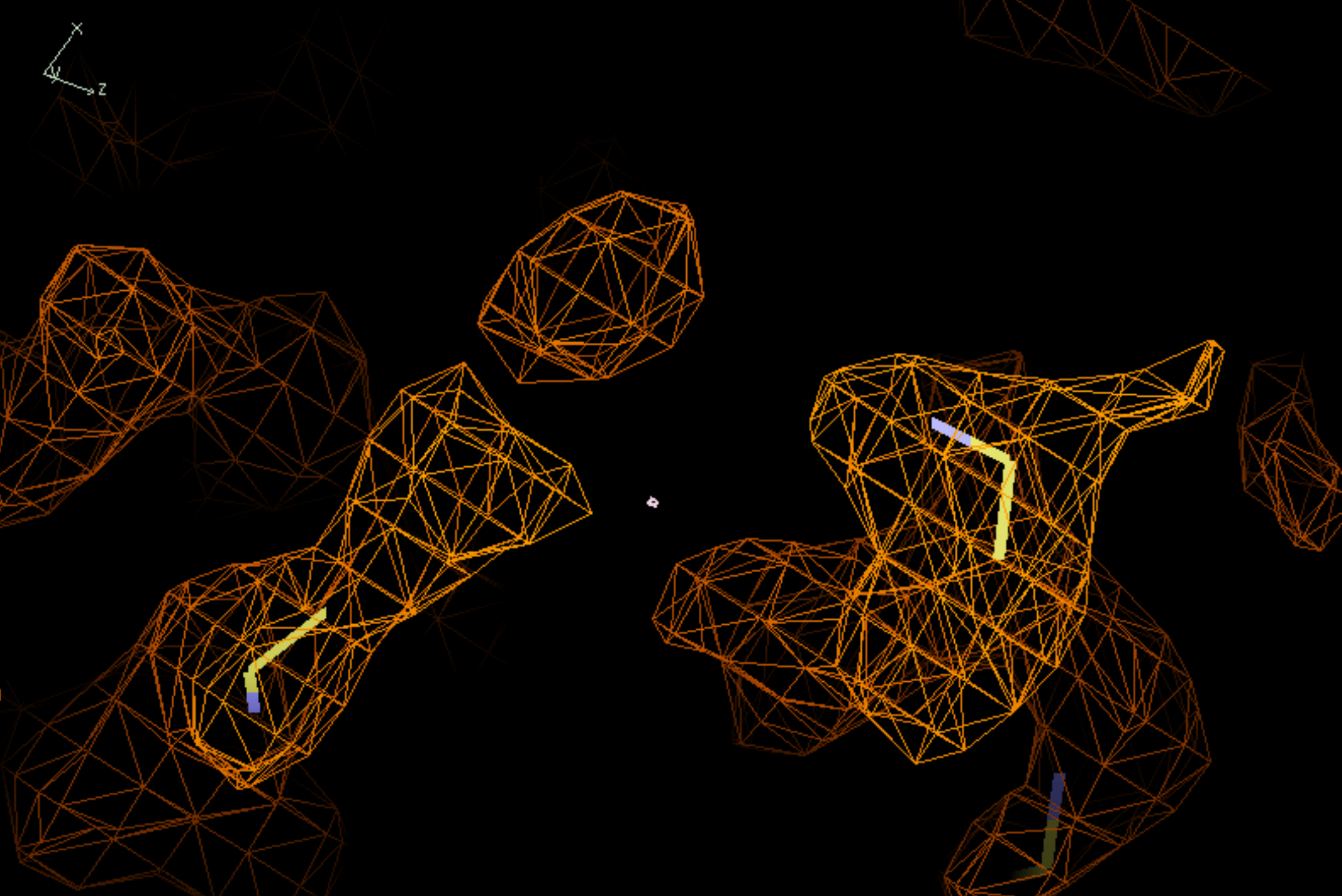
10 stages:

- **Find** candidate C-alpha positions
- **Grow** them into chain fragments
- **Join** and merge the fragments, resolving branches
- **Link** nearby N and C terminii (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

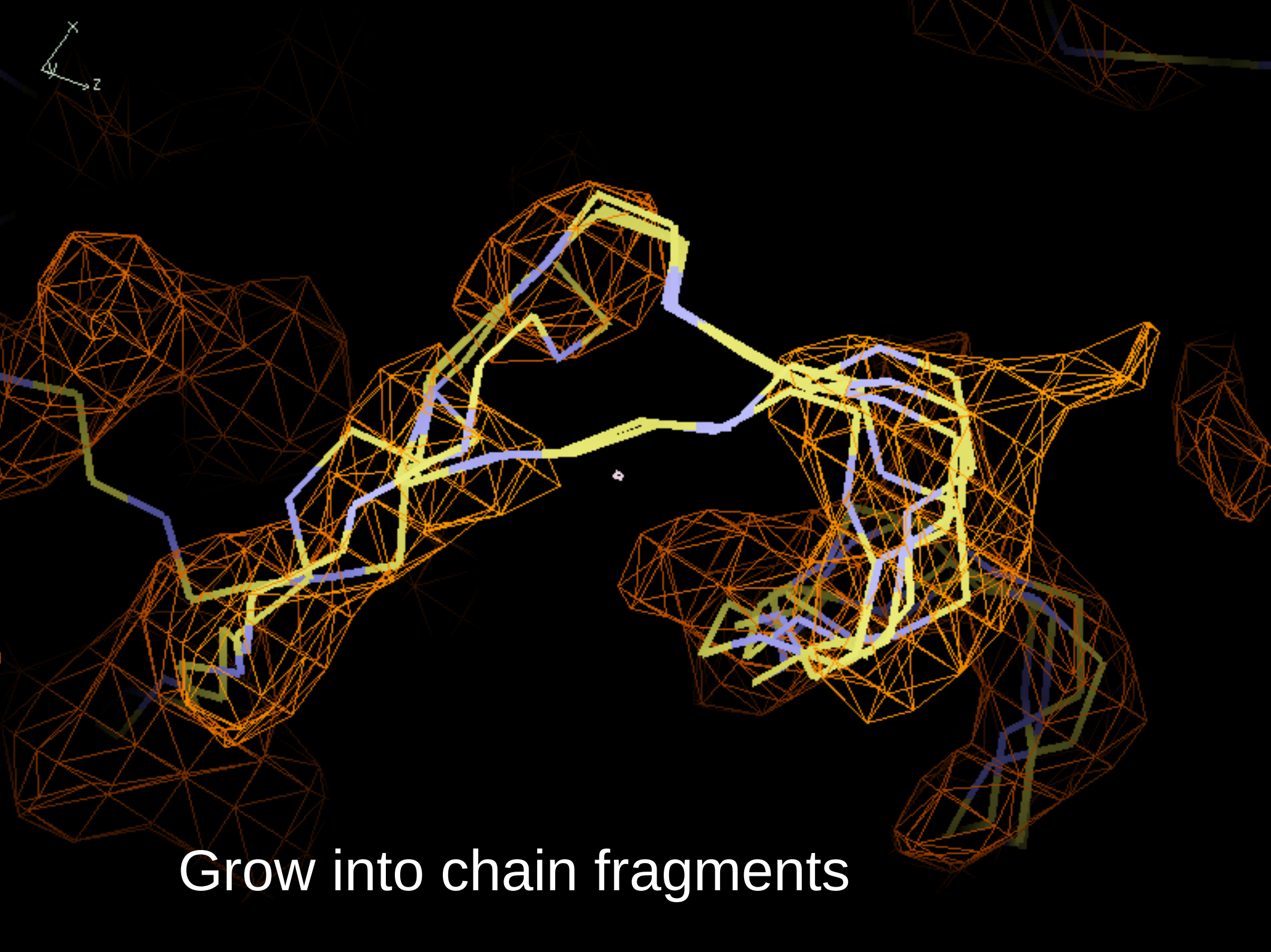
Buccaneer

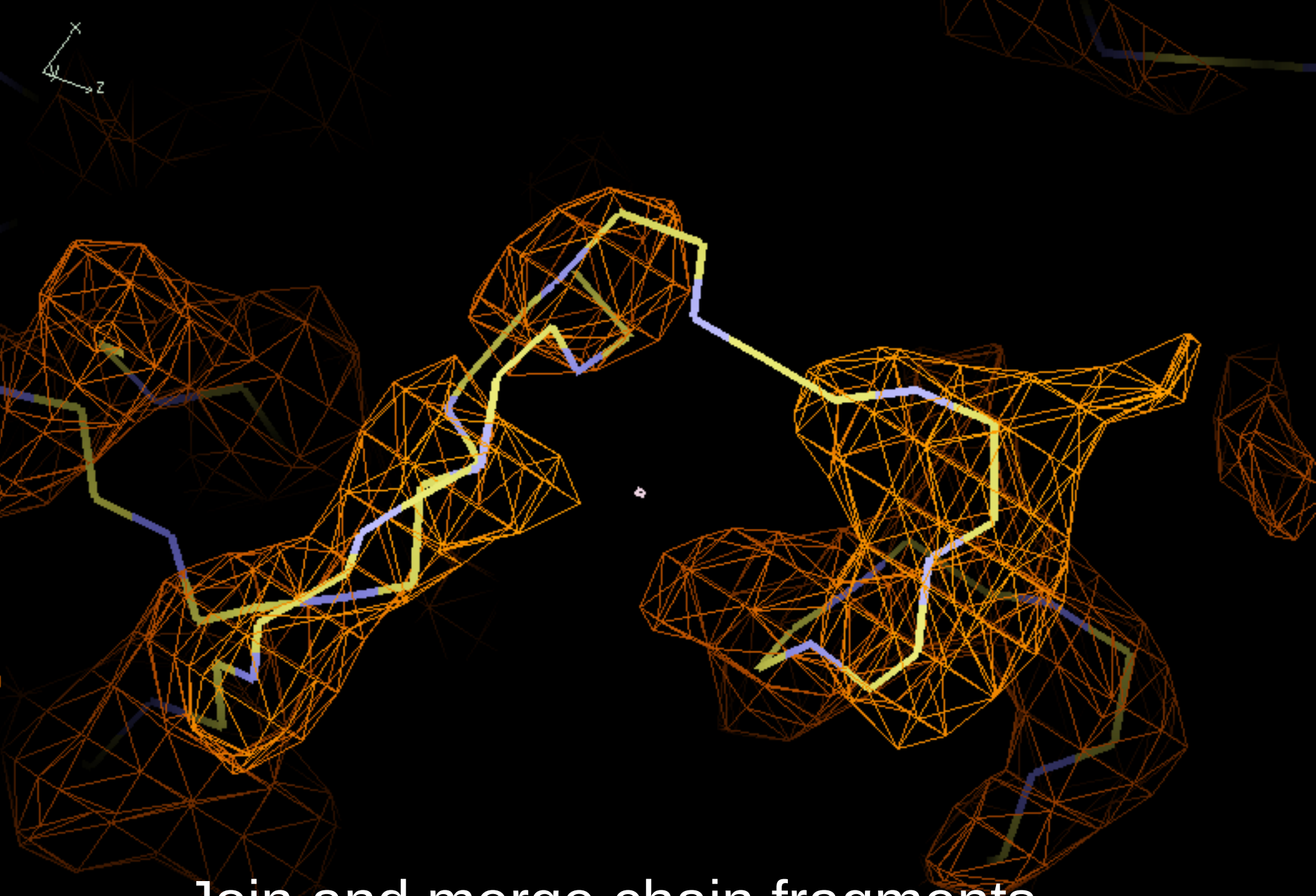
Case Study:

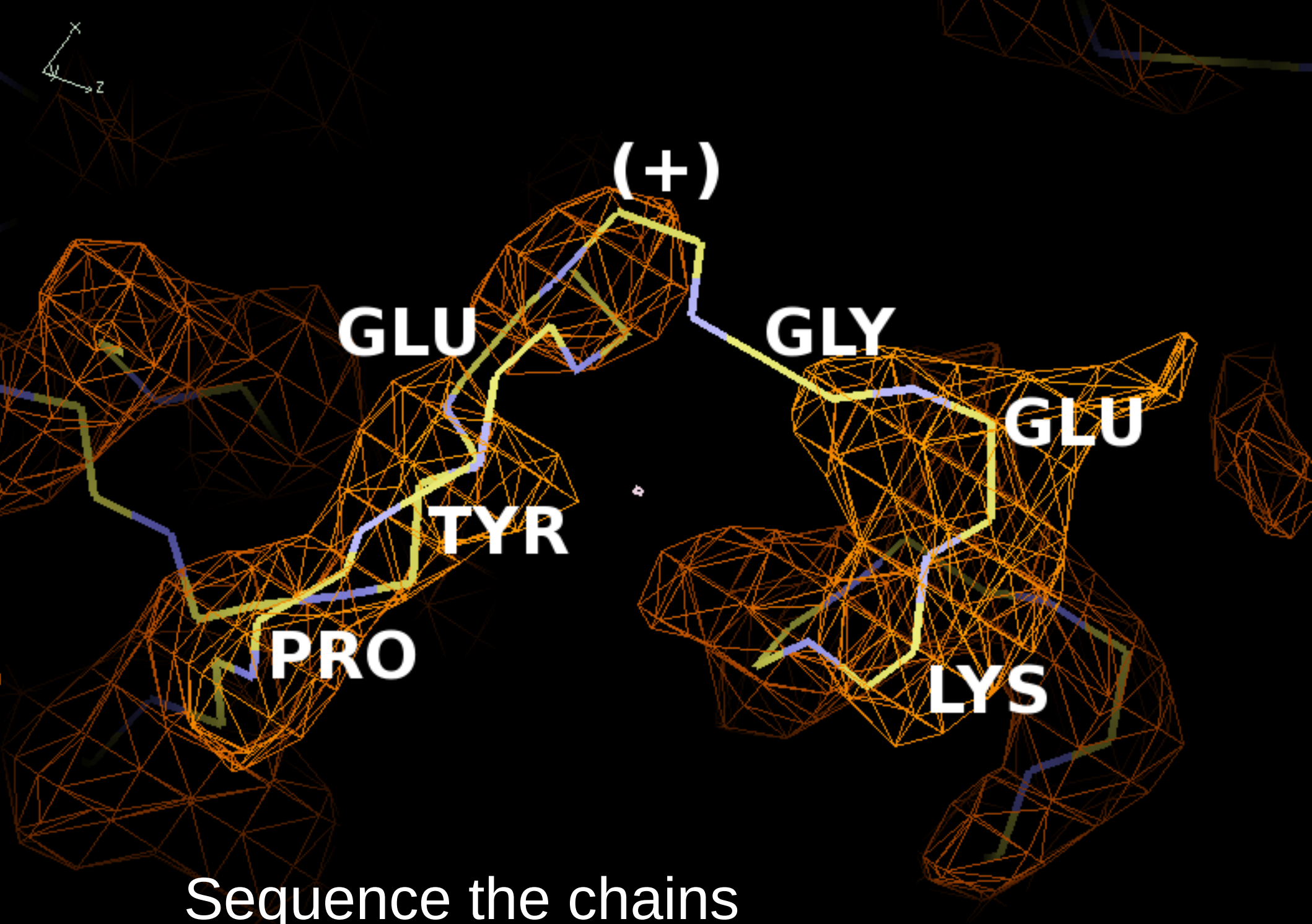
A difficult loop in a 2.9A map, calculated using real data from the JCSG.



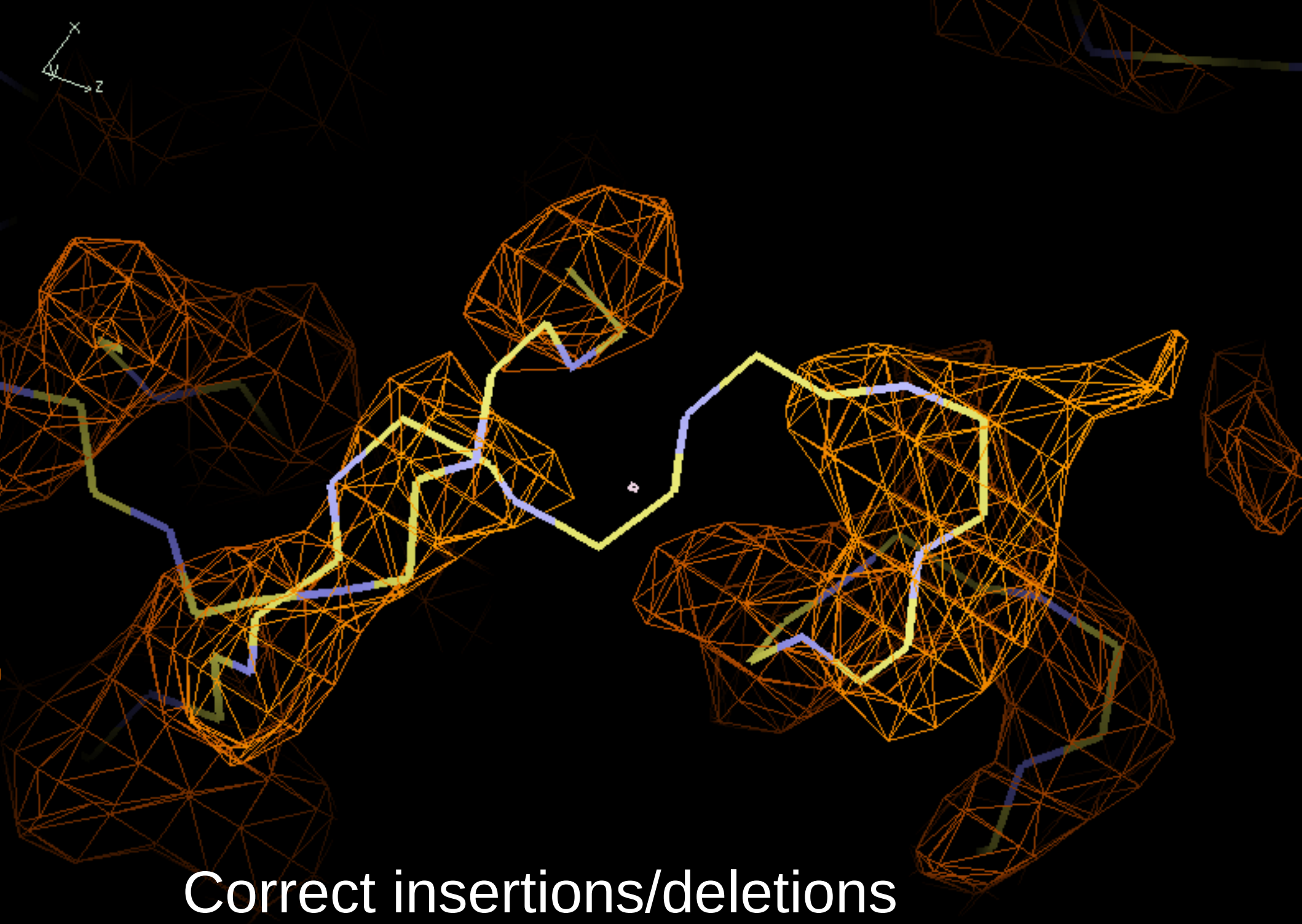
Find candidate C-alpha positions


X
Z
Grow into chain fragments

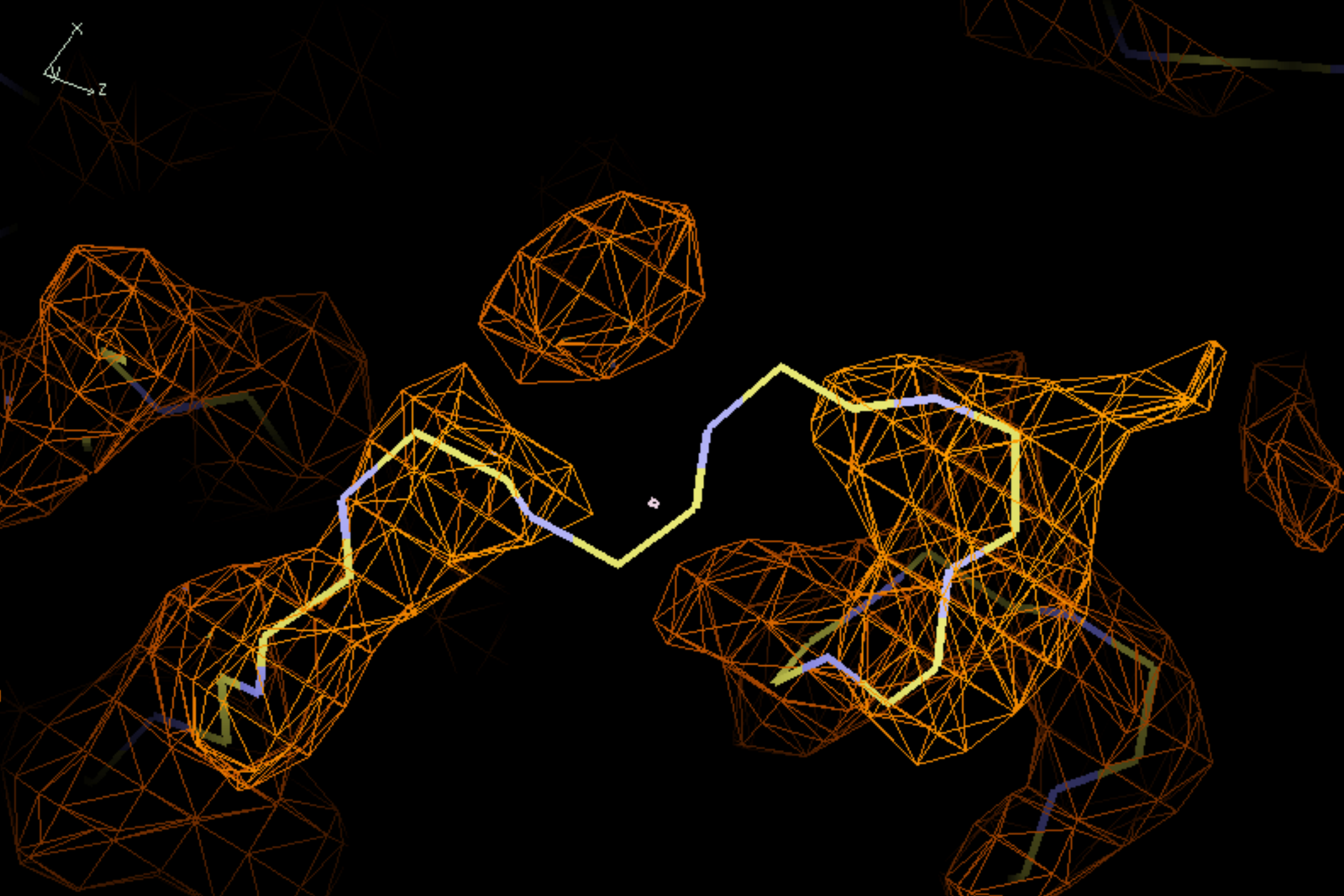

Join and merge chain fragments



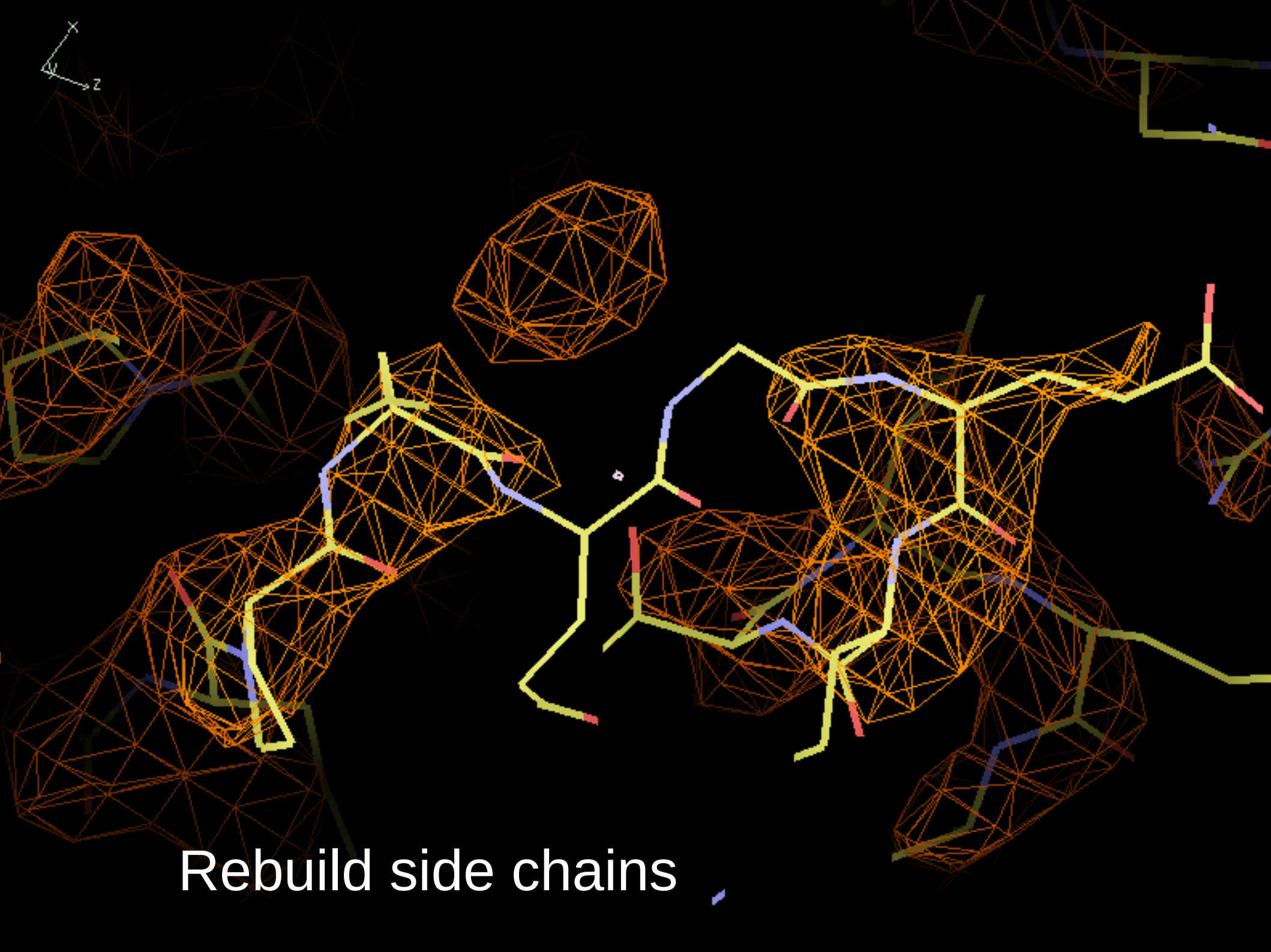
Sequence the chains

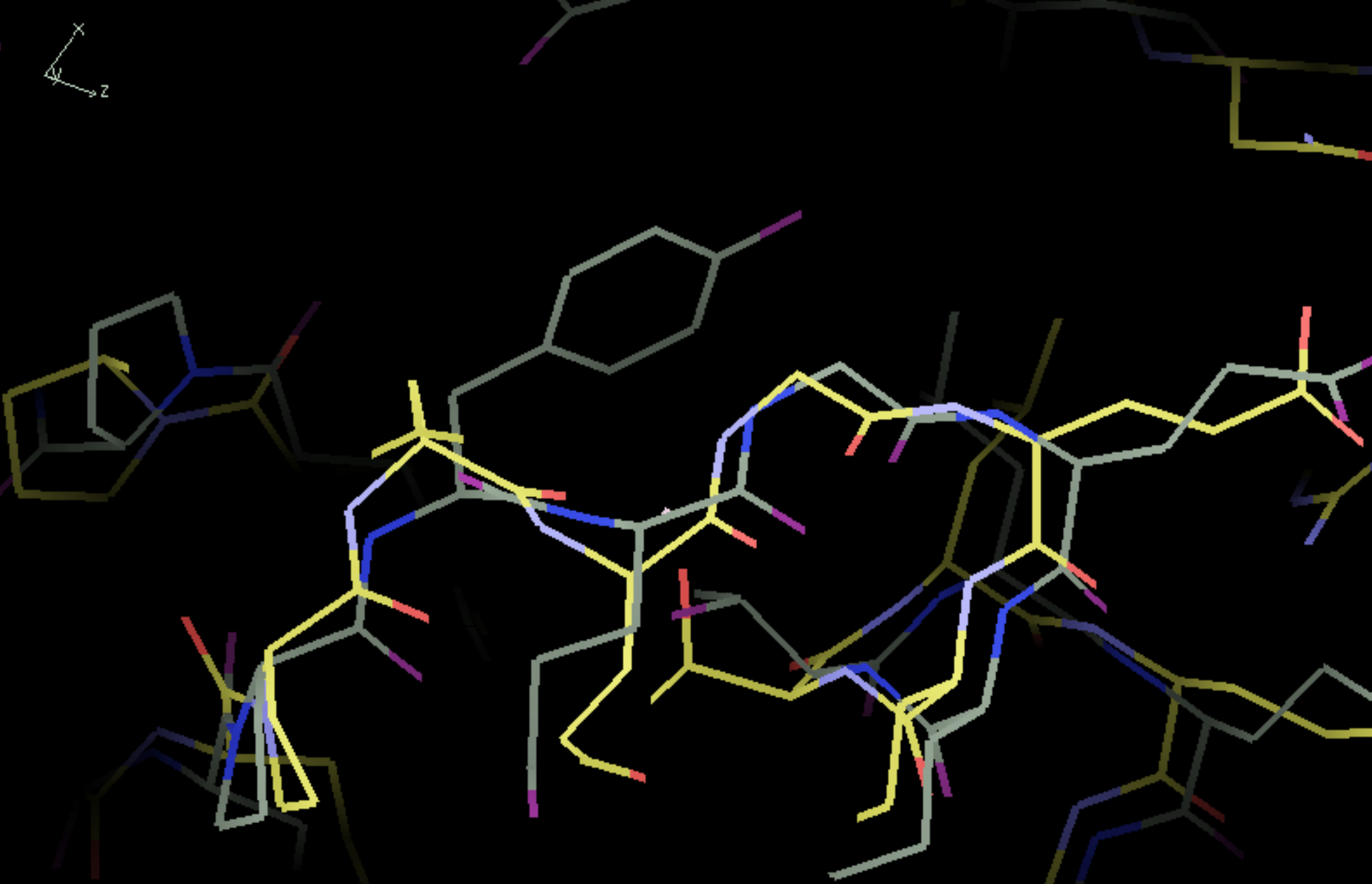


Correct insertions/deletions



Prune any remaining clashing chains


Rebuild side chains

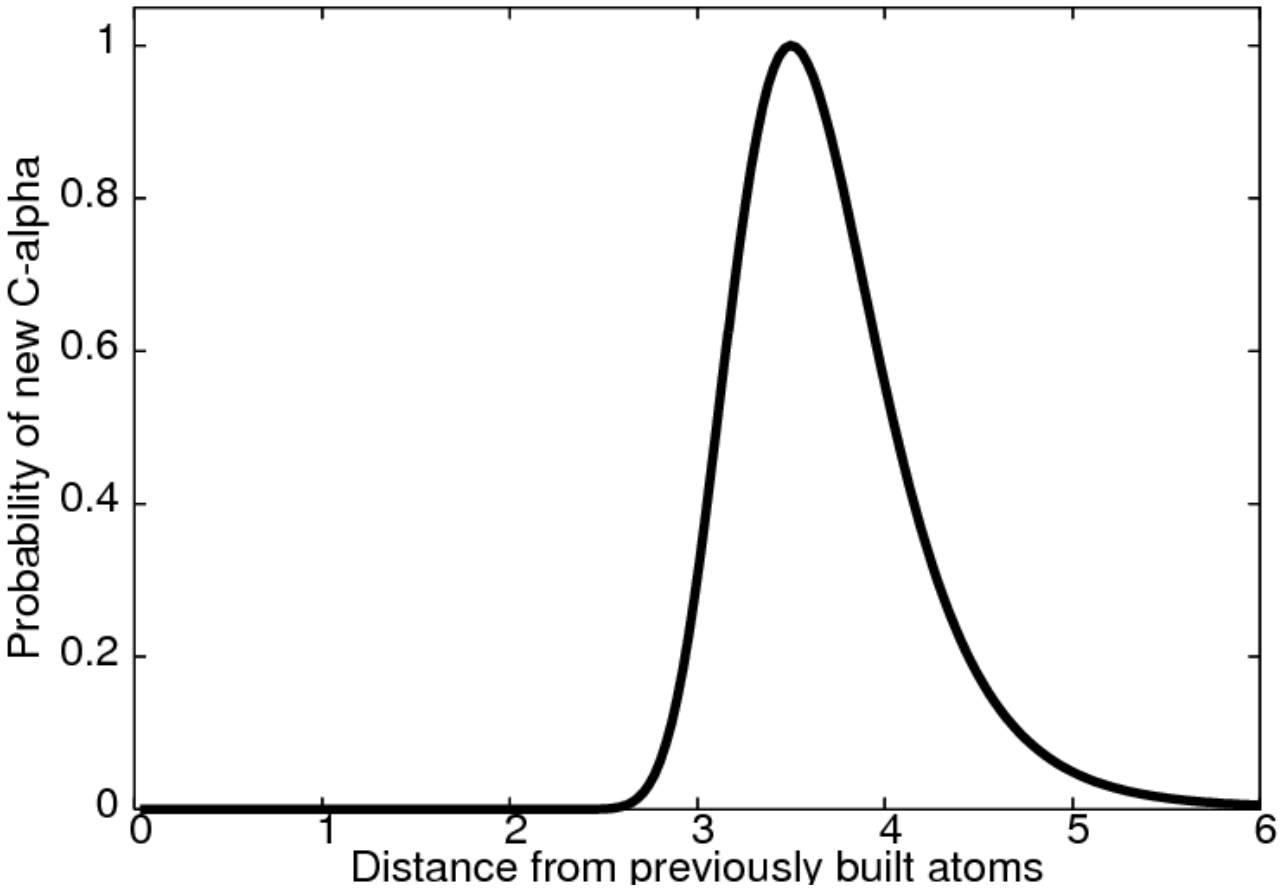


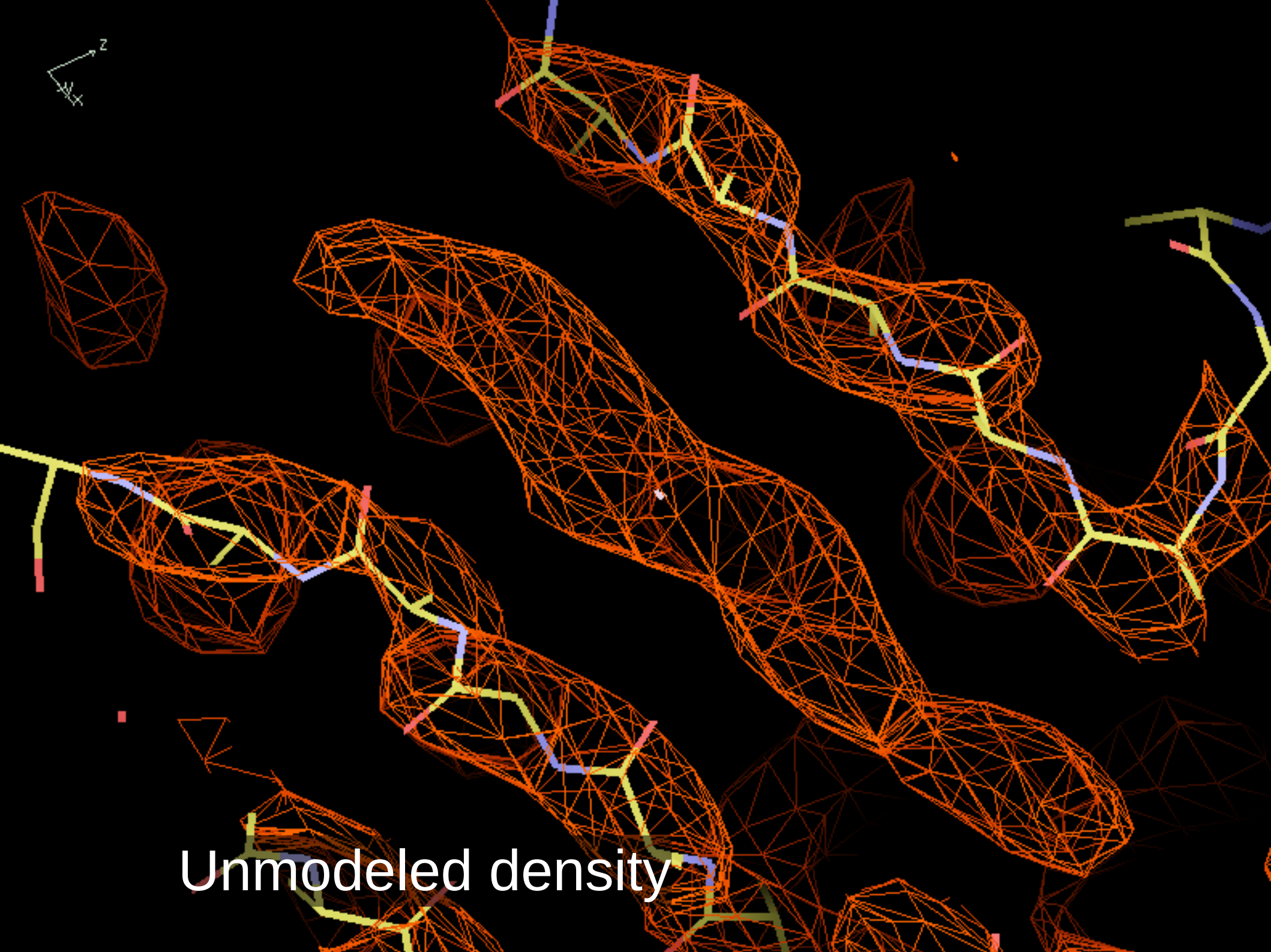
Comparison to the final model

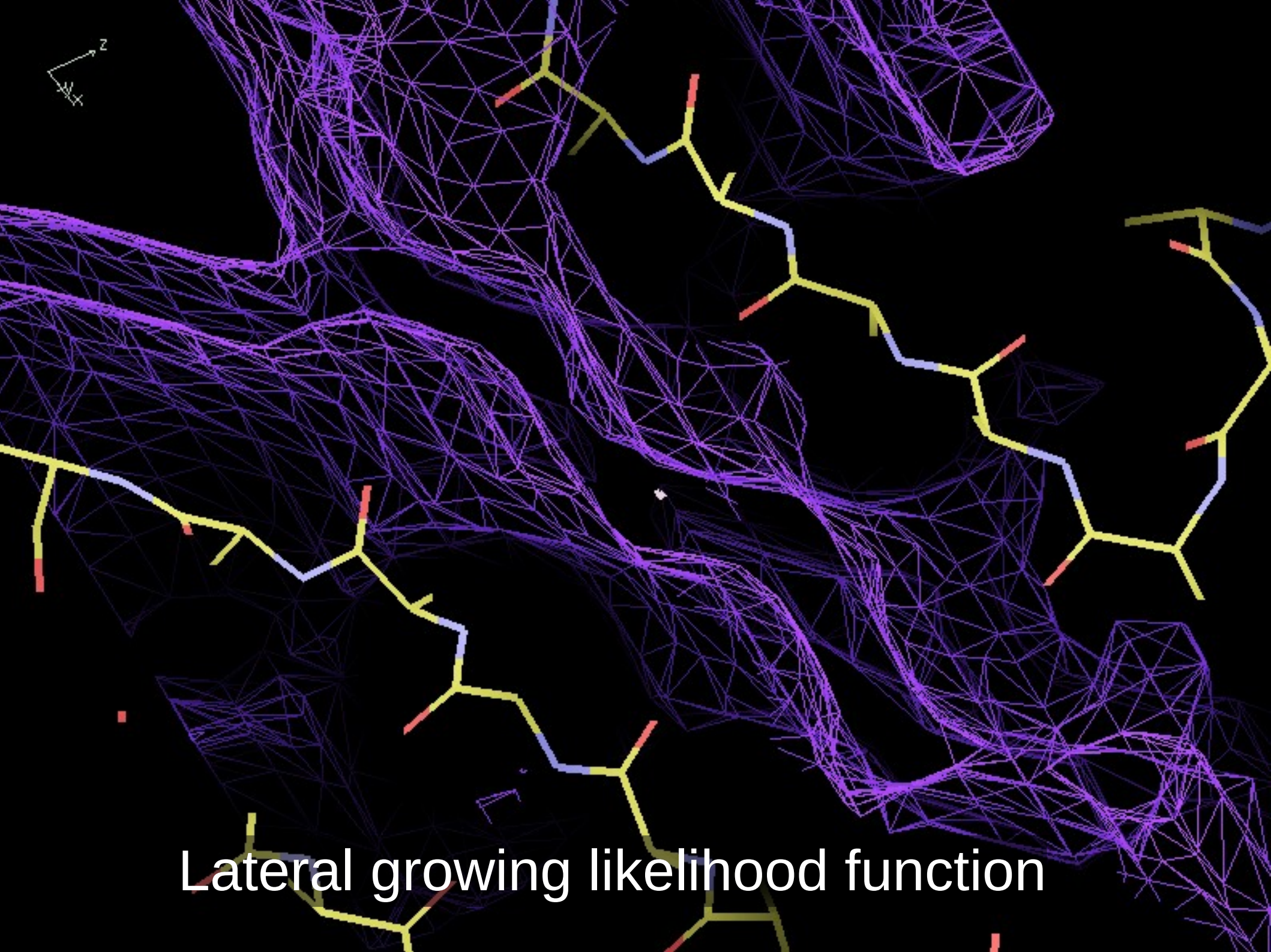
Buccaneer

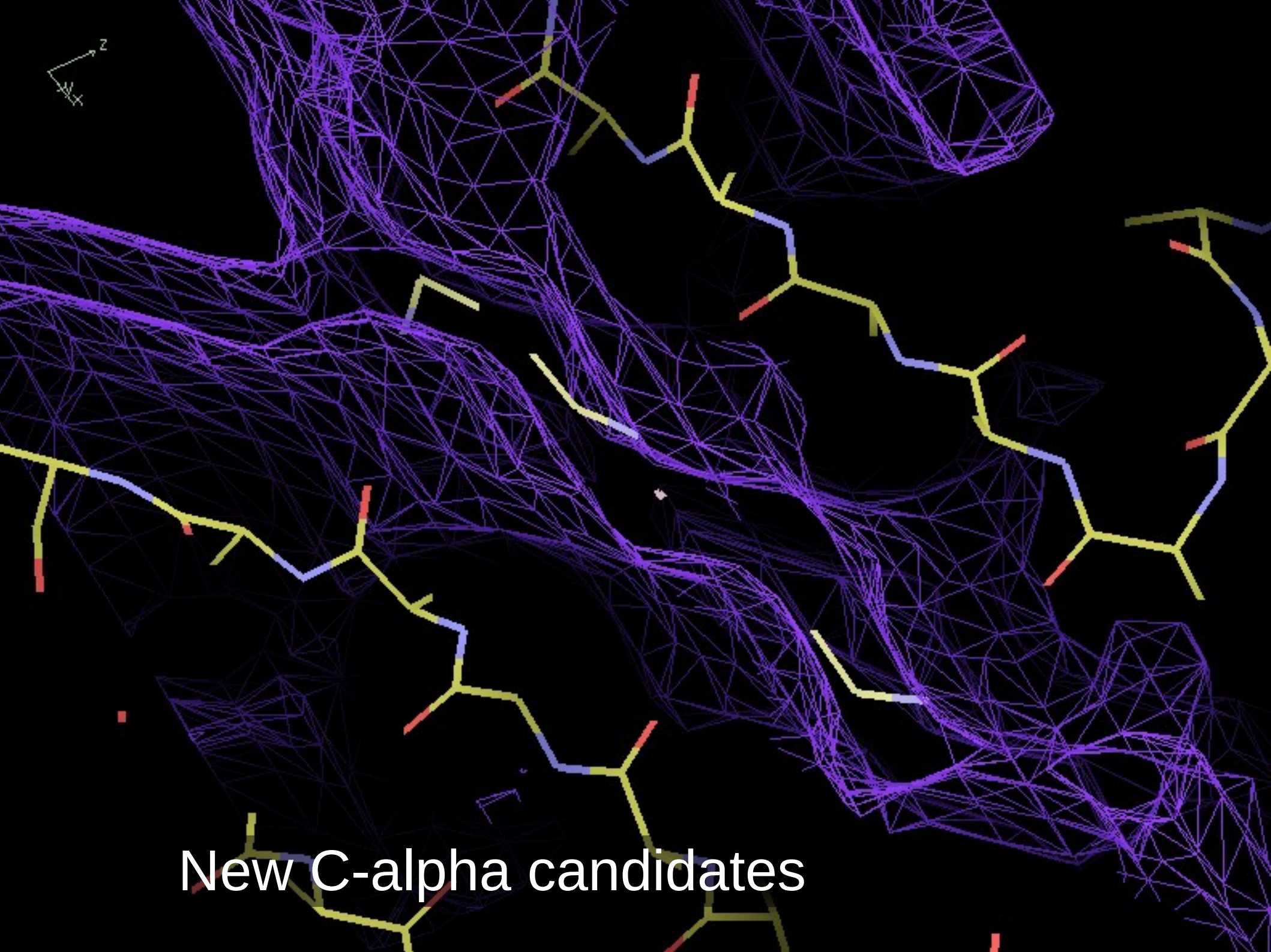
Model completion uses "**Lateral growing**":

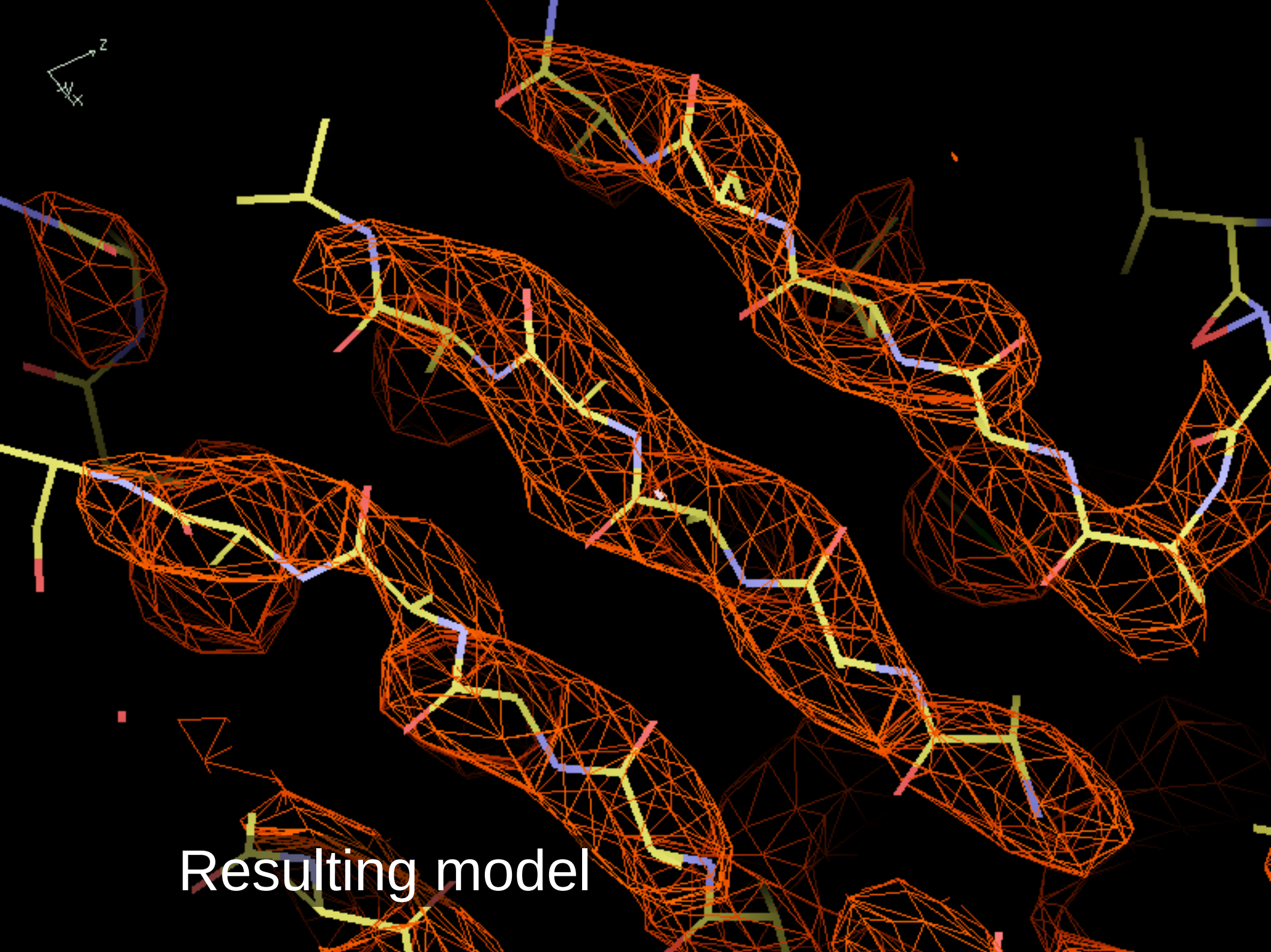
Grow sideways from existing chain fragments by looking for new C-alphas at an appropriate distance "sideways" from the existing chain:




Unmodeled density

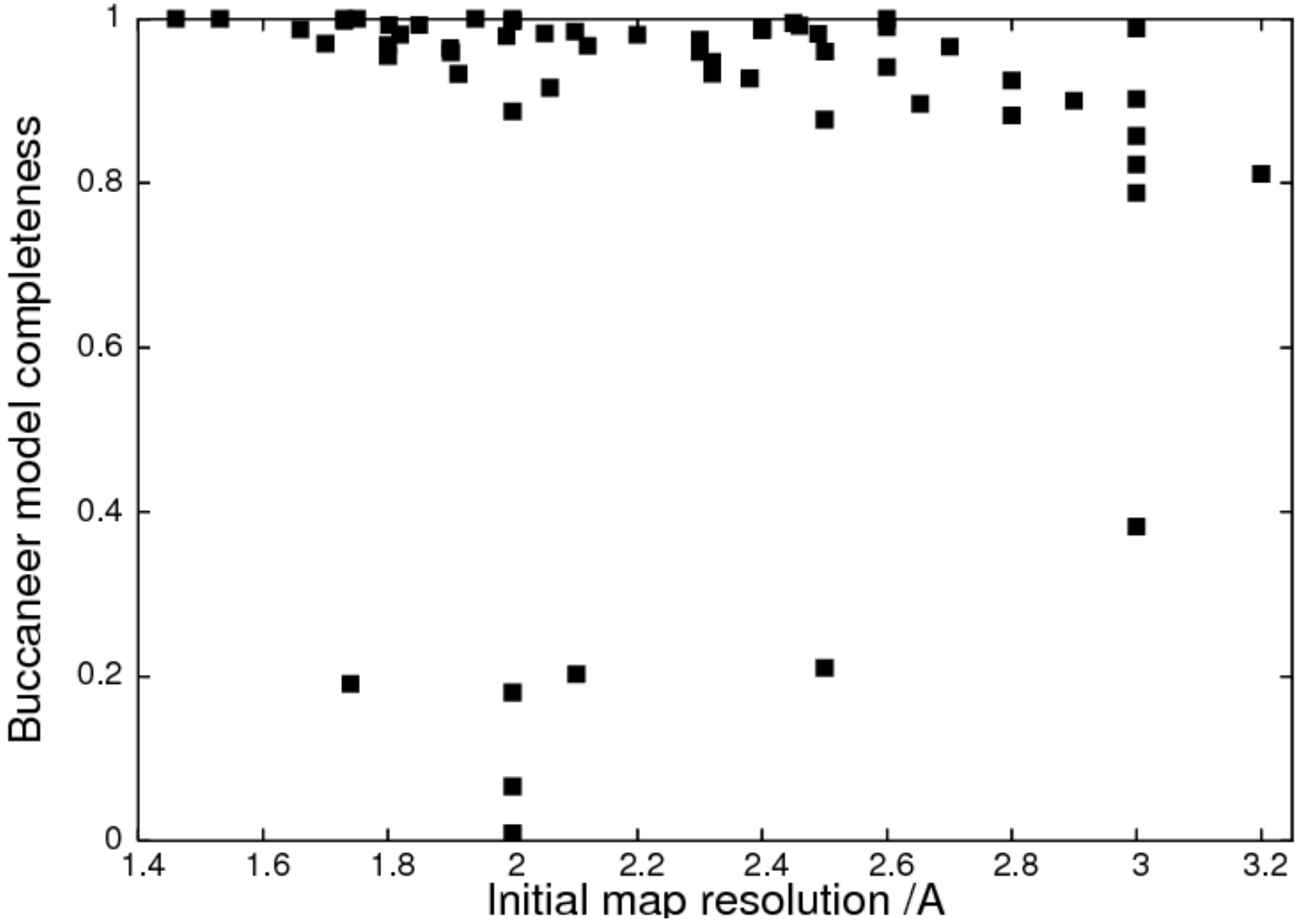

Lateral growing likelihood function


New C-alpha candidates


Z
X
Resulting model

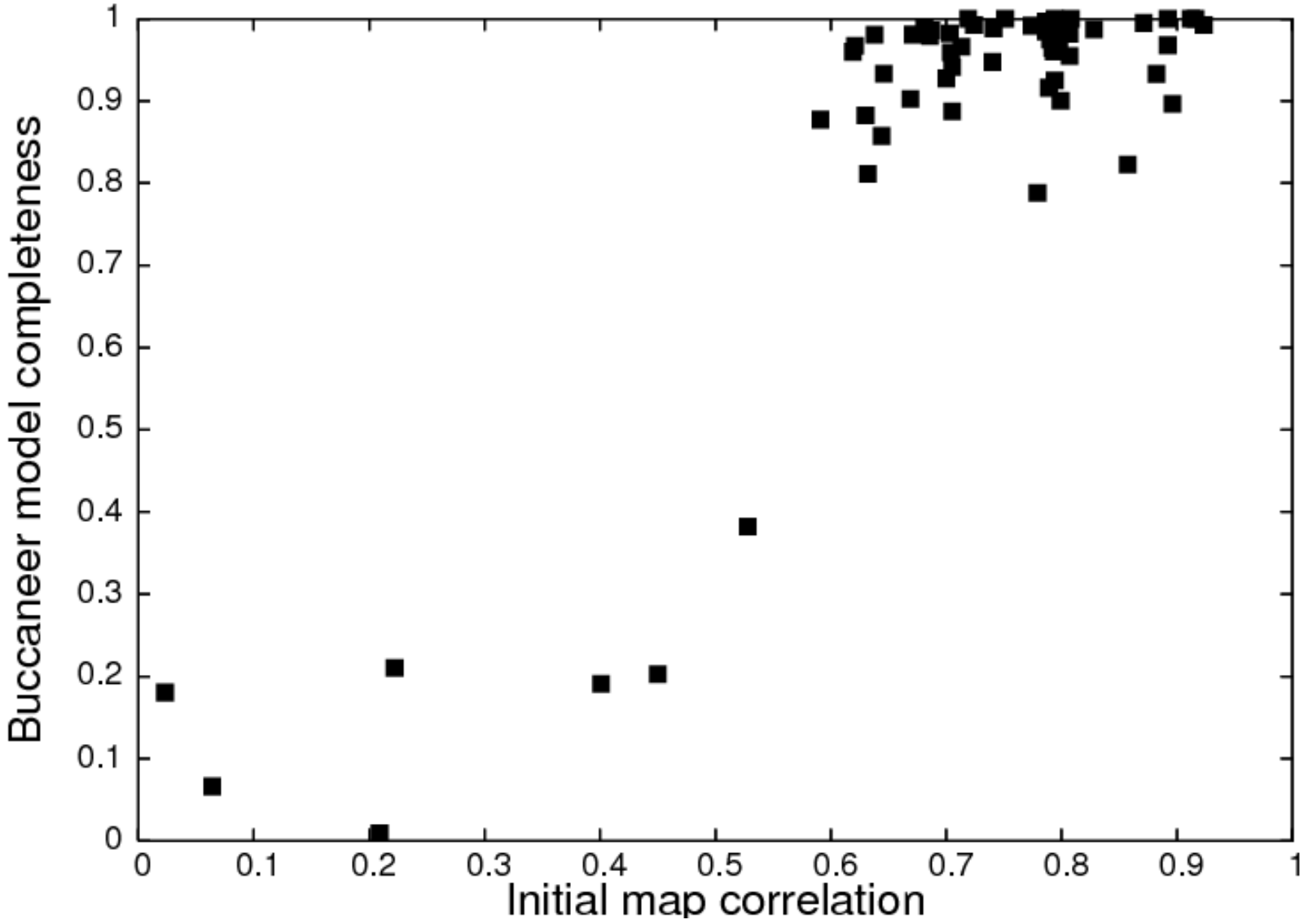
Buccaneer: Results

Model completeness not very dependent on resolution:

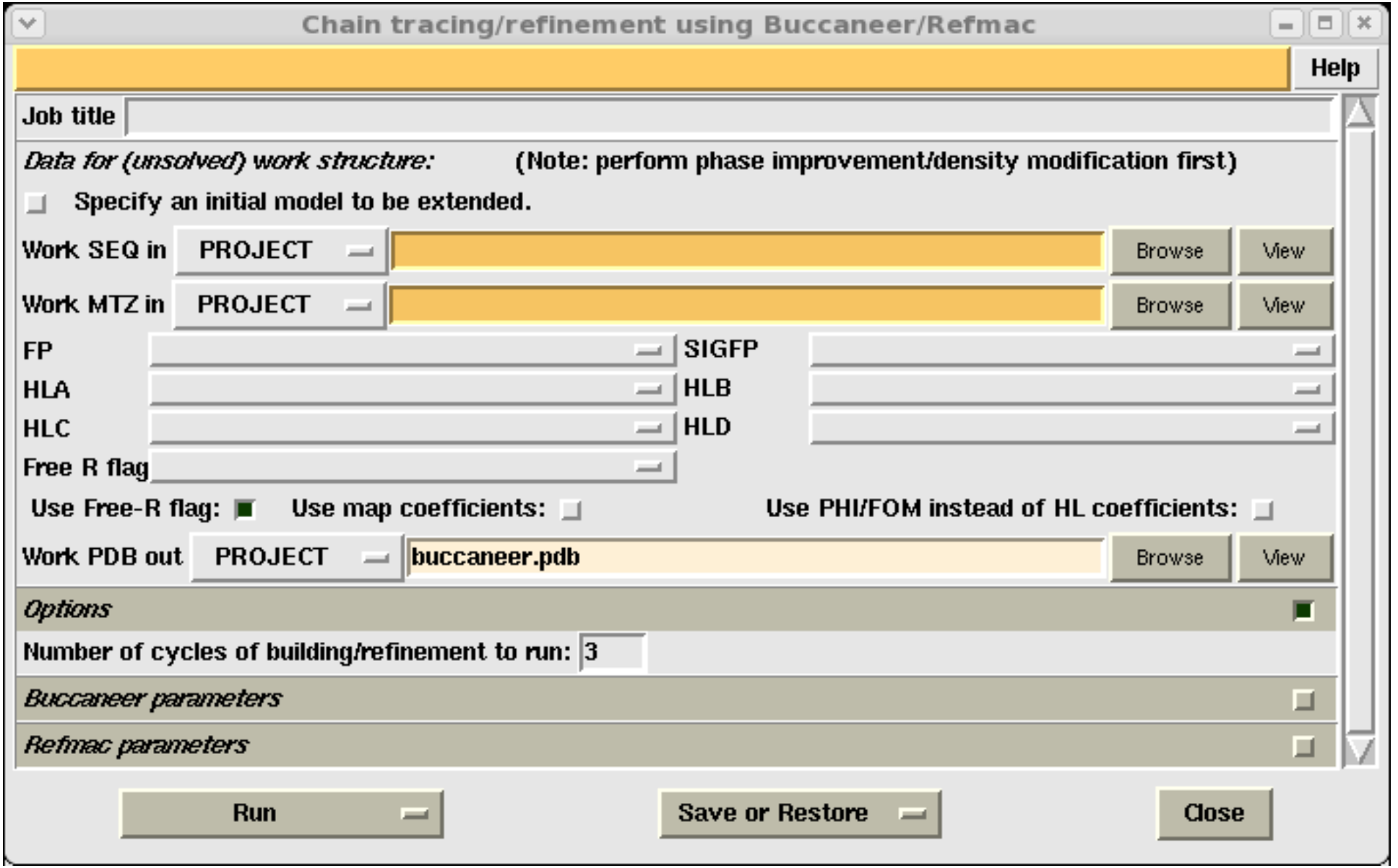


Buccaneer: Results

Model completeness dependent on initial phases:



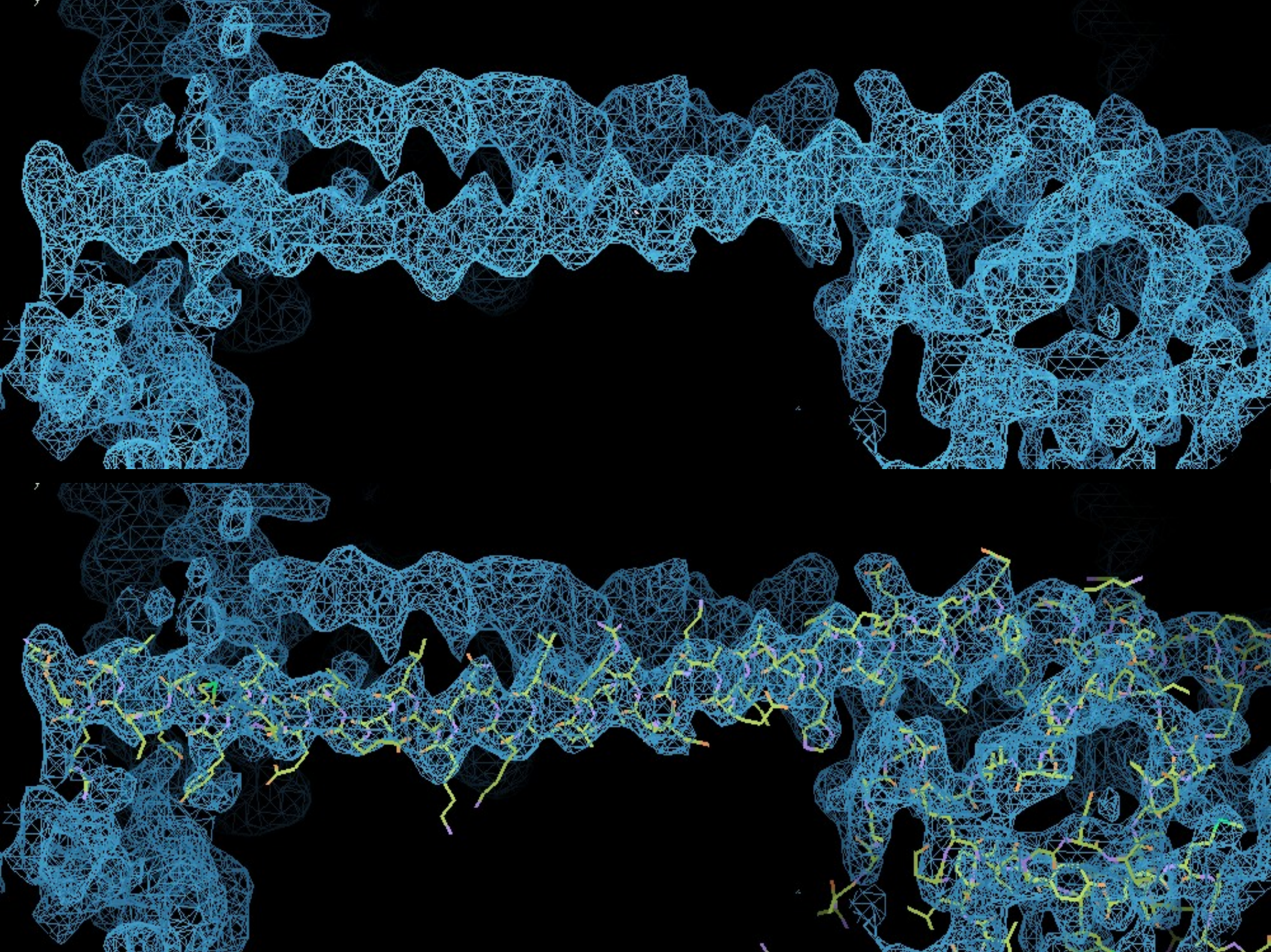
Buccaneer


Chain tracing/refinement using Buccaneer/Refmac
Help
Job title
Data for (unsolved) work structure:
(Note: perform phase improvement/density modification first)
Specify an initial model to be extended.
Work SEQ in
PROJECT
Browse
View
Work MTZ in
PROJECT
Browse
View
FP
SIGFP
HLA
HLB
HLC
HLD
Free R flag
Use Free-R flag:
Use map coefficients:
Use PHI/FOM instead of HL coefficients:
Work PDB out
PROJECT
buccaneer.pdb
Browse
View
Options
Number of cycles of building/refinement to run:
3
Buccaneer parameters
Refmac parameters
Run
Save or Restore
Close

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 un-modeled blobs..
  - Re-refine, examine difference maps.



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 3.7A 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!

Nucleic Acid Building

Nautilus:

- A new tool for nucleic acid model building
- Automated (CCP4i) or interactive (Coot)
- Starting from:
  - Experimental phasing
  - Molecular replacement
  - Protein complexes

Nautilus

The task:

- To build continuous nucleic acid chains into electron density.
- To assign sequence to those chains.
- To allow addition of nucleotide chains to non-nucleotide structures.

Nautilus

'Fingerprint' detection:

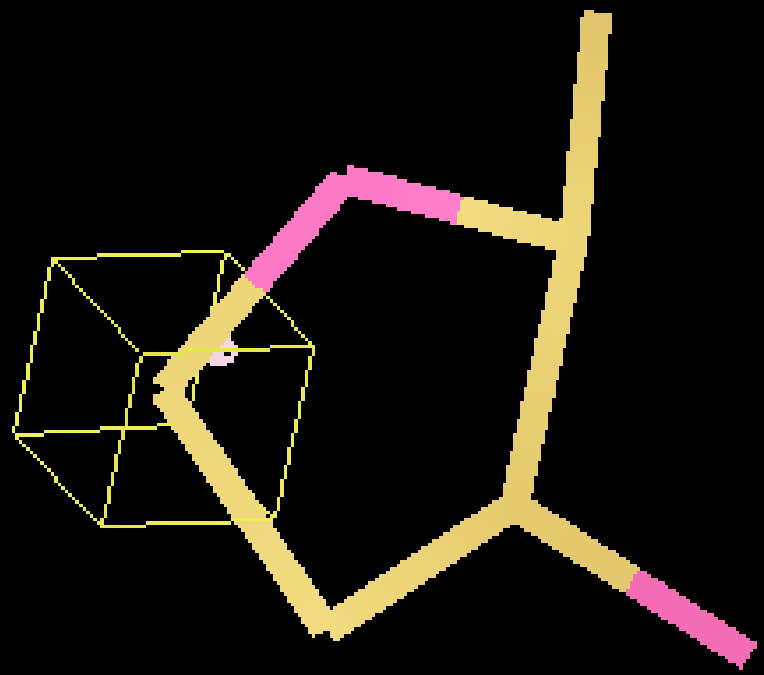
- Identify high and low density features consistent with the presence of nucleic acid features.
- Very fast.
- Related to 'Essens' (Kleywegt and Jones), but with looks at both ridges and troughs.

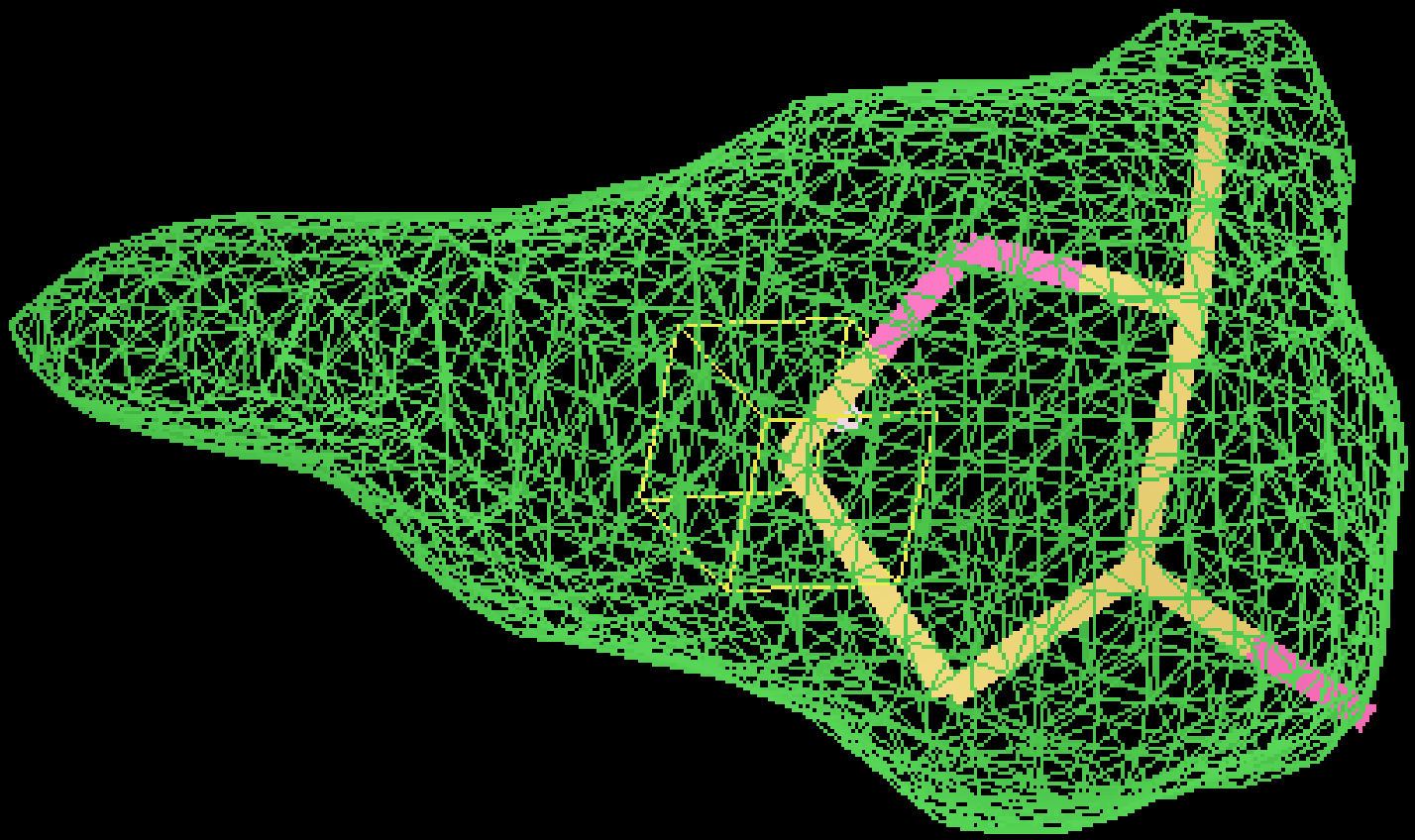
http://www.youtube.com/watch?v=QGN6tF-zKOE

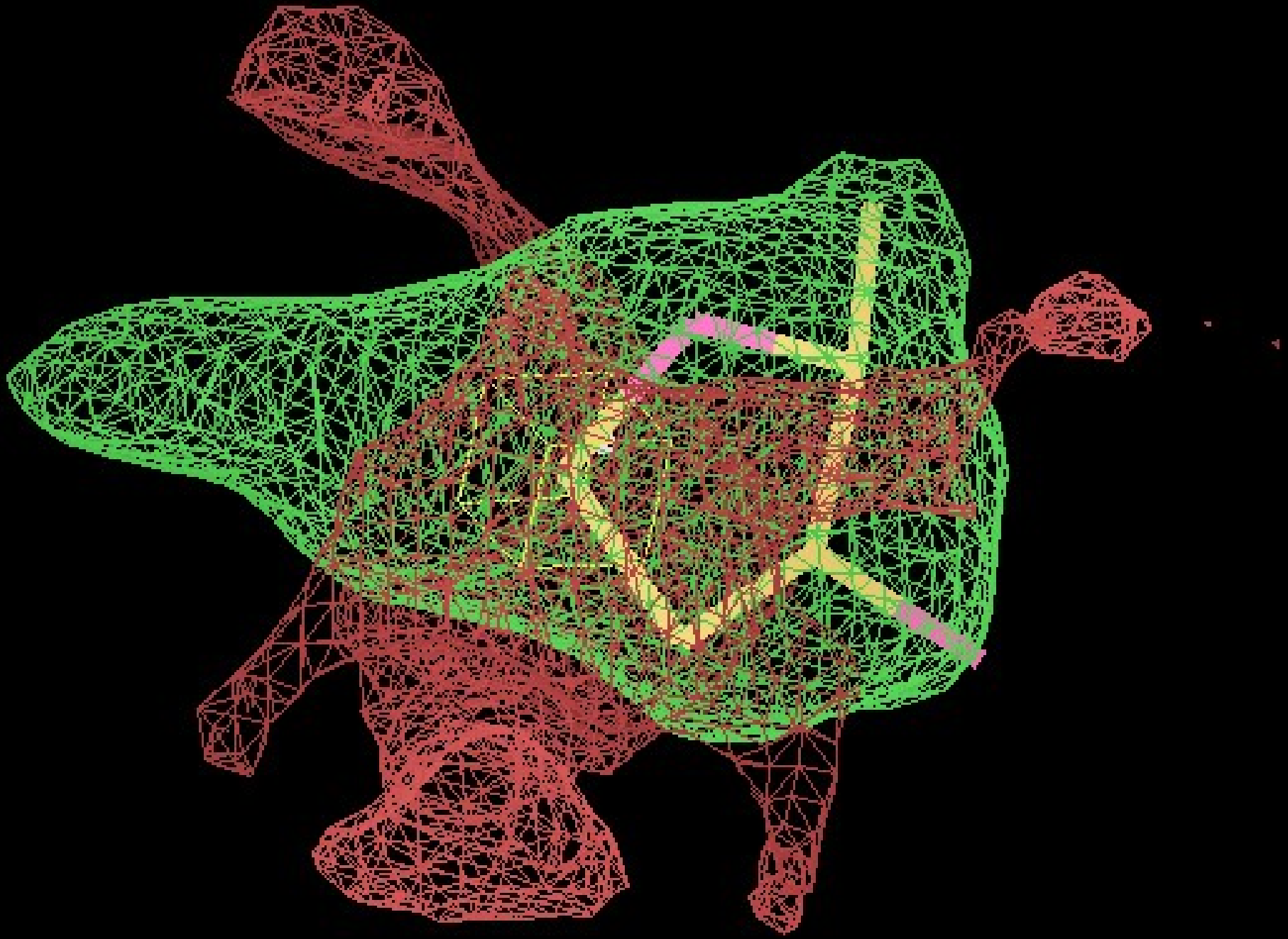
Nautilus

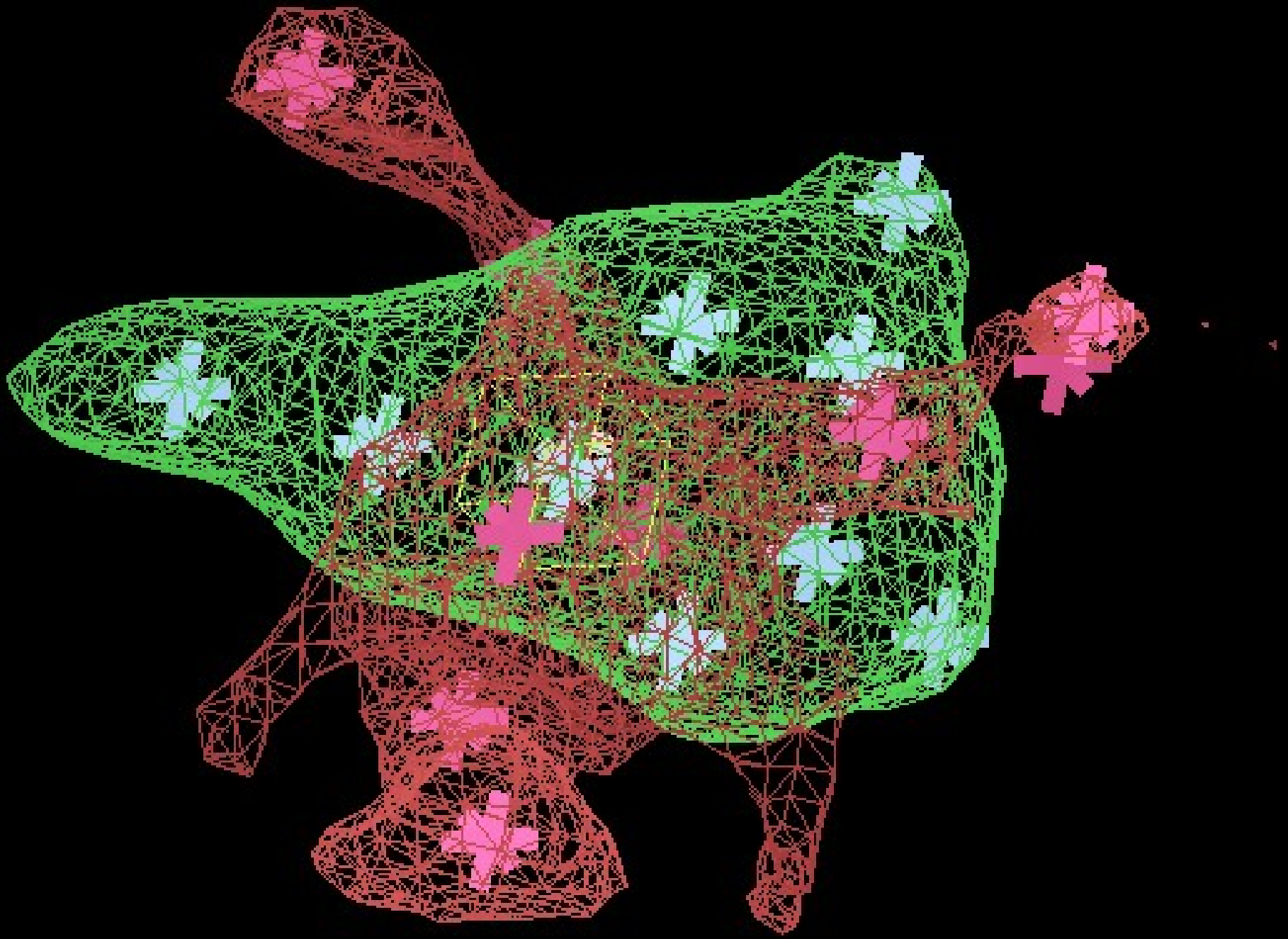
Types of fingerprint:

- Sugar
- Phosphate
- Base type (A/C/G/U)

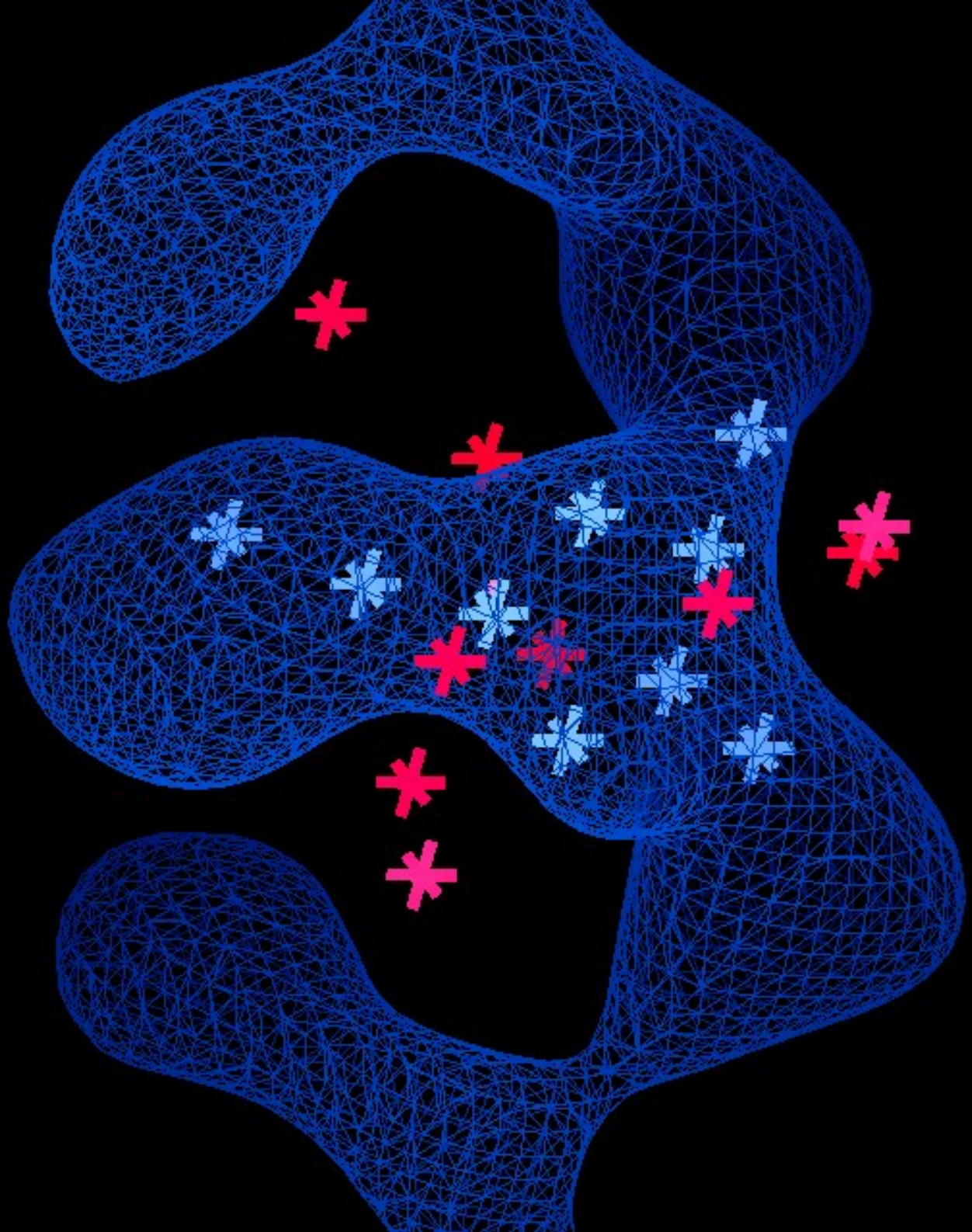




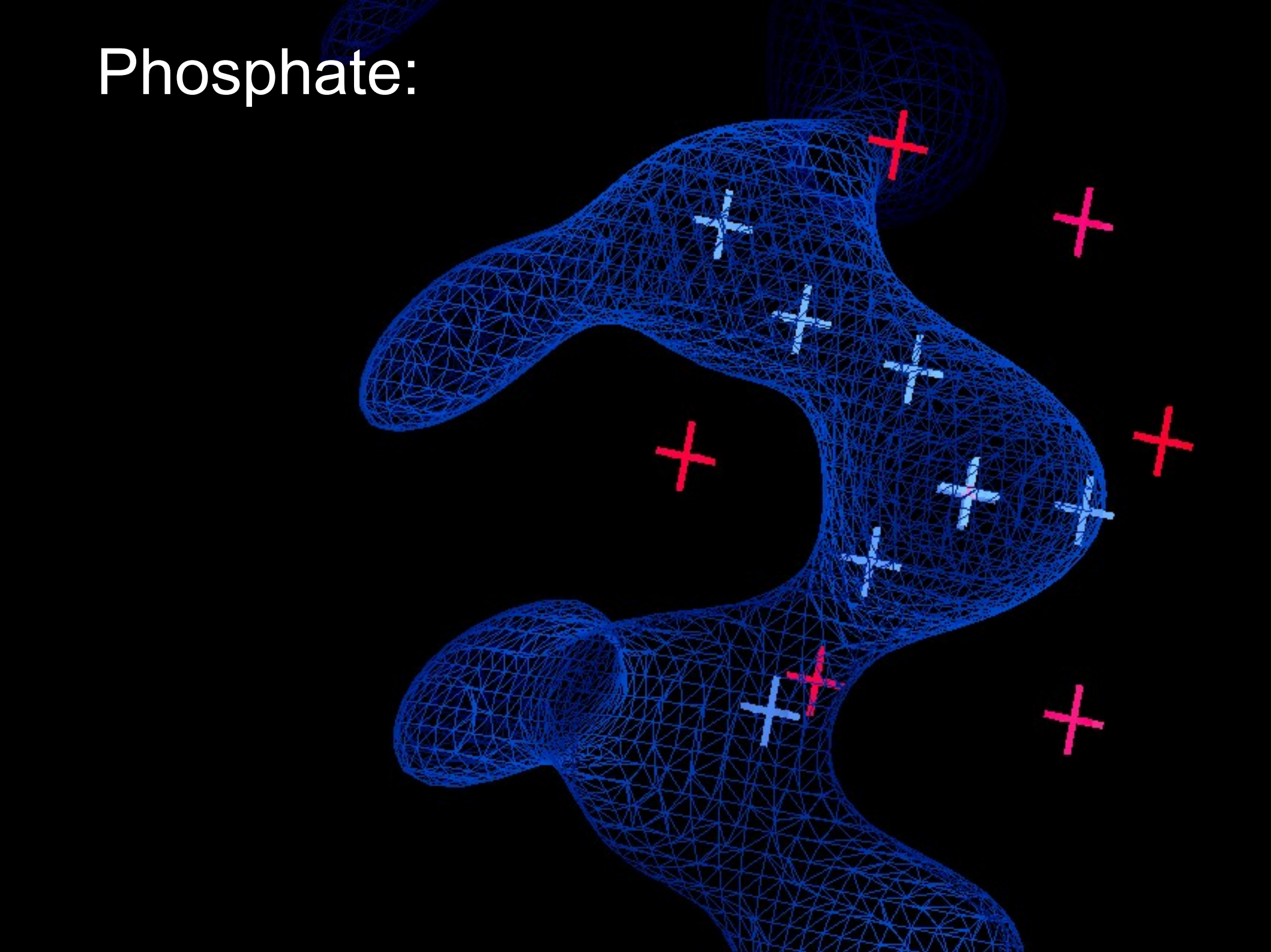




Sugar:



Phosphate:



Nautilus

Use the difference between the mean of the 'high' points and the mean of the 'low' points as a score indicating how likely it is the given group is present at a given position and orientation.

Need to search positions and orientations – a more optimized version of the same target uses the minimum of the highs minus the maximum of the lows – can often stop the calculation before testing all the sample points.

Nautilus

Steps:

- Find chain seeds
- Grow into chains
- Join overlapping chains
- Link nearby chains
- Prune clashing chains
- Rebuild chains to ensure connectivity
- Assign sequence
- Build bases

Nautilus

Find:

- Optimised 6-d rotation-translation using the sugar or phosphate fingerprint.
  - ~5 seconds for whole ASU
- Sugar:
  - Build a single nucleic acid using the best matching equivalent from the database, scored by 1 x sugar + 2 x phosphate fingerprints
- Phosphate:
  - Build a pair of nucleic acids using the best matching equivalent from the database, scored by 1 x phosphate + 2 x sugar fingerprints

Nautilus

Grow:

- Try adding additional nucleic acids to either end of each fragment, scored by the sugar fingerprint and the intermediate phosphate fingerprint.
  - ~1-2 seconds

Join:

- Merge overlapping fragments into longer fragments
  - <0.1 second

Link:

- Join fragments with nearby 3' and 5' terminii
  - ~0.5 second

Nautilus

Prune:

- Eliminate clashing regions
  - <0.1 second

Rebuild chains:

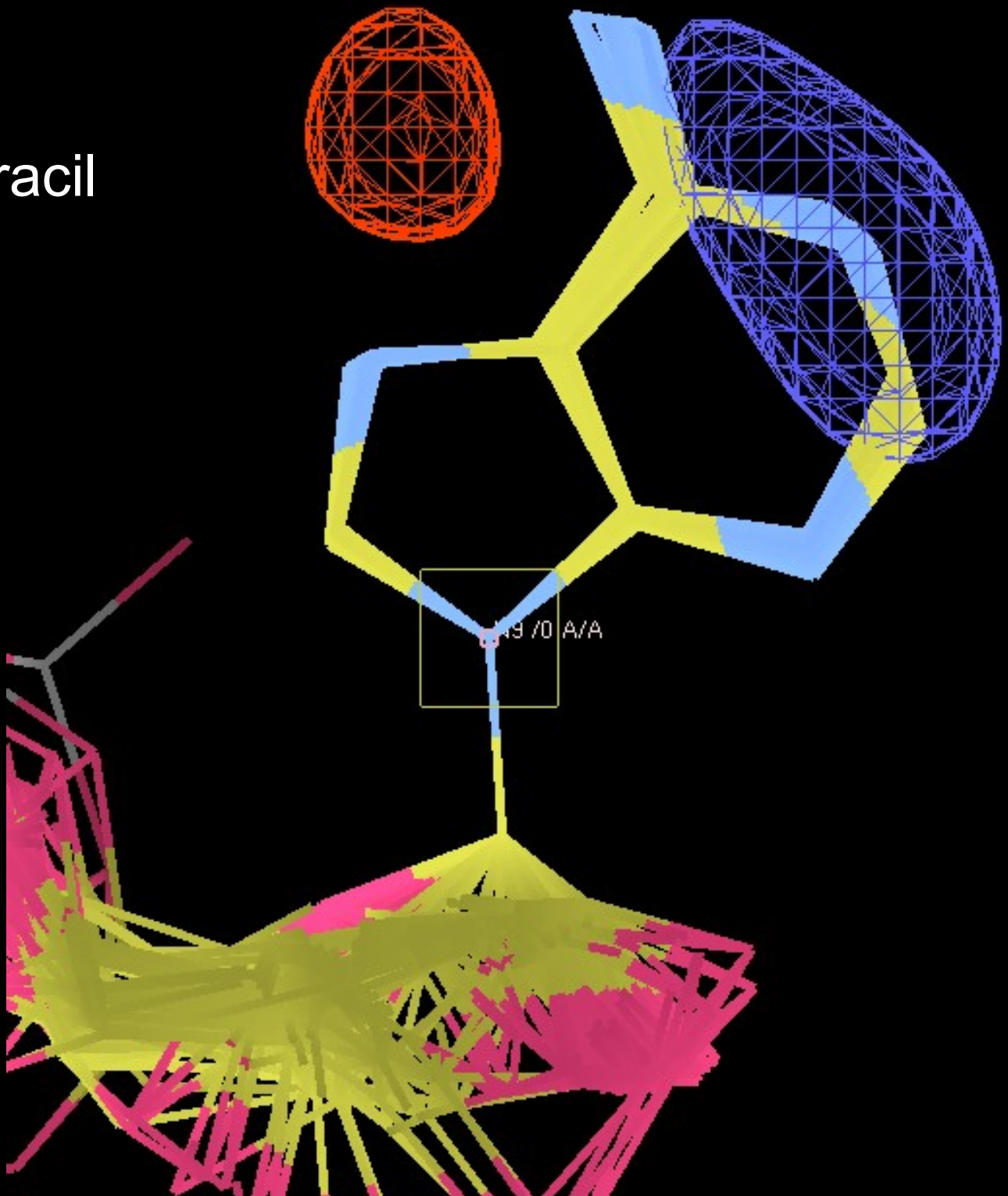
- Rebuild each sugar-sugar link using a fragment from the database
  - ~0.3 seconds

Sequence:

- Score base-type fingerprints at each position and assign sequence
  - <0.1 second

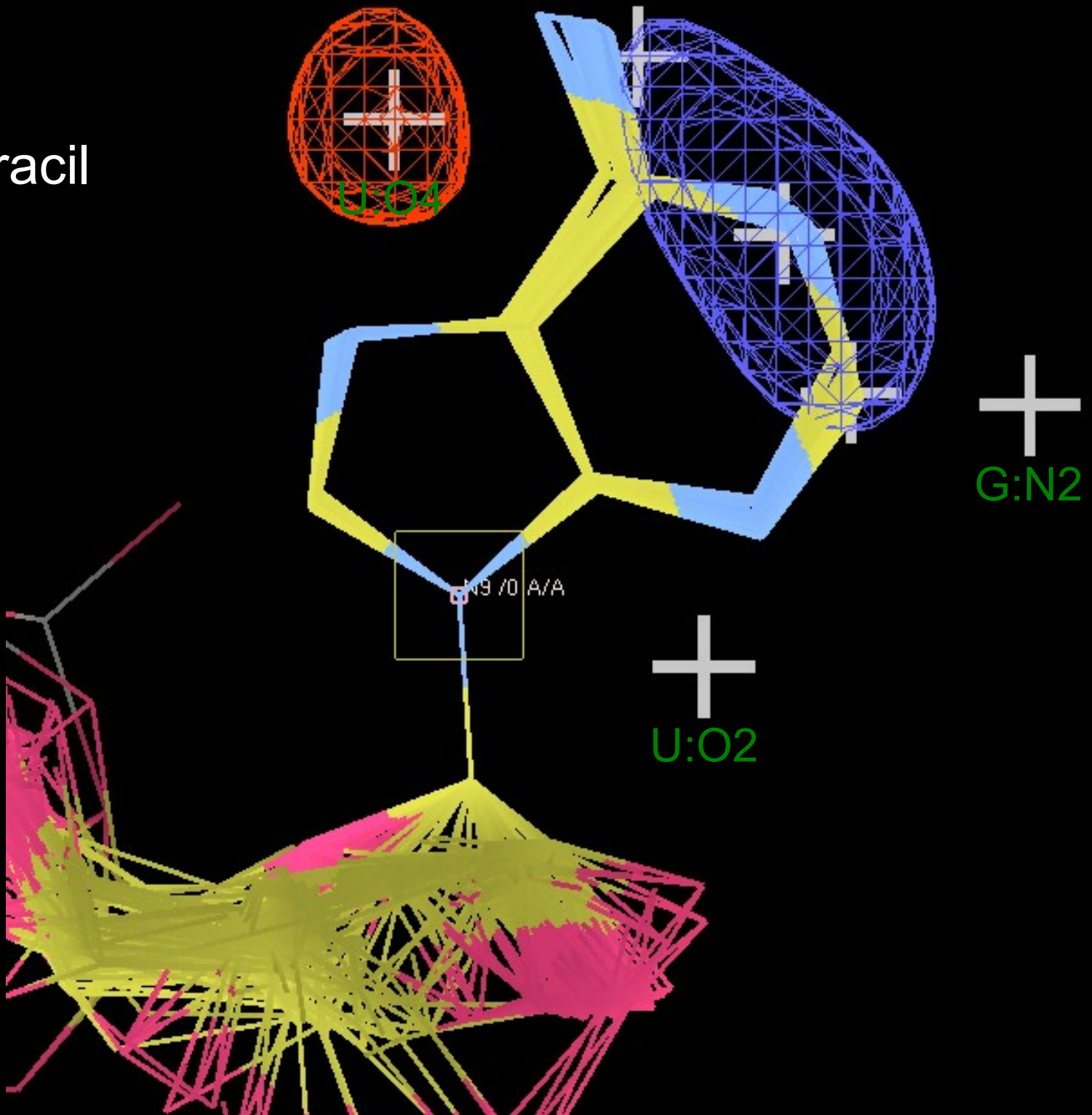
Base:

Adenine-Uracil



Base:

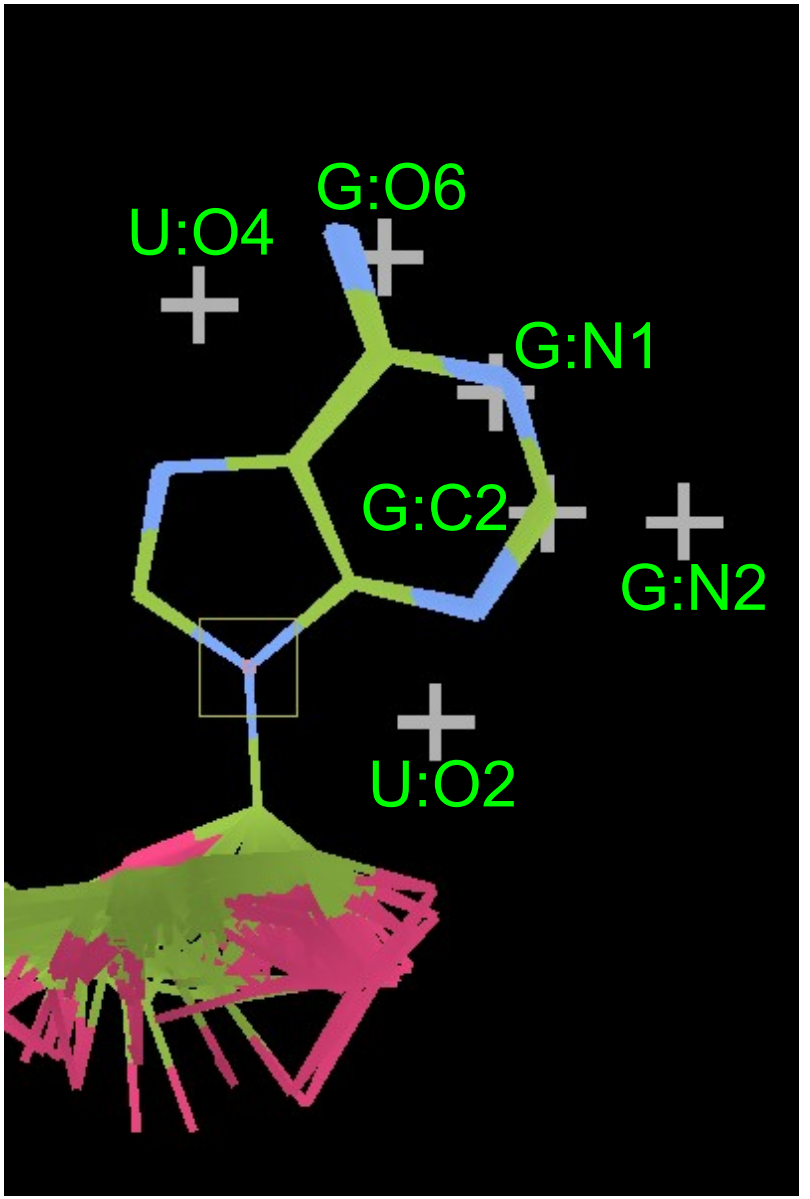
Adenine-Uracil



Nautilus

Adenine:

	U: O4	U: O2	G: O6	G: N1	G: C2	G: N2
A	-	-	+	+	+	-
C						
G						
U						

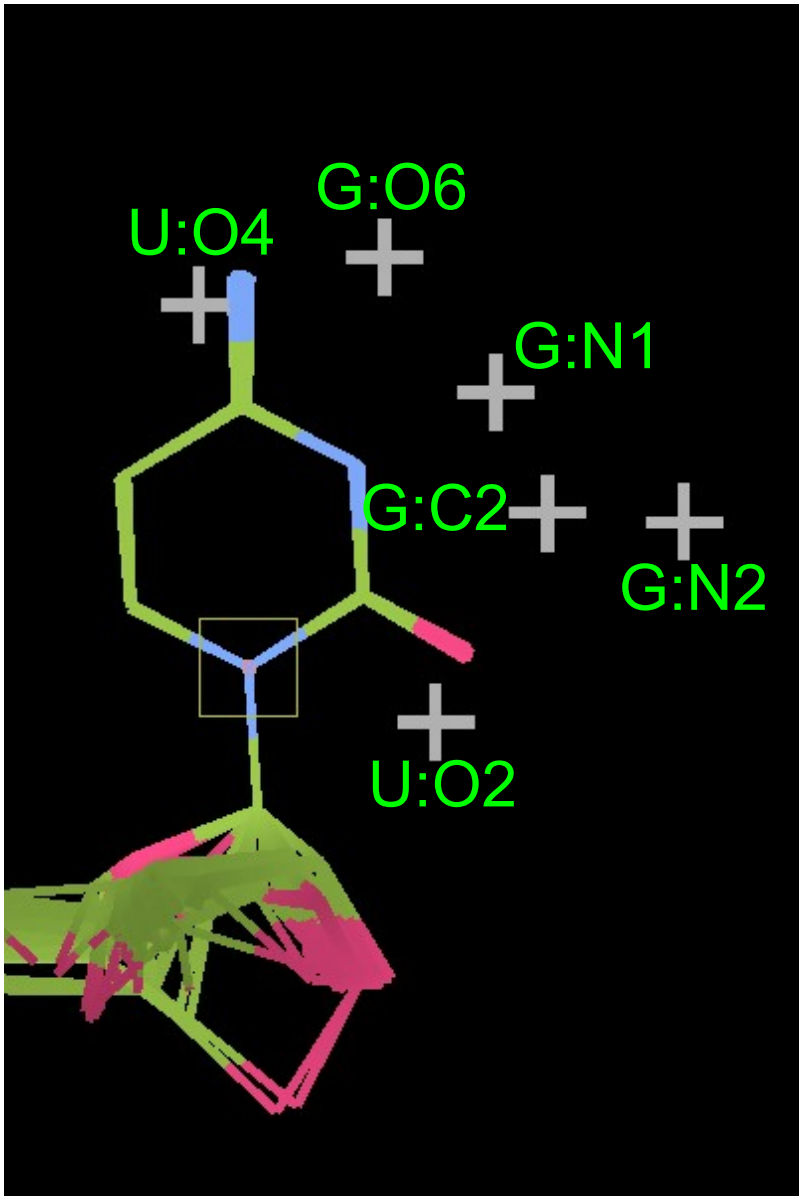


Kevin Cowtan, cowtan@ysbl.york.ac.uk

Nautilus

Cytosine:

	U: O4	U: O2	G: O6	G: N1	G: C2	G: N2
A	-	-	+	+	+	-
C	+	+	-	-	-	-
G						
U						

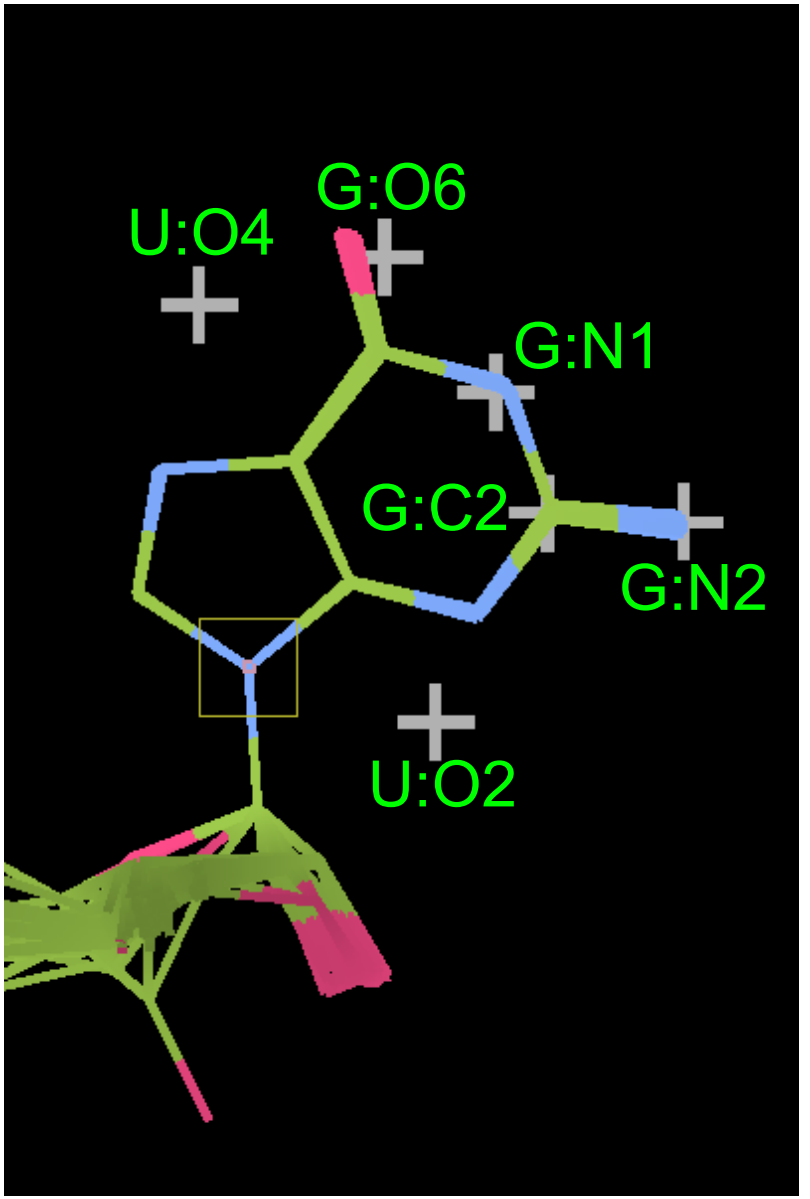


Kevin Cowtan, cowtan@ysbl.york.ac.uk

Nautilus

Guanine:

	U: O4	U: O2	G: O6	G: N1	G: C2	G: N2
A	-	-	+	+	+	-
C	+	+	-	-	-	-
G	-	-	+	+	+	+
U						

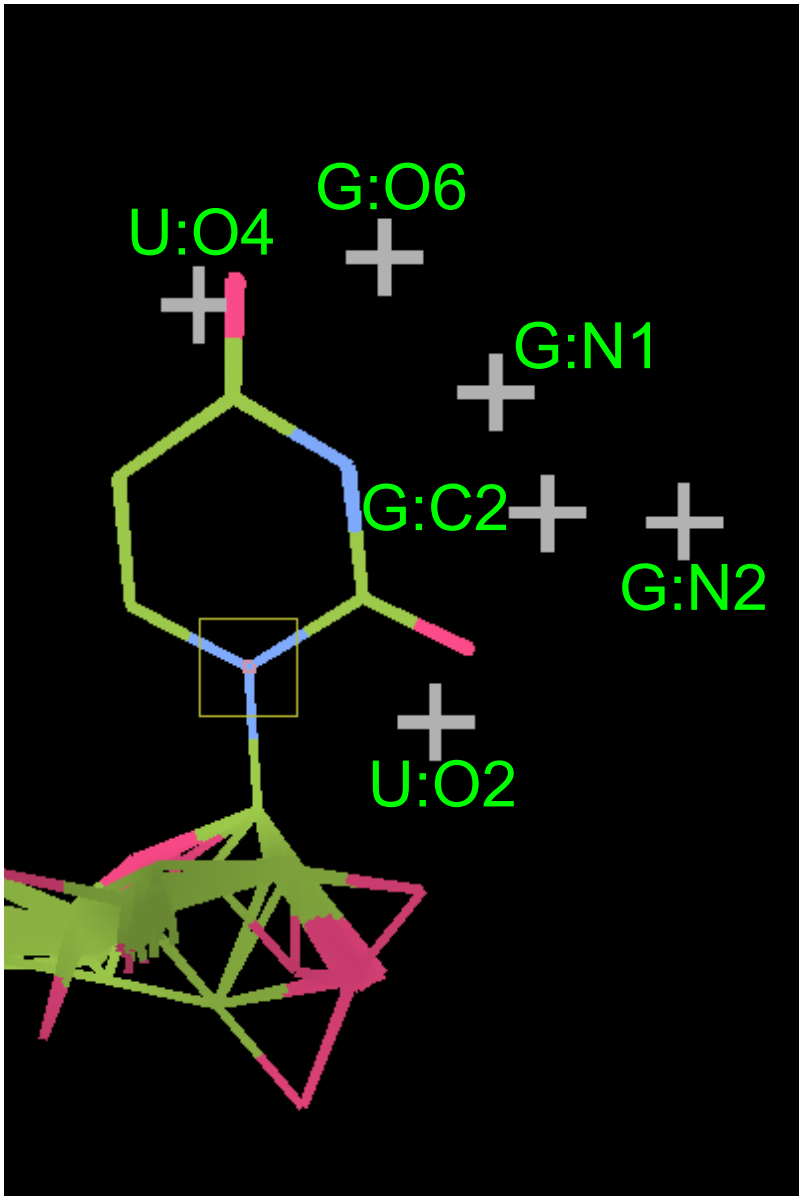


Kevin Cowtan, cowtan@ysbl.york.ac.uk

Nautilus

Guanine:

	U: O4	U: O2	G: O6	G: N1	G: C2	G: N2
A	-	-	+	+	+	-
C	+	+	-	-	-	-
G	-	-	+	+	+	+
U	+	+	-	-	-	-



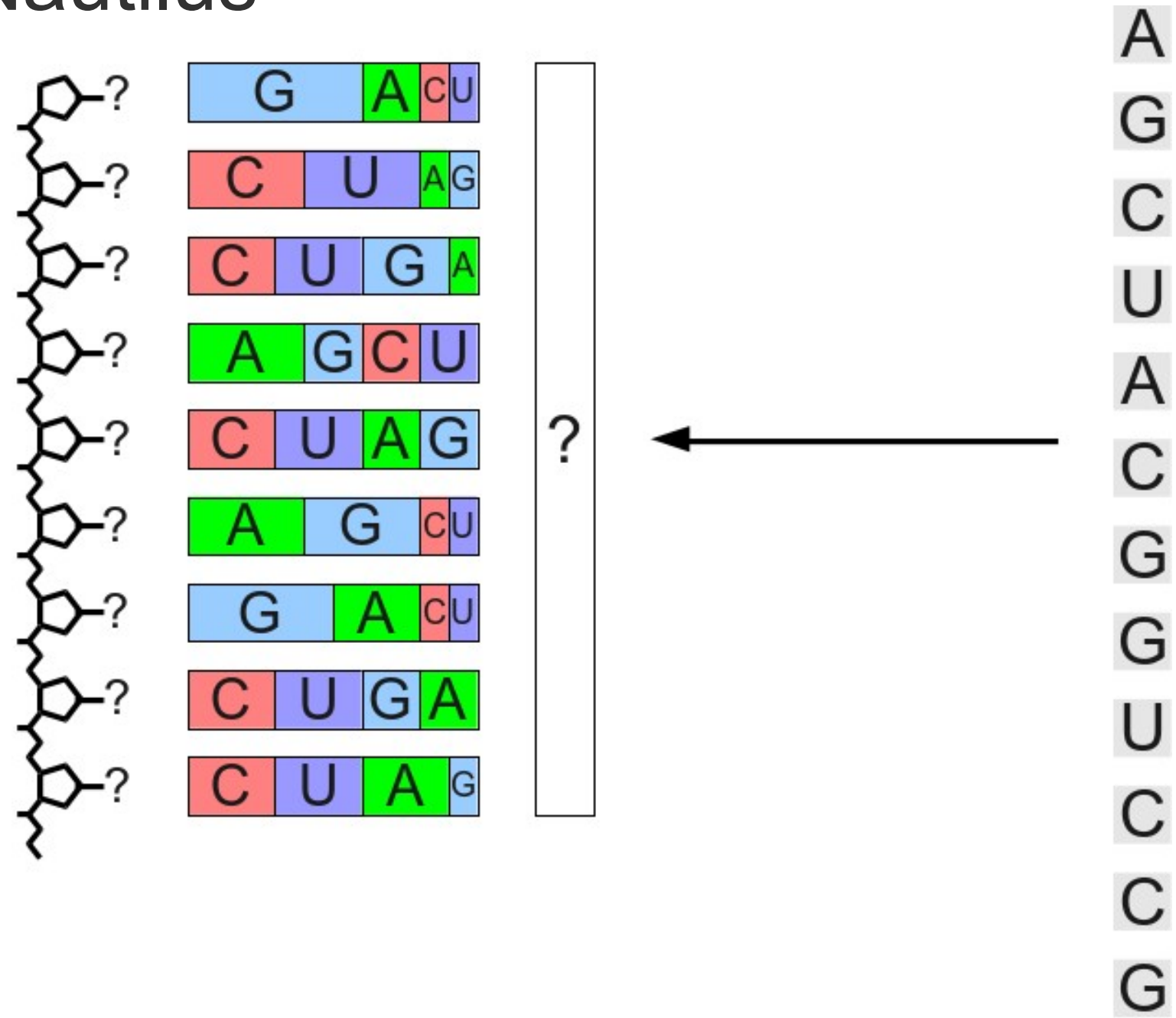
Nautilus

But the real world isn't black and white. Ideally we want a probability of a base being of a particular type.

- Calculate z-scored densities for the density at each of the 6 sample positions for 200 bases (50 of each type), to form a sample database.
- Calculate z-scored densities for the 6 sample positions of the unknown base.
- Find the 50 closest matches to the unknown base from the database.
- Assign probability of being A/C/G/U on the basis of the proportion of of the 50 closest matches being of each type (+ an error term).

Google: k-NN (k-Nearest Neighbour)

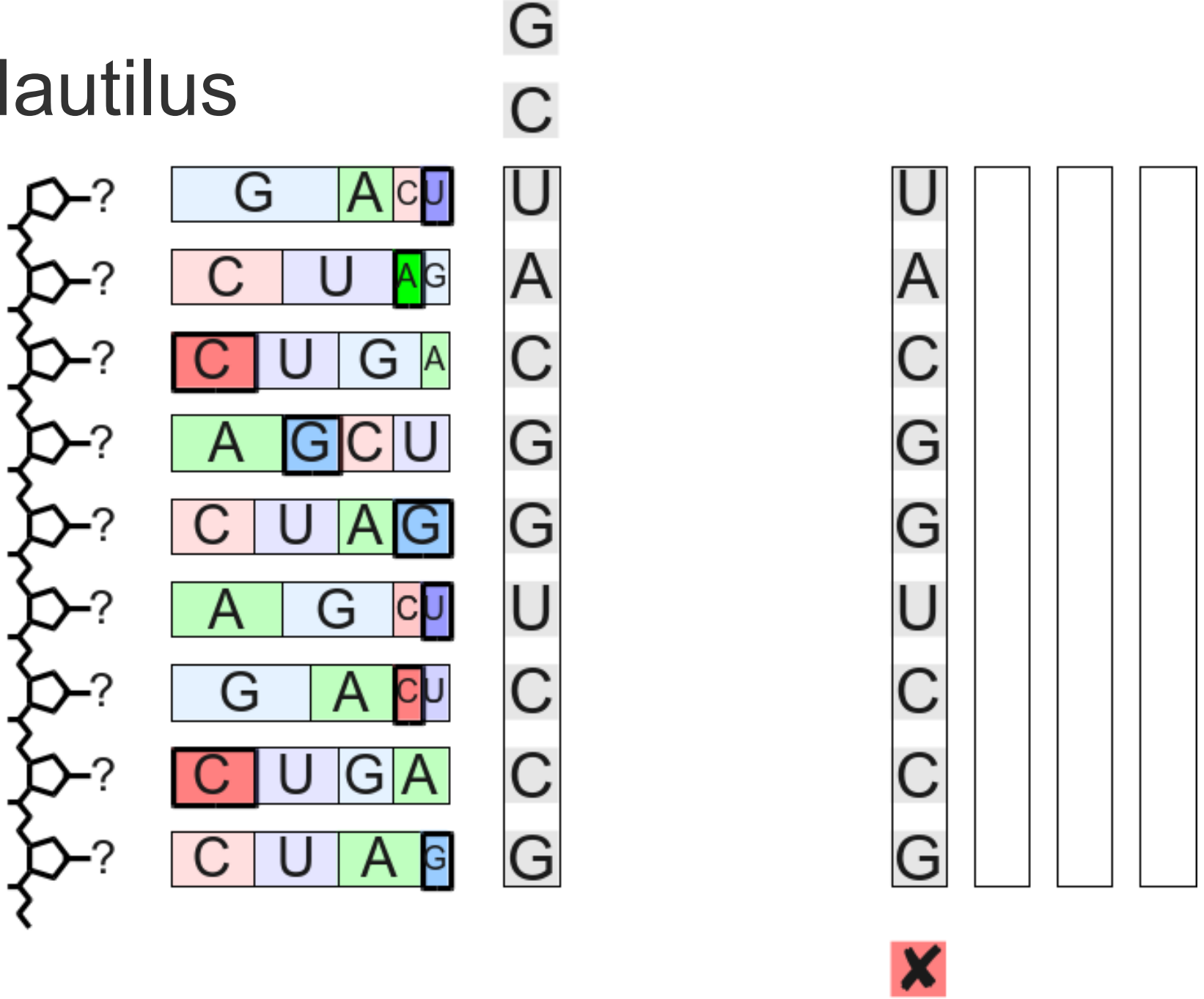
Nautilus



Kevin Cowtan, cowtan@ysbl.york.ac.uk

2012

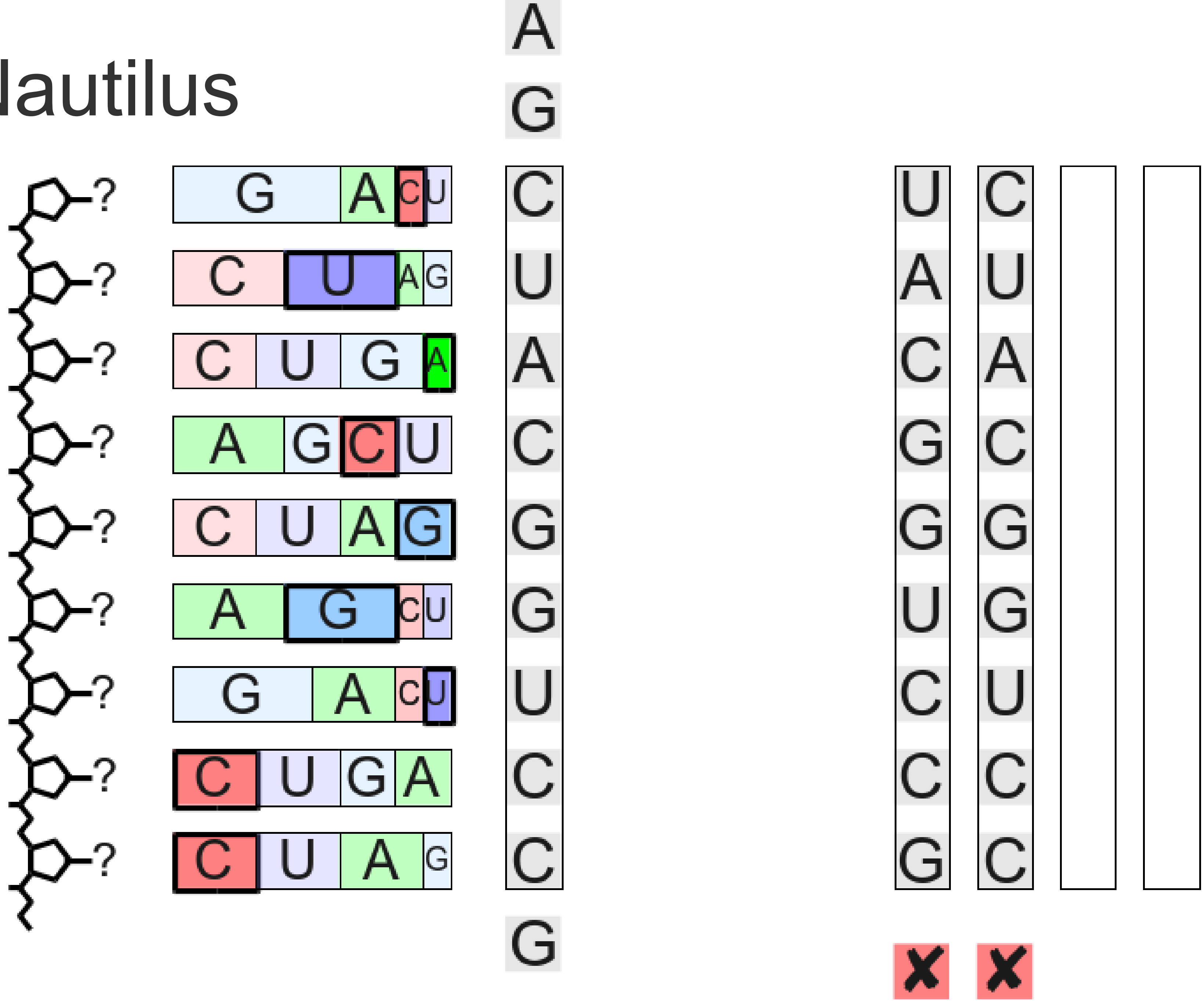
Nautilus



Kevin Cowtan, cowtan@ysbl.york.ac.uk

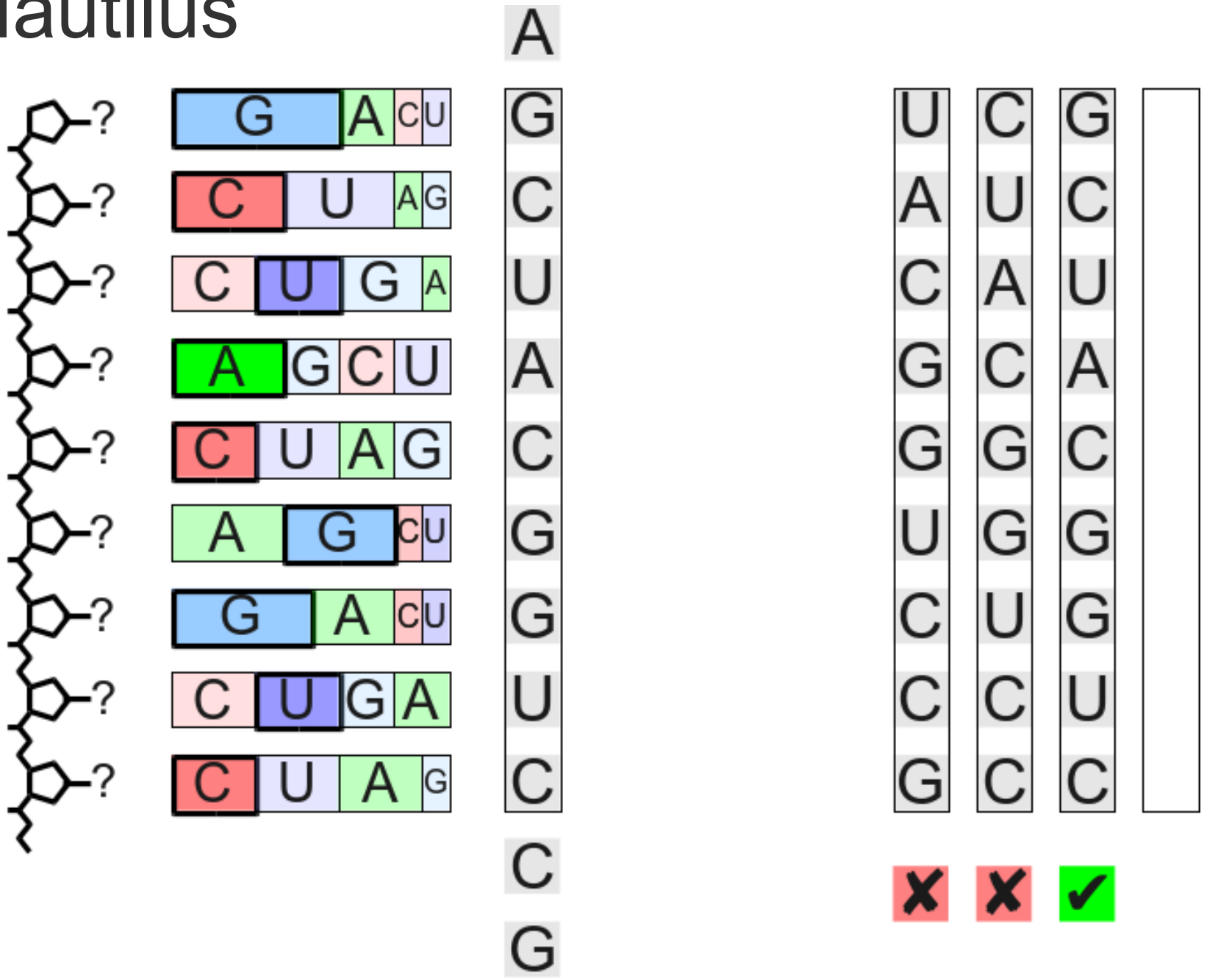
2012

Nautilus



Kevin Cowtan, cowtan@ysbl.york.ac.uk

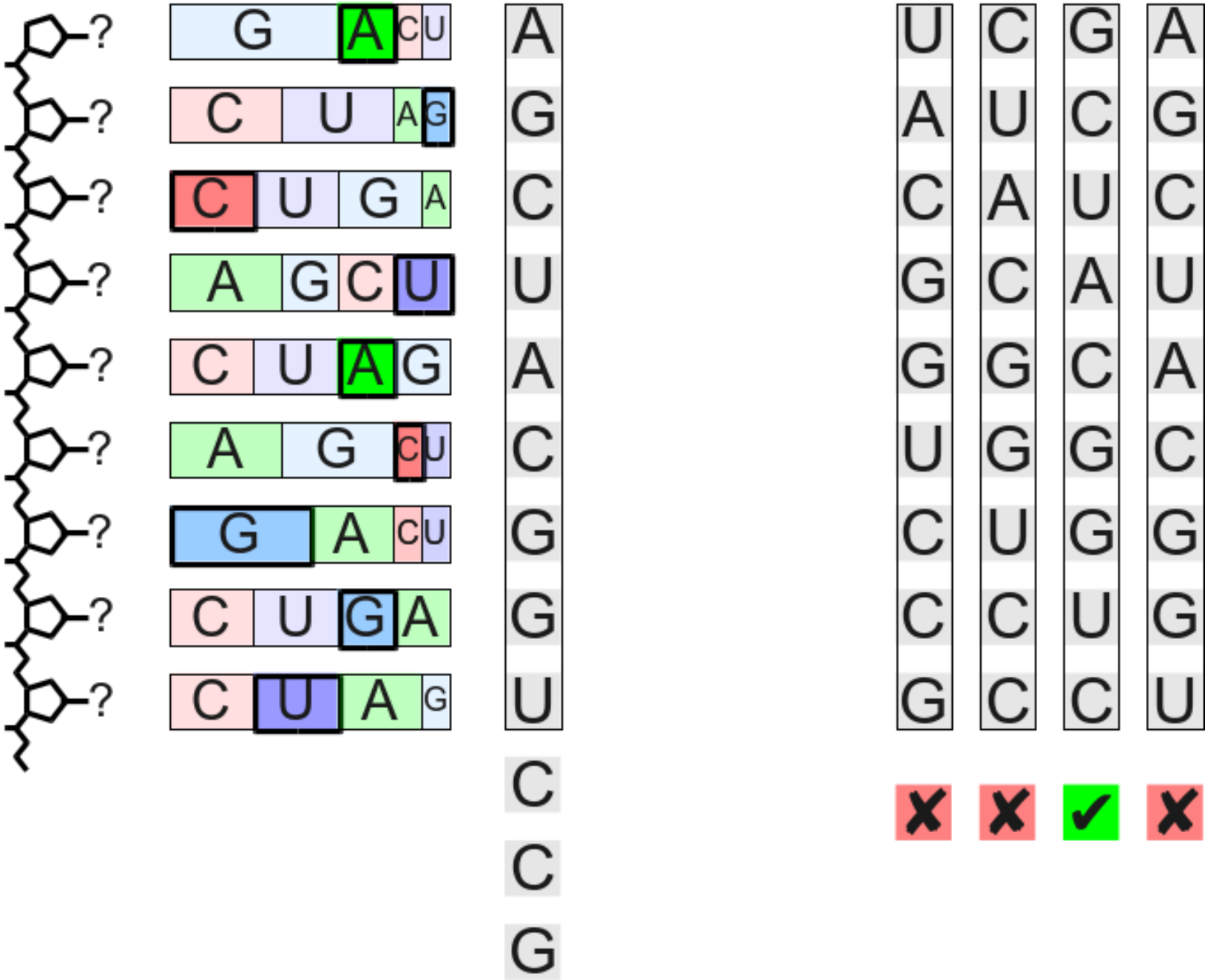
Nautilus



Kevin Cowtan, cowtan@ysbl.york.ac.uk

2012

Nautilus


G A C U
C U A G
C U G A
A G C U
C U A G
A G C U
G A C U
C U G A
C U A G
A G C C G G
U C G A
A U C G
C A U C
G C A U
G G C A
U G G C
C U G G
C C U G
G C C U
✗ ✗ ✓ ✗

Nautilus

Results:

- Good results on synthetic noisy data at 3.5A and user reports on real data at 3.8A.
  - Need more data
- Like '*buccaneer*', phases are more important than resolution.
- Failed on a quadruplex structure with good phases.
  - Try a different database?

Achnowledgements

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