

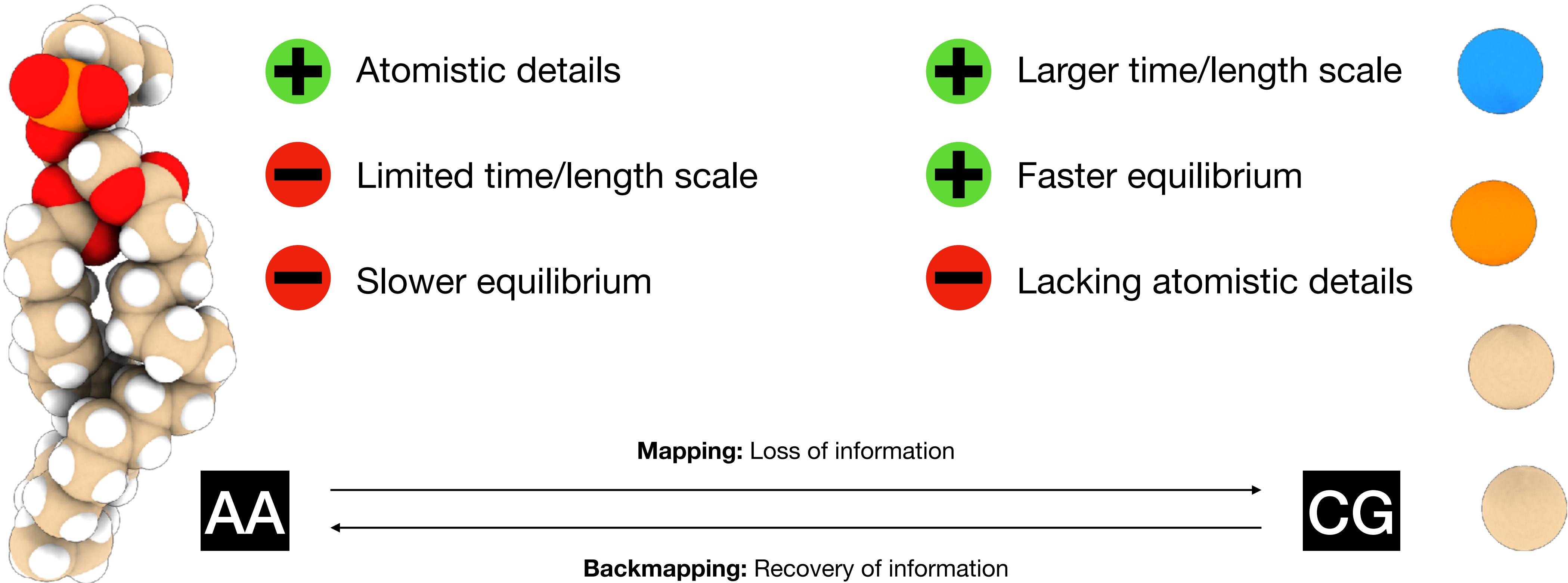
# **Multiscale Simulation Tool**

## **Backmapping with mapping schemes**

**Siyoung Kim**

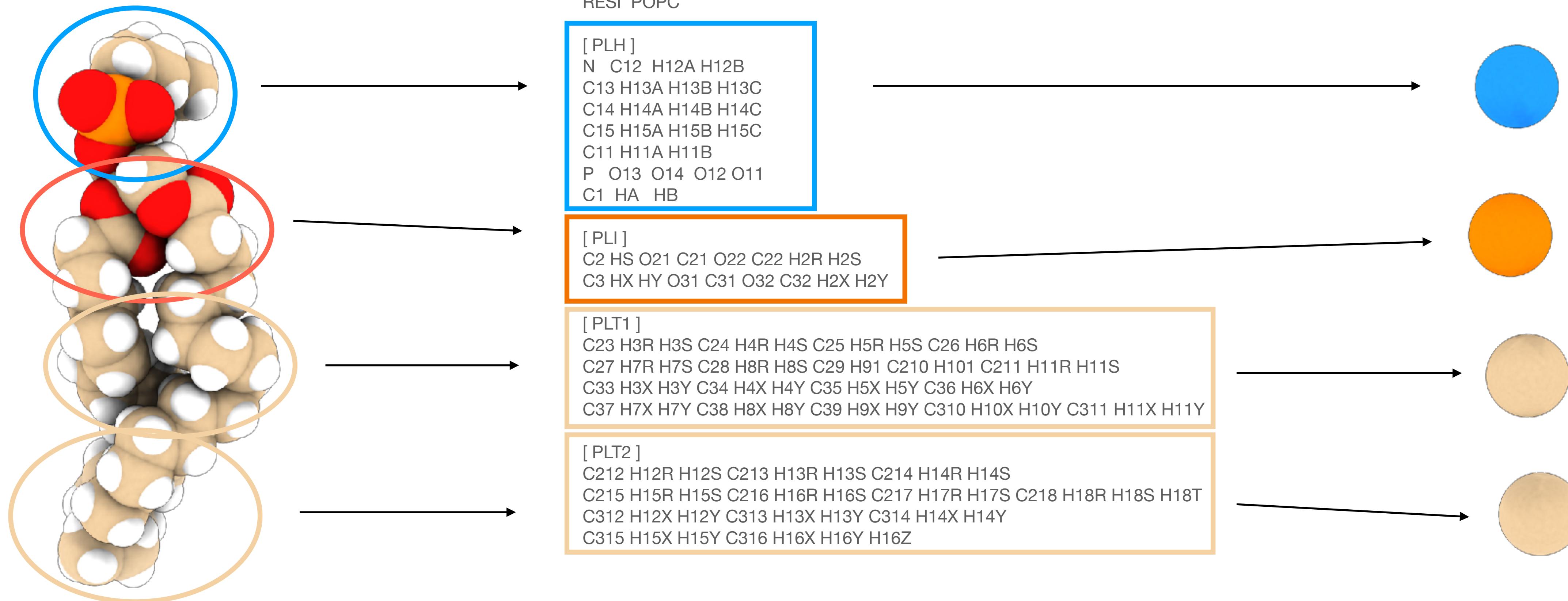
# Strength and limitation of resolutions

## All-atom (AA) vs Coarse-grained (CG)

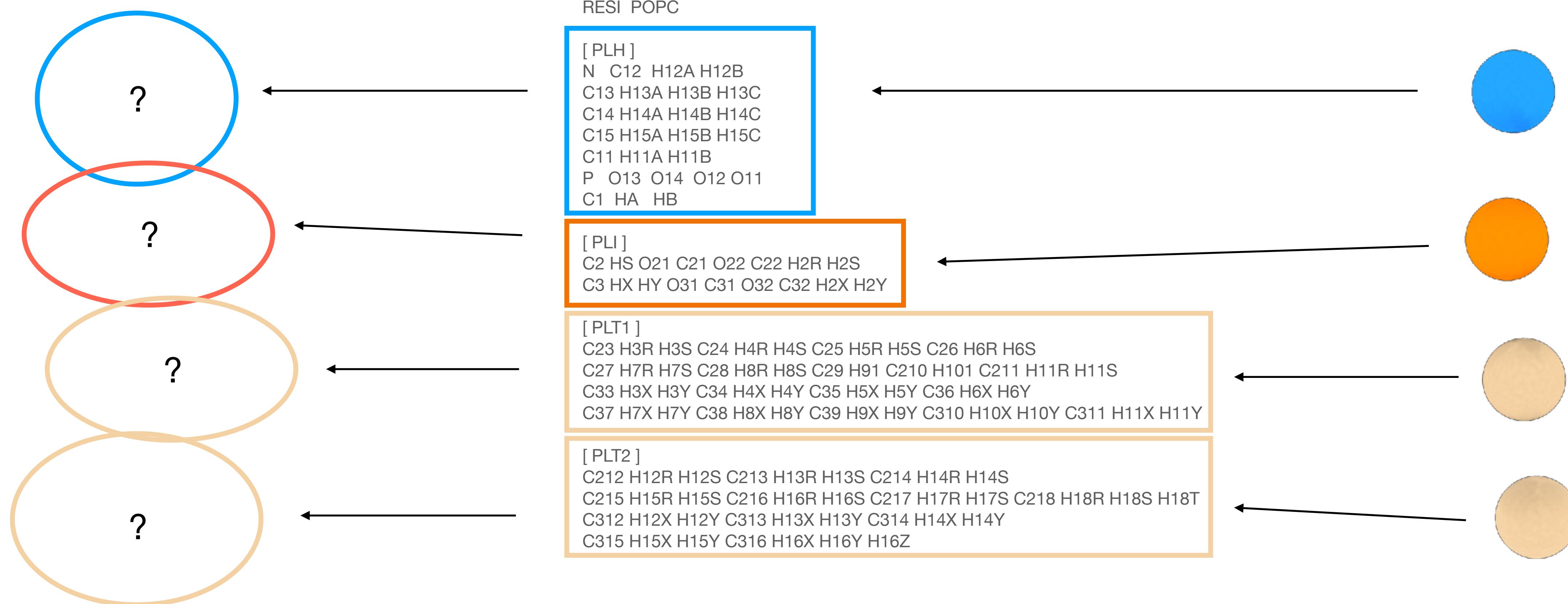


# Mapping is straightforward

## Mapping scheme: Atoms and their corresponding CG beads



# Backmapping is NOT straightforward



# mstool: Backmapping with mapping scheme

**Geometrical approach**  
Marrink group

[ atoms ]

1	N	BB
2	HN	BB
3	CA	BB
4	HA	BB
5	CB	SC1 BB
6	HB1	SC1 BB
7	HB2	SC1 BB
8	CG	SC1 SC1 SC1 SC1 SC3 SC3
9	CD1	SC1
10	HD1	SC1
11	NE1	SC2 SC1
12	HE1	SC2
13	CE2	SC2 SC2 SC3
14	CD2	SC3 SC3 SC2
15	CE3	SC3 SC3 SC4
16	HE3	SC3
17	CZ3	SC4 SC4 SC3
18	HZ3	SC4
19	CZ2	SC2 SC2 SC4
20	HZ2	SC2
21	CH2	SC4 SC4 SC2
22	HH2	SC4
23	C	BB
24	O	BB

[ trans ]

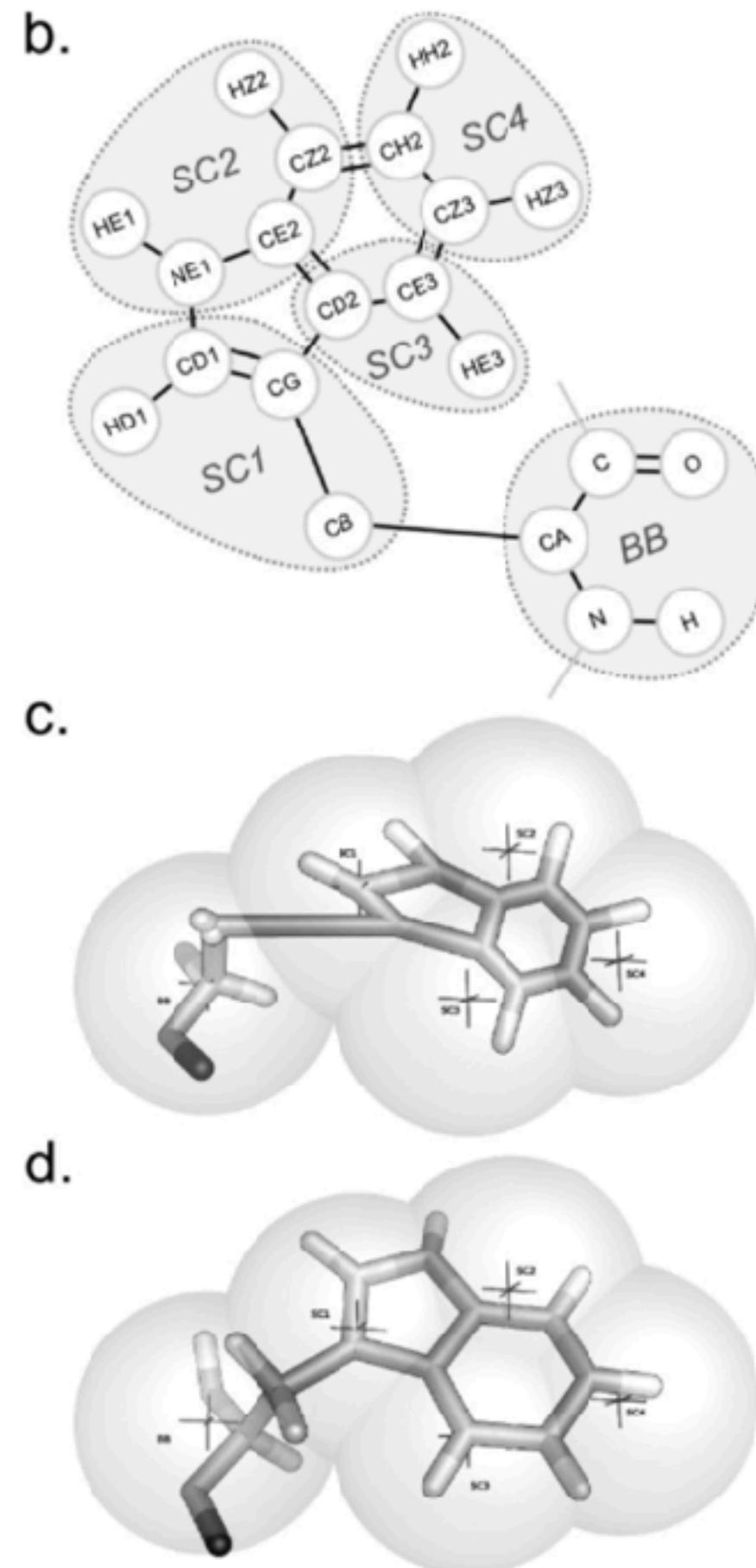
CB	CG	CD2	CE2
HD1	CD1	NE1	CE2
HE1	NE1	CD1	CG
HE3	CE3	CD2	CE2
HZ2	CZ2	CE2	CD2
HZ3	CZ3	CE3	CD2
HH2	CH2	CZ3	CE3

[ chiral ]

CB	CA	N	C
HB1	CA	N	C
HB2	CA	N	C

[ chiral ]

HA	CA	N	CB	C
; L-Trp				



**mstool**  
Siyoung Kim

RESI TRP

[ BB ]  
N HN CA HA C O

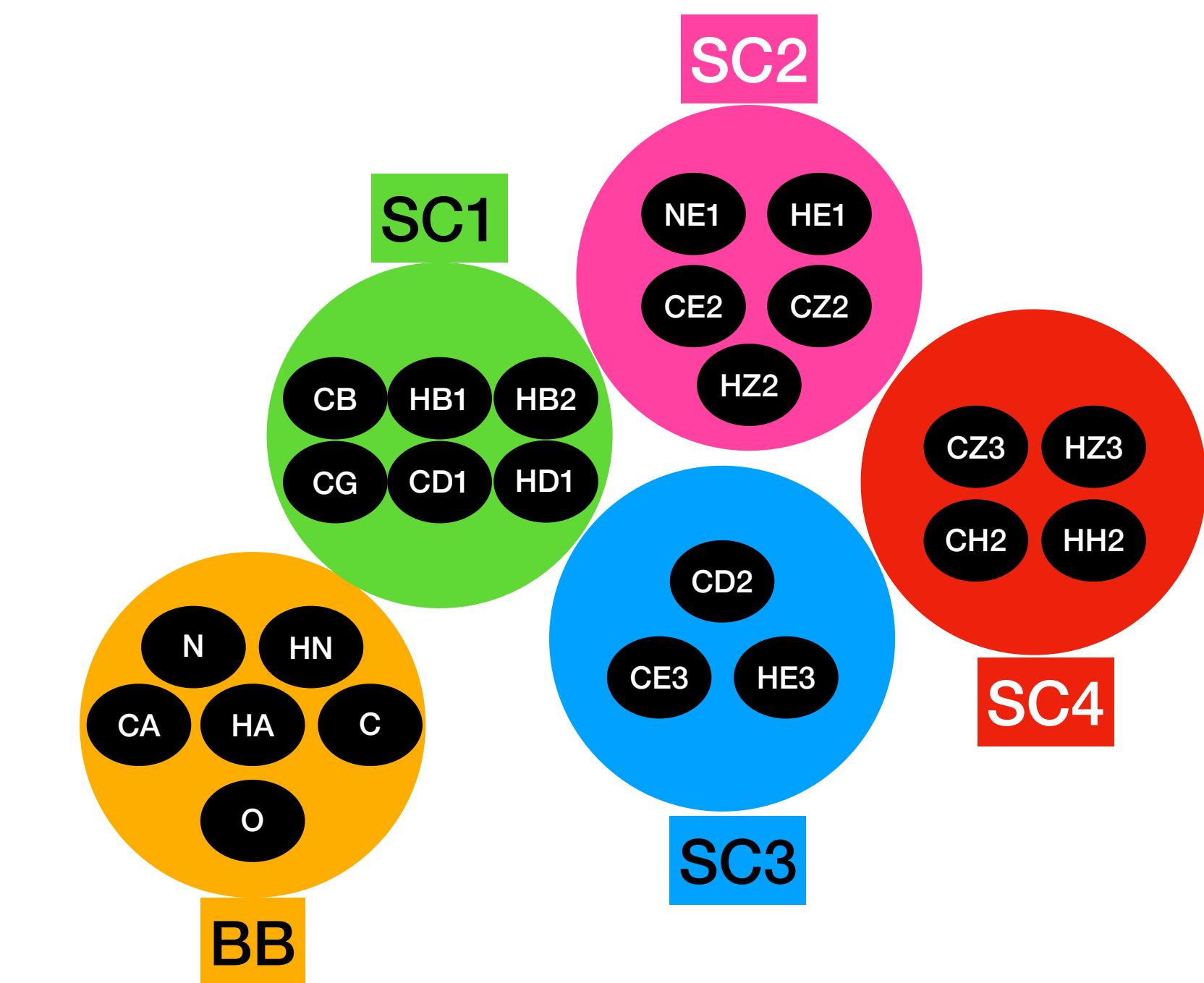
[ SC1 ]  
CB HB1 HB2  
CG CD1 HD1

[ SC2 ]  
NE1 HE1 CE2 CZ2 HZ2

[ SC3 ]  
CD2 CE3 HE3

[ SC4 ]  
CZ3 HZ3 CH2 HH2

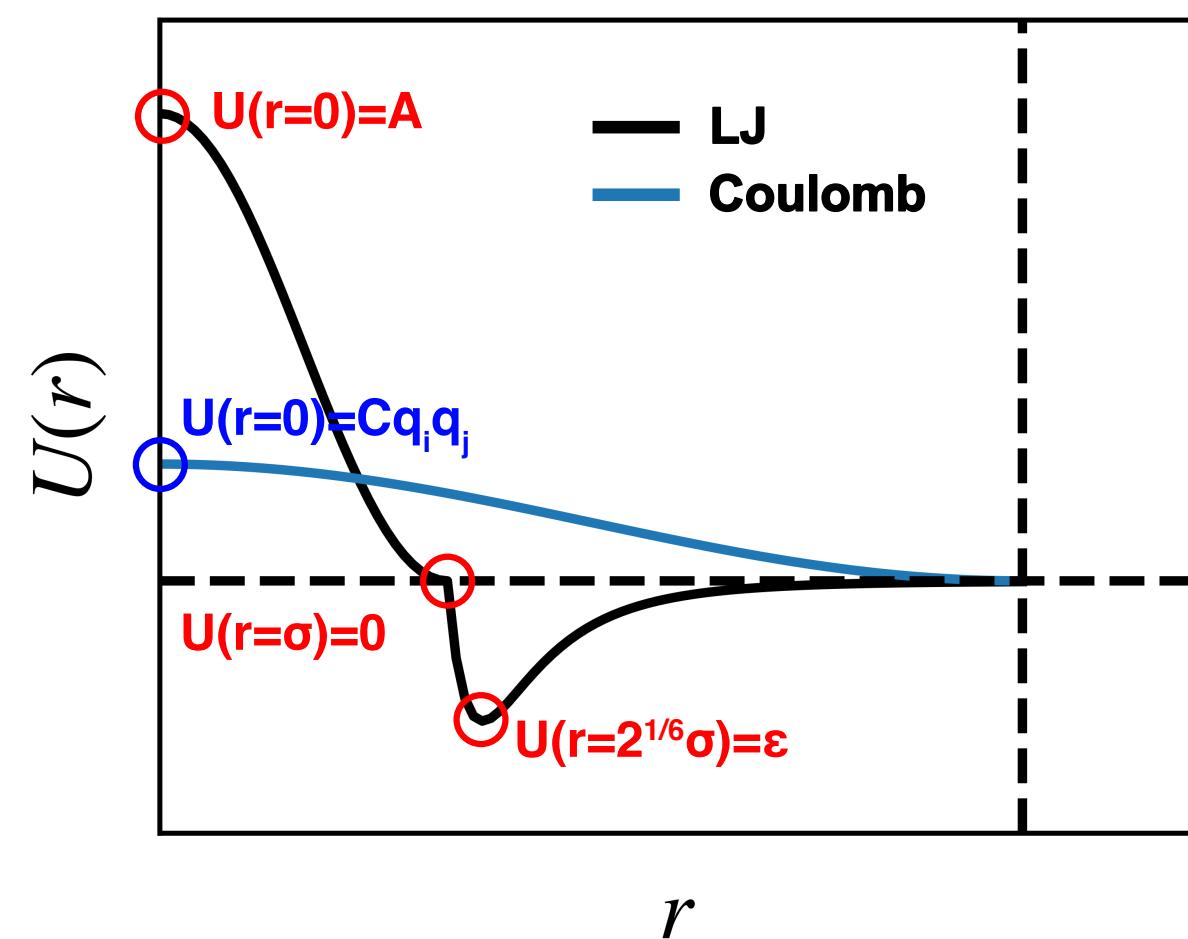
[ chiral ]  
HA CA N CB C



isomeric properties of a molecule should be indicated: **chiral/cis/trans**

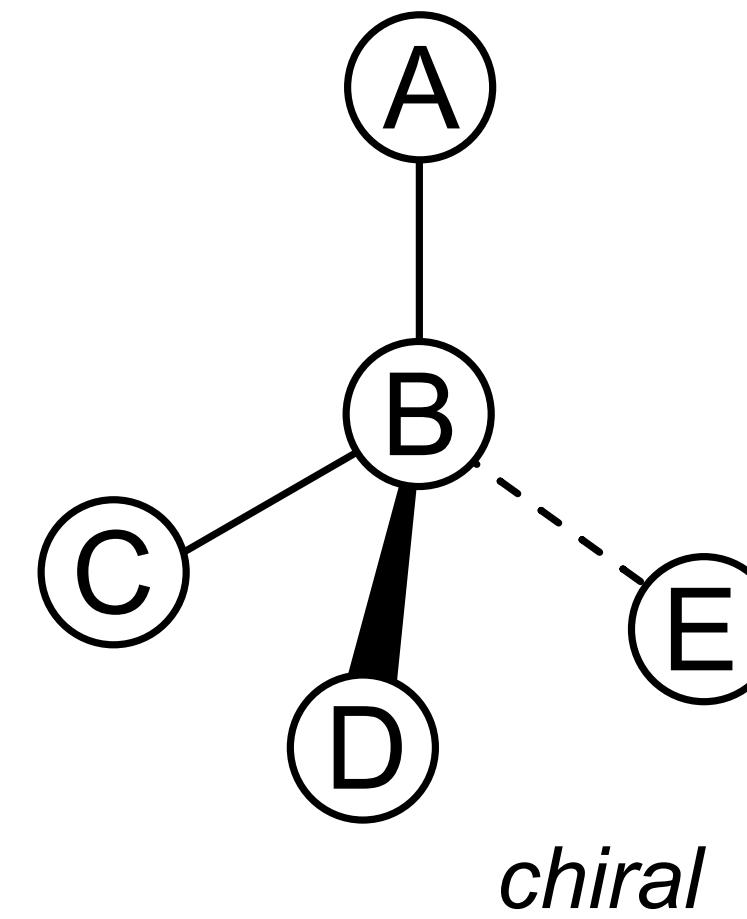
# mstool

1. Place atoms randomly near their corresponding CG bead.
2. Carry out a special energy minimization (REM).
  - Reduce nonbonded interactions
  - Apply dihedrals for chiral/cis/trans



$$U_{LJ}(r) = \min \left( A \cos^2 \left( \frac{\pi}{2\sigma} r \right), 4\epsilon \left( \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right) \right)$$

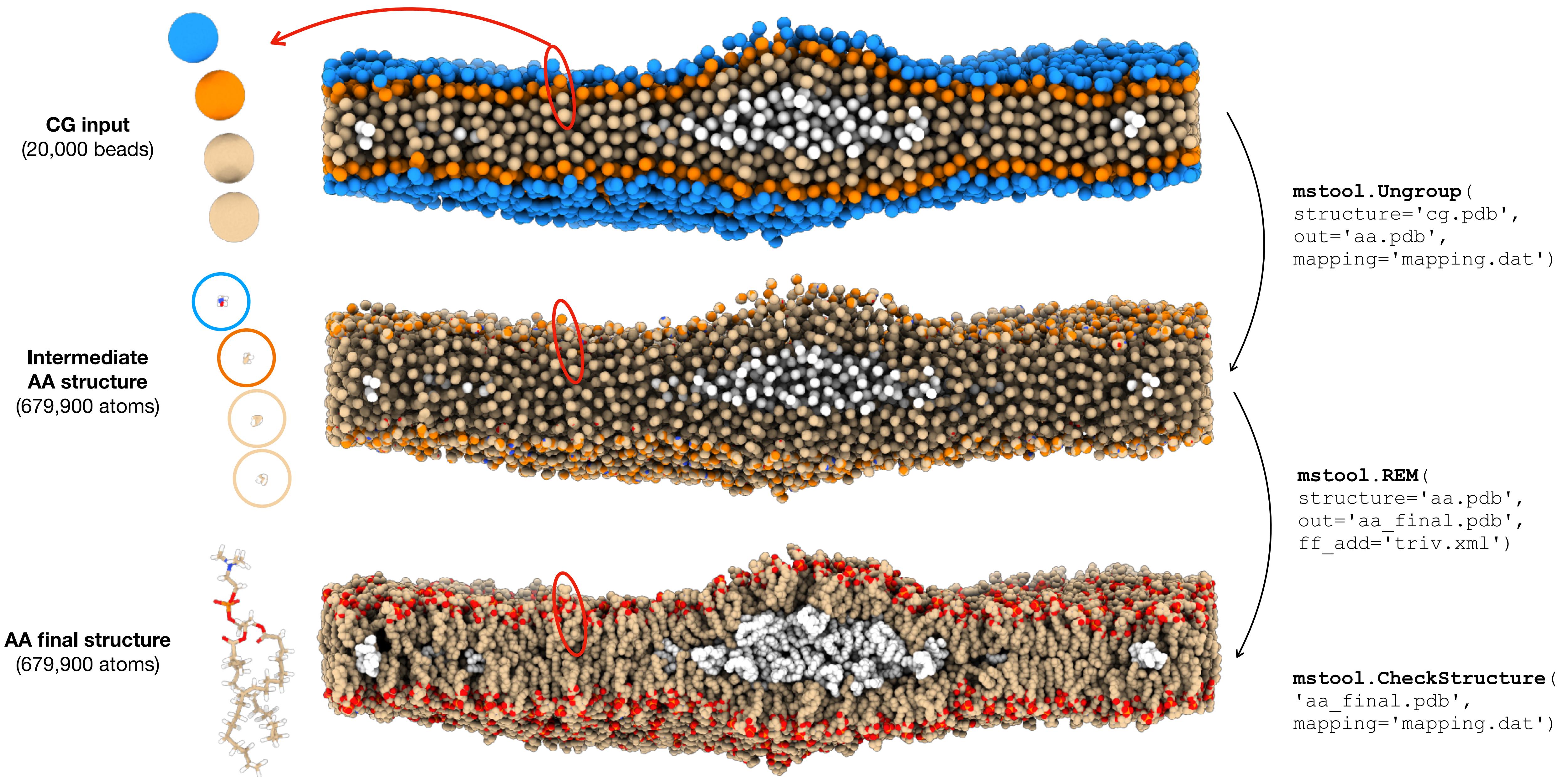
$$U_C(r) = C q_i q_j \cos^2 \left( \frac{\pi}{2r_c} r \right)$$



Custom Torsion Force  
 $V = K * (\xi - \xi_0)^2$   
 $K = 300 \text{ kJ/mol/rad}^2$   
 $\xi_0$  (rad) written below

B	C	D	E	-0.615
B	D	E	C	-0.615
B	E	C	D	-0.615
A	C	D	E	-1.231
A	D	E	C	-1.231
A	E	C	D	-1.231

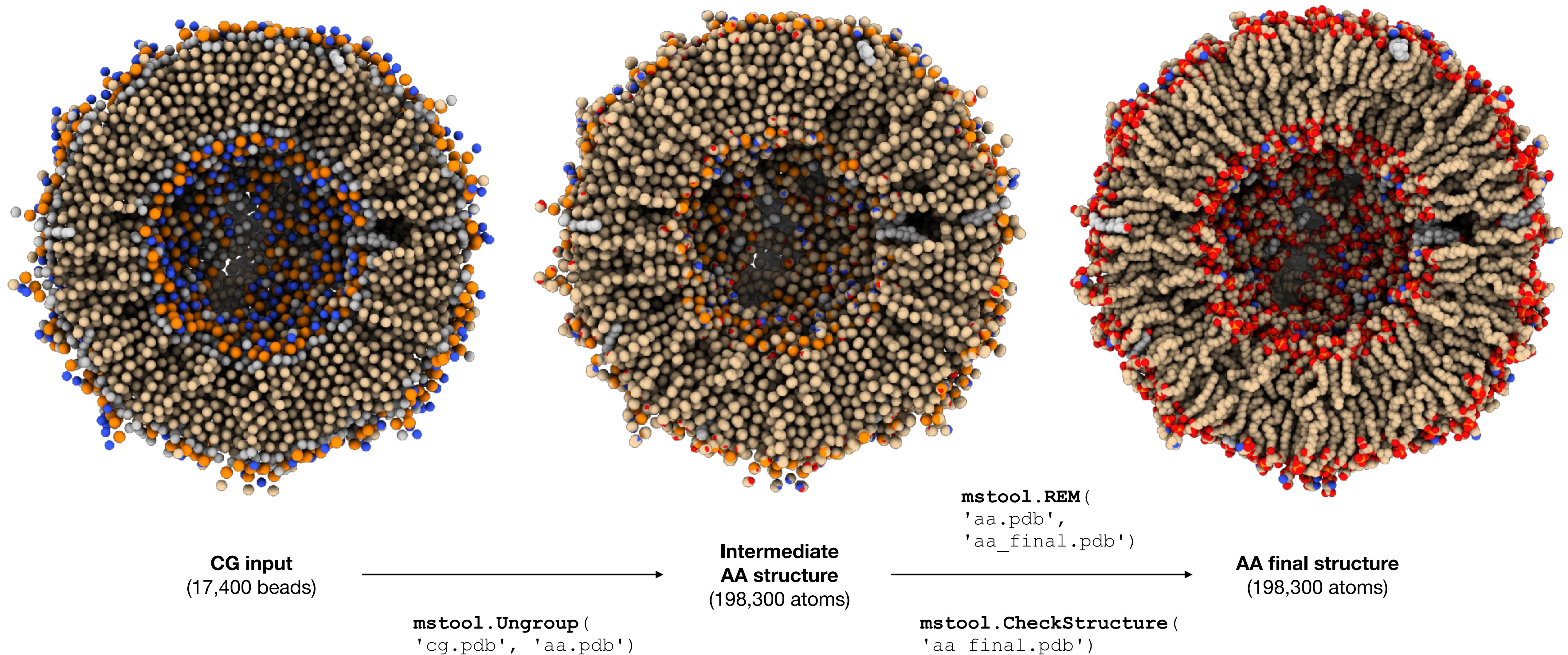
# folder: \$mstool/examples/Fig5A\_hCG



# folder: \$mstool/examples/Fig5B\_SphereMartini

Martini lipids:

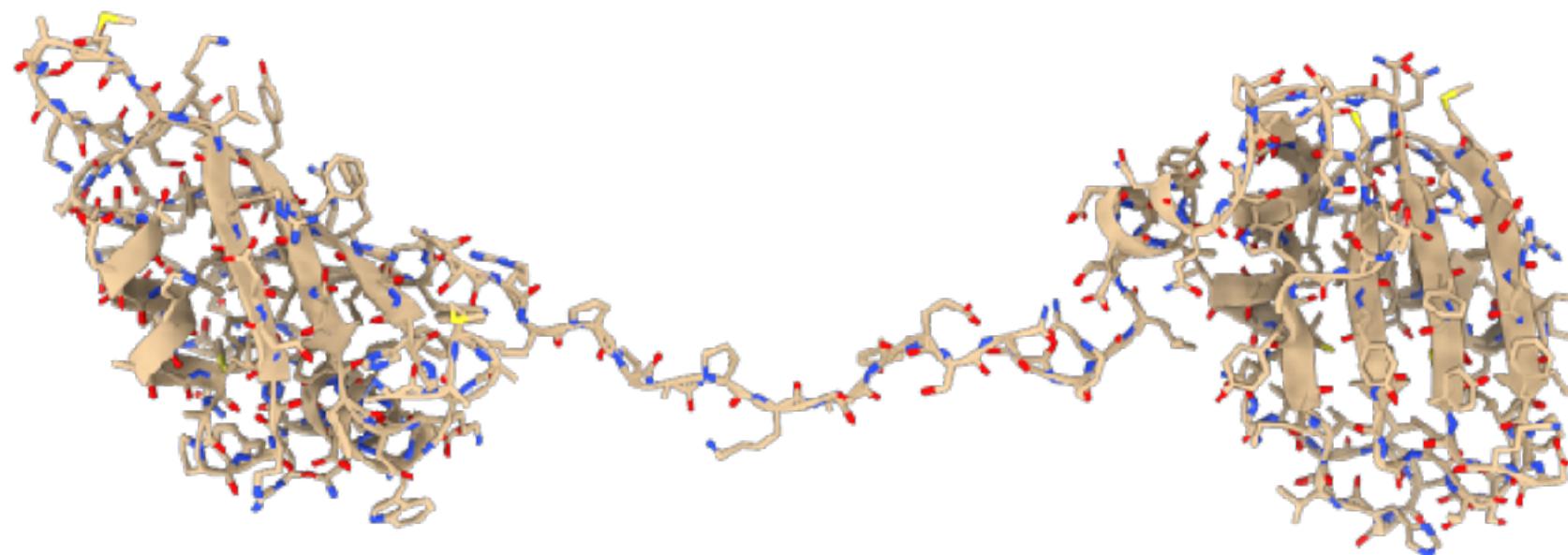
POPS DPPC CHL1 DMPC DOPG DSPC POPA DOPS POPC DOPE POPG POPE DOPA DOPC



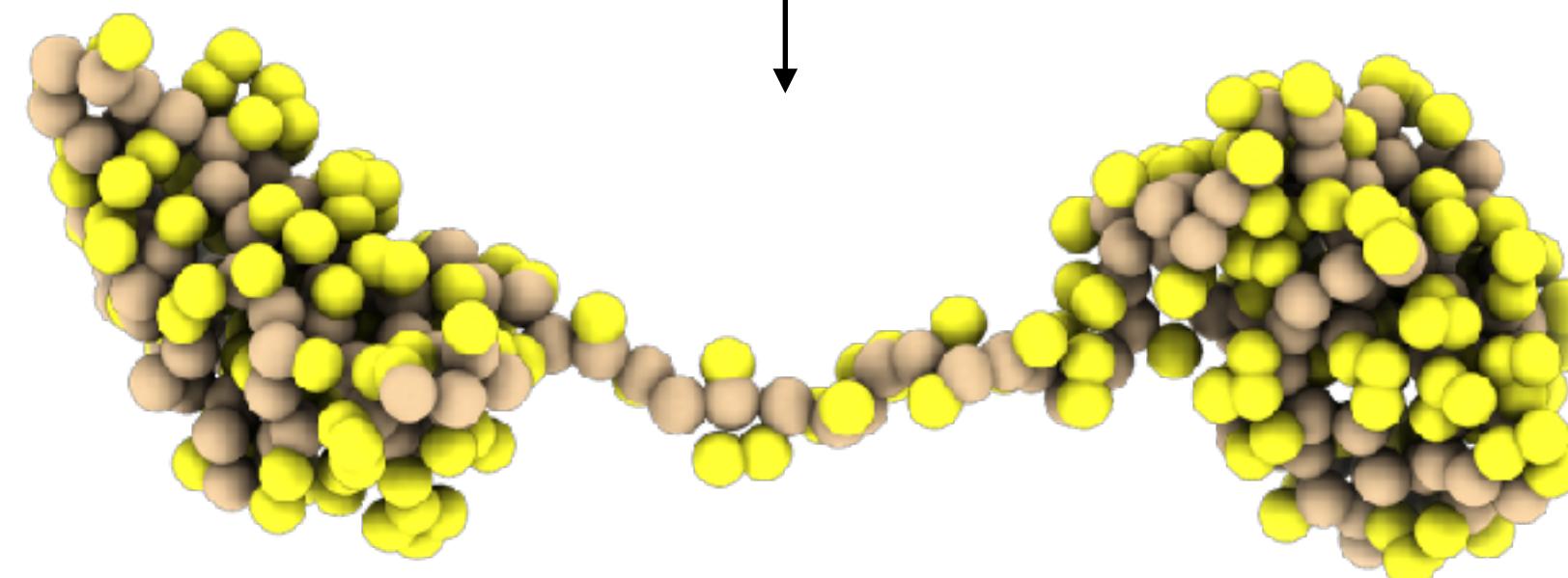
**folder: \$mstool/examples/Fig6\_TIA1**

**Martini simulation**

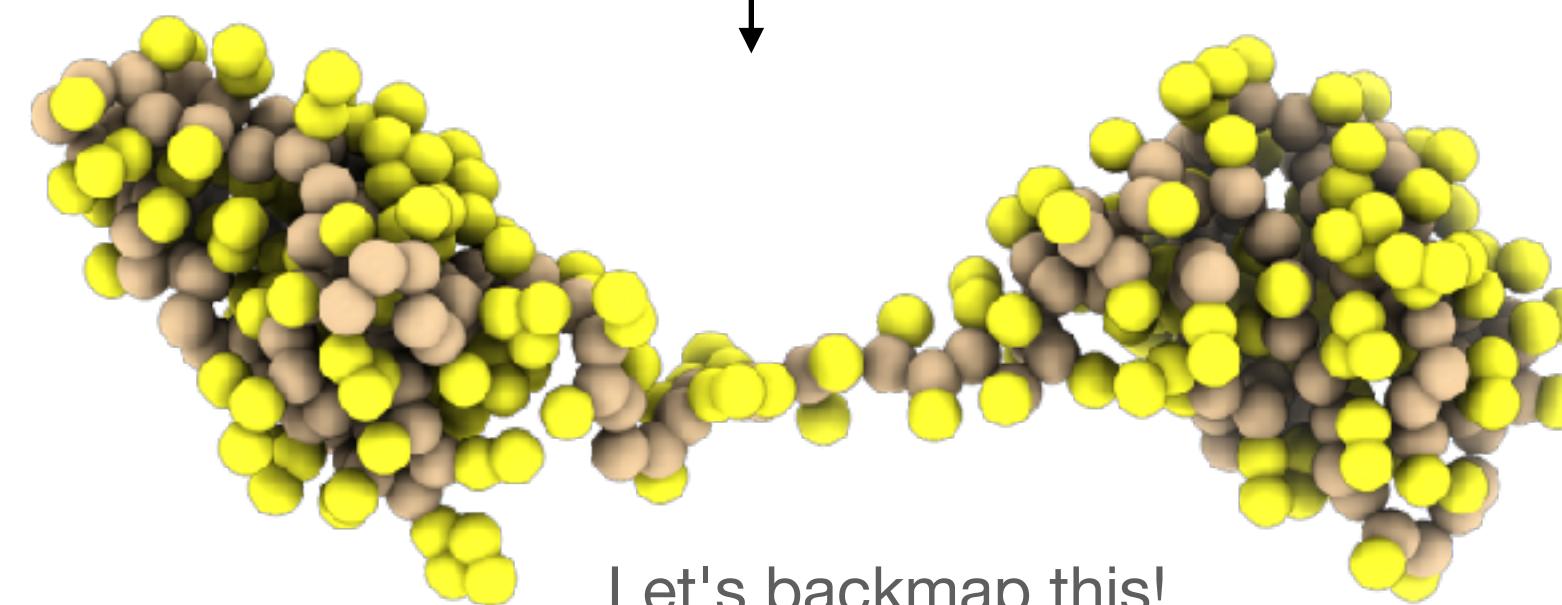
AA structure



Mapped CG structure



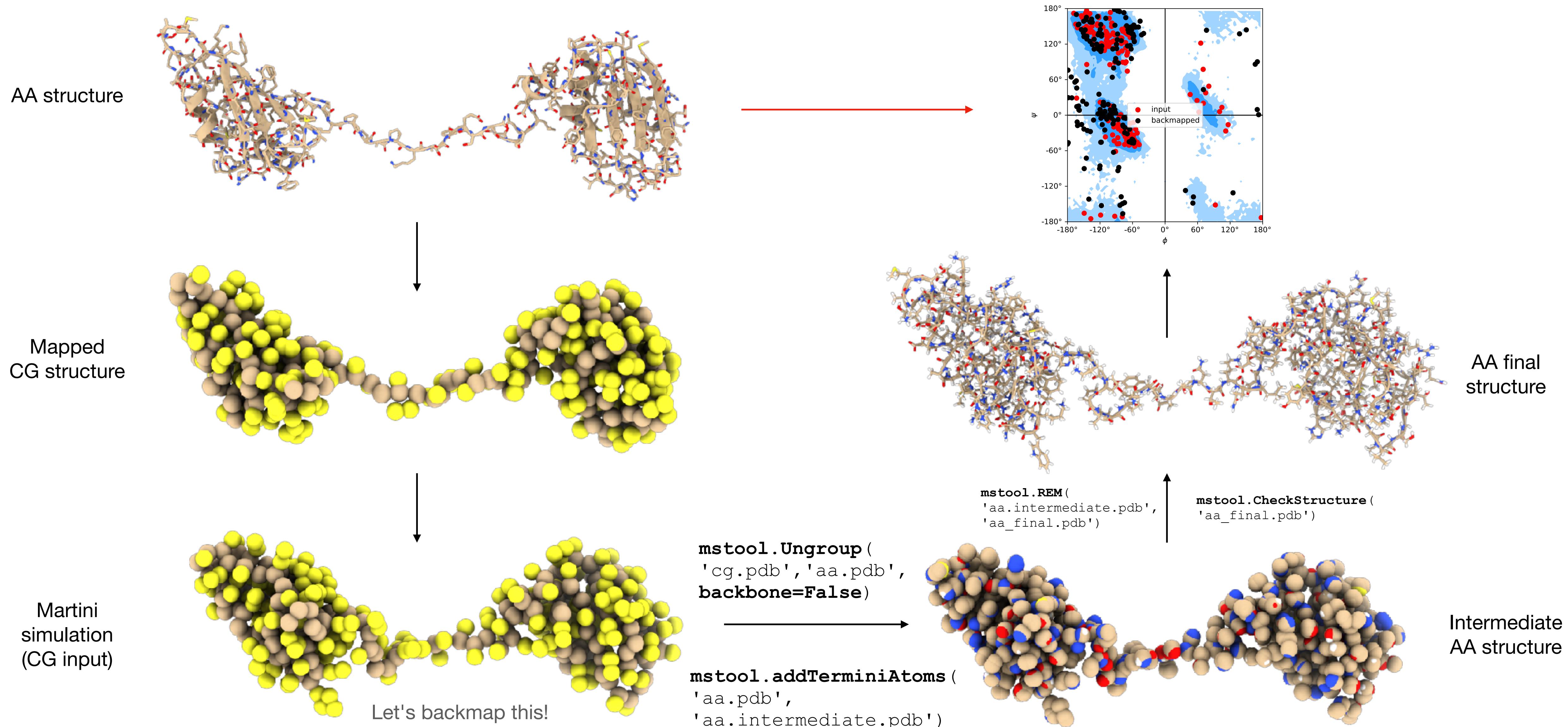
Martini  
simulation  
(CG input)



Let's backmap this!

folder: \$mstool/examples/Fig6\_TIA1/step4\_backmap\_B

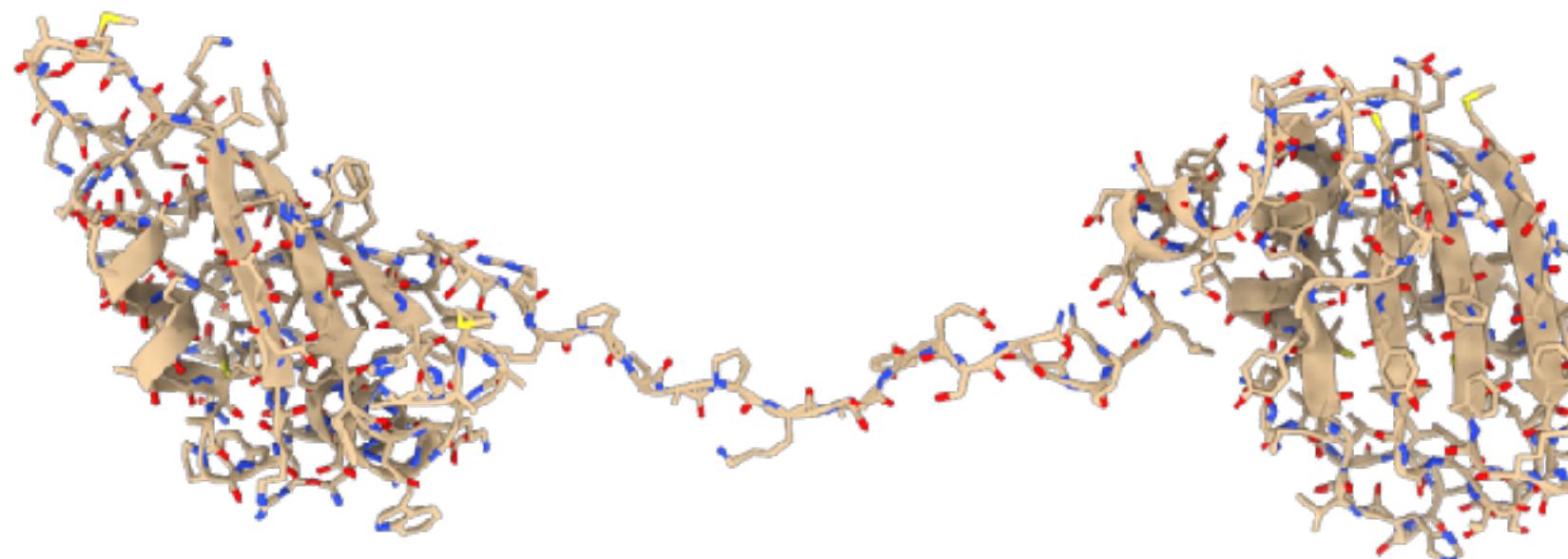
## Method 1: Randomly place atoms around their CG bead



folder: \$mstool/examples/Fig6\_TIA1/step4\_backmap\_C

## Method 2: Prebuild backbone using 3 consecutive CA atoms

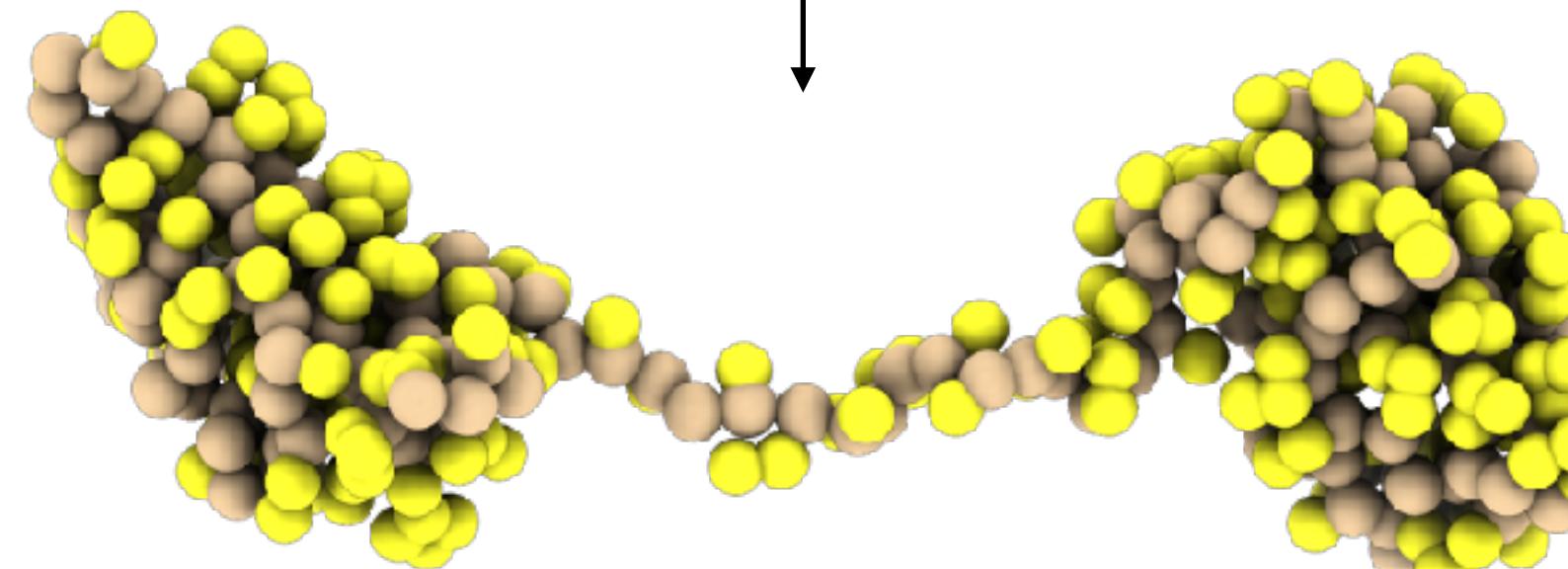
AA structure



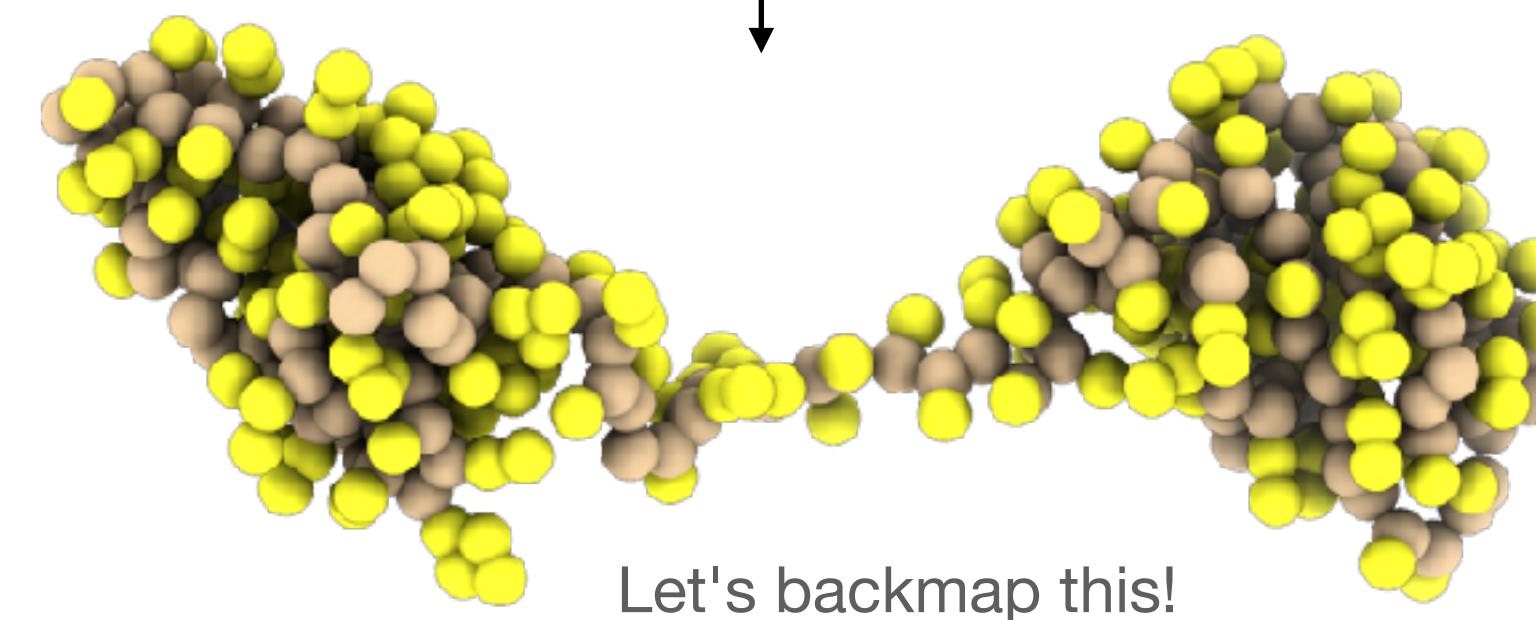
JCTC 2014, 10, 2, 676–690.

Bioinformatics 2016, 32, 8, 1235–1237.

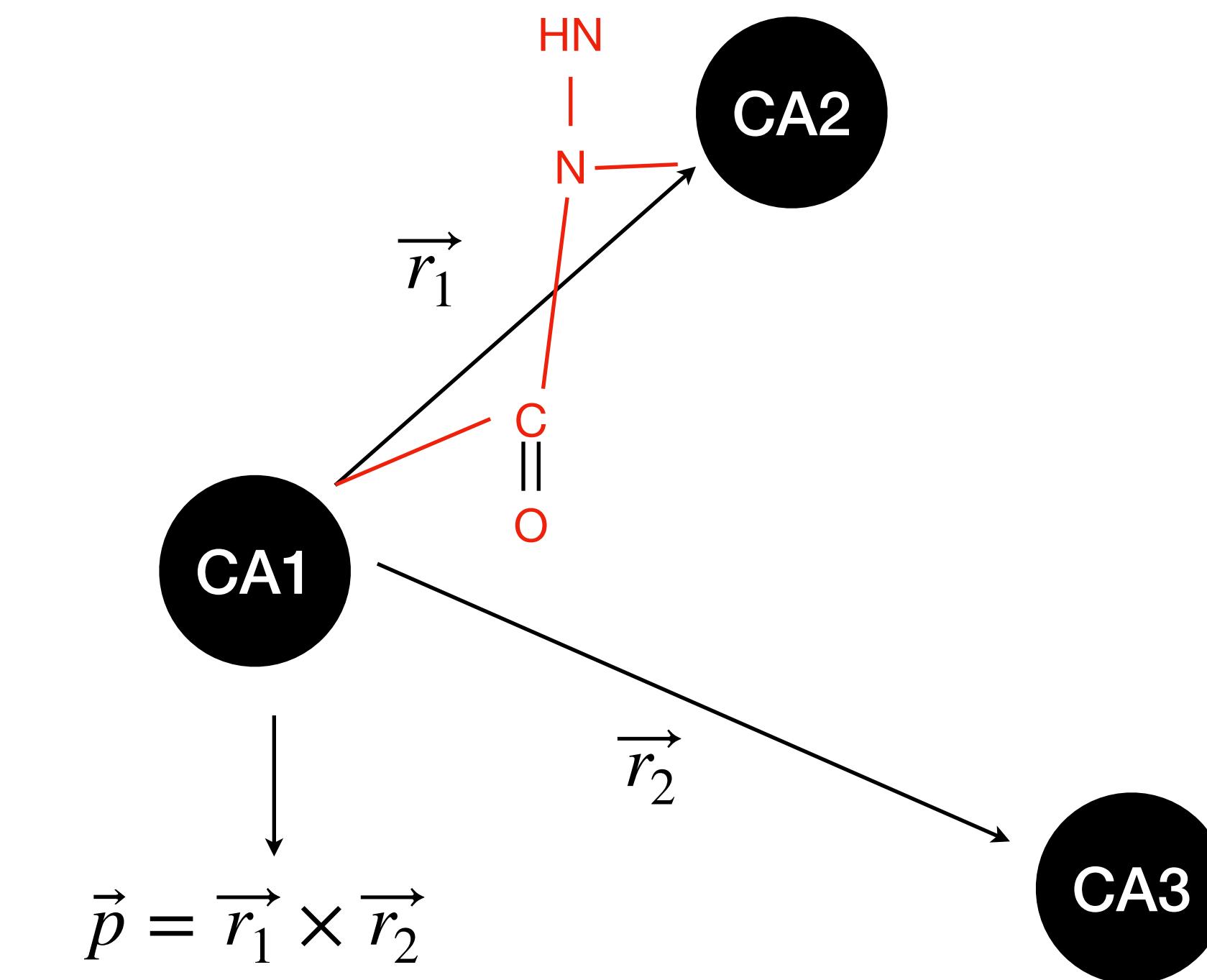
Mapped CG structure



Martini simulation  
(CG input)

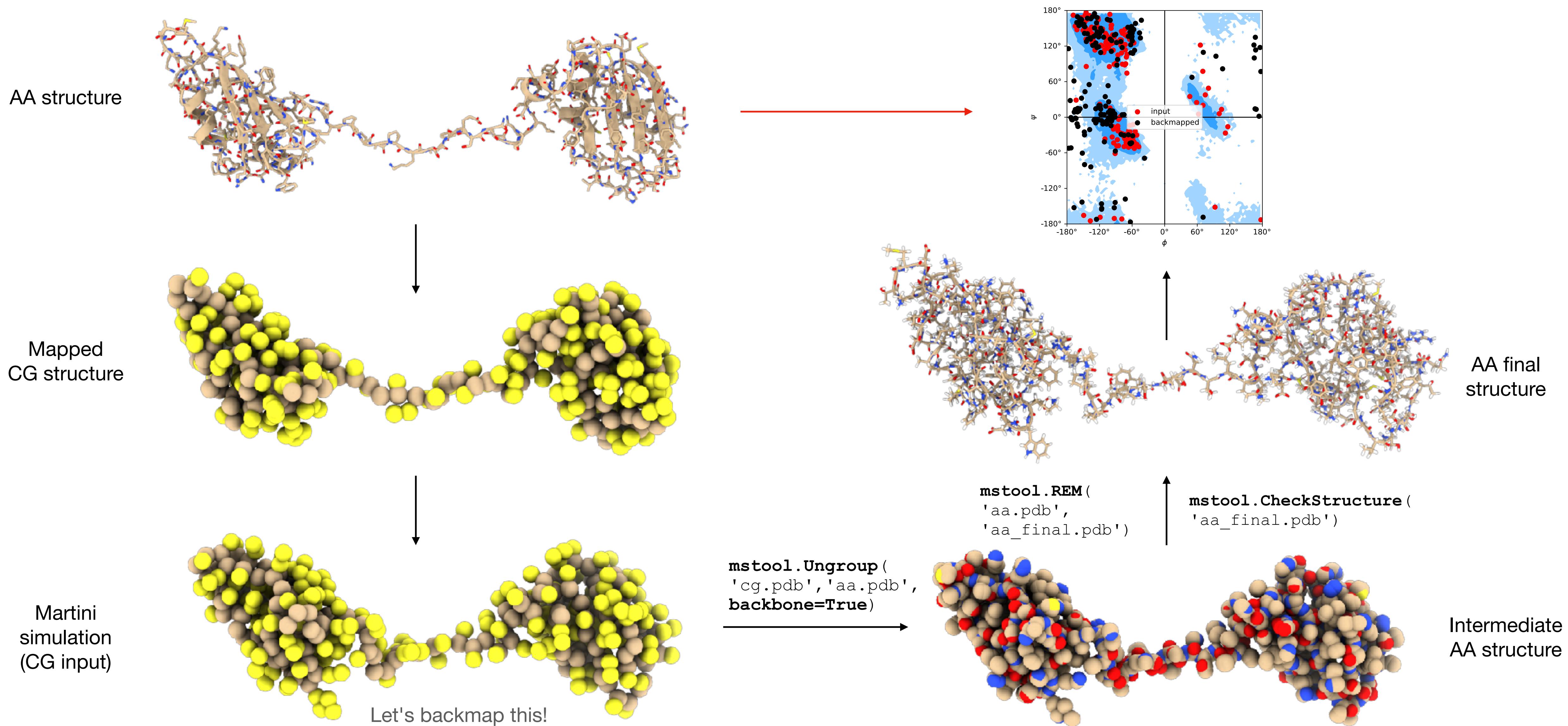


Let's backmap this!



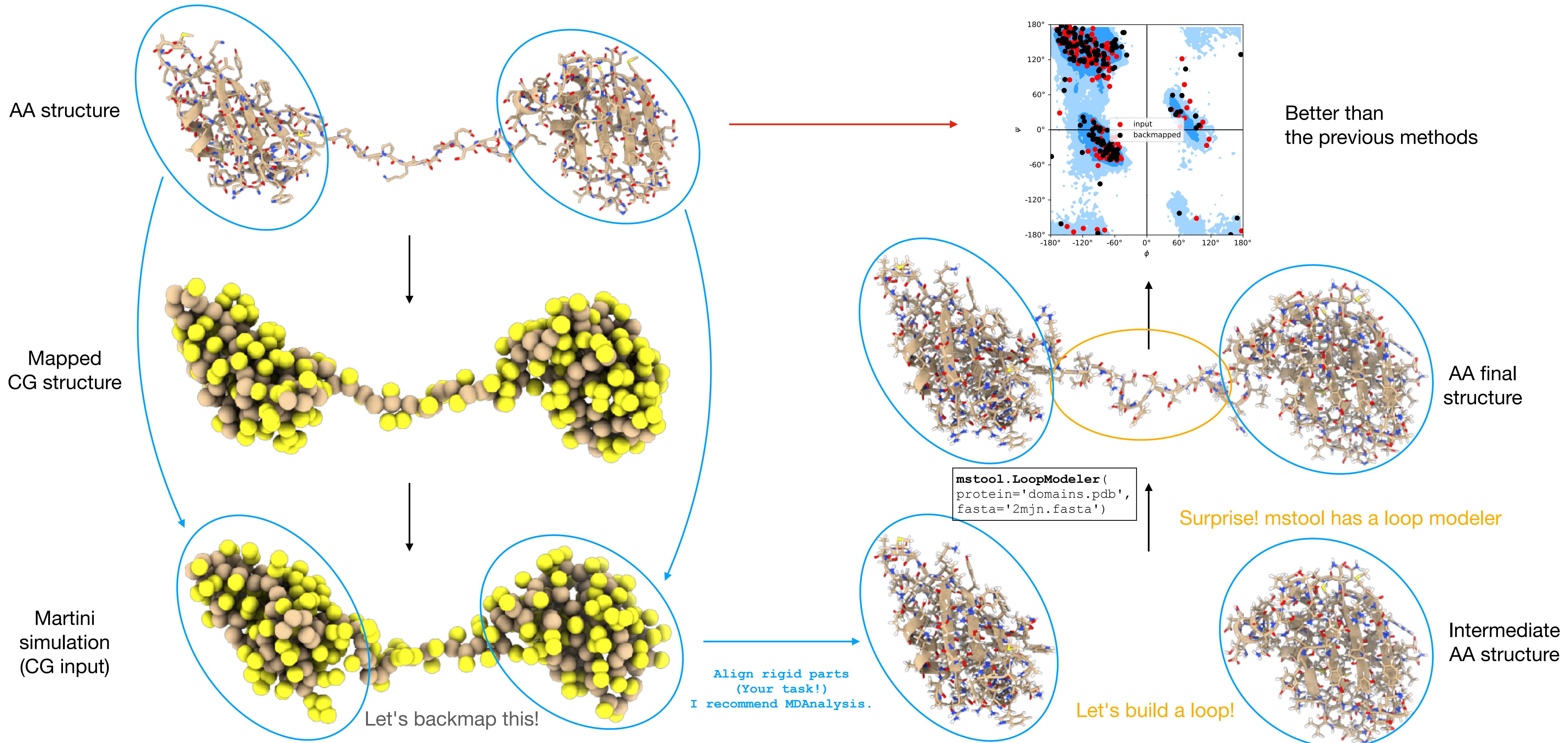
folder: \$mstool/examples/Fig6\_TIA1/step4\_backmap\_C

## Method 2: Prebuild backbone using 3 consecutive CA atoms



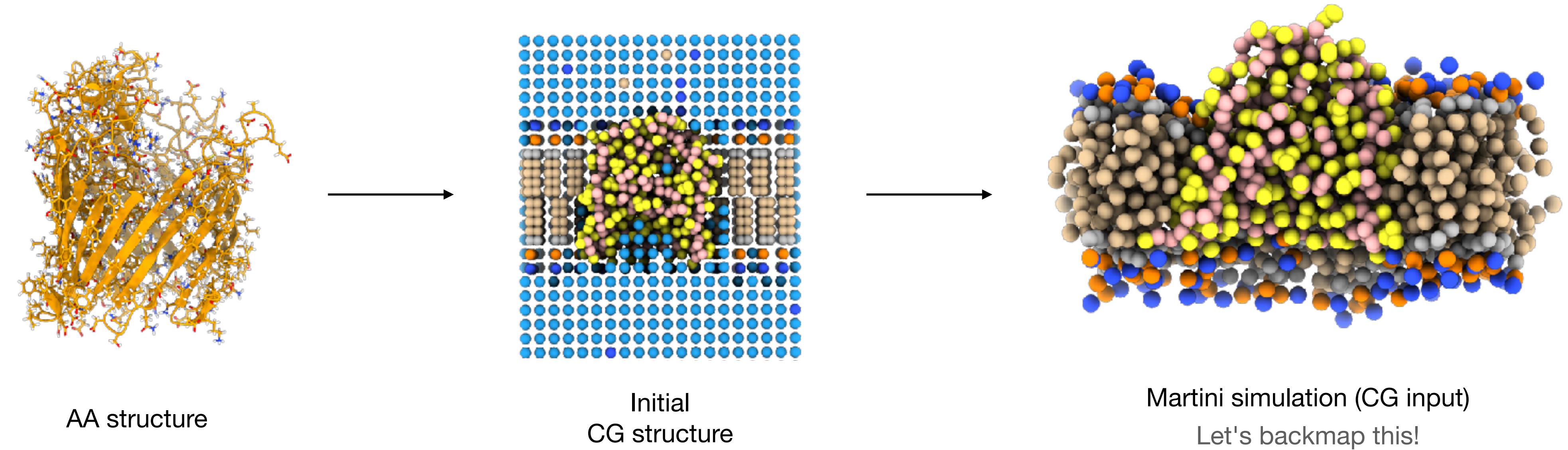
folder: \$mstool/examples/Fig6\_TIA1/step4\_backmap\_D

## Method 3: Use AA structure AS MUCH AS POSSIBLE



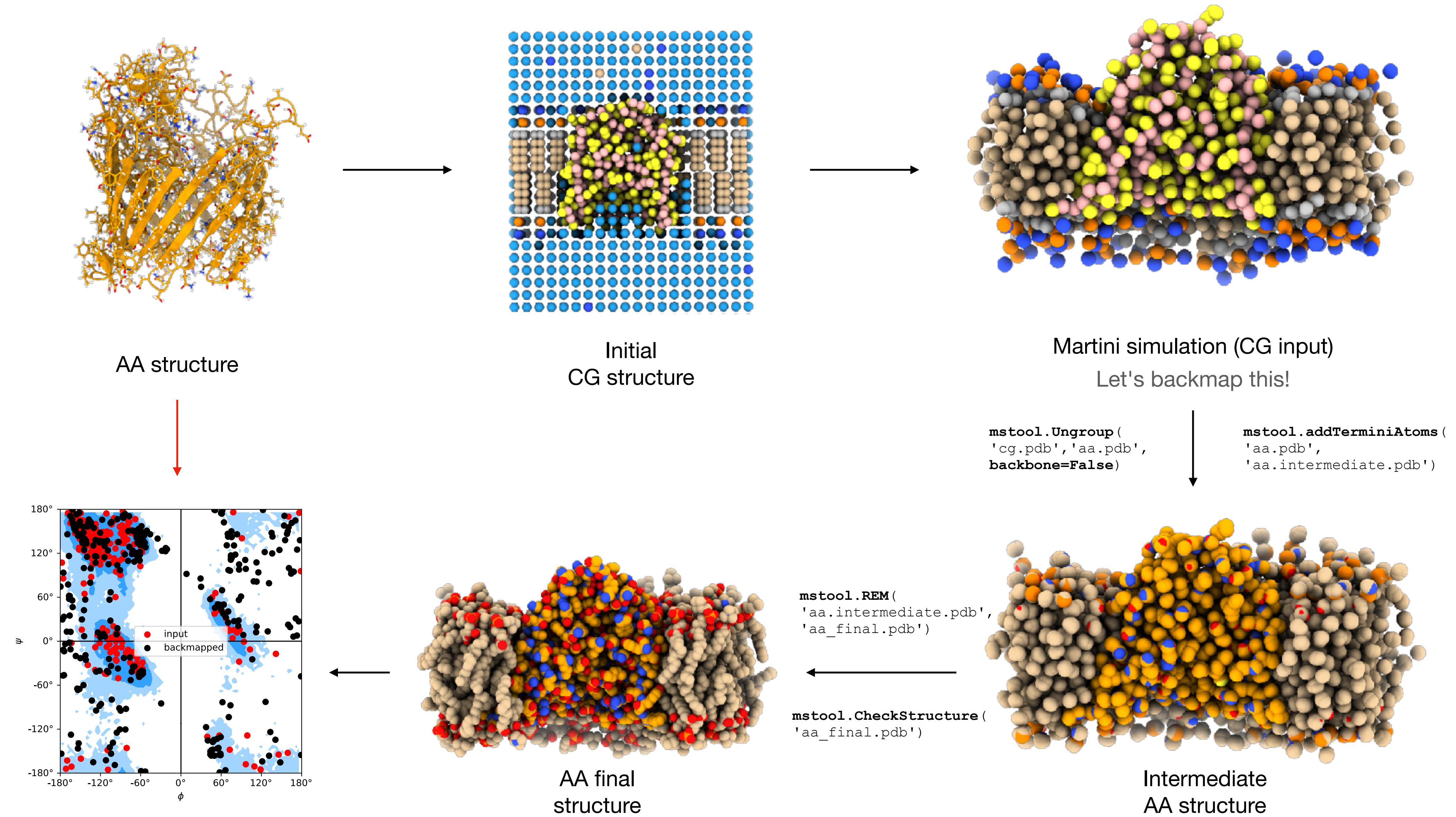
**folder: \$mstool/examples/Fig7\_ompF**

# Martini simulation



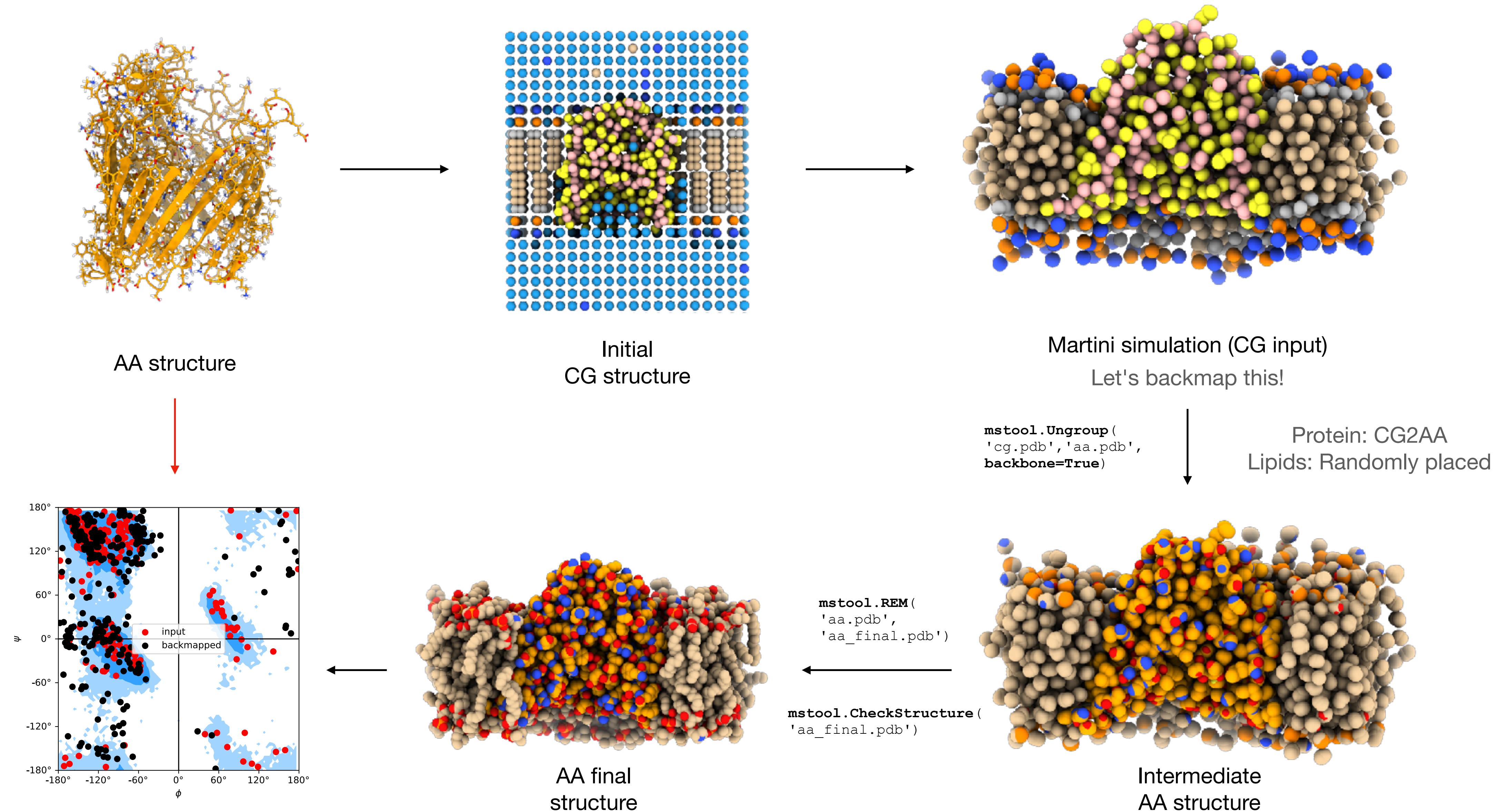
folder: \$mstool/examples/Fig7\_ompF/step4\_backmap\_B

## Method 1: Randomly place atoms around their CG bead



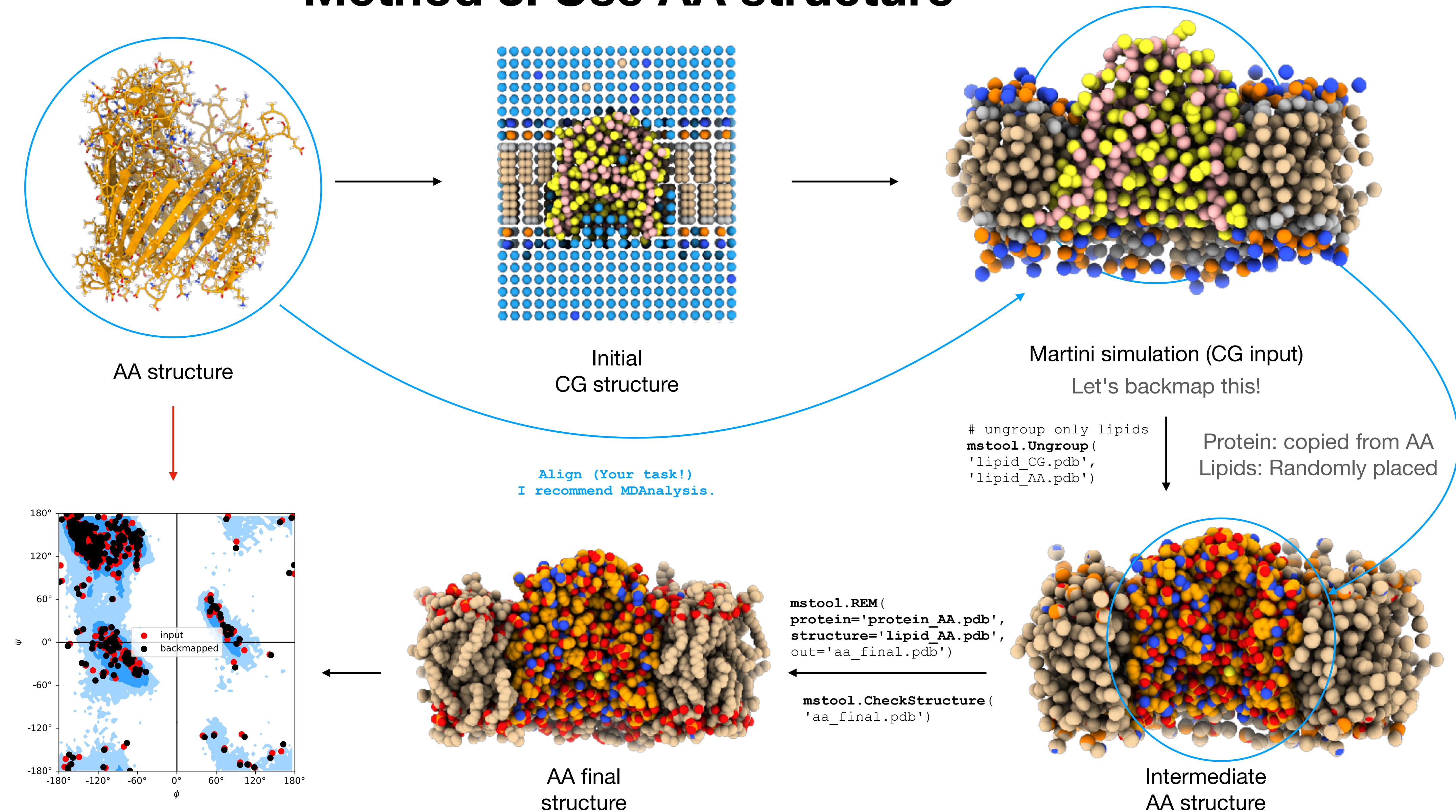
folder: \$mstool/examples/Fig7\_ompF/step4\_backmap\_C

## Method 2: Prebuild backbone using 3 consecutive CA atoms



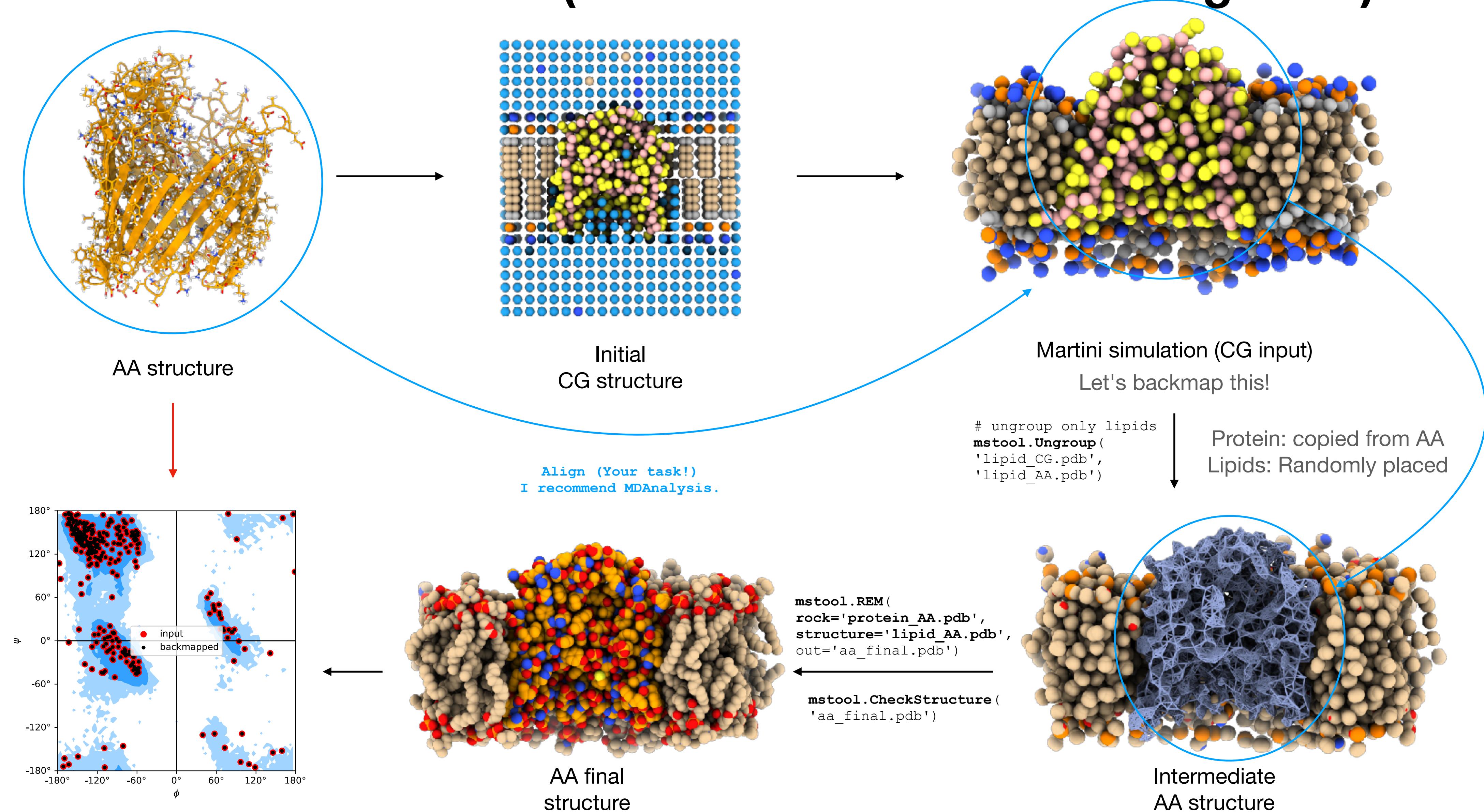
folder: \$mstool/examples/Fig7\_ompF/step4\_backmap\_D

## Method 3: Use AA structure

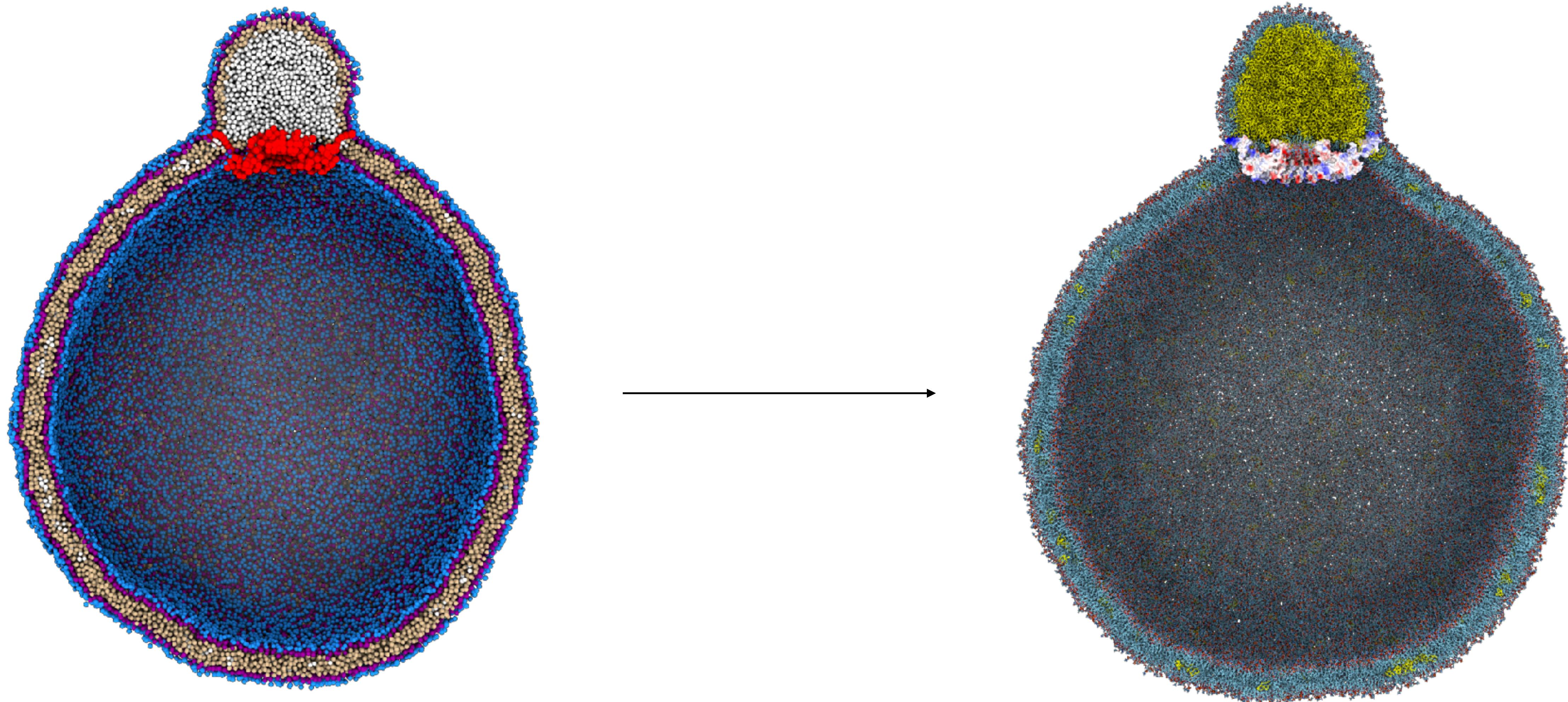


folder: \$mstool/examples/Fig7\_ompF/step4\_backmap\_EF

## Method 4: Use AA structure (when a residue is not recognized)



**folder: \$mstool/examples/Fig7\_seipin**  
**Heroic backmapping**



**CG input**  
(154,884 beads)

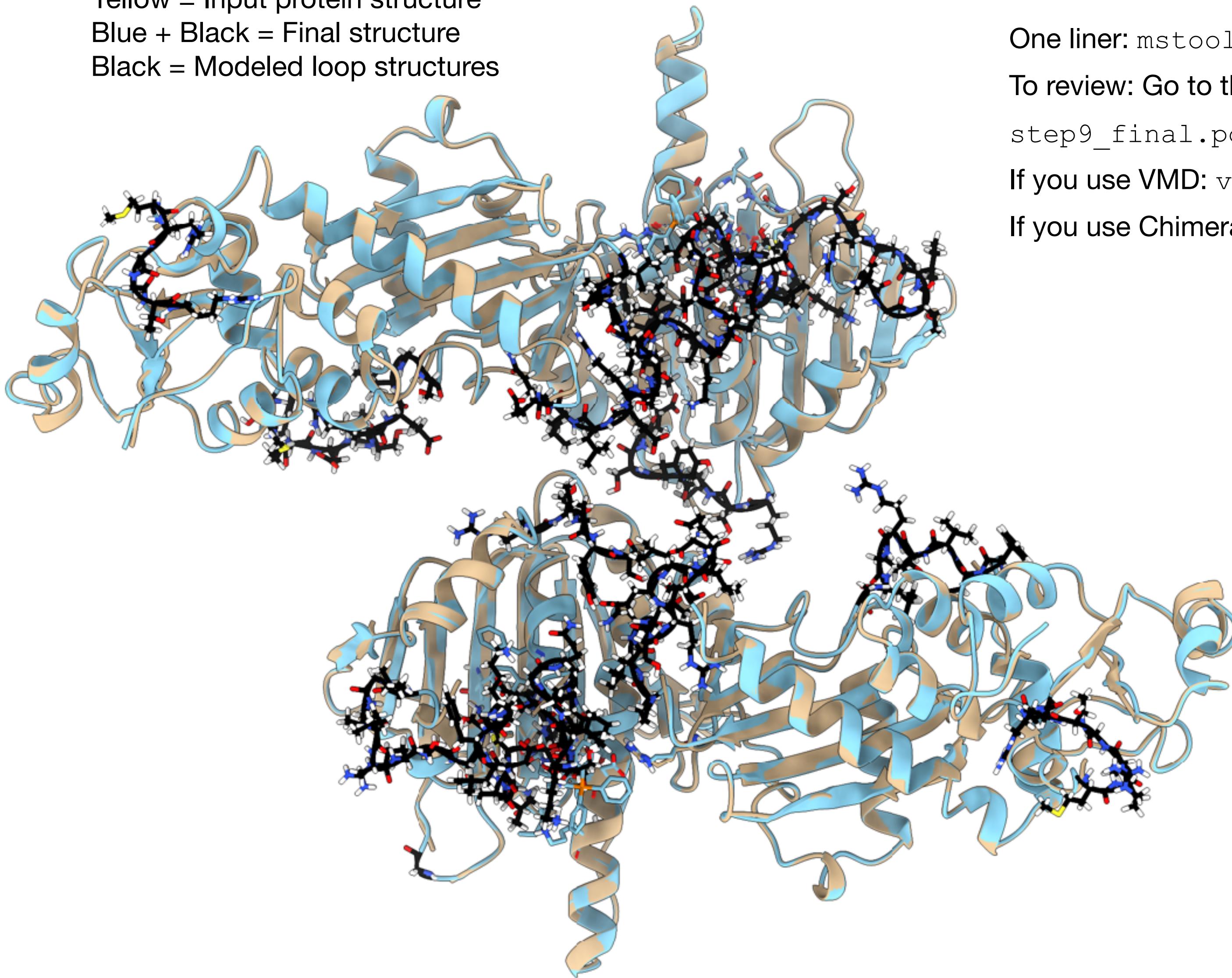
*Elife* 11 (2022): e75808

**AA final structure**  
(5,306,303 atoms)

# folder: \$mstool/examples/LoopModeler/7U8U

## LoopModeler Example

Yellow = Input protein structure  
Blue + Black = Final structure  
Black = Modeled loop structures



One liner: `mstool.LoopModeler(protein='7u8u.pdb', fasta='q12931.fasta')`  
To review: Go to the workdir (`$mstool/examples/LoopModeler/7U8U/workdir`)  
step9\_final.pdb is the final structure.  
If you use VMD: `vmd -e vis.tcl`  
If you use ChimeraX: `chimearx vis.cxc`

# folder: \$mstool/examples/LoopModeler/7U8U

## LoopModeler Example

Yellow = Input protein structure

Blue + Black = Final structure

Black = Modeled loop structures

One liner: `mstool.LoopModeler(protein='7u8u.pdb', fasta='q12931.fasta')`

SCALE1	0.015068	0.000000	0.000228	0.000000		
SCALE2	0.000000	0.008433	0.000000	0.000000		
SCALE3	0.000000	0.000000	0.014412	0.000000		
ATOM	1 N PRO A	70	8.795 27.391 -27.377	1.00 95.43	N	
ANISOU	1 N PRO A	70	12029 14345 9884 5329 2674 2129		N	
ATOM	2 CA PRO A	70	8.108 28.312 -26.453	1.00101.41	C	
ANISOU	2 CA PRO A	70	12806 14803 10923 4944 2408 2323		C	
ATOM	3 C PRO A	70	7.103 27.639 -25.506	1.00120.03	C	
ANISOU	3 C PRO A	70	15614 16633 13358 4886 2185 2078		C	
ATOM	4 O PRO A	70	6.011 27.279 -25.945	1.00118.35	O	
ANISOU	4 O PRO A	70	15726 16299 12941 4786 2163 1905		O	

↑  
Resid starts from 70

mstool will think the first amino acid  
in the fasta file has a resid of 1.

You have to provide two sequences in a single fasta file,  
as your PDB structure has two chains in this example.

The order of the chains in the fasta file should be  
consistent with the order in the structure file.

For instance, chain A appears first in the structure file.  
The first sequence in the fasta file should be about chain A

Resid = 1  
of the first chain

Resid = 70  
of the first chain

Resid = 1  
of the second chain

>sp|Q12931|TRAP1\_HUMAN Heat shock protein 75 kDa, mitochondrial  
1 PE=1 SV=3  
MARELALLWGRRLRPLLRAPALAAVPGGKPILCPRRTAQLGPRRNPAWSLQAGRLFS  
TQTAEDKEEPLHSIISSTESVQGSTSKHEFQAETKKLLDIVARSLYSEKEVFIRELISNA  
SDALEKLRHKLVSDGQALPEMEIHLQTNAEKGTTITIQDTGIGMTQEELVSNLGTIARSGS  
KAFLDALQNQAEASSKIIGQFGVGFYSAFMVADRVEVYSRSAAPGSLGYQWLSDGSGVFE  
IAEASGVRTGTTKIIHLKSDCKEFSSEARVRDVVTKYSNFVFPLYLNNGRRMNTLQAIWM  
MDPKDVREWQHEEFYRYVAQAHDKPRYTLHYKTDAVLNIRSIFYVPDMKPSMFDSRELG  
SSVALYSRKVLIQTKATDILPKWLRFIRGVVDSEIDPLNLRELLQESALIRKLRLDVQQ  
RLIKFFIDQSKKDAEKYAKFFEDYGLFMREGIVTATEQEVKEDIAKLLRYESSALPSQL  
TSLSEYASRMRAAGRTRNIYYLCAPNRH LAEHSPYYEAMKKDTEVLFCEQFDETLLLHRL  
EFDKKKLISVETDIVVDHYKEEKFEDRSPAAECLSEKETEELMAWMRNVLGSRTNVKVT  
LRLDTHPAMVTVLEMGAARHFLRMQQQLAKTQEERAQLLQPTLEINPRHALIKKLNLQRAS  
EPGLAQQLLVQIYENAMIAAGLVDDPRAMVGRNLNLVKALERH

>sp|Q12931|TRAP1\_HUMAN Heat shock protein 75 kDa, mitochondrial  
1 PE=1 SV=3  
MARELALLWGRRLRPLLRAPALAAVPGGKPILCPRRTAQLGPRRNPAWSLQAGRLFS  
TQTAEDKEEPLHSIISSTESVQGSTSKHEFQAETKKLLDIVARSLYSEKEVFIRELISNA  
SDALEKLRHKLVSDGQALPEMEIHLQTNAEKGTTITIQDTGIGMTQEELVSNLGTIARSGS  
KAFLDALQNQAEASSKIIGQFGVGFYSAFMVADRVEVYSRSAAPGSLGYQWLSDGSGVFE  
IAEASGVRTGTTKIIHLKSDCKEFSSEARVRDVVTKYSNFVFPLYLNNGRRMNTLQAIWM  
MDPKDVREWQHEEFYRYVAQAHDKPRYTLHYKTDAVLNIRSIFYVPDMKPSMFDSRELG  
SSVALYSRKVLIQTKATDILPKWLRFIRGVVDSEIDPLNLRELLQESALIRKLRLDVQQ  
RLIKFFIDQSKKDAEKYAKFFEDYGLFMREGIVTATEQEVKEDIAKLLRYESSALPSQL  
TSLSEYASRMRAAGRTRNIYYLCAPNRH LAEHSPYYEAMKKDTEVLFCEQFDETLLLHRL  
EFDKKKLISVETDIVVDHYKEEKFEDRSPAAECLSEKETEELMAWMRNVLGSRTNVKVT  
LRLDTHPAMVTVLEMGAARHFLRMQQQLAKTQEERAQLLQPTLEINPRHALIKKLNLQRAS  
EPGLAQQLLVQIYENAMIAAGLVDDPRAMVGRNLNLVKALERH

# folder: \$mstool/examples/LoopModeler/7U8U

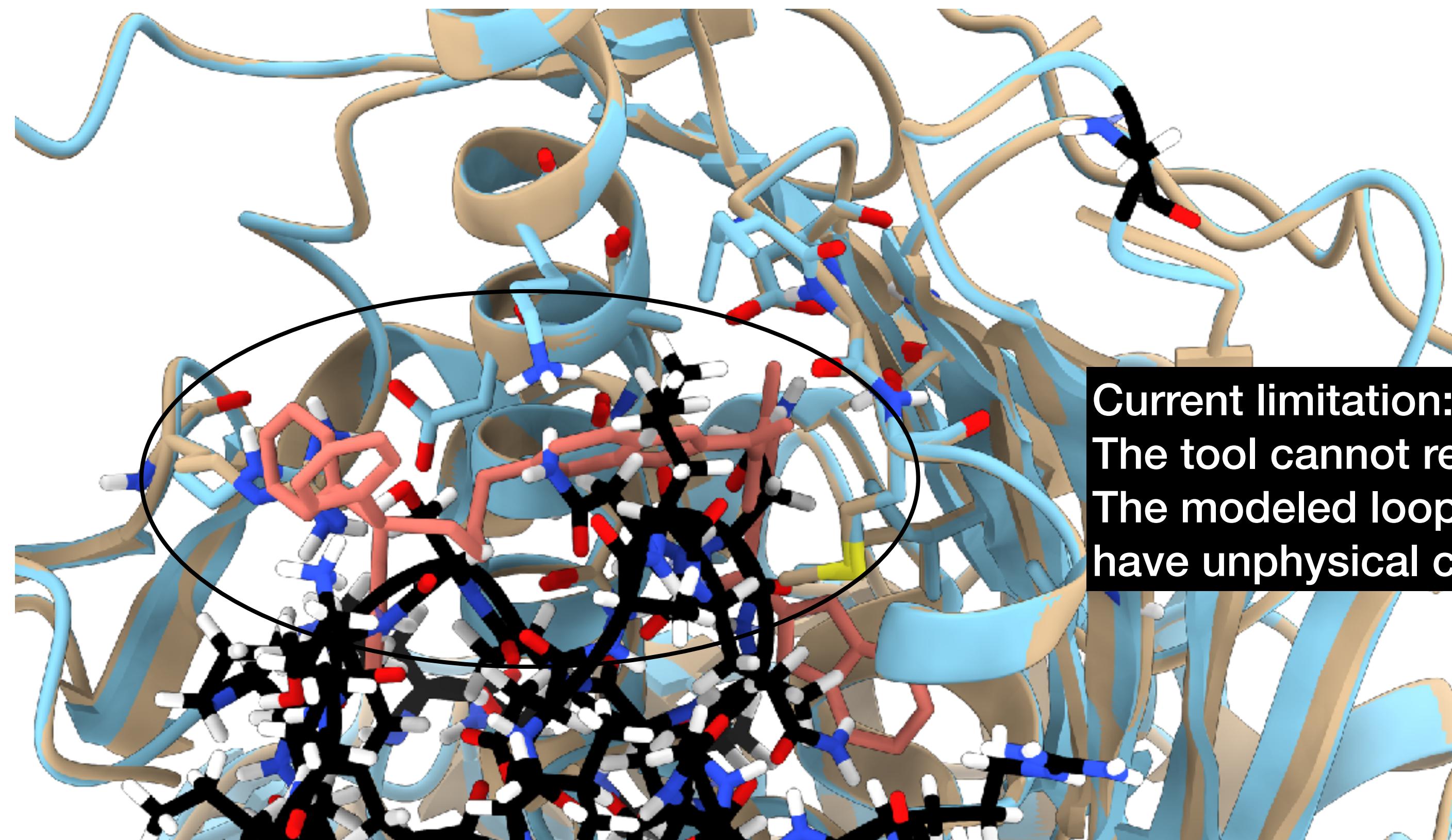
## LoopModeler Example

Yellow = Input protein structure

Blue + Black = Final structure

Black = Modeled loop structures

Salmon = Non-protein molecules



### Current limitation:

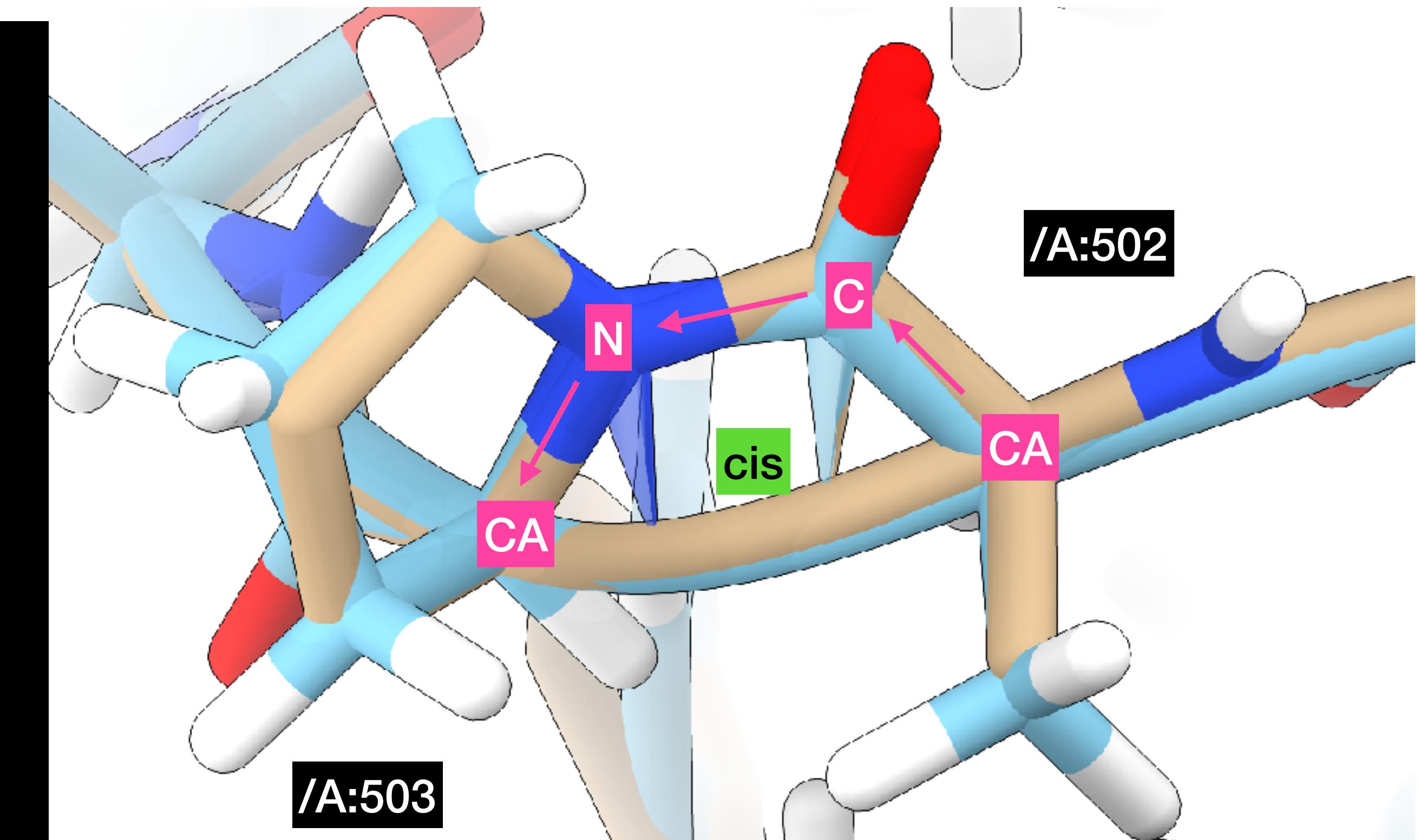
The tool cannot recognize non-protein residues (e.g., ligand).  
The modeled loop structures can possibly have unphysical contacts with non-protein residues.

# folder: \$mstool/examples/LoopModeler/7U8U

## LoopModeler Example

```
#####
workdir/step8_minimized.pdb was reviewed
#####

The following isomers were reviewed:
peptide bond cis/trans: chain A
peptide bond cis/trans: chain B
chiral: resname TYR - HA CA N CB C
chiral: resname CYS - HA CA N CB C
chiral: resname ALA - HA CA N CB C
chiral: resname VAL - HA CA N CB C
chiral: resname MET - HA CA N CB C
chiral: resname GLN - HA CA N CB C
chiral: resname GLU - HA CA N CB C
chiral: resname HIS - HA CA N CB C
chiral: resname ILE - HA CA N CB C
chiral: resname ILE - HB CB CG2 CG1 CA
chiral: resname ASN - HA CA N CB C
chiral: resname LYS - HA CA N CB C
chiral: resname THR - HA CA N CB C
chiral: resname THR - HB CB CG2 OG1 CA
chiral: resname TRP - HA CA N CB C
chiral: resname PRO - HA CA N CB C
chiral: resname ARG - HA CA N CB C
chiral: resname LEU - HA CA N CB C
chiral: resname ASP - HA CA N CB C
chiral: resname PHE - HA CA N CB C
chiral: resname SER - HA CA N CB C
#####
The following molecules had flipped isomers:
peptide bond cis/trans: (chain A and name C 0 and resid 502 and resname ALA) or (chain A and name CD N and resid 503 and resname PRO)
#####
```

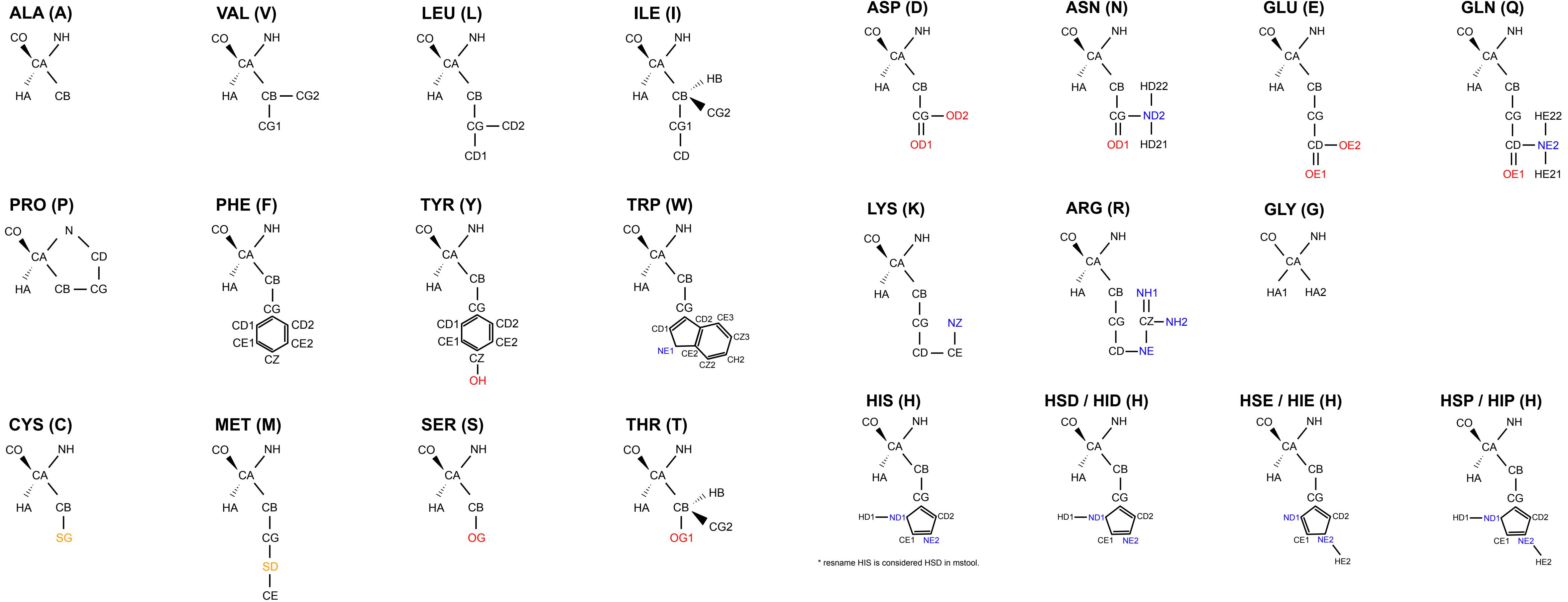


At the end of the step,  
you will likely see this warning.  
`/A:502,503 is not the loop that I built.`  
`This cis peptide bond is already included  
in the PDB file.`  
`Therefore, this warning is harmless.`

But if you see more than this one,  
it is probably my bad. Run it again.

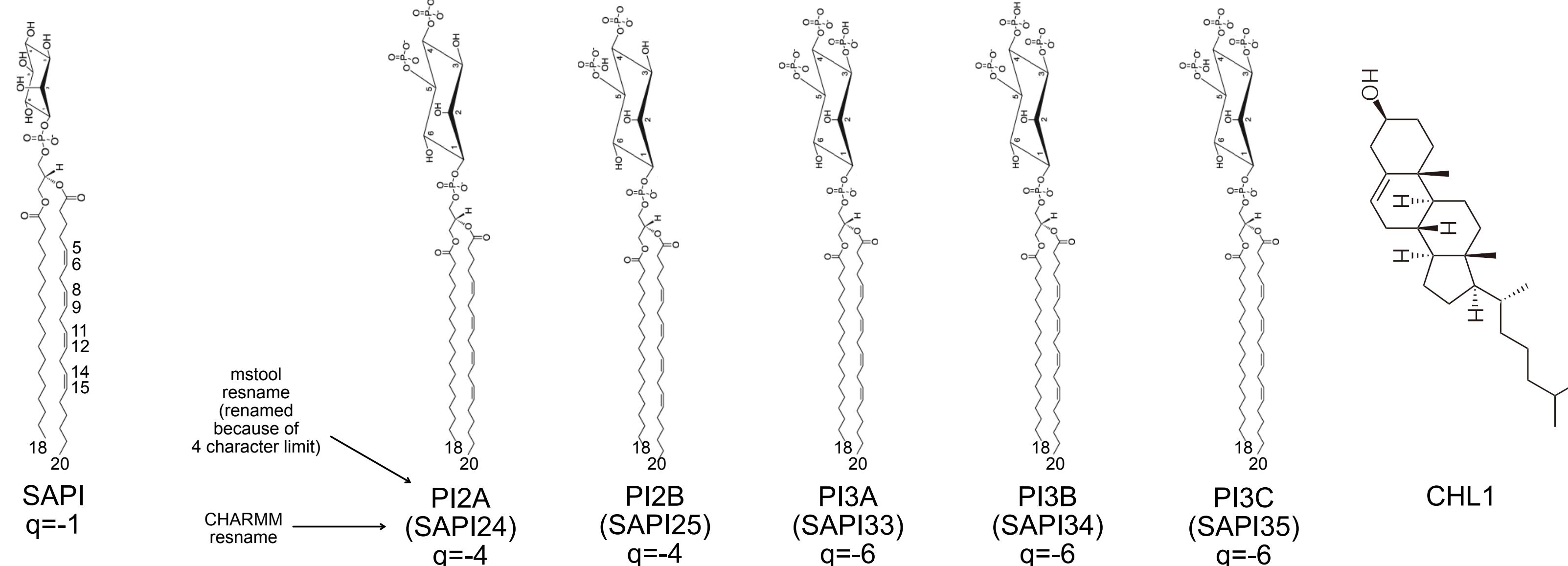
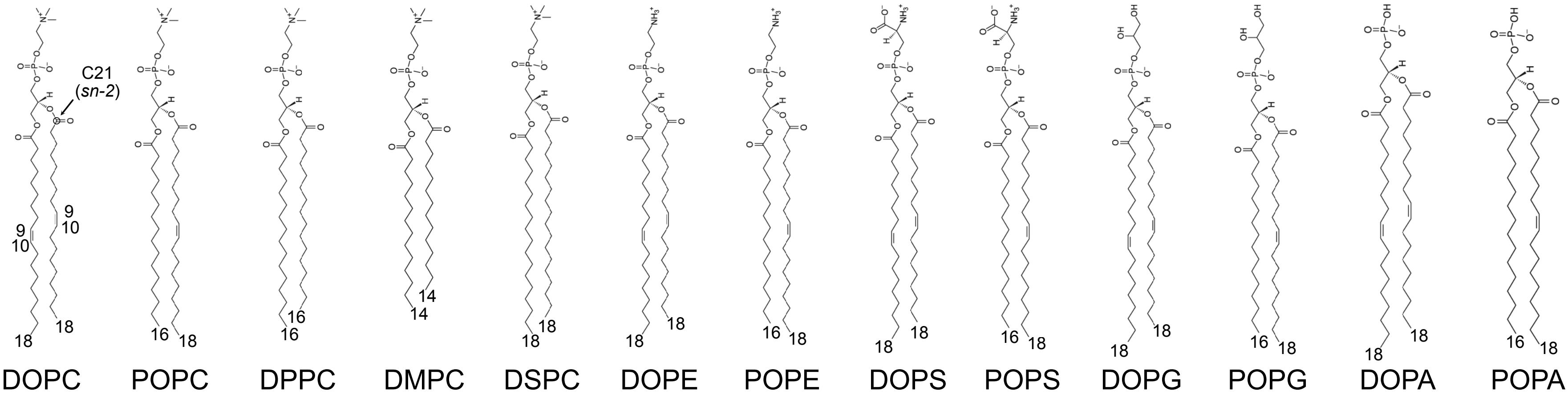
# Amino acids in default mapping schemes

\$mstool/mapping/martini.protein.c36m.dat (Martini resolution)



# Lipids in default mapping schemes

\$mstool/mapping/martini.lipid.c36.dat (Martini resolution)



# Notes and troubleshooting

- mstool can input/output pdb or dms .  
(no gro. sorry not sorry. gro has no chain field - use **MDAnalysis** to change a format).

 **Protein** topology is internally built by **openMM**.

If **mstool.REM** complains about protein (e.g., No template found for residue 1 (ALA)), you might have missing atoms (remember termini atoms!), non-consecutive resid, etc. Check whether you can do the following thing:

```
from openMM import *
pdb = PDBFile('protein_AA.pdb') #contains only protein atoms just for checking
forcefield = ForceField('charmm36.xml')
system = forcefield.createSystem(pdb.topology)
```

- To-do: **Nucleic acids** are not supported yet.  
If openMM can build a topology internally, it might be fine... I have not checked.
- Topology of **other molecules**, each containing only one resname and resid, which are non-protein and non-nucleic acid, non-polymer molecules, will be built by mstool based on their **resnames** and **atomic names**.  
(Yes, resnames and names are very important! - More in the next slide)

# Notes and troubleshooting



**resnames and names should be consistent between a structure file, mapping schemes, and openMM-compatible all-atom forcefield.**  
e.g., resname of cholesterol in Martini is CHOL. The resname of cholesterol in charmm36 is CHL1.  
Change CHOL to CHL1 in your CG structure.

CG names should be consistent

```
ATOM 1 ROH CHL1 1 -42.060 -41.080 17.020 1.00 0.00
ATOM 2 R1 CHL1 1 -42.070 -40.380 14.390 1.00 0.00
ATOM 3 R2 CHL1 1 -40.270 -40.330 12.480 1.00 0.00
ATOM 4 R3 CHL1 1 -42.280 -41.440 11.000 1.00 0.00
ATOM 5 R4 CHL1 1 -40.290 -40.550 9.040 1.00 0.00
ATOM 6 R5 CHL1 1 -42.110 -40.200 9.290 1.00 0.00
ATOM 7 C1 CHL1 1 -42.200 -41.560 5.620 1.00 0.00
ATOM 8 C2 CHL1 1 -39.500 -40.830 3.140 1.00 0.00
```

Your structure file

resname should be consistent and should be equal to less than 4 characters because of PDB formatting issue

RESI CHL1

```
[ ROH 1
C3 H3 O3 H3'
C2 H2A H2B
C4 H4A H4B
[ R1 ]
C1 H1A H1B
C6 H6
C19 H19A H19B H19C
C10
...
...
```

<Residue name="CHL1">

```
<Atom charge="0.14" name="C3" type="CTL1"/>
<Atom charge="-0.66" name="O3" type="OHL"/>
<Atom charge="0.43" name="H3'" type="HOL"/>
...
```

1. AA names should be consistent

2. All the atoms must be defined - and defined only once in a mapping scheme (no duplicate please).

\$mstool/mapping/martini.lipid.c36.dat

\$mstool/FF/charmm36/charmm36.xml



**Each residue should have a unique (resname, resid, chain) in a structure file**

This is often not the case if your system is very big so that resid (> 9999) is chopped off in a pdb file.

The solution would be changing chain or making a dms file.

```
ATOM 1 ROH CHL1X 1 -42.060 -41.080 17.020 1.00 0.00
ATOM 2 R1 CHL1X 1 -42.070 -40.380 14.390 1.00 0.00
ATOM 3 R2 CHL1X 1 -40.270 -40.330 12.480 1.00 0.00
ATOM 4 R3 CHL1X 1 -42.280 -41.440 11.000 1.00 0.00
ATOM 5 R4 CHL1X 1 -40.290 -40.550 9.040 1.00 0.00
ATOM 6 R5 CHL1X 1 -42.110 -40.200 9.290 1.00 0.00
ATOM 7 C1 CHL1X 1 -42.200 -41.560 5.620 1.00 0.00
ATOM 8 C2 CHL1X 1 -39.500 -40.830 3.140 1.00 0.00
ATOM 9 ROH CHL1X 2 -42.690 -35.990 16.130 1.00 0.00
ATOM 10 R1 CHL1X 2 -42.870 -36.060 13.310 1.00 0.00
ATOM 11 R2 CHL1X 2 -44.710 -37.290 11.850 1.00 0.00
ATOM 12 R3 CHL1X 2 -44.710 -34.890 10.550 1.00 0.00
ATOM 13 R4 CHL1X 2 -45.980 -36.780 8.690 1.00 0.00
ATOM 14 R5 CHL1X 2 -44.370 -35.480 8.410 1.00 0.00
ATOM 15 C1 CHL1X 2 -46.560 -34.350 5.500 1.00 0.00
ATOM 16 C2 CHL1X 2 -45.870 -32.930 2.950 1.00 0.00
```

Good - Two molecules have the same resname (CHL1) and chain (X) but a distinct resid (1, 2)

```
ATOM 1 ROH CHL1X 1 -42.060 -41.080 17.020 1.00 0.00
ATOM 2 R1 CHL1X 1 -42.070 -40.380 14.390 1.00 0.00
ATOM 3 R2 CHL1X 1 -40.270 -40.330 12.480 1.00 0.00
ATOM 4 R3 CHL1X 1 -42.280 -41.440 11.000 1.00 0.00
ATOM 5 R4 CHL1X 1 -40.290 -40.550 9.040 1.00 0.00
ATOM 6 R5 CHL1X 1 -42.110 -40.200 9.290 1.00 0.00
ATOM 7 C1 CHL1X 1 -42.200 -41.560 5.620 1.00 0.00
ATOM 8 C2 CHL1X 1 -39.500 -40.830 3.140 1.00 0.00
ATOM 9 ROH CHL1X 1 -42.690 -35.990 16.130 1.00 0.00
ATOM 10 R1 CHL1X 1 -42.870 -36.060 13.310 1.00 0.00
ATOM 11 R2 CHL1X 1 -44.710 -37.290 11.850 1.00 0.00
ATOM 12 R3 CHL1X 1 -44.710 -34.890 10.550 1.00 0.00
ATOM 13 R4 CHL1X 1 -45.980 -36.780 8.690 1.00 0.00
ATOM 14 R5 CHL1X 1 -44.370 -35.480 8.410 1.00 0.00
ATOM 15 C1 CHL1X 1 -46.560 -34.350 5.500 1.00 0.00
ATOM 16 C2 CHL1X 1 -45.870 -32.930 2.950 1.00 0.00
```

Bad - Two molecules have the same resname (CHL1), chain (X), and resid (1)

```
ATOM 1 ROH CHL1X 1 -42.060 -41.080 17.020 1.00 0.00
ATOM 2 R1 CHL1X 1 -42.070 -40.380 14.390 1.00 0.00
ATOM 3 R2 CHL1X 1 -40.270 -40.330 12.480 1.00 0.00
ATOM 4 R3 CHL1X 1 -42.280 -41.440 11.000 1.00 0.00
ATOM 5 R4 CHL1X 1 -40.290 -40.550 9.040 1.00 0.00
ATOM 6 R5 CHL1X 1 -42.110 -40.200 9.290 1.00 0.00
ATOM 7 C1 CHL1X 1 -42.200 -41.560 5.620 1.00 0.00
ATOM 8 C2 CHL1X 1 -39.500 -40.830 3.140 1.00 0.00
ATOM 9 NC3 POPCX 1 -30.850 35.370 23.530 1.00 0.00
ATOM 10 PO4 POPCX 1 -30.570 31.820 21.260 1.00 0.00
ATOM 11 GL1 POPCX 1 -29.350 35.520 18.720 1.00 0.00
ATOM 12 GL2 POPCX 1 -26.120 36.540 19.300 1.00 0.00
ATOM 13 C1A POPCX 1 -31.610 37.540 16.020 1.00 0.00
ATOM 14 D2A POPCX 1 -34.940 39.180 13.670 1.00 0.00
ATOM 15 C3A POPCX 1 -34.390 39.260 9.030 1.00 0.00
ATOM 16 C4A POPCX 1 -34.720 35.830 6.650 1.00 0.00
ATOM 17 C1B POPCX 1 -27.570 39.410 14.880 1.00 0.00
ATOM 18 C2B POPCX 1 -28.950 39.190 10.570 1.00 0.00
ATOM 19 C3B POPCX 1 -30.150 36.410 8.310 1.00 0.00
ATOM 20 C4B POPCX 1 -30.330 33.090 6.560 1.00 0.00
```

Acceptable (openMM will complain but this is harmless) -  
Two molecules have the same chain (X) and resid (1)  
but a distinct resname (CHL1 and POPC)

# Notes on installation

- The following python (>=3.7) packages are required for mstool.
  - **openMM**
    - To install openMM: conda install -c conda-forge openmm
    - It is ok if you are not familiar with openMM. Users do not directly interact with openMM.
  - **cython**
    - To install cython: conda install cython
  - **pandas, numpy, sqlite3** <--- I think you probably have them already installed.
- To install mstool:

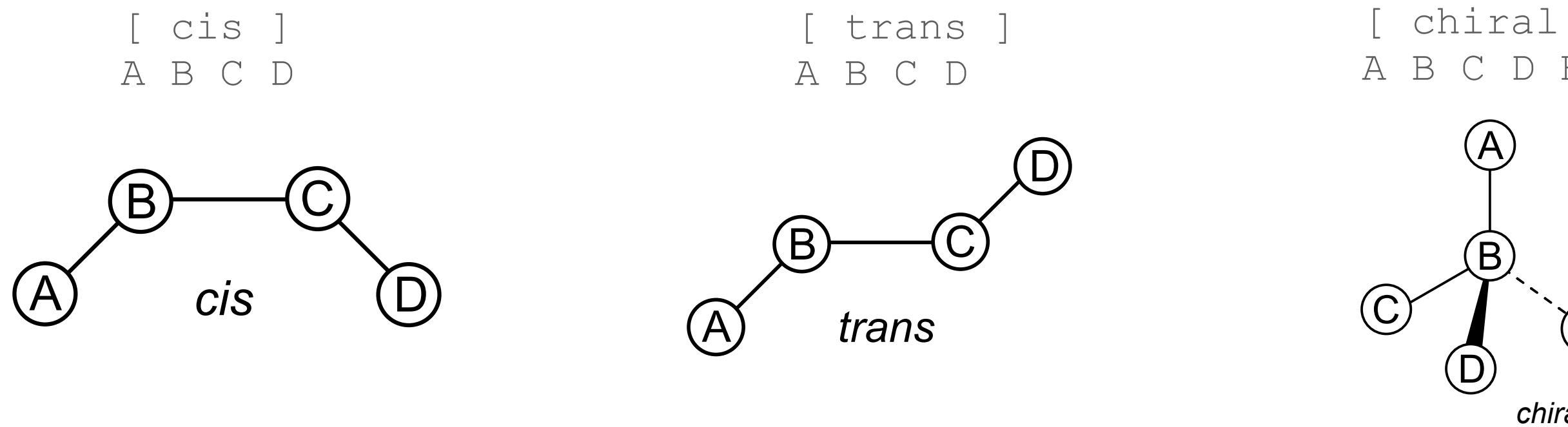
```
cd ~ (if you want to download a package into the home directory)
git clone https://github.com/ksy141/mstool.git
cd mstool
sh install.sh
```

# Notes on mapping files

- The mapping schemes are required for `mstool.Backmap`, `mstool.REM`, and `mstool.CheckStructure`.
- The following mapping files are tailored toward martini forcefield and read by default in `mstool.Backmap`, `mstool.REM`, and `mstool.CheckStructure`.
  - `$mstool/mapping/martini.protein.c36m.dat`
  - `$mstool/mapping/martini.lipid.c36.dat`
- The default mapping files contain the mapping schemes and isomeric properties (cis/trans/chiral) of the following molecules.
  - GLY, ALA, VAL, ILE, LEU, MET, SER, THR, ASN, GLN, CYS, ARG, LYS, ASP, GLU, PRO, PHE, TYR, TRP, HIS
  - POPC, DPPC, POPS, DPPC, POPS, DOPS, DOPG, POPG, DOPA, POPA, DOPC, DOPE, POPE, DSPC, DMPC, CHL1
- If you want to add a new molecule, you can make a new mapping file and pass it to mstool. The read mapping files will be **the default files (above) + the new files** that you provide.
  - `mstool.Backmap(..., mapping_add=['TRIO.dat', 'CHYO.dat'])`
  - `mstool.REM(..., mapping_add=['TRIO.dat', 'CHYO.dat'])`
  - `mstool.CheckStructure(..., mapping_add=['TRIO.dat', 'CHYO.dat'])`
- If your system is not at martini resolution, the default mapping schemes (tailored toward Martini forcefield) will not be very meaningful to you. The default mapping schemes will be overridden if you provide anything to mapping. The read mapping files will be only **the files that you provide**.
  - `mstool.Backmap(..., mapping=['mapping.dat'])`
  - `mstool.REM(..., mapping=['mapping.dat'])`
  - `mstool.CheckStructure(..., mapping=['mapping.dat'])`

# Notes on mapping files

- Format of a mapping scheme
  - Each molecule should have a resname defined next to RESI.
  - Each CG bead name should be written in a bracket [ ].
  - Each AA atom that belongs to its corresponding CG bead should be defined under the CG bead name.
  - The orders do not matter - the order of CG beads, the order of AA atoms, the order of brackets [ ].
  - If a molecule has isomeric properties, define them in [cis], [trans], or [chiral].
  - A resname of a molecule in a mapping file should be consistent with a resname in your CG structure file and openMM-compatible FF files. (e.g., \$mstool/FF/charmm36/charmm36.xml)
  - CG bead name in a mapping scheme should be consistent with a name in your CG structure file.
  - AA name defined in a mapping scheme should be consistent with openMM-compatible FF files. (e.g., \$mstool/FF/charmm36/charmm36.xml)



RESI DOPC

resname

[ NC3 ] ← CG bead name  
N C12 H12A H12B  
C13 H13A H13B H13C  
C14 H14A H14B H14C  
C15 H15A H15B H15C

AA name

[ PO4 ]  
C11 H11A H11B  
P O13 O14 O12 O11  
C1 HA HB

[ GL1 ]  
C2 HS O21 C21 O22 C22 H2R H2S

[ GL2 ]  
C3 HX HY O31 C31 O32 C32 H2X H2Y

[ C1A ]  
C23 H3R H3S C24 H4R H4S C25 H5R H5S C26 H6R H6S

[ D2A ]  
C27 H7R H7S C28 H8R H8S C29 H9R C210 H10R C211 H11R H11S

[ C3A ]  
C212 H12R H12S C213 H13R H13S C214 H14R H14S

[ C4A ]  
C215 H15R H15S C216 H16R H16S C217 H17R H17S C218 H18R H18S H18T

[ C1B ]  
C33 H3X H3Y C34 H4X H4Y C35 H5X H5Y C36 H6X H6Y

[ D2B ]  
C37 H7X H7Y C38 H8X H8Y C39 H9X C310 H10X C311 H11X H11Y

[ C3B ]  
C312 H12X H12Y C313 H13X H13Y C314 H14X H14Y

[ C4B ]  
C315 H15X H15Y C316 H16X H16Y C317 H17X H17Y C318 H18X H18Y H18Z

[ cis ]  
C28 C29 C210 C211  
C38 C39 C310 C311

[ chiral ]  
HS C2 O21 C1 C3  
O13 P O11 O14 O12

Isomeric properties

# Notes

- Supra CG water models are supported (several water molecules per each water CG bead)  
`mstool.Ungroup(water_resname='W', water_number=4).`  
**water\_resname** is the residue name of water;  
**water\_number** is the number of water molecules included in each CG bead.
- There is no reason why you should not run this on a cluster.  
But there is no advantage of running backmapping on a cluster either.
- All of the backmapping was performed locally on my Macbook M1 Pro.  
I found that apple M1 chip works really well with mstool and is so fast!
- I have tested my tool on CentOS (Linux) but not on windows.
- Do not run multiple mstool runs at the same time.  
I don't understand why, but running multiple mstool at the same time sometimes gives you outputs with flipped isomers

# FAQ

**Q.** When writing a mapping scheme, I am not sure whether this atom should go to this coarse-grained bead or that coarse-grained bead.

**A.** Does not matter. Put it in the either coarse-grained bead (but not both).

**Q.** Can I omit hydrogen atoms in a mapping scheme?

**A.** No, all the atoms must be shown in a mapping scheme. They should be defined only once in a mapping scheme. No duplicate is allowed.

The name of the each atomistic atom should be consistent with the name written in the openMM-compatible forcefield (`$mstool/FF/charmm36/charmm36.xml`).

The name of the each coarse-grained bead should be consistent with the name written in your structure file.

Importantly, `resname` should be consistent between your structure file, mapping scheme, and openMM-compatible forcefield.

(Please see Very important notes and troubleshooting)

**Q.** I have many flipped isomers (cis is flipped to trans, trans is flipped to cis, or chirality is flipped) in the output.

**A.** This is not what you should expect. Something went wrong. Report it to me. I will debug.

**Q.** I receive an error message related to protein - No template found for residue 1 (ALA)

**A.** Your protein probably has missing atoms, non-continuous resids, or etc. Check whether you can do the followings:

```
from openMM import *
pdb = PDBFile('protein_AA.pdb') #contains only protein atoms just for checking
forcefield = ForceField('charmm36.xml')
system = forcefield.createSystem(pdb.topology)
```

**Q.** I downloaded mstool, but `import mstool` does not work.

**A.** Make sure you install it properly. Go to the `mstool` folder and execute `sh install.sh`.

In your `.bashrc` or `~/.zshrc`, confirm whether `mstool` is added to `PYTHONPATH`.