

Lattice Identification Protocol

ChimeraX

- <https://www.cgl.ucsf.edu/chimerax/download.html>
- We are using Version 1.1 (2020-10-07)

Capsid Lattice Database:

https://docs.google.com/spreadsheets/d/15QC2P9mTebc_43gU3zhI_OUjwixvOILNi32MVk0_YG4/edit?usp=sharing

Notes:

- ChimeraX command line commands will be written in the format:
 - ChimeraX cmdline: `chimerax command`
 - Any portion of the command in the format `[fill-in]` should be replaced with the relevant information specified within the brackets
- File names will be written in the format
 - Filename: `[FileName]_[fill-in].filetype`
 - Any portion of the command in the format `[fill-in]` should be replaced with the relevant information specified within the brackets

PART 1: Collect relevant literature information

1. We're currently collecting all of this information in a spreadsheet that will be converted to database format: [Capsid Catalog Luque lab 2021](#).
2. The assumption is that, currently when we have a virus to model, we'll start with the PDB ID. Record this in the "PDB_ID" field
3. Record the name of the virus in the "Virus_name" field, as well as a name in the "Virus_short_name" field.
 - a. The Virus short name is used to generate file names, so do not use spaces or special characters, e.g.:
 - Virus name: Cowpea Chlorotic Mottle Virus (CCMV)
 - Virus short name: CCMV
4. Use the PDB ID to open the page in viperdb (<http://viperdb.scripps.edu/>)
5. Record the Viperdb address. From the ViperDB link, record the following information:
 - a. Host
 - b. Genome Type (ssDNA, dsDNA, RNA, etc.)
 - c. Outer Radius (in Å)
 - d. T-number
 - e. number of subunits (column "Subunits")
6. Record the full taxonomy information, which may be found by following the Uniprot ID link in the Viper DB entry or by looking up the phage up in the NCBI taxonomy browser (<https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi>)
 - a. Realm, Kingdom, Phylum, Class, Order, Family, SubFamily (if applicable), Genus
 - b. Google searching "[Virus Name] taxonomy" or "[PDB ID] taxonomy" is an alternative if the virus is difficult to find in the taxonomy browser

7. Record the fold type in the “Fold_type” field.
 - a. Krupovic and Koonin, 2017 (<https://doi.org/10.1073/pnas.1621061114>), include a table in their Supplemental Information (SI) that identifies the fold type for some specific PDBs, but more importantly, by virus family
 - b. If your virus is not on that list, search for “[virus name] protein fold” and search for the paper in which the virus model was published
 - c. [insert information here about analyzing the capsid fold]
 - d. Record the reference for the fold in the “Fold_ref” field.
8. Record the Genome length in the “Genome_size(bp)” field.
 - a. Make sure the measure is in base pairs (not kbp)
 - b. Record the reference for the size in the “GS_ref” field.

PART 2: Assemble the PDB into a full capsid

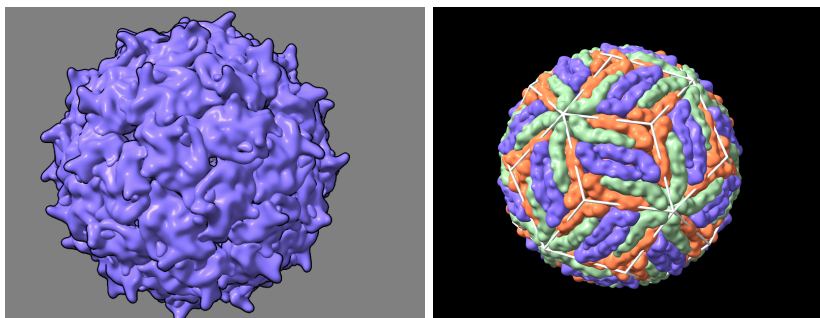
1. Open the structure in Chimera
 - a. ChimeraX cmdline: `open [PDB ID]`
 - b. This should create open the asymmetric unit, but in some instances, it may open a fully assembled capsid
 - c. This first item will be labeled by ChimeraX as model ID 1
2. Record the number of unique capsid protein chains that contribute to the asymmetric unit (DB col: “**capsid_protein**”) using the Chains table.
 - a. Each row represents a different/unique type of protein chain unless otherwise specified. Each letter in the right column represents a chain instance.
 - b. We’re trying to determine which capsid proteins contribute to the structure. To that end, we want to classify the proteins as one of the following:
 - i. Major capsid protein: proteins that make up the majority of the capsid
 - ii. Minor capsid protein: proteins that are important to the structural integrity of the capsid but are in the minority
 - iii. Other capsid protein: proteins that exist on the capsid but do not contribute significantly to the main structure (cement proteins, decoration proteins, internal RNA strands, etc.)
 - c. Note:
 - i. How many proteins of each kind make up the asymmetric unit?
 - ii. Are there multiple species of protein that make up the asymmetric unit. i.e. minor capsid proteins or scaffolding proteins.
 - d. WARNING: sometimes the pdb entry will contain the fully assembled capsid rather than the asymmetric unit ready to be assembled. Keep this in mind when counting chain instances within the asymmetric unit
3. Complete the Icosahedral Assembly (if necessary)
 - a. Verify that in the “[PDB ID] mmCIF Assemblies” table, #1 corresponds to “complete icosahedral assembly”
 - b. We will be hiding specific proteins in this assembly and may need to translate the lattice, so it’s usually best to use the ChimeraX cmdline: `sym #1 assembly 1 copies true`
 - i. This option makes each individual protein available for selection
4. Rainbow color the proteins

We'll need this later, but it's sometimes complicated once surfaces are made, so let's do it ahead of time

- a. Use the command `show #2 cartoons` or click "show cartoons" in the Home menu
- b. Use the command `rainbow #2 target c` or click "Rainbow" the Molecule Display menu
- c. Use the command `hide #2 cartoons` or click "hide cartoons" in the Home menu

PART 3: Set the capsid display settings

1. Create a surface that accentuates the protein/chain boundaries
 - a. ChimeraX cmdline: `surface #[ModelNum] resolution [integer >0] level [number 0-1]`
 - b. Adjust the resolution and level until the smoothness of the protein's surface looks similar to the images below



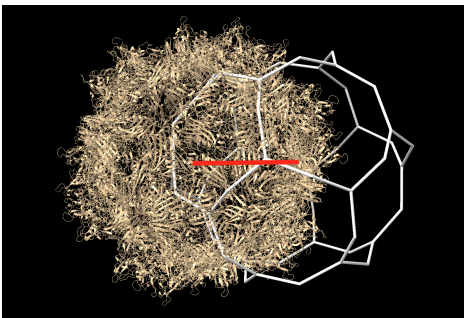
- c. Record the resolution and level used
2. Change the background color if necessary using the ChimeraX cmdline: `set bgColor [color]`
 - a. We'll need screenshots using both black and white backgrounds, but you may find it easier to work in black
3. Change the color of the chains to a colorblind-friendly palette using the ChimeraX cmdline: `color /[chain] #[6-digit RGB color picker code] target s`
 - a. Colorblind-friendly palettes can be found here: [insert link to Meghan's sheet, or add as an appendix]
 - b. Color only major/minor chains. We'll be removing the other chains.
4. Select the icosahedral assembly and turn on the graphics silhouettes using the ChimeraX cmdline: `graphics silhouettes true`

PART 4: Create the lattice cage

1. Use the `hkcage` command to overlay a lattice onto the capsid using the following format:
 - a. `hkcage [h] [k] alpha [lattice] radius [A] orientation [O] sphereFactor [sf]`
 - i. `[h]` and `[k]` are taken from the T# on the Viperdb page
 1. Read the Nature paper to determine how to figure out these numbers
 - ii. Different options for `[lattice]` will be tested to see which fit the best. The hexagonal option is the default.
 1. hexagonal, hexagonal-dual, rhomb, rhomb-dual, snub, snub-dual, trihex, trihex-dual)

- iii. [A] is the radius in angstroms. A starting point can be taken from the viperdb outer radius, but this will not necessarily be exact for the cage.
 - iv. [O] is the orientation
 - 1. n25, n25r, 222, 222r, 2n3, 2n3r, 2n5, 2n5r, n25, n25r
 - v. [sf] is the sphere factor of the cage. This is a decimal value from 0 (not spherical at all) to 1 (spherical)
- b. Select the lattice cage and turn on the graphics silhouettes using the ChimeraX cmdline: `graphics silhouettes true`
- c. Notes:
- i. If only a single species of protein is present, the T number chosen should be equal to the number of proteins in the asymmetric unit.
 - ii. Each polygon in the cage should contain a consistent number of proteins.
 - iii. If there are multiple kinds of proteins present, determine where the secondary proteins lie by coloring them a distinct color

Part 4, Note 1: Lattice and Capsid not aligned



1. Measure the center mass of your assembly (you need to use 'copies true' for this to work) using ChimeraX command `measure center #[assembly model #]`
2. Measure the center mass of your lattice using ChimeraX command `measure center #[lattice model #]`
3. Move the lattice to the assembly one coordinate at a time using the ChimeraX commands:
 - a. `move x [lattice xcoord value change] coordinateSystem #[assembly model #] models #[lattice model #]`
 - b. `move y [lattice ycoord value change] coordinateSystem #[assembly model #] models #[lattice model #]`
 - c. `move z [lattice zcoord value change] coordinateSystem #[assembly model #] models #[lattice model #]`
 - d. Remember, the goal is to bring one center to the other, so if you have to move 324 to 0, you have to use -324 as your coord value change
4. More information can be found in the documentation here. [\[insert link\]](#)
5. Alternatively, you can manually move the lattice onto the capsid by:
 - a. Right Mouse > Move Model
 - b. Using the Model panel, select the lattice by clicking the right check box
 - c. Right click, hold down and drag the lattice so it overlaps the capsid
 - d. Rotate the lattice/capsid pair to view from various angles to make sure the lattice and capsid overlap from all angles

- e. Top Menu Bar > Right Mouse > Translate
 - i. Reset the right click to translate all models together, so you don't accidentally mess up the alignment

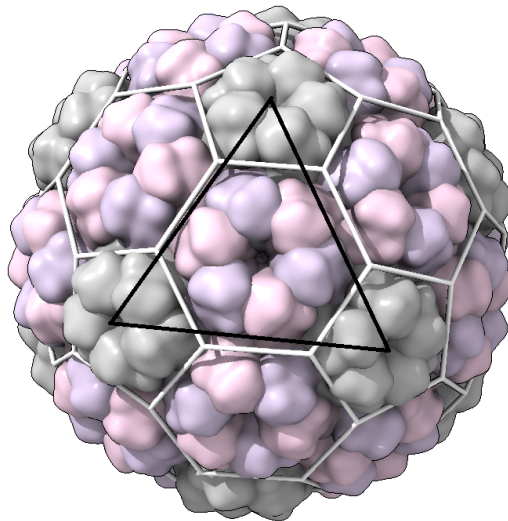
Part 4, Note 2: Lattice and Capsid have non-standard orientation

If none of the above orientations work, the cage can be rotated and aligned manually

1. Top Menu Bar > Right Mouse > Rotate Model
2. Using the Model panel, select the lattice by clicking the right check box
3. Right click, hold down and drag the lattice rotating it until there are minimal protein chains spanning two lattice faces and the orientation of the lattice matches the major landmarks of the capsid

PART 5: Take snapshots and save the ChimeraX session

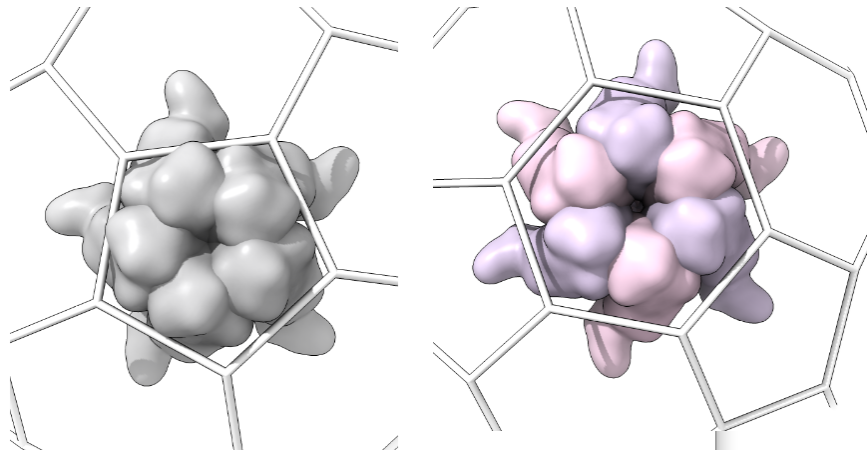
1. We need several snapshots for each capsid, with each shot against both a black and a white background. The naming convention for saving each file is as follows:
`[PDBID]_[Virus_Short_Name]_[h]_[k]_[lattice]_[orientation]_[spheresfactor*100]_[snapshot_type]_[bgcolor].png`
2. Snapshot types:
 - a. **Full_surface**: a full snapshot of the capsid. Take these snapshots with a clear view of one of the triangular faces of the icosahedron in view, e.g.:



- i. (black line is for emphasis here; this should not be in your snapshot)
- NOTE:** If there are other capsid proteins on the surface that do not contribute to the lattice, select and remove them from view. You can do this by using the ChimeraX cmdline: `sym #1 assembly [desired assembly number] copies true`
- ii. This will make each individual protein able to be selected in the Models window and hidden individually.
 - iii. Select the assembly and use Hide Atoms and Hide Cartoons

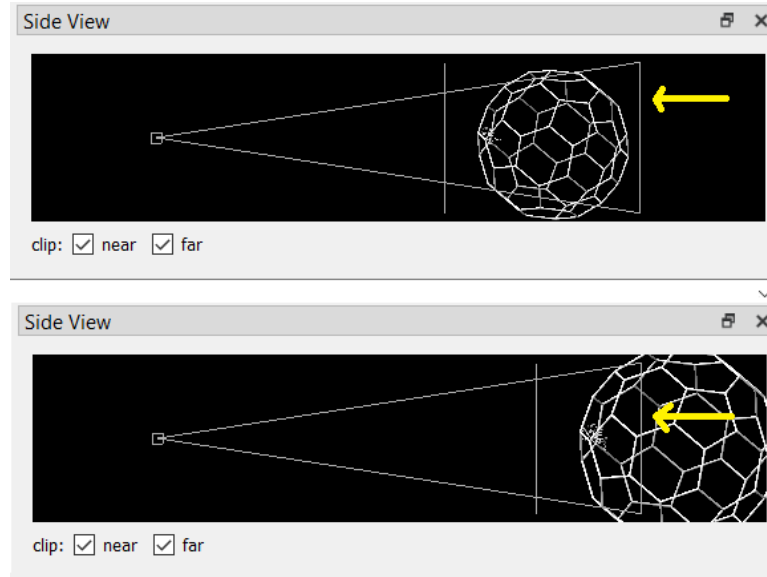
- iv. If you hover the mouse over the protein you want to hide, Chimera will pop up its protein number, so you know where to look in the Models window
- b. **[Capsomer_unit]_outside**: For each type of shape in the lattice, centered in the shot. Each lattice type requires a specific number of capsomer shots, as follows:
 - i. Hexagonal lattices: a pentamer and a hexamer (if $T>1$)
 - ii. Trihexagonal: a pentamer, a hexamer (if $T>1$), and a trimer
 - iii. Rhomb: a pentamer, a hexamer (if $T>1$), a rhomb, and a trimer
 - iv. Snub: a pentamer, a hexamer (if $T>1$), and a trimer
 - v. Hex dual: a trimer
 - vi. Trihex dual: a dimer
 - vii. Rhomb dual: a kite
 - viii. Snub dual: a floret

For the hexagonal example above, these would be the shapes we're looking for:

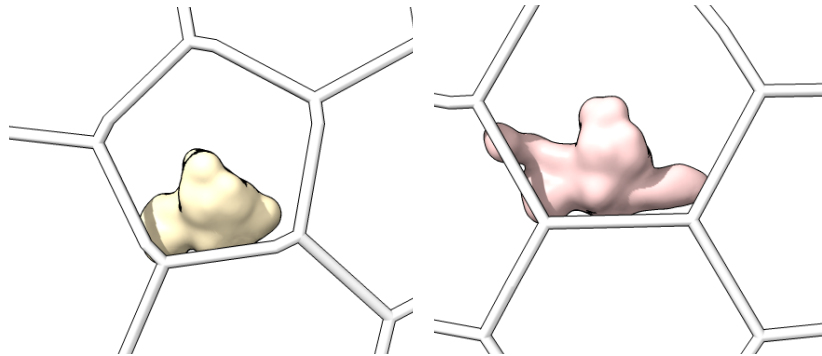


NOTE: to isolate single capsomers, you must use the ChimeraX cmdline: `sym #1 assembly [desired assembly number] copies true`

1. This will make each individual protein able to be selected in the Models window and hidden individually.
2. Select the assembly and use Hide Atoms and Hide Cartoons
3. If you hover the mouse over the protein you want to hide, Chimera will pop up its protein number, so you know where to look in the Models window
4. It may be advantageous to hide the full assembly, then create a copies-true assembly of the pentamer or hexamer if available, that way there are fewer individual proteins to hide.
5. To hide the back side of the lattice, use Graphics → Side View to open the side view window. Click and drag the viewing location (the line indicated by the yellow arrow below) through the capsid until the underside is clear.



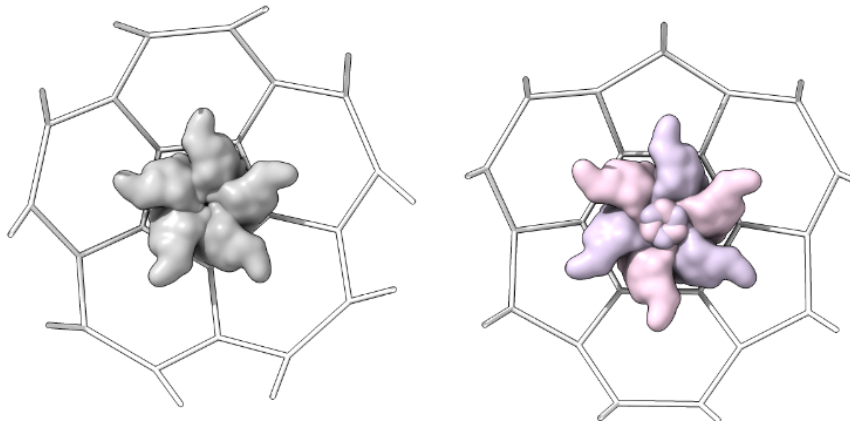
- c. **Single_protein_surface_outside:** For each type of shape in the lattice, centered in the shot, one shot for each major/minor protein as a surface:



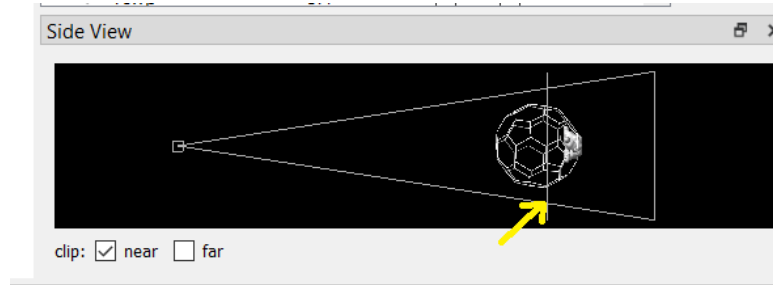
- i. Use the ChimeraX cmdline: `sym #1 assembly 2 copies true`. This will give you a single asymmetric unit with each individual protein able to be selected individually.
 - ii. Adjust the surface to match what you used for the full icosahedral assembly by using `surface #[ModelNum] resolution [integer >0] level [number 0-1]` as above
 1. Hide all but the protein in question
 - iii. Snapshot the surface
- d. **Single_protein_ribbon_outside:** one shot for each major/minor protein as a ribbon without the lattice cage



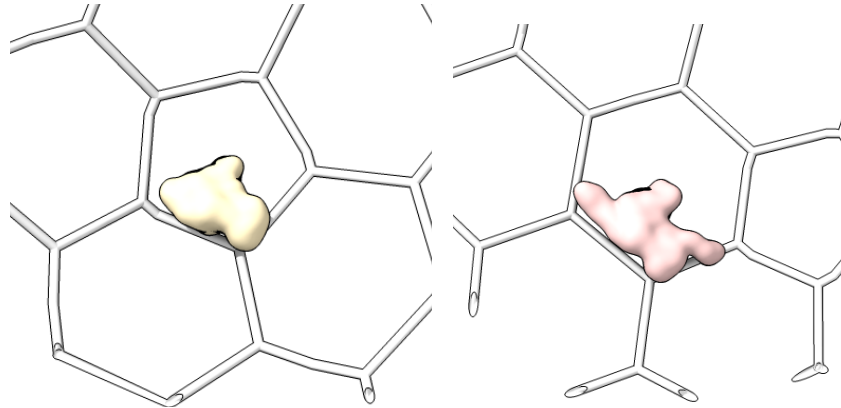
- i. Make another copy of the pentamer assembly using the ChimeraX cmdline: `sym #1 assembly 2 copies true`.
 - ii. Select the assembly and show cartoons, then hide atoms.
 1. You can adjust the ribbon style if necessary by using `cartoon style width 1.5 thick 0.3` where the width and thickness are values you can change.
 - iii. Hide all but the protein you are interested in by using `hide /[chain ID] cartoons`
 1. You can hide multiple chains by specifying a range, e.g.: `/A-D` will hide all chains between A and D (inclusive)
 - iv. From the Molecule Display menu, choose “rainbow” coloring.
 - v. Snapshot the ribbon in the same position as the corresponding surface
- e. **[Capsomer_unit]_inside**: one shot for each type of shape in the lattice from inside the structure as determined above, centered in the shot. This is to illustrate the interconnectedness of the proteins.



- i. To do this:
 1. Rotate the structure so that the capsomer is on the back side
 2. Use Graphics → Side View to open the side view window
 3. Click and drag the viewing location (the line indicated by the yellow arrow below) through the capsid until the underside is clear.

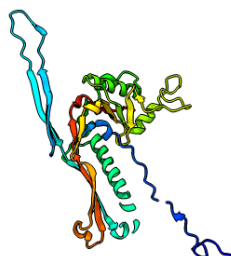


- f. **Single_protein_surface_inside:** For each type of shape in the lattice, centered in the shot, one shot for each major/minor protein as a surface:



- i. Use the ChimeraX cmdline: `sym #1 assembly 2 copies true`. This will give you a single asymmetric unit with each individual protein able to be selected individually.
- ii. Adjust the surface to match what you used for the full icosahedral assembly by using `surface #[ModelNum] resolution [integer >0] level [number 0-1]` as above
 1. Hide all but the protein in question
- iii. Snapshot the surface

- g. **Single_protein_ribbon_inside:** one shot for each major/minor protein as a ribbon without the lattice cage



- i. Make another copy of the pentamer assembly using the ChimeraX cmdline: `sym #1 assembly 2 copies true`.
- ii. Select the assembly and show cartoons, then hide atoms.

1. You can adjust the ribbon style if necessary by using `cartoon style width 1.5 thick 0.3` where the width and thickness are values you can change.
 - iii. Hide all but the protein you are interested in by using `hide /[chain ID] cartoons`
 1. You can hide multiple chains by specifying a range, e.g.: `/A-D` will hide all chains between A and D (inclusive)
 - iv. From the Molecule Display menu, choose “rainbow” coloring.
 - v. Snapshot the ribbon in the same position as the corresponding surface
3. Save the ChimeraX session
 - a. `[PDBID]_[VirusCapsidNameNoSpaces]_[h]_[k]_[lattice]_[orientation]_[sphericity*100].cxs`