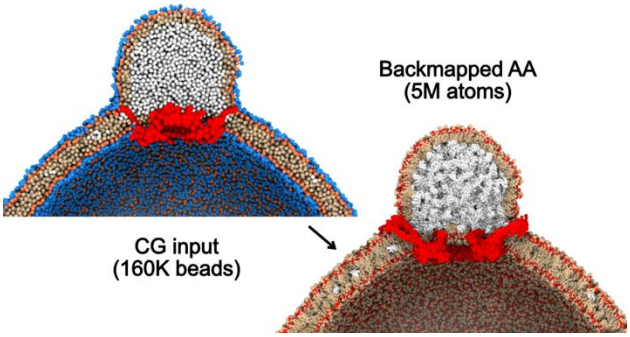
Backmapping with Mapping and Isomeric Information

Siyoung Kim[1]
[1]Pritzker School of Molecular Engineering, University of Chicago, Chicago, IL 60637 USA.

ABSTRACT

I present a powerful and flexible backmapping tool named Multiscale Simulation Tool (`mstool`) that converts a coarse-grained (CG) system into all-atom (AA) resolution and only requires AA to CG mapping and isomeric information (*cis/trans/dihedral/chiral*). The backmapping procedure includes two simple steps: a) AA atoms are randomly placed near the corresponding CG beads according to the provided mapping scheme. b) Energy minimization is performed with two modifications in the AA force field (FF). First, nonbonded interactions are replaced with cosine functions to ensure numerical stability. Second, additional torsions are imposed to maintain molecules' isomeric properties. To test the simplicity and robustness of the tool, I backmapped multiple membrane and protein CG structures into AA resolution, including a four-bead CG lipid model (resolution increased by a factor of 34) without using intermediate resolution. The tool is freely available at github.com/ksy141/mstool.

TABLE OF CONTENTS (TOC)



INTRODUCTION

CG molecular dynamics (MD) simulations have become a popular method in studying biological processes with time and length scales beyond the reach of AA-MD simulations,[1–6] such as large-scale membrane deformation, organelle biogenesis, and the complete virion or capsid of a virus.[7–10] In this context, resolution transformation from CG to AA, or backmapping, is necessary to obtain detailed atomistic interactions between molecules, a level of detail not available in CG resolution. Therefore, CG-MD and backmapping can complement each other: one can perform preliminary CG-MD simulations to navigate systems' equilibrium distribution, followed by additional AA-MD simulations, using initial structures obtained from backmapping.

While it is straightforward to map AA structures into CG ones, the reverse requires recovery of the degree of freedom that has been integrated away.[11,12] One of the

approaches is fragment-based reconstruction, which replaces CG beads with the corresponding atomistic fragment. This popular approach has been used in protein structure modeling by reconstructing full atomistic protein structures from their alpha carbon (CA) positions. For example, a four-residue backbone fragment that fits the four consecutive CA atoms from the Protein Data Bank (PDB) search is used to construct backbone atoms, followed by searching for the most probable side chain conformation.[13,14] The same approach has been used on protein and DNA complexes.[15] A more flexible and general fragment-based approach has been developed by Stansfeld and colleagues.[16,17] This approach can convert CG-MD structures of membrane protein and lipids at Martini resolution[18–21] to AA structures. Throughout this manuscript, the latest version of their tool will be referred to as `CG2AT2`.[17]

There is another approach in which atoms are positioned based on local geometrical information that users provide. For instance, `CG2AA` uses the positions of three consecutive CA atoms to construct backbone and side chains based on a simple geometrical algorithm.[22] Wassenaar et al. have defined the relative positions of atoms not only for protein but also for lipids, extending applications of the geometrical approach.[23] This tool will be referred to as `Backward` in this manuscript. Finally, there are recent studies that use machine learning for resolution transformation.[24–27]

The above approaches have focused on constructing initial AA structures from CG structures, followed by standard energy minimization. This paper presents a backmapping approach that uses a modified FF in energy minimization. It is powerful enough that the initial positions of atoms can be random without fragment alignment or geometrical projection, greatly simplifying user inputs into a minimal set of information: mapping scheme and isomer properties. The tool replaces nonbonded interactions with cosine functions to ensure numerical stability of energy minimization while keeping bonded interactions intact. In other words, a system is relaxed with higher priority given to bonded interactions, while cosine nonbonded interactions ensure no atoms overlap. In addition, a set of torsion potentials is imposed during relaxation to maintain the provided isomeric properties of molecules (*cis/trans/chiral/dihedral*). While the idea is simple, the tool is powerful and capable of constructing complete AA structures from large-scale, highly CG systems.

The rest of the paper is structured as follows. I will first describe a modified FF. In Results, I will test whether torsions applied to the provided isomeric properties work as intended. Then, I will compare the backmapping performance of `mstool` with the popular geometrical projection approach and fragment-based approach, `Backward`[23] and `CG2AT2`,[17] respectively. I will then present more examples of resolution transformation of CG lipid and/or protein systems. The examples described in this paper are available at the Github link with a step-by-step guideline.

METHODS

For each type of molecule (residue) present in CG structures, a mapping scheme describes which AA atoms belong to which CG beads. It also should include molecules' isomeric properties such as *cis*, *trans*, *dihedral*, and/or *chiral*. The predefined mapping files based on the Martini FF are read if no user mapping scheme is provided.[18–21]

The backmapping procedure consists of three steps: Ungrouping, relaxing a system, and checking a backmapped structure. The first step places AA atoms near their corresponding CG beads according to the provided mapping scheme, implemented as `mstool.Ungroup`. The output of this step is an intermediate AA structure, but not at energy minimum because AA atoms are clustered at the locations of CG beads. Water CG beads are treated separately in this step because each water bead represents more than one water molecule in most CG models. For instance, the Martini FF contains four water molecules per each water bead. Users can provide the number of water molecules that are represented by each water CG bead and the residue name of CG water. If no user input is given for water, they are set to 4 and W, respectively, consistent with the Martini FF convention.

The next step is called Reduced Nonbonded Energy Minimization (REM), implemented as `mstool.REM`. A system is energy minimized with a modified FF. First, CHARMM36m protein[28] and CHARMM36 lipid[29] FFs are applied to the output structure of `mstool.Ungroup`. Then, the tool replaces the positive part of LJ potential (U > 0) with cosine repulsion, expressed as

$$U_{LJ}(r) = \min\left[A\cos^2\left(\frac{\pi}{2\sigma}r\right), 4\epsilon\left(\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right)\right] \#(1)$$

A charged interaction is described by

$$U_c(r) = Cq_iq_j\cos^2\left(\frac{\pi}{2r_c}r\right) \#(2)$$

By default, $A = 100\ kJ/mol$, $C = 50\ kJ/mol$. Nonbonded interactions are cut-offed at $r_c = 1.2\ nm$. Eqs. 1 and 2 are illustrated in Fig. 1A. The modified nonbonded interactions ensure energy and force values do not diverge even if atoms are very close and smooth potential energy surfaces, making energy minimization easier.

In addition to using modified nonbonded interactions, isomeric torsion terms are added during REM to maintain the provided isomeric properties of molecules, which should be written in mapping files. A set of improper dihedral terms is applied for *chiral* (Fig. 1B). A single improper dihedral term is applied for each *cis*, *trans*, or *dihedral* (Fig. 1C), described by $V = K\ (\xi - \xi_0)^2$. For *dihedral*, ξ_0 is determined by user input. For *cis* or *trans*, $\xi_0 = 0^\circ\ (cis)\ or\ 180^\circ\ (trans)$. K is set to $300\ kJ/mol/rad^2$.

Every amino acid except for glycine has a *chiral* center at the carbon alpha (CA) atom, and their chirality should be denoted in a mapping scheme. However, a peptide bond, which is dominantly in the *trans* configuration, cannot be specified in a mapping scheme because it is a cross-residue property and cannot be defined within a single residue. To prevent any *cis* peptide bonds, `mstool` applies a *trans* dihedral (Fig. 1C) in every

peptide bond that it detects using atomic names during REM. With the modified FF, `mstool` uses openMM for energy minimization.[30,31] The search of a local minimum is performed with the L-BFGS algorithm until the root-mean-square value of all force components reaches 10 kJ/mol/nm.[32]

The last step in the backmapping procedure is checking the resulting AA structures, implemented as `mstool.CheckStructure`. It reports which isomeric properties are reviewed (written in mapping schemes) and flipped (inconsistent with mapping schemes) isomeric properties. This function only checks the defined isomeric properties in the provided mapping schemes. Unspecified isomeric properties will not be detected or checked. However, `mstool.CheckStructure` automatically detects and checks *cis* peptide bonds because a peptide bond is a cross-residue property and cannot be specified in a mapping file.

When using the `mstool` backmapping, a system should not have two or more residues with the same residue name, residue number, and chain name. This may be an issue for a very large CG system because the residue number is limited to up to only four digits (0-9999), and the chain name can be defined with only one character in a PDB format. To solve this issue, `mstool` also accepts a Desmond structure file (DMS), which has no limits on the length of residue name, residue number, and chain name.[33] Finally, one CG bead should not represent two or more molecules except water, which is treated separately inside the tool. Therefore, resolution transformation of supra-CG or mesoscale CG models is not supported.

All the backmapping and simulations were performed on a MacBook Pro (16-inch, 2021) with an Apple M1 Pro chip. For NPT simulations, Langevin dynamics was used with a target temperature of 310 K and a friction coefficient of 1 ps^{-1}. A semi-isotropic or isotropic Monte Carlo barostat was used with a target pressure of 1 bar and a pressure coupling frequency of 100 steps.[34,35] A cutoff of 1.2 nm was used for the direct space interactions, and particle mesh Ewald was used for the long-range electrostatic interactions.[36] All bonds involving hydrogen atoms were constrained.[37] Simulations were evolved with a 2 fs time step. Simulations and protein dihedrals were analyzed using MDAnalysis.[38,39] The C36-c parameters were used for triolein.[40] Structures were visualized with ChimeraX.[41,42]

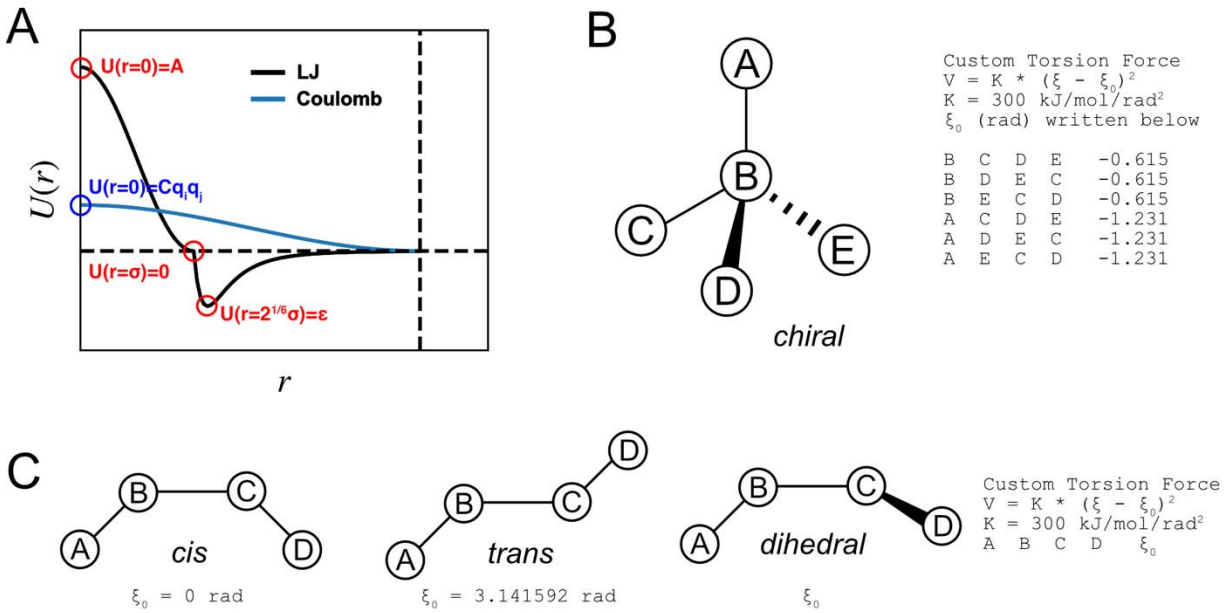


Figure 1. Illustration of the modified force field. (A) Nonbonded interactions. (B) Torsions for *chiral*. (C) Torsions for *cis*, *trans,* or *dihedral*.

RESULTS

Enantiomers and Configurational Isomers

Due to their coarse nature, CG models contain less isomeric information than their AA counterparts. For example, the CA atom of every amino acid except for glycine is a *chiral* center, and peptide bonds are dominantly in *trans* configuration at AA resolution. However, many CG models, including Martini, have only one CG backbone bead, removing both enantiomeric and configurational isomerism. For this reason, care must be taken when reconstructing AA models from CG ones because the resulting AA structures should have the correct isomeric properties.

In `mstool`, torsions are imposed to ensure backmapped molecules have correct isomeric properties during REM. To test whether these torsions work properly, I backmapped CG systems with and without providing isomeric properties. The first test system was a Martini bilayer membrane with each leaflet constructed from 20 POPC molecules. Each POPC molecule has one *chiral* center and one *cis* configuration (Fig. 2A). The CG and backmapped structures are shown in Fig. 2B.

When the *chiral* center was given in a mapping file, either R or S, all the backmapped POPC molecules were enantiomerically pure with their chirality as provided. If the *chiral* center was unspecified, the tool produced a racemic mixture (Fig. 2C). Similarly, when the geometric isomerism was specified, either *cis* or *trans*, all the molecules had homoconfiguration. When configuration was not specified, an equal mixture of *cis* and *trans* molecules was created (Fig. 2C). The important conclusion from this test is twofold: A) Enantiomeric and configurational isomeric properties must be provided to

have desired isomeric properties in resulting AA systems. B) Imposed torsions play a critical role in reconstructing molecules with desired enantiomeric and configurational properties.

Lipid Properties

I performed 40 ns of NPT simulation of a POPC bilayer membrane, backmapped with the correct enantiomeric (R) and configurational (*cis*) properties, and then analyzed the physical properties of the membrane. The initial values of the area per lipid (Fig. 2D) and distance between phosphate atoms across the bilayer (Fig. 2E) were already in the equilibrium region. This is because the Martini FF agrees with the experimental and AA-MD simulation data.[18–21] The order parameters, $S_{CD} = 0.5 \times |< 3cos^2\Theta - 1 >|$, where Θ is the angle between CH vectors with the bilayer normal,[43] are more directly related to the backmapping performance. In this POPC bilayer membrane example, the initial structure, backmapped from the CG structure, already had the reasonable order parameters and could distinguish *sn*-1 and *sn*-2 chains (dashed lines in Fig. 2F). This is a reinsuring result given that the backmapping procedure does not involve any NPT/NVT simulations but only energy minimization. Furthermore, the first 10 ns simulation could already equilibrate the bilayer membrane (Fig. 2F).

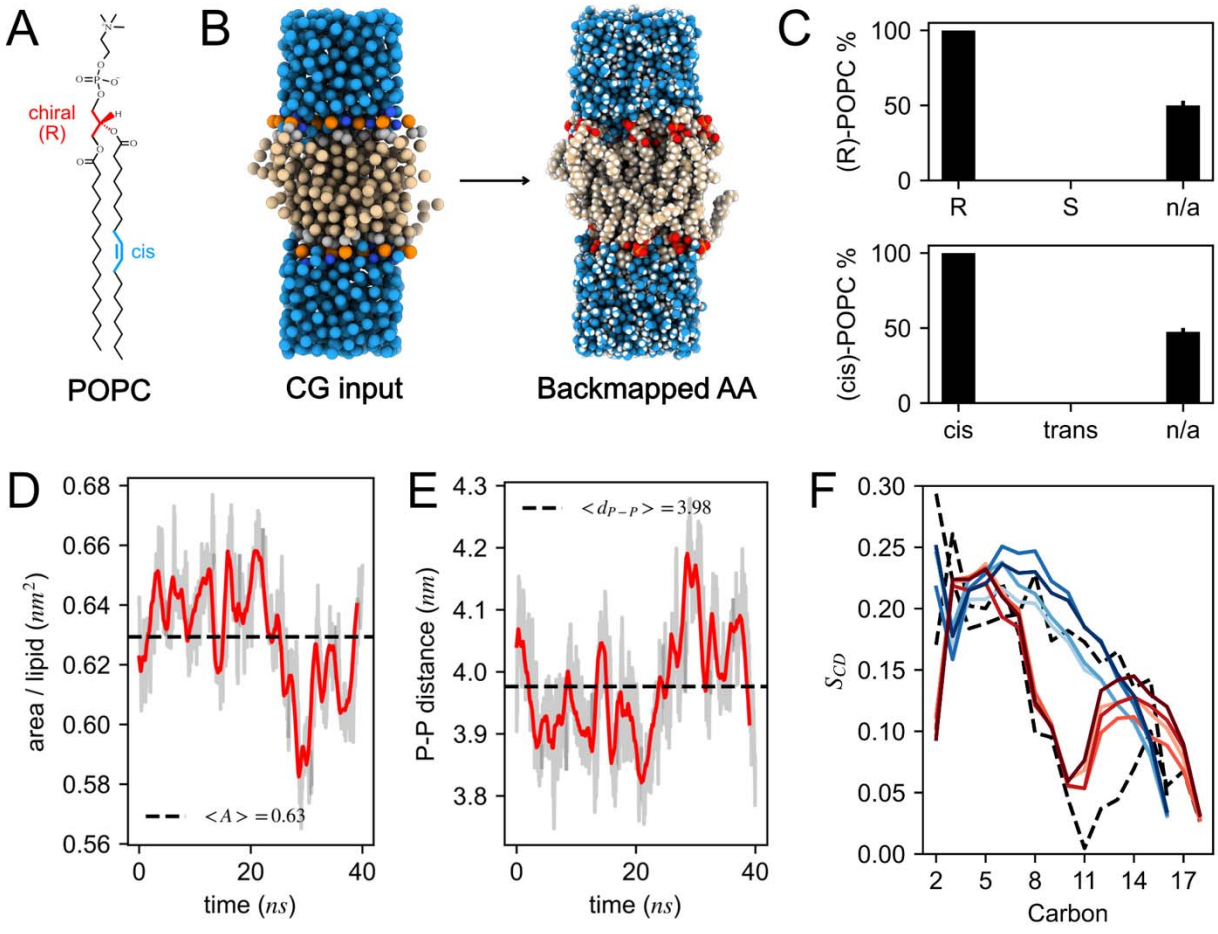


Figure 2. Backmapping and simulation of POPC bilayer membrane. (A) Molecular structure of POPC (B) CG and AA structures (C) Isomeric properties of backmapped POPC molecules (y-axis) with given isomeric properties (x-axis). n/a represents unspecified isomeric properties. Error bars are standard errors of four replicates. (D) Area per lipid. (E) Distance of phosphate across bilayer in AA-MD simulation. Gray lines are the instantaneous values at 20 ps intervals. Red lines are the moving average with 1 ns window. Dashed lines are the averaged values. (F) Order parameters for *sn*-1 (blue) and *sn*-2 (red) chains, depicted every 10 ns, with darker color representing later simulation time. Dashed lines indicate the order parameters of the initial structure.

Comparison with Backward and CG2AT2

The backmapping performance of `mstool` was compared with the two other popular tools, `Backward`[23] and `CG2AT2`.[17] The first system was a Martini bilayer membrane of POPC and DOPC of various sizes (Fig. 3A). For each system size, four equilibrated Martini bilayer membranes were prepared and then backmapped with `Backward`, `CG2AT2`, and `mstool`. While all tools can backmap the CG systems into AA structures, `Backward` produced flipped configuration, and `CG2AT2` flipped enantiomers. In contrast, all molecules converted with `mstool` preserved their isomeric properties.

The second system was the WzmWzt ABC transporter (PDB: 6M96), a membrane protein system.[44] Its CG structure was provided by `CG2AT2`, with the protein surrounded by 504 POPE and 125 POPG lipid molecules (Fig. 3B).[17] The CG structure was backmapped four times using the three tools. Consistent with the previous example, cis-to-trans flipped configurations and flipped enantiomers were observed in AA structures when backmapped with `Backward` and `CG2AT2`, respectively. In the `Backward` structures, *cis* peptide bonds were also detected. There were no flipped isomeric properties for the `mstool` structures.

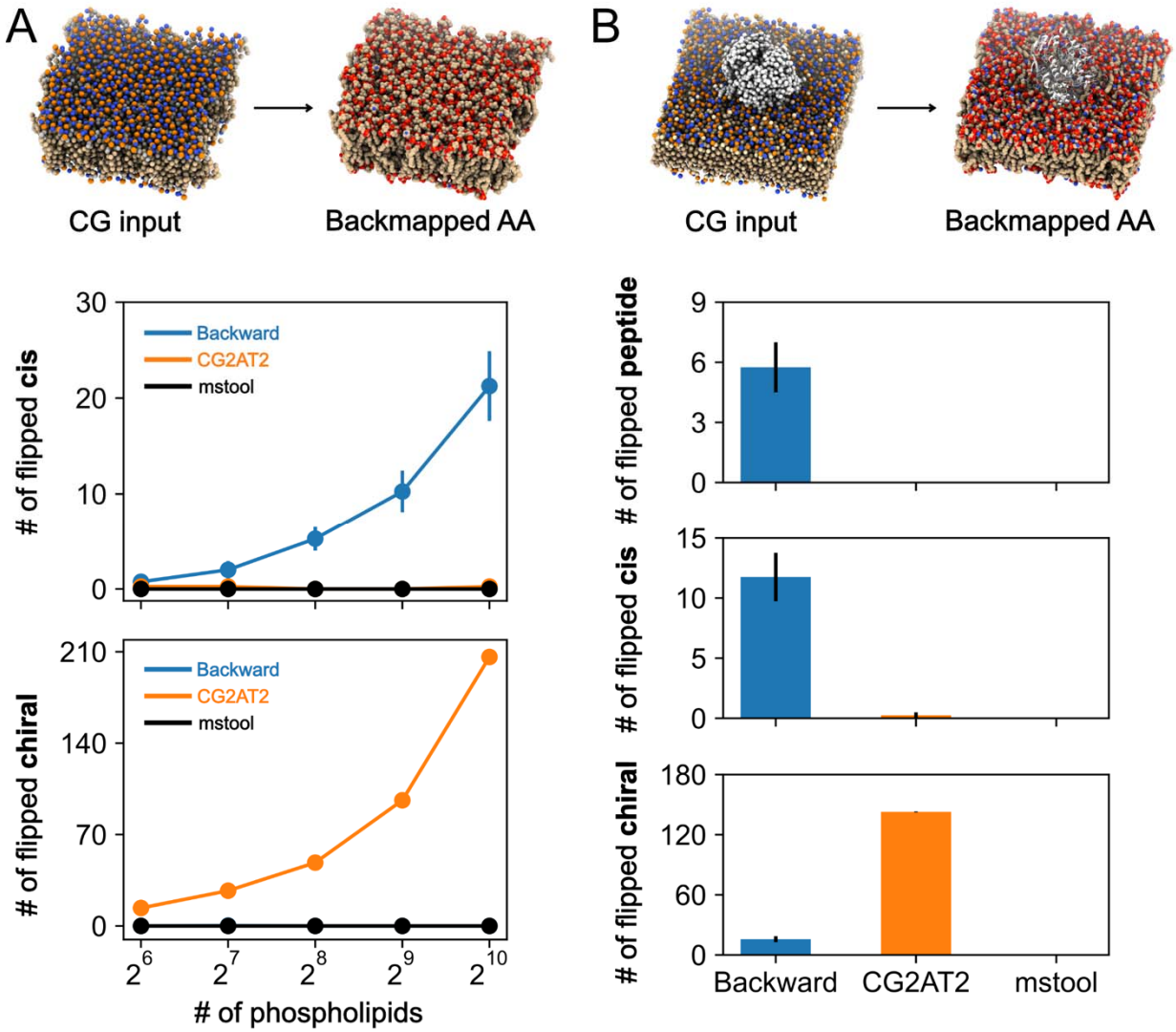


Figure 3. Comparison of backmapping performance. (A) DOPC and POPC bilayer membrane (B) WzmWzt ABC transporter in POPE and POPG membrane. The number of flipped enantiomerical and configurational properties are shown. Peptide on the right means the number of *cis* peptide bonds. Error bars represent standard errors of four backmapping results. Hydrogen atoms in backmapped AA structures are omitted for visual clarity from now on.

Conformation

Unlike enantiomeric and configurational isomerism, conformational isomerism changes during unbiased AA-MD simulations. In fact, the boat-to-chair conversion or the interconversion of twist-boat is frequent. However, neither the ring inversion (chair-to-chair) nor the chair-to-boat conversion is likely probable in unbiased AA-MD simulations because of the stability of the chair state. Therefore, the most probable conformation should be specified using *dihedral* torsions during backmapping.

Cyclohexane was used as a test system to evaluate the backmapping tool's performance of conserving conformational isomerism. Cyclohexane has multiple conformations: chair (C), boat (B), twist-boat (TB), and half-chair (HC) as shown in Fig. 4A. A total of 125 cyclohexane molecules, each of which was represented by one CG bead, was backmapped with and without *dihedral* torsions and equilibrated for 10 ns in NPT ensemble (Fig. 4B). When *dihedral* torsions were provided, an initial conformation of all cyclohexane molecule was consistent with the provided torsions (Figs. 4C-4E). For instance, if a chair conformation was specified in a mapping file, all the cyclohexane molecules were initiated with the chair conformation (red dots in Figs. 4C and 4D). Given the stability of the chair conformation, all the molecules maintained their chair conformation during the NPT simulation (heatmap in Figs. 4C and 4D). If a twist boat conformation was specified, the initial conformation of cyclohexane was a twist boat. However, during the NPT equilibrium, cyclohexane changed its conformation to the most stable chair conformation, either C1 or C2 (Fig. 4E).

When isomeric properties were not given, unlike the previous example that had an equal mixture of *cis/trans* and *R/S* (Fig. 2C), the conformations of cyclohexane were not equally distributed: twist-boat was 80.8%, chair 14.4%, half-chair 3.2%, and boat 1.6% (Fig. 4F). The chair conformation is the most stable state and should be the most probable state. However, the twist boat conformation was the most frequent conformation when torsions were not provided, suggesting that the resulting structure was not Boltzmann distributed.

There are multiple reasons for the non-Boltzmann distribution of conformations. First, the 1-4 repulsion is a large energetic component that makes the chair conformation lower energy than the boat conformation. In the `mstool` pipeline, the repulsion in the modified FF is much softer than the original LJ. Second, backmapping involves energy minimization but no dynamics. Therefore, no information on temperature is given during REM. Therefore, the softer repulsion in the modified FF combined with the lack of dynamics results in the non-Boltzmann distribution of conformations.

However, during the following NPT simulation, all the conformations changed to the chair conformation (C1 or C2), as shown in the heatmap of Fig. 4F. Therefore, it should be noted that backmapping does not produce Boltzmann distributed initial conformations and should be only used to make an initial structure for AA-MD simulations which then can be further equilibrated using NPT or NVT simulation.

Finally, it should be noted that cyclohexane is rather a simple molecule, and its chair conformations are all equivalent. Other ring molecules (e.g., glycosylated residues or inositol-containing lipids) have multiple chair conformations with different energetic levels. Users should provide the *dihedral* torsions that describe the most probable conformation because it is likely that the ring flip does not occur in AA-MD simulations, as shown in Figs. 4C and 4D.

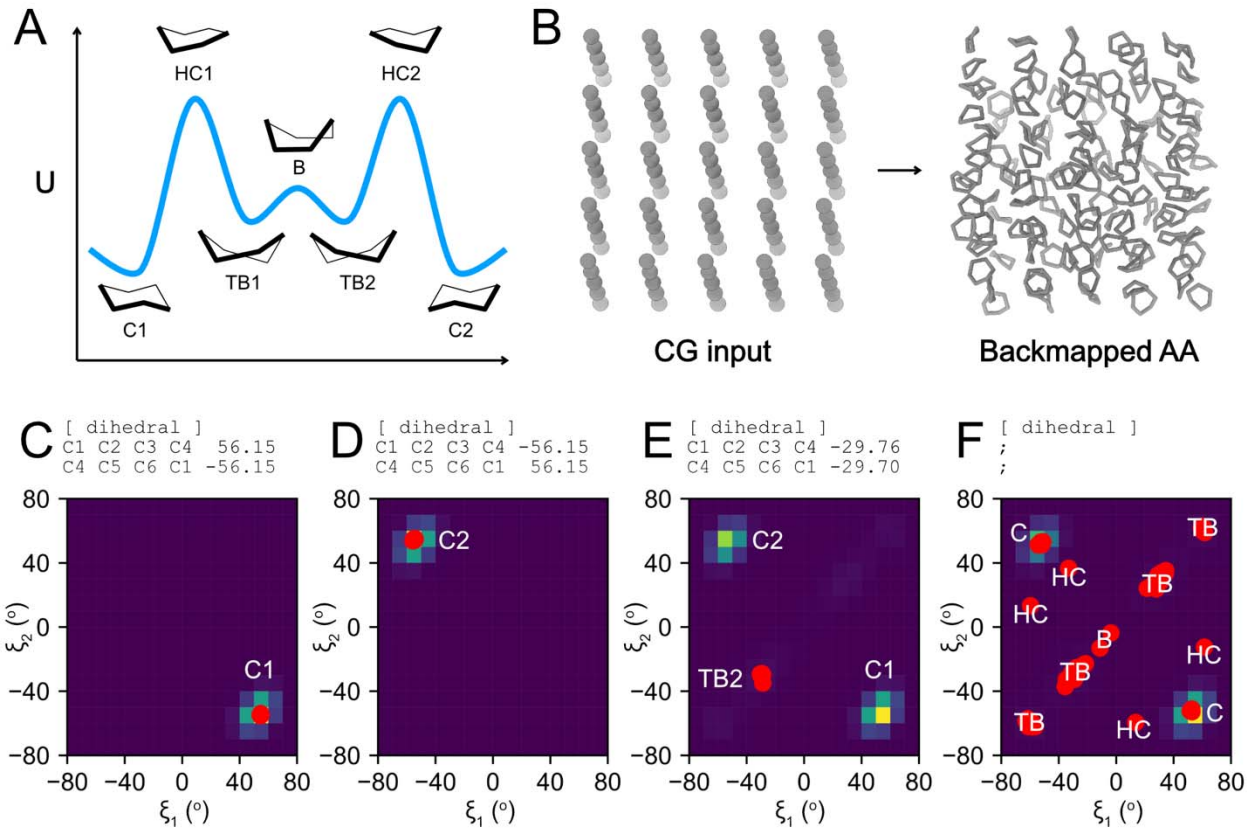


Figure 4. Conformation of cyclohexane. (A) Schematic illustration of cyclohexane conformation energy landscape. (B) CG and AA structures. (C-F) Dihedrals of cyclohexane molecules. Red dots indicate the conformations of backmapped structures. Heatmaps indicate the conformations during the following NPT simulations. Brighter colors represent more populated states in the heatmaps.

Lipid Backmapping

To test the performance of `mstool`, two CG lipid systems were transformed into AA ones. The first example was a low-resolution CG model in which four CG beads represent one lipid.[45,46] Two types of lipids were involved in this example. One was a phospholipid, and another one was a neutral lipid triolein. I backmapped a CG membrane structure in which triolein nucleated inside a POPC bilayer (Fig. 5A). Although the resolution of the CG model was low, making a backmapping input file for this system was straightforward because `mstool` simply requires a mapping scheme and isomeric information of these two types of lipids. The conversion increased the resolution of the structure 34-fold without any bridging resolutions, and the resulting structure had no flipped isomers, demonstrating the robustness of `mstool`.

Martini lipids are one of the most popular CG models.[18–21] It has been shown that they accurately capture the physics and chemistry of many biological processes and reproduce the bilayer properties such as thickness and area per lipid.[47,48] To test the efficacy of `mstool`, I backmapped a spherical bilayer at Martini resolution with 14 different Martini lipids into AA resolution (Fig. 5B). The backmapped structure had no

flipped isomers. This shows the robustness of isomeric torsions applied during REM (Fig.1B and Fig. 1C) as each cholesterol molecule (residue name CHL1) has eight *chiral* centers, and a majority of lipids contain one or more *cis* bonds.

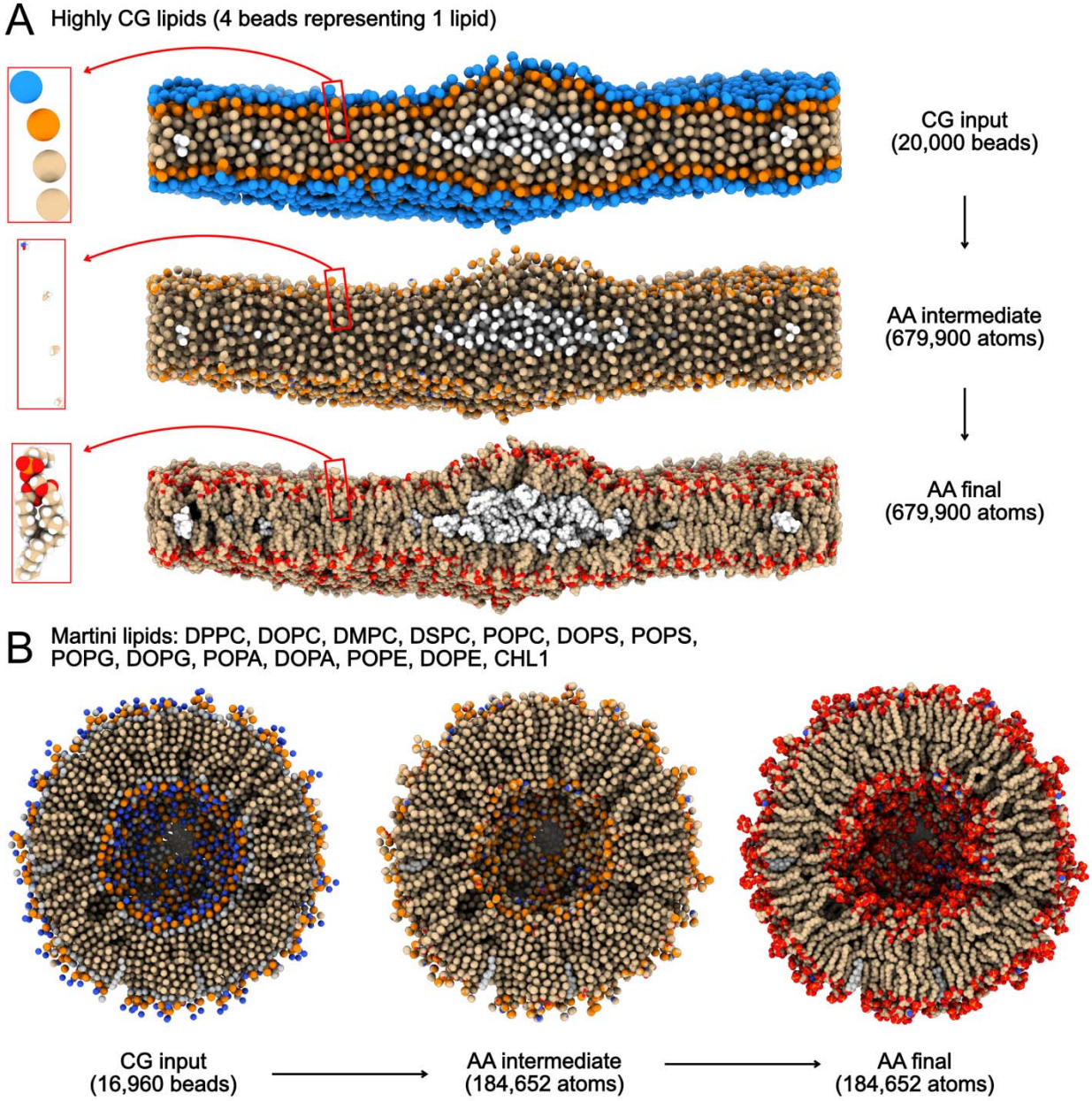


Figure 5. Lipid backmapping of (A) highly CG lipids and (B) Martini lipids.

Protein Backmapping

`mstool` supports multiple backmapping options for protein. The equilibration timescale of protein is much longer than that of lipids and usually beyond timescales attainable by AA-MD. Therefore, one should carefully choose a protein backmapping scheme that best fits the problem at hand. If there is an initial or reference AA protein structure, it is beneficial to incorporate it into the backmapping procedure. If there are no big

conformational changes during CG-MD simulations, users can directly align the AA protein structure against the CG protein structure and then provide the aligned AA structure into `mstool` during REM. If there is a conformational change during CG-MD simulations, one can align only the rigid parts of protein and then build loops using a loop modeler provided in `mstool` or other software. However, random placement or geometrical projection can be an option if there is no initial AA protein structure, or the final CG structure is very different from the AA structure. Backmapping of T-cell intracellular antigen 1 (PDB: 2MJN) from Martini resolution to AA resolution was used as a test case.[49] A short CG-MD simulation was performed to stray away from the initial AA structure (Fig. 6A).

The first option is to randomly place atoms of each amino acid at their corresponding CG beads, just like lipid backmapping, through `mstool.Ungroup`. For instance, the Martini resolution has one backbone bead (BB) per residue; Therefore, backbone atoms (N, HN, CA, HA, C, O) are initially placed at the location of their corresponding BB bead. Side chain atoms are also ungrouped similarly. An intermediate AA structure is relaxed (`mstool.REM`) and reviewed (`mstool.CheckStructure`).

The second option, the default option in `mstool`, is only slightly different from the first option in ungrouping. Instead of randomly placing AA atoms from CG beads, it predicts the positions of backbone atoms using the previously published geometrical algorithm (`CG2AA` and `Backward`).[22,23] In short, it uses the positions of three consecutive backbone beads to obtain the positions of backbone atoms. Side chain atoms are placed near their corresponding CG beads. The rest of the procedure (`mstool.REM` and `mstool.CheckStructure`), is the same.

Backmapped protein structures using the first and second options are shown in Fig. 6B and Fig. 6C, respectively, along with their Ramachandran plots. Although both options provide a reasonable structure, there are some Ramachandra outliers. Therefore, I highly recommend the last option, which leverages the initial AA structure.

Most of cases, protein AA structures are present before running CG simulations. The last option of backmapping CG protein structures into biologically sound AA ones is to use these reference AA structures. While the Martini protein FF allows global changes in protein tertiary structures, it is not expected to change any secondary structures by design. If CG-MD simulations are run with an elastic network model (ENM), it is more obvious that protein structure changes little. Therefore, one can copy a rigid domain of an atomistic structure and align it with its corresponding CG counterpart. Loops that connect the rigid domains can be built using `mstool.LoopModeler` or other loop modelers. For instance, T-cell intracellular antigen 1 has two rigid domains, connected by a single loop. Therefore, it is reasonable to use the reference AA structure of the two domains in the backmapping process. The backmapped structure using this approach has fewer Ramachandran outliers (Fig. 6D).

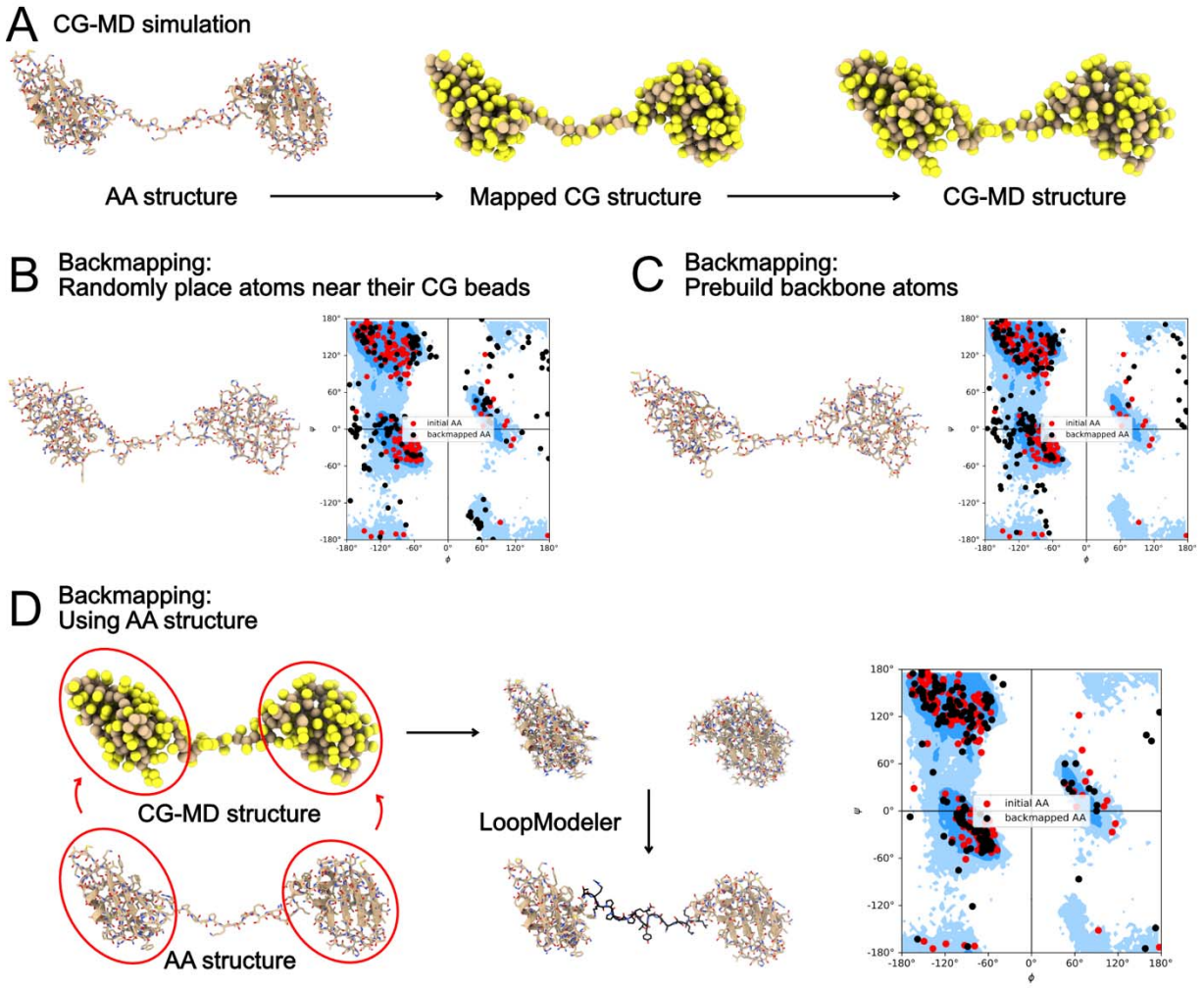


Figure 6. Protein backmapping. (A) CG-MD simulation of T-cell intracellular antigen 1. (B) First option. (C) Second option (default in `mstool.Ungroup`). (D) Third option using AA structure and `mstool.LoopModeler`.

Membrane Protein and Lipid Backmapping

Backmapping a membrane protein and lipid system is not different from backmapping either lipid (Fig. 5) or protein (Fig. 6). However, it is important to note that protein and lipid should not be backmapped separately and then combined because a resulting structure will likely have unphysical contact between protein and lipid. All the atoms should be present during REM to get a final backmapped structure with no bad contacts. In this section, I used outer membrane protein F (PDB: 2OMF),[50] facilitating the diffusion of molecules in the outer membrane of *E. coli*, as an example. A short CG-MD simulation was performed to stray away from an initial CG structure (Fig. 7A).

The first approach randomly places protein and lipid atoms near their corresponding CG beads. The second option, the default, is the same as the first option except for protein backbone atoms, prebuilt based on three consecutive protein backbone beads in the ungrouping step. Both approaches gave a reasonable structure for the protein;

However, as seen in the previous session, there were non-negligible Ramachandran outliers with these two options (Fig. 7B and Fig. 7C).

Considering the minimal conformational changes of the protein during the CG-MD simulation, it is reasonable to align the initial protein AA structure to the final CG structure and ungroup lipid AA atoms from lipid CG beads. However, because the initial AA protein structure is slightly different from the final CG protein structure, there are inevitably bad contacts between the initial AA protein atoms aligned to the final CG protein structure and lipid atoms ungrouped from CG lipid beads. Fortunately, REM is capable of relaxing such systems with unphysical contacts. The Ramachandran plot of a backmapped structure using the third option (Fig. 7D) showed fewer outliers compared to the first two options (Fig. 7B and Fig. 7C).

In the third option, protein topology is built by openMM during REM. If an input protein structure is not complete, has a modified protein residue such as glycosylation or phosphorylation, or has a ligand, openMM cannot create a protein topology. To deal with such cases, `mstool` has the last option to treat an input protein structure as a space-filler. In this case, the tool does not apply an AA protein FF to a provided protein structure but creates ENM for an input structure with moderate LJ parameters assigned to input protein atoms. During REM, non-protein atoms do not overlap protein atoms while protein maintains its position. Therefore, the Ramachandran plot of the final structure is identical to the initial AA protein structure (Fig. 7E); However, there are no bad contacts between protein and lipid atoms in a backmapped structure. The procedure of this option is illustrated in Fig. 7F.

`mstool` can work for larger and more complicated systems. Heroic backmapping of lipid droplet biogenesis is shown in Fig. 7G. It is a gigantic spherical bilayer with a diameter of 60 nm that shows seipin-mediated lipid droplet maturation.[51] Protein is a seipin oligomer (PDB: 6DS5),[52] facilitating triolein nucleation and proper lipid droplet maturation. This CG system is also a good example of protein backmapping using `mstool.LoopModeler`. The luminal domain of the seipin oligomer is rigid and changes little during CG-MD simulations. Therefore, I cut the luminal part of the AA protein structure and aligned it to the CG protein structure. In contrast, the transmembrane segments were flexible and became open up during CG-MD.[51] Therefore, each transmembrane segment of the AA structure was aligned into its corresponding CG structure. A total of 22 loop structures that connect the luminal domain and the transmembrane segments were made using `mstool.LoopModeler` all at once. This way, while the AA structure of the luminal domain and transmembrane segments were preserved, the backmapped model also could capture the opening of seipin transmembrane segments during lipid droplet maturation.

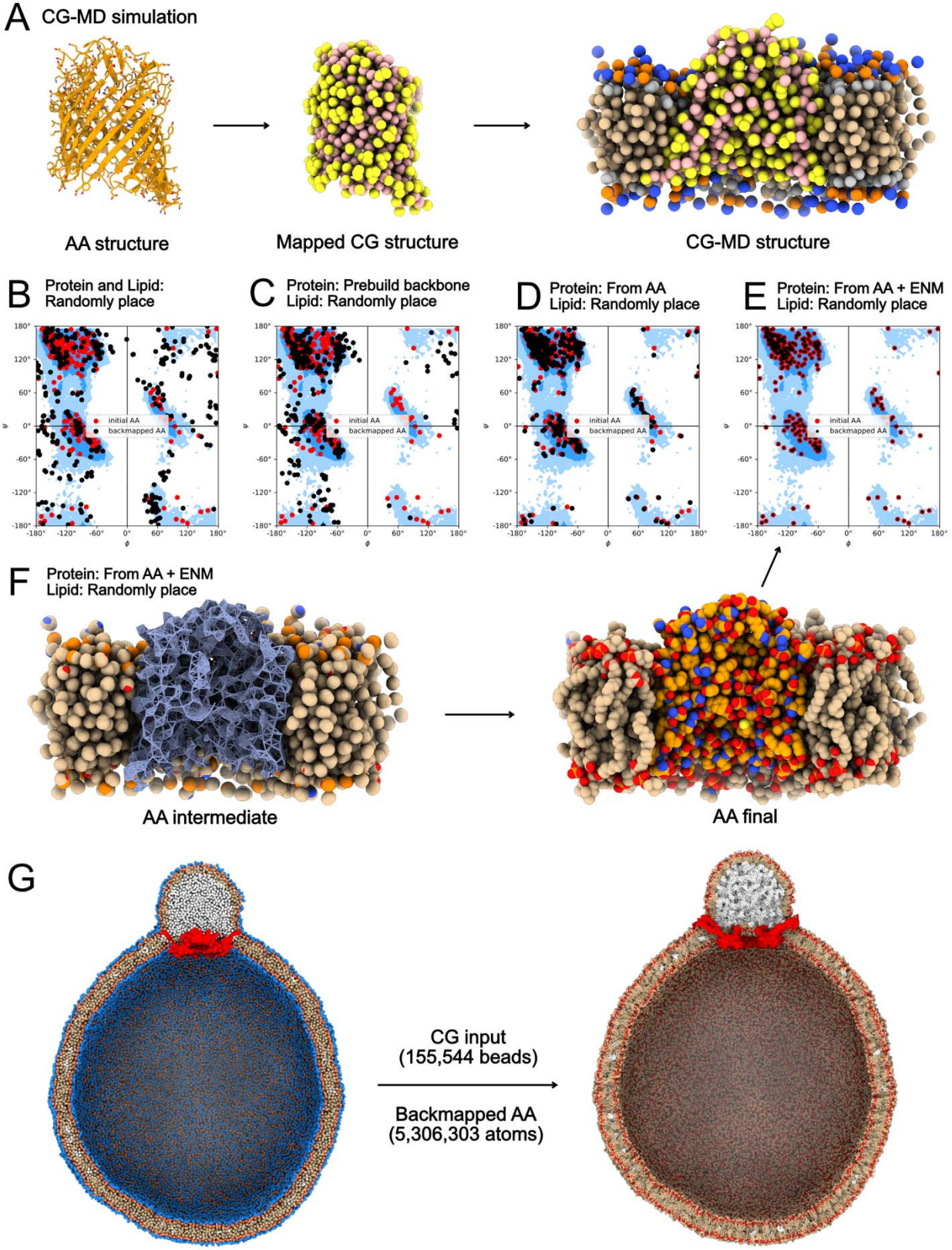


Figure 7. Lipid and protein backmapping. (A) CG-MD simulation of outer membrane protein F. (B-E) Ramachandran plots of the backmapped protein structures using

various options. (F) Workflow that treats an input protein structure as a space-filling rock structure. (G) Heroic backmapping of lipid droplet biogenesis model. Red protein is seipin, and white is triolein.

DISCUSSION

CG-MD simulations are helpful in understanding complex biological processes by allowing access to time and length scales beyond the scales typical of AA-MD simulations. However, CG models lack atomistic details critical in understanding protein structures, protein-lipid interactions, and protein-ligand interactions. Therefore, converting CG structures into AA ones in a biologically and physically sound manner is an important tool in the field of CG modeling of biophysical systems.

In this work, I have developed a backmapping tool that converts CG structures into AA ones and implemented it into a package called `mstool`. The procedure consists of three steps for resolution conversion. First, `mstool.Ungroup` simply places atoms near their corresponding beads. Second, `mstool.REM` applies a modified FF to a system and then performs energy minimization. The modified FF replaces the positive part of LJ with a cosine function and adds additional torsions to defined isomers. The former modification ensures numerical stability during energy minimization, and the latter does consistency in isomeric properties. Finally, a backmapped structure is reviewed by `mstool.CheckStructure`.

These three steps are decoupled. Therefore, each step can be combined with other backmapping tools. For instance, one can consider using approaches to create a better intermediate AA structure before energy minimization, such as fragment alignment[16,17] or geometrical projection.[23] Once an intermediate structure is obtained using either of these approaches or other methods, `mstool.REM` can relax the structure. Obviously, one can backmap CG structures into AA structures with other backmapping tools and then review flipped isomers with `mstool.CheckStructure`.

It should be noted that `mstool.REM` can be used for other purposes. For example, if an AA structure has multiple bad contacts, standard energy minimization will fail because of high potential energy. In such cases, `mstool.REM` can help resolve all the unphysical contacts so that standard energy minimization can be run afterward.

There are potential applications of this backmapping procedure that leverage the efficiency and fast equilibration of CG models. For example, `mstool.LoopModeler` can build a structure of missing loops without using structure templates. It is powerful to model any lengths of missing and multiple loops simultaneously. It is also simple to use, taking only a fasta file of protein chains and a protein structure file. The loop modeler automatically detects which parts are not present in the provided AA structure (except for the missing N-termini and C-termini) by comparing the sequence and residue number and builds loop structures. The details of the workflow and quality of loops will be discussed in a separate paper.

Another application is a membrane builder. CG membranes are easier to construct and equilibrate than AA membranes. One can build a CG membrane with a desired lipid composition at the Martini resolution, equilibrate the membrane at the CG level, and then backmap into the AA one. A prototype of a membrane builder is included in the repository.

CONCLUSIONS

I present a robust and flexible backmapping tool with a minimal user input set: mapping and isomeric information. Leveraging the efficacy of the modified FF, the tool is capable of converting CG systems to AA ones.

ACKNOWLEDGMENTS

I thank my Ph.D. advisor, Gregory A. Voth, for his mentorship and support. I am appreciative of my coworkers at D. E. Shaw Research for helpful discussion, Avi Robinson-Mosher, Peter Skopp, Robert McGibbon, Stefano Piana-Agostinetti, Qi Wang, Virginia Jiang, Dan Kozuch, Jim Valcourt, Parker de Waal, Jacob Pessin, and Brent Gregersen. Finally, I appreciate Won Hee Ryu, Gregory A. Voth, and anonymous reviewers for their time to review this paper.

REFERENCES

(1) Kmiecik, S.; Gront, D.; Kolinski, M.; Wieteska, L.; Dawid, A. E.; Kolinski, A. Coarse-Grained Protein Models and Their Applications. *Chem. Rev.* **2016**, *116* (14), 7898–7936. https://doi.org/10.1021/acs.chemrev.6b00163.

(2) Marrink, S. J.; Corradi, V.; Souza, P. C. T.; Ingólfsson, H. I.; Tieleman, D. P.; Sansom, M. S. P. Computational Modeling of Realistic Cell Membranes. *Chem. Rev.* **2019**, *119* (9), 6184–6226. https://doi.org/10.1021/acs.chemrev.8b00460.

(3) Corradi, V.; Sejdiu, B. I.; Mesa-Galloso, H.; Abdizadeh, H.; Noskov, S. Yu.; Marrink, S. J.; Tieleman, D. P. Emerging Diversity in Lipid–Protein Interactions. *Chem. Rev.* **2019**, *119* (9), 5775–5848. https://doi.org/10.1021/acs.chemrev.8b00451.

(4) Jin, J.; Pak, A. J.; Durumeric, A. E. P.; Loose, T. D.; Voth, G. A. Bottom-up Coarse-Graining: Principles and Perspectives. *J. Chem. Theory Comput.* **2022**, *18* (10), 5759–5791. https://doi.org/10.1021/acs.jctc.2c00643.

(5) Kim, S.; Swanson, J. M. J.; Voth, G. A. Computational Studies of Lipid Droplets. *J. Phys. Chem. B* **2022**, *126* (11), 2145–2154. https://doi.org/10.1021/acs.jpcb.2c00292.

(6) Álvarez, D.; Sapia, J.; Vanni, S. Computational Modeling of Membrane Trafficking Processes: From Large Molecular Assemblies to Chemical Specificity. *Current Opinion in Cell Biology* **2023**, *83*, 102205. https://doi.org/10.1016/j.ceb.2023.102205.

(7) Simunovic, M.; Srivastava, A.; Voth, G. A. Linear Aggregation of Proteins on the Membrane as a Prelude to Membrane Remodeling. *Proceedings of the National Academy of Sciences* **2013**, *110* (51), 20396–20401. https://doi.org/10.1073/pnas.1309819110.

(8) Jarin, Z.; Pak, A. J.; Bassereau, P.; Voth, G. A. Lipid-Composition-Mediated Forces Can Stabilize Tubular Assemblies of I-BAR Proteins. *Biophysical Journal* **2021**, *120* (1), 46–54. https://doi.org/10.1016/j.bpj.2020.11.019.

(9) Yu, A.; Pak, A. J.; He, P.; Monje-Galvan, V.; Casalino, L.; Gaieb, Z.; Dommer, A. C.; Amaro, R. E.; Voth, G. A. A Multiscale Coarse-Grained Model of the SARS-CoV-2 Virion. *Biophysical Journal* **2021**, *120* (6), 1097–1104. https://doi.org/10.1016/j.bpj.2020.10.048.

(10) Yu, A.; Lee, E. M. Y.; Briggs, J. A. G.; Ganser-Pornillos, B. K.; Pornillos, O.; Voth, G. A. Strain and Rupture of HIV-1 Capsids during Uncoating. *Proceedings of the National Academy of Sciences* **2022**, *119* (10), e2117781119. https://doi.org/10.1073/pnas.2117781119.

(11) Peter, C.; Kremer, K. Multiscale Simulation of Soft Matter Systems – from the Atomistic to the Coarse-Grained Level and Back. *Soft Matter* **2009**, *5* (22), 4357. https://doi.org/10.1039/b912027k.

(12) Badaczewska-Dawid, A. E.; Kolinski, A.; Kmiecik, S. Computational Reconstruction of Atomistic Protein Structures from Coarse-Grained Models. *Computational and Structural Biotechnology Journal* **2020**, *18*, 162–176. https://doi.org/10.1016/j.csbj.2019.12.007.

(13) Milik, M.; Kolinski, A.; Skolnick, J. Algorithm for Rapid Reconstruction of Protein Backbone from Alpha Carbon Coordinates. *J. Comput. Chem.* **1997**, *18* (1), 80–85. https://doi.org/10.1002/(SICI)1096-987X(19970115)18:1<80::AID-JCC8>3.0.CO;2-W.

(14) Rotkiewicz, P.; Skolnick, J. Fast Procedure for Reconstruction of Full-atom Protein Models from Reduced Representations. *J Comput Chem* **2008**, *29* (9), 1460–1465. https://doi.org/10.1002/jcc.20906.

(15) Shimizu, M.; Takada, S. Reconstruction of Atomistic Structures from Coarse-Grained Models for Protein–DNA Complexes. *J. Chem. Theory Comput.* **2018**, *14* (3), 1682–1694. https://doi.org/10.1021/acs.jctc.7b00954.

(16) Stansfeld, P. J.; Sansom, M. S. P. From Coarse Grained to Atomistic: A Serial Multiscale Approach to Membrane Protein Simulations. *J. Chem. Theory Comput.* **2011**, *7* (4), 1157–1166. https://doi.org/10.1021/ct100569y.

(17) Vickery, O. N.; Stansfeld, P. J. CG2AT2: An Enhanced Fragment-Based Approach for Serial Multi-Scale Molecular Dynamics Simulations. *J. Chem. Theory Comput.* **2021**, *17* (10), 6472–6482. https://doi.org/10.1021/acs.jctc.1c00295.

(18) Marrink, S. J.; De Vries, A. H.; Mark, A. E. Coarse Grained Model for Semiquantitative Lipid Simulations. *J. Phys. Chem. B* **2004**, *108* (2), 750–760. https://doi.org/10.1021/jp036508g.

(19) Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; de Vries, A. H. The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations. *J. Phys. Chem. B* **2007**, *111* (27), 7812–7824. https://doi.org/10.1021/jp071097f.

(20) Monticelli, L.; Kandasamy, S. K.; Periole, X.; Larson, R. G.; Tieleman, D. P.; Marrink, S.-J. The MARTINI Coarse-Grained Force Field: Extension to Proteins. *J. Chem. Theory Comput.* **2008**, *4* (5), 819–834. https://doi.org/10.1021/ct700324x.

(21) De Jong, D. H.; Singh, G.; Bennett, W. F. D.; Arnarez, C.; Wassenaar, T. A.; Schäfer, L. V.; Periole, X.; Tieleman, D. P.; Marrink, S. J. Improved Parameters for the Martini Coarse-Grained Protein Force Field. *J. Chem. Theory Comput.* **2013**, *9* (1), 687–697. https://doi.org/10.1021/ct300646g.

(22) Lombardi, L. E.; Martí, M. A.; Capece, L. CG2AA: Backmapping Protein Coarse-Grained Structures. *Bioinformatics* **2016**, *32* (8), 1235–1237. https://doi.org/10.1093/bioinformatics/btv740.

(23) Wassenaar, T. A.; Pluhackova, K.; Böckmann, R. A.; Marrink, S. J.; Tieleman, D. P. Going Backward: A Flexible Geometric Approach to Reverse Transformation from Coarse Grained to Atomistic Models. *J. Chem. Theory Comput.* **2014**, *10* (2), 676–690. https://doi.org/10.1021/ct400617g.

(24) Li, W.; Burkhart, C.; Polińska, P.; Harmandaris, V.; Doxastakis, M. Backmapping Coarse-Grained Macromolecules: An Efficient and Versatile Machine Learning Approach. *The Journal of Chemical Physics* **2020**, *153* (4), 041101. https://doi.org/10.1063/5.0012320.

(25) Louison, K. A.; Dryden, I. L.; Laughton, C. A. GLIMPS: A Machine Learning Approach to Resolution Transformation for Multiscale Modeling. *J. Chem. Theory Comput.* **2021**, *17* (12), 7930–7937. https://doi.org/10.1021/acs.jctc.1c00735.

(26) Christofi, E.; Chazirakis, A.; Chrysostomou, C.; Nicolaou, M. A.; Li, W.; Doxastakis, M.; Harmandaris, V. A. Deep Convolutional Neural Networks for Generating Atomistic Configurations of Multi-Component Macromolecules from Coarse-Grained Models. *The Journal of Chemical Physics* **2022**, *157* (18), 184903. https://doi.org/10.1063/5.0110322.

(27) Jones, M. S.; Shmilovich, K.; Ferguson, A. L. DiAMoNDBack: Diffusion-Denoising Autoregressive Model for Non-Deterministic Backmapping of C{\alpha} Protein Traces. arXiv July 23, 2023. http://arxiv.org/abs/2307.12451 (accessed 2023-10-01).

(28) Huang, J.; Rauscher, S.; Nawrocki, G.; Ran, T.; Feig, M.; De Groot, B. L.; Grubmüller, H.; MacKerell, A. D. CHARMM36m: An Improved Force Field for Folded and Intrinsically

Disordered Proteins. *Nat Methods* **2017**, *14* (1), 71–73. https://doi.org/10.1038/nmeth.4067.

(29) Klauda, J. B.; Venable, R. M.; Freites, J. A.; O'Connor, J. W.; Tobias, D. J.; Mondragon-Ramirez, C.; Vorobyov, I.; MacKerell, A. D. Jr.; Pastor, R. W. Update of the CHARMM All-Atom Additive Force Field for Lipids: Validation on Six Lipid Types. *J. Phys. Chem. B* **2010**, *114* (23), 7830–7843. https://doi.org/10.1021/jp101759q.

(30) Eastman, P.; Friedrichs, M. S.; Chodera, J. D.; Radmer, R. J.; Bruns, C. M.; Ku, J. P.; Beauchamp, K. A.; Lane, T. J.; Wang, L.-P.; Shukla, D.; Tye, T.; Houston, M.; Stich, T.; Klein, C.; Shirts, M. R.; Pande, V. S. OpenMM 4: A Reusable, Extensible, Hardware Independent Library for High Performance Molecular Simulation. *J. Chem. Theory Comput.* **2013**, *9* (1), 461–469. https://doi.org/10.1021/ct300857j.

(31) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L.-P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; Wiewiora, R. P.; Brooks, B. R.; Pande, V. S. OpenMM 7: Rapid Development of High Performance Algorithms for Molecular Dynamics. *PLOS Computational Biology* **2017**, *13* (7), e1005659. https://doi.org/10.1371/journal.pcbi.1005659.

(32) Liu, D. C.; Nocedal, J. On the Limited Memory BFGS Method for Large Scale Optimization. *Mathematical Programming* **1989**, *45* (1–3), 503–528. https://doi.org/10.1007/BF01589116.

(33) Bowers, K. J.; Chow, E.; Xu, H.; Dror, R. O.; Eastwood, M. P.; Gregersen, B. A.; Klepeis, J. L.; Kolossvary, I.; Moraes, M. A.; Sacerdoti, F. D.; Salmon, J. K.; Shan, Y.; Shaw, D. E. Scalable Algorithms for Molecular Dynamics Simulations on Commodity Clusters. In *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*; SC '06; Association for Computing Machinery: New York, NY, USA, 2006; pp 84-es. https://doi.org/10.1145/1188455.1188544.

(34) Chow, K.-H.; Ferguson, D. M. Isothermal-Isobaric Molecular Dynamics Simulations with Monte Carlo Volume Sampling. *Computer Physics Communications* **1995**, *91* (1–3), 283–289. https://doi.org/10.1016/0010-4655(95)00059-O.

(35) Åqvist, J.; Wennerström, P.; Nervall, M.; Bjelic, S.; Brandsdal, B. O. Molecular Dynamics Simulations of Water and Biomolecules with a Monte Carlo Constant Pressure Algorithm. *Chemical Physics Letters* **2004**, *384* (4–6), 288–294. https://doi.org/10.1016/j.cplett.2003.12.039.

(36) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. A Smooth Particle Mesh Ewald Method. *The Journal of Chemical Physics* **1995**, *103* (19), 8577–8593. https://doi.org/10.1063/1.470117.

(37) Eastman, P.; Pande, V. S. Constant Constraint Matrix Approximation: A Robust, Parallelizable Constraint Method for Molecular Simulations. *J. Chem. Theory Comput.* **2010**, *6* (2), 434–437. https://doi.org/10.1021/ct900463w.

(38) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. MDAnalysis: A Toolkit for the Analysis of Molecular Dynamics Simulations. *Journal of Computational Chemistry* **2011**, *32* (10), 2319–2327. https://doi.org/10.1002/jcc.21787.

(39) Gowers, R.; Linke, M.; Barnoud, J.; Reddy, T.; Melo, M.; Seyler, S.; Domański, J.; Dotson, D.; Buchoux, S.; Kenney, I.; Beckstein, O. MDAnalysis: A Python Package for the Rapid Analysis

of Molecular Dynamics Simulations; Austin, Texas, 2016; pp 98–105. https://doi.org/10.25080/Majora-629e541a-00e.

(40) Campomanes, P.; Prabhu, J.; Zoni, V.; Vanni, S. Recharging Your Fats: CHARMM36 Parameters for Neutral Lipids Triacylglycerol and Diacylglycerol. *Biophysical Reports* **2021**, *1* (2), 100034. https://doi.org/10.1016/j.bpr.2021.100034.

(41) Goddard, T. D.; Huang, C. C.; Meng, E. C.; Pettersen, E. F.; Couch, G. S.; Morris, J. H.; Ferrin, T. E. UCSF ChimeraX: Meeting Modern Challenges in Visualization and Analysis: UCSF ChimeraX Visualization System. *Protein Science* **2018**, *27* (1), 14–25. https://doi.org/10.1002/pro.3235.

(42) Pettersen, E. F.; Goddard, T. D.; Huang, C. C.; Meng, E. C.; Couch, G. S.; Croll, T. I.; Morris, J. H.; Ferrin, T. E. UCSF ChimeraX: Structure Visualization for Researchers, Educators, and Developers. *Protein Science* **2021**, *30* (1), 70–82. https://doi.org/10.1002/pro.3943.

(43) Seelig, J. Deuterium Magnetic Resonance: Theory and Application to Lipid Membranes. *Quart. Rev. Biophys.* **1977**, *10* (3), 353–418. https://doi.org/10.1017/S0033583500002948.

(44) Caffalette, C. A.; Corey, R. A.; Sansom, M. S. P.; Stansfeld, P. J.; Zimmer, J. A Lipid Gating Mechanism for the Channel-Forming O Antigen ABC Transporter. *Nat Commun* **2019**, *10* (1), 824. https://doi.org/10.1038/s41467-019-08646-8.

(45) Grime, J. M. A.; Madsen, J. J. Efficient Simulation of Tunable Lipid Assemblies Across Scales and Resolutions. arXiv October 11, 2019. http://arxiv.org/abs/1910.05362 (accessed 2023-08-14).

(46) Kim, S.; Li, C.; Farese, R. V.; Walther, T. C.; Voth, G. A. Key Factors Governing Initial Stages of Lipid Droplet Formation. *J. Phys. Chem. B* **2022**, *126* (2), 453–462. https://doi.org/10.1021/acs.jpcb.1c09683.

(47) Marrink, S. J.; Risselada, J.; Mark, A. E. Simulation of Gel Phase Formation and Melting in Lipid Bilayers Using a Coarse Grained Model. *Chemistry and Physics of Lipids* **2005**, *135* (2), 223–244. https://doi.org/10.1016/j.chemphyslip.2005.03.001.

(48) Ingólfsson, H. I.; Melo, M. N.; van Eerden, F. J.; Arnarez, C.; Lopez, C. A.; Wassenaar, T. A.; Periole, X.; de Vries, A. H.; Tieleman, D. P.; Marrink, S. J. Lipid Organization of the Plasma Membrane. *J. Am. Chem. Soc.* **2014**, *136* (41), 14554–14559. https://doi.org/10.1021/ja507832e.

(49) Wang, I.; Hennig, J.; Jagtap, P. K. A.; Sonntag, M.; Valcárcel, J.; Sattler, M. Structure, Dynamics and RNA Binding of the Multi-Domain Splicing Factor TIA-1. *Nucleic Acids Research* **2014**, *42* (9), 5949–5966. https://doi.org/10.1093/nar/gku193.

(50) Cowan, S. W.; Schirmer, T.; Rummel, G.; Steierf, M.; Ghosh, R.; Pauptitt, R. A.; Jansonius, J. N.; Rosenbusch, J. P. Crystal Structures Explain Functional Properties of Two E. Coli Porins. *Nature* **1992**, *358*, 727–733.

(51) Kim, S.; Chung, J.; Arlt, H.; Pak, A. J.; Farese, R. V.; Walther, T. C.; Voth, G. A. Seipin Transmembrane Segments Critically Function in Triglyceride Nucleation and Lipid Droplet Budding from the Membrane. *eLife* **2022**, *11*, e75808. https://doi.org/10.7554/eLife.75808.

(52) Yan, R.; Qian, H.; Lukmantara, I.; Gao, M.; Du, X.; Yan, N.; Yang, H. Human SEIPIN Binds Anionic Phospholipids. *Developmental Cell* **2018**, *47* (2), 248-256.e4. https://doi.org/10.1016/j.devcel.2018.09.010.