PACKMOL-Memgen: A simple-to-use generalized workflow for membrane-protein/lipid-bilayer system building

Stephan Schott-Verdugo[1,2] and Holger Gohlke[1,3*]

[1]Institute for Pharmaceutical and Medicinal Chemistry, Heinrich Heine University Düsseldorf, 40225 Düsseldorf, Germany

[2]Centro de Bioinformática y Simulación Molecular (CBSM), Faculty of Engineering, University of Talca, 1 Poniente 1141, Casilla 721, Talca, Chile

[3]John von Neumann Institute for Computing (NIC), Jülich Supercomputing Centre (JSC) & Institute for Complex Systems - Structural Biochemistry (ICS 6), Forschungszentrum Jülich GmbH, 52425 Jülich, Germany

[*]Address: Universitätsstr. 1, 40225 Düsseldorf, Germany.

Phone: (+49) 211 81 13662; Fax: (+49) 211 81 13847

E-mail: gohlke@uni-duesseldorf.de

ABSTRACT We present PACKMOL-Memgen, a simple-to-use generalized workflow for automated building of membrane-protein/lipid-bilayer systems based on open-source tools including packmol, memembed, pdbremix, and AmberTools. Compared to webinterface-based related tools, PACKMOL-Memgen allows setting up multiple configurations of a system in a user-friendly and efficient manner within minutes. The generated systems are well packed and thus well suited as starting configurations in MD simulations under periodic boundary conditions, requiring only moderate equilibration times. PACKMOL-Memgen is distributed with AmberTools, runs on most computing platforms, and its output can also be used for CHARMM or adapted to other molecular simulation packages.

INTRODUCTION

More than half of the current drug targets are integral membrane proteins, although they only represent about a third of the human proteome[1]. This highlights the functional importance of membrane proteins for physiology: Membrane proteins participate in essential processes such as ion and molecule transport, signal transduction and enzymatic catalysis[2]. These characteristics, together with an increase in available structures and computational resources, has motivated the proliferation of computational studies focusing on this type of proteins[3] and the lipid bilayer they are embedded in or associated with[4].

One of the challenges of studying membrane systems or membrane proteins by computational techniques is building a system that is representative of the *in vivo* conditions. This is particularly true when one considers the anisotropic nature of the lipid bilayer, making the assembly of lipids to form a membrane and the placement of a protein within or at a membrane a non-trivial task ("membrane-protein packing problem") [5-6]. Multiple tools have been developed to help in the process of generating such systems [5-6]. In general, they can be subdivided into methods that pack the systems from scratch (self-assembly, grid-, or geometry-based) and methods that use pre-equilibrated membrane systems to merge them with the components of interest (insertion or replacement methods)[5]. The former have the advantage of letting the user generate any composition they need, at the cost of requiring longer simulation times in order to equilibrate the membrane system. This challenge is partially overcome by the latter methods, at the cost of having only a limited amount of pre-equilibrated membrane patches available, while still having to equilibrate the protein-membrane interface.

Popular tools that are widely used to solve the membrane-protein packing problem are CHARMM-GUI[7], VMD (through the Membrane plugin)[8], Maestro[9], and Packmol[10]. Of these tools, CHARMM-GUI stands out for having multiple lipids available and a user-friendly web interface, making the packing process easier for newcomers but at the same time dependent on

the web interface [7]. This dependency turns into a challenge when a user needs to generate multiple systems with different proteins or different membrane compositions, or requires different starting configurations, as the process becomes time-consuming. Packmol stands out for having functionalities that allow building complex and intricate simulation systems[10]. These functionalities build upon solving the packing problem with the GENCAN optimizer, which minimizes an objective function that describes the molecular overlap[11]. To do so, appropriate PDB files for the individual system components and system-specific geometric constraints to restrict the packing are required as input. Together with the requirement of command line-usage, this makes applying Packmol difficult for non-expert users.

Therefore, we developed PACKMOL-Memgen, a Python-based program that uses Packmol as the packing engine but wraps the main procedures required to build complex membrane systems, such that only single-line commands of the user are required. Further advantages of PACKMOL-Memgen are that its protocol is easily parallelizable when working with multiple proteins, membrane compositions, or starting configurations, the packing is achieved on the order of minutes for common system sizes, and it is user-friendly and easy to grasp. PACKMOL-Memgen generates formatted PDB files that can be used both with AMBER's LeAP with the Lipid17 forcefield[12-13] and with VMD's psfgen with the CHARMM c36 forcefield[14].

The program is distributed under a GPL license together with AmberTools[13] (http://ambermd.org/AmberTools.php).

WORKFLOW DESCRIPTION

The general workflow of PACKMOL-Memgen is depicted in Figure 1. It follows similar principles as the CHARMM-GUI membrane builder[7]. The user can decide to either pack a membrane-only system, or embed a protein structure into the membrane. For the latter, a PDB

file of the protein structure following the conventional formatting is used as input (step 1). The user needs to make sure that this PDB file represents the system of interest adequately, e.g., the protein configuration is representative of the process to be studied, ligands of interest are included (which can be kept for further processing with the flag `--keep`), and/or relevant protonation states are set. Next, unless otherwise specified, hydrogens are added to the input structure with **reduce** [15], as this helps to correctly determine the protein volume in later steps (step 2). This step can be skipped by the user if the structure was previously protonated. To determine the placement of the protein within the membrane, the structure is oriented with **memembed** [16], which employs a knowledge-based potential function to mimic embedding of the protein in a membrane slab centered at the origin of the coordinate system (step 3). By default, the orientation is optimized during five cycles of the memembed genetic algorithm.

From the placement of the protein, the size limits of the system to be packed are determined (step 4). Unless specified differently by the user, a distance of at least 15 Å to the boundaries in the x-y plane (`--dist`) and a water slab with a thickness of at least 17.5 Å above and below the membrane are used (`--dist_wat`). The determined geometry is used to prepare the placement of lipids, waters, ions, and possible additional solutes, which, together with specified molecule concentrations (e.g., `--sol_con` or `--salt_con`) and lipid ratios (`--lipids`, `--ratio`), determines the number of molecules to be used. The placement of the center of the membrane slab is set to be on the z-coordinate origin, with the lipids oriented parallel to this axis and a leaflet width of 23 Å by default (`--leaflet`). To ensure the correct leaflet orientation, restricted volumes for I) the phosphorous atom together with the terminal atom of the headgroup (`--headplane`) and II) the carbon atoms of the terminal methyl groups of the tails (`--tailplane`) are set (step 4.1).

The default behavior of the program is to estimate the number of lipids per leaflet based on the calculated leaflet area and the area per lipid (APL). APL values are obtained for systems at

or close to 300 K from the literature[17-22], taken from the maximum value of lipids with the same headgroup, or set to 75 Å^2 otherwise. If lipid mixtures are used, the APL is computed as a weighted average according to the composition ratio. The number of water molecules in the upper and lower water slabs is determined based on the respective volume and the density of water. The volume occupied by the protein in each water slab, as well as in the membrane leaflets, is calculated by a grid approach derived from **pdbremix** [23] (step 4.2). The algorithm adds equidistant grid points (0.5 Å by default) within the radius of the atoms in the structure, representing the volume of the protein as a sum of cubes, with the grid spacing as the cube edge. This volume is used to reduce the previously calculated number of molecules in the different system sections by the numbers of lipids and waters that would occupy the protein volume. For this, the lipid molecular volume is obtained from literature[22, 24-27] or estimated based on the length of the acyl chains as described previously[28]. If solutes are added, a similar approach is applied by using the pdbremix-computed volume to adapt the numbers of lipids or waters as appropriate. By default, potassium and chloride are added as counterions to reach electroneutrality with respect to the charges in the protein and the lipids. Extra ions can be included by specifying the `--salt` flag, and a concentration of 0.15 M in the water volume is used by default (`--salt_con`).

The user can select multiple lipids (`--lipids`) and define complex lipid mixtures at desired ratios (`--ratio`), including different compositions per leaflet. For this, corresponding colon-separated lists of lipids and ratios have to be provided, separated by two consecutive slashes ("`//`") in case a different composition per leaflet is desired. For example, `--lipids POPE:POPC:POPS//POPC:POPE --ratio 3:1:1//4:1` would add a lower leaflet with POPE, POPC, and POPS in a 3:1:1 ratio, and an upper leaflet with POPC and POPE in a 4:1 ratio. Table 1 lists lipids available in the current version, which can be extended with new lipids as long as parameters such as APL and molecular volume are provided. For example,

parameters for phosphoinositides and lysophospholipids have been added to the AmberTools19 PACKMOL-Memgen release, extending the range of lipids in the Lipid17 force field towards that available in other lipid force fields [29] (Table S1).

In addition, multiple bilayers can be generated by calling the corresponding flags once per bilayer. This is useful if, e.g., an electrochemical gradient is required to study a system as in the case of "computational electrophysiology"[30].

To initiate the molecule packing (step 5), the calculated number of molecules of each type and the geometrical constraints associated with each section of the system are written to a Packmol input file, and Packmol is started. If a protein is included, its position is considered fixed during the packing, and all other molecules will be packed such that they accommodate to the protein. By default, the system is packed with 20 iterations of the optimization algorithm per *molecule section* (or per entry in the Packmol input file) and 100 iterations system-wide. This is considerably less than the Packmol default value for iterations of 200 times the number of *molecule sections*, but is usually more than enough to obtain the best possible solution from the GENCAN algorithm for the given input constraints. This is evidenced by the fact that, after the system-wide packing starts, usually only few cycles are required for the objective function of Packmol to reach a plateau (see Figure S1 for an example). Still, as the number of iterations required to pack a system depends on the system size and the complexity of the mixture, the number of iterations can be specified by the user if required (`--nloop` and `--nloop_all`).

The Packmol PDB file output is, by default, transformed into an AMBER-compatible file using an adapted version of charmmlipid2amber.py from Benjamin Madej[13]; this step can be skipped if desired. The resulting assembled system has the best possible packing given the input molecules and geometrical restraints but might not be an overlap-free solution. Thus, prior to using the system in MD simulations, usually a thorough energy minimization is required, which, if using the AMBER suite of programs, should be performed with the CPU code of pmemd[31].

To simplify this step, functions to parametrize (using, by default, the ff14SB parameters [32]for the protein, LIPID17 from Skjevik *et al.*[12-13] for lipids, and the TIP3P water with AMBER 18 default ion parameters [33-35]) and minimize simple protein-membrane systems after the packing are included.

Example case 1: Bacteria-like membrane

As a first example, packing of a membrane bilayer composed of a mixture of DOPE:DOPG lipids with a 3:1 ratio is shown. This composition has been proposed as a representative model of the inner membrane of Gram-negative bacteria [36]. The packing of such a system can be performed with a single command line:

```
packmol-memgen --ratio 3:1 --lipids DOPE:DOPG --distxy_fix 100
```

Since no protein is included, a desired length in the xy dimension has to be set (`--distxy_fix`, using 100 Å for this particular example). The resulting packed system is depicted in Figure 2A. The whole packing process took less than 13 minutes on a single i5-4590 CPU. In order to evaluate the stability of this system, MD simulations were performed as described in the Supporting Information, and a representative structure after 500 ns of production time is shown in Figure 2B. Parameters including the APL[36], the S_{CD} order parameter[12, 37], membrane thickness[38], electron density[38-39], and the average lipid mean square displacement[12] correspond with values reported previously (see Supplementary Methods and Figures S2-S4); this result demonstrates that the packed system is appropriate for performing all-atom MD simulations without requiring extensive equilibration times. Along these lines, the average xy cross sectional area is less than 5% smaller than the size of the system at the beginning of the production run, suggesting that the amount of lipids and geometrical constraints used for the packing are close to the equilibrium values.

Example case 2: KcsA potassium channel

The packing is more challenging when a protein is included, because the other system components need to adapt to the protein during the equilibration phase of the MD simulations, and there may be different volumes in the membrane leaflets or water sections. As an example, we use the potassium channel of *Streptomyces lividans,* KcsA. The structure with PDB ID 1BL8 was capped in the termini and a protonated glutamic acid in position 71 was used, as reported previously[40-41]. As `--keep` was not used, crystallization ions were removed. The resulting packed system including the protein, lipids (DOPE:DOPG 3:1), water, and ions is shown in Figure 2C. The packing of this system was performed with the following command line:

```
packmol-memgen --pdb 1BL8.pdb --lipids DOPE:DOPG --ratio 3:1
```

The packing time is comparable to the one of the membrane-only example. A script to pack this system is included with the software as a test case. Furthermore, animation of the packing process performed by PACKMOL is shown in Videos S1 and S2. MD simulations were performed to investigate whether the packed system is suitable as a starting structure. A representative structure after 500 ns of production is shown in Figure 2D. The protein structure shows slight deviations during the MD simulations, as indicated by C_α-atom RMSD to the crystal structure < 2.5 Å and RMS average correlation reaching 1 Å within 200 ns (Figure S5). The membrane maintains its bilayer structure as in the case of simulations with lipids only (Figures S4A and B). The KcsA channel has been investigated by MD simulations in detail before, particularly regarding the binding and flux of potassium ions into and through the selectivity filter[42]. Yet, the exact mechanism is still a matter of debate[43-44]. While a thorough discussion of the dynamics of the protein and protein/potassium interactions is beyond the scope of the present work, the simulation results reveal that potassium stays bound mainly to the exit of the channel (Figures S4B and S6). These results resemble those of previously performed simulations of the protein started without potassium bound in the selectivity filter, where the

protein stayed in an inactivated state[45]. Additionally, the selectivity filter adopts a conformation closer to a structure crystallized at low K^+ concentrations where the selectivity filter is unoccupied [46] (Figure S7). This finding is consistent with the observation that the channel becomes inactivated due to carbonyl repulsions in the selectivity filter and the hydration that follows[45]. Overall, the simulations show previously described structural dynamics for this protein, indicating that the packed system is a suitable starting structure.

DISCUSSION

In this study, we developed PACKMOL-Memgen, a simple-to-use generalized workflow for automated building of membrane-protein/lipid-bilayer systems based on open-source tools including packmol, memembed, pdbremix, and AmberTools. As demonstrated, the built systems are well suited as starting configurations in MD simulations under periodic boundary conditions, requiring only moderate equilibration times. As discussed before[5-6], there is more than one way of building a membrane system for performing computational studies, and the choices may depend on the molecular simulation package and the force field used, or the user preference. Until recently, users of the AMBER biomolecular simulation programs were primarily dependent on webservers such as CHARMM-GUI to build membrane-protein/lipid-bilayer systems. While user-friendly, providing support for many lipids, including glycolipids and lipopolysaccharides[47], and yielding input for current molecular simulation packages[48-49], the use of a web interface becomes inconvenient if very many systems need to be built, e.g., for performing umbrella sampling simulations, when different membrane compositions are to be tested, or replicas are to be started from different configurations. A command line implementation helps to save time when such broad studies are performed, allowing to script the system generation and, thereby, reducing the chances to make input errors. As shown in the example cases, the PACKMOL-Memgen workflow is both user-friendly and efficient, generating well-packed starting structures in minutes for typical simulation systems.

While the workflow automatizes a large part of the building process, the user is still required to critically evaluate if the built system is representative from a molecular and physiological point of view. As an example, for packing a porin/lipid-bilayer system, one needs to consider the appropriate knowledge-based potential in memembed for the β-barrel type protein (available through the `--barrel` flag). Furthermore, since the packing of lipids depends

exclusively on reducing the molecular overlap, lipids may be inserted in the protein tunnel, which needs to be fixed manually prior to performing simulations.

PACKMOL-Memgen generates output with atom and residue names compatible with AMBER's Leap with the Lipid17 force field and with VMD's psfgen with the CHARMM c36 forcefield. Finally, the current implementation of the Windows Subsystem for Linux (WSL) in Windows 10 makes AmberTools and the included PACKMOL-Memgen workflow available on most computing platforms, including common Linux distributions and MacOS.

ACKNOWLEDGEMENTS

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Projektnummer 267205415 – SFB 1208 (project A03 to HG) and, in part, FOR 2518 (GO 1367/2-1). We are grateful for computational support and infrastructure provided by the "Zentrum für Informations- und Medientechnologie" (ZIM) at the Heinrich Heine University Düsseldorf and the computing time provided by the John von Neumann Institute for Computing (NIC) to HG on the supercomputer JURECA at Jülich Supercomputing Centre (JSC) (user IDs: HDD18 and HKF7). We acknowledge the developers of AMBER, Packmol, memembed, and PDBREMIX for making their software available to the scientific community

SUPPORTING INFORMATION

The Supporting Information is available free of charge on the ACS Publications website at DOI: [TO BE FILLED IN LATER].

Supplementary Methods on the molecular dynamics simulations and the lipid diffusion calculation; Supplementary Figures showing the changes in the target function during the packing process (Figure S1), plots of area per lipid and membrane thickness (Figure S2), lipid order parameter (Figure S3), electron density profile (Figure S4), RMSD and RAC of the performed MD simulations (Figure S5), and 3D histograms of the potassium ions and water molecules (Figure S6), and structural comparison with crystal structures of the KcsA channel (Figure S7); Supplementary Tables showing additional Lipid17 headgroups (Table S1) and the calculated lipid diffusion with periodic boundary corrections (Table S2); Supplementary Videos showing the packing process of the KcsA channel from two different perspectives (Video S1, S2).

REFERENCES

1. Rask-Andersen, M.; Almen, M. S.; Schioth, H. B., Trends in the exploitation of novel drug targets. *Nat. Rev. Drug Discov.* **2011,** *10*, 579-590.
2. Brown, D. G., Drug discovery strategies to outer membrane targets in Gram-negative pathogens. *Bioorg. Med. Chem.* **2016,** *24*, 6320-6331.
3. Almeida, J. G.; Preto, A. J.; Koukos, P. I.; Bonvin, A.; Moreira, I. S., Membrane proteins structures: A review on computational modeling tools. *Biochim. Biophys. Acta Biomembr.* **2017,** *1859*, 2021-2039.
4. Stansfeld, P. J.; Sansom, M. S., Molecular simulation approaches to membrane proteins. *Structure* **2011,** *19*, 1562-1572.
5. Sommer, B., Membrane Packing Problems: A short Review on computational Membrane Modeling Methods and Tools. *Computational and Structural Biotechnology Journal* **2013,** *5*, 1-13.
6. Javanainen, M.; Martinez-Seara, H., Efficient preparation and analysis of membrane and membrane protein systems. *Biochim. Biophys. Acta* **2016,** *1858*, 2468-2482.
7. Jo, S.; Lim, J. B.; Klauda, J. B.; Im, W., CHARMM-GUI Membrane Builder for Mixed Bilayers and Its Application to Yeast Membranes. *Biophys. J.* **2009,** *97*, 50-58.
8. Humphrey, W.; Dalke, A.; Schulten, K., VMD: Visual molecular dynamics. *J. Mol. Graphics* **1996,** *14*, 33-38.
9. *Maestro*, Schrödinger, LLC: New York, NY, 2019.
10. Martinez, L.; Andrade, R.; Birgin, E. G.; Martinez, J. M., PACKMOL: a package for building initial configurations for molecular dynamics simulations. *J. Comput. Chem.* **2009,** *30*, 2157-2164.
11. Martínez, J. M.; Martínez, L., Packing optimization for automated generation of complex system's initial configurations for molecular dynamics and docking. *J. Comput. Chem.* **2003,** *24*, 819-825.
12. Dickson, C. J.; Madej, B. D.; Skjevik, Å. A.; Betz, R. M.; Teigen, K.; Gould, I. R.; Walker, R. C.; Skjevik, A. A.; Betz, R. M.; Teigen, K.; Gould, I. R.; Walker, R. C., Lipid14: The Amber Lipid Force Field. *J. Chem. Theory Comput.* **2014,** *10*, 865-879.
13. D.A. Case, I. Y. B.-S., S.R. Brozell, D.S. Cerutti, T.E. Cheatham, III, V.W.D. Cruzeiro, T.A. Darden, R.E. Duke, D. Ghoreishi, M.K. Gilson, H. Gohlke, A.W. Goetz, D. Greene, R Harris, N. Homeyer, S. Izadi, A. Kovalenko, T. Kurtzman, T.S. Lee, S. LeGrand, P. Li, C. Lin, J. Liu, T. Luchko, R. Luo, D.J. Mermelstein, K.M. Merz, Y. Miao, G. Monard, C. Nguyen, H. Nguyen, I. Omelyan, A. Onufriev, F. Pan, R. Qi, D.R. Roe, A. Roitberg, C. Sagui, S. Schott-Verdugo, J. Shen, C.L. Simmerling, J. Smith, R. Salomon-Ferrer, J. Swails, R.C. Walker, J. Wang, H. Wei, R.M. Wolf, X. Wu, L. Xiao, D.M. York and P.A. Kollman *AMBER 2018*, University of California: San Francisco, 2018.
14. Best, R. B.; Zhu, X.; Shim, J.; Lopes, P. E.; Mittal, J.; Feig, M.; Mackerell, A. D., Jr., Optimization of the additive CHARMM all-atom protein force field targeting improved sampling of the backbone phi, psi and side-chain chi(1) and chi(2) dihedral angles. *J. Chem. Theory Comput.* **2012,** *8*, 3257-3273.
15. Word, J. M.; Lovell, S. C.; Richardson, J. S.; Richardson, D. C., Asparagine and glutamine: using hydrogen atom contacts in the choice of side-chain amide orientation. *J. Mol. Biol.* **1999,** *285*, 1735-1747.
16. Nugent, T.; Jones, D. T., Membrane protein orientation and refinement using a knowledge-based statistical potential. *BMC bioinformatics* **2013,** *14*, 1-10.
17. Hung, W. C.; Lee, M. T.; Chen, F. Y.; Huang, H. W., The condensing effect of cholesterol in lipid bilayers. *Biophys. J.* **2007,** *92*, 3960-3967.
18. Petrache, H. I.; Dodd, S. W.; Brown, M. F., Area per Lipid and Acyl Length Distributions in Fluid Phosphatidylcholines Determined by 2H NMR Spectroscopy. *Biophys. J.* **2000,** *79*, 3172-3192.
19. Kucerka, N.; Nagle, J. F.; Sachs, J. N.; Feller, S. E.; Pencer, J.; Jackson, A.; Katsaras, J., Lipid bilayer structure determined by the simultaneous analysis of neutron and x-ray scattering data. *Biophys. J.* **2008,** *95*, 2356-2367.
20. Hills, R. D.; McGlinchey, N., Model parameters for simulation of physiological lipids. *J. Comput. Chem.* **2016,** *37*, 1112-1118.

21. McIntosh, T. J.; Advani, S.; Burton, R. E.; Zhelev, D. V.; Needham, D.; Simon, S. a., Experimental tests for protrusion and undulation pressures in phospholipid bilayers. *Biochemistry* **1995,** *34*, 8520-8532.
22. Pan, J.; Heberle, F. A.; Tristram-Nagle, S.; Szymanski, M.; Koepfinger, M.; Katsaras, J.; Kučerka, N., Molecular structures of fluid phase phosphatidylglycerol bilayers as determined by small angle neutron and X-ray scattering. *Biochim. Biophys. Acta Biomembr.* **2012,** *1818*, 2135-2148.
23. Ho, B. pdbremix Git-Hub repository. https://github.com/boscoh/pdbremix.
24. Murugova, T. N.; Balgavy, P., Molecular volumes of DOPC and DOPS in mixed bilayers of multilamellar vesicles. *Phys. Chem. Chem. Phys.* **2014,** *16*, 18211-18216.
25. Greenwood, A. I.; Tristram-Nagle, S.; Nagle, J. F., Partial molecular volumes of lipids and cholesterol. *Chem. Phys. Lipids* **2006,** *143*, 1-10.
26. Nagle, J. F.; Tristram-Nagle, S., Structure of lipid bilayers. *Biochim. Biophys. Acta Biomembr.* **2000,** *1469*, 159-195.
27. Petrache, H. I.; Feller, S. E.; Nagle, J. F., Determination of component volumes of lipid bilayers from simulations. *Biophys. J.* **1997,** *72*, 2237-2242.
28. Koenig, B. W.; Gawrisch, K., Specific volumes of unsaturated phosphatidylcholines in the liquid crystalline lamellar phase. *Biochim. Biophys. Acta Biomembr.* **2005,** *1715*, 65-70.
29. Wu, E. L.; Cheng, X.; Jo, S.; Rui, H.; Song, K. C.; Davila-Contreras, E. M.; Qi, Y.; Lee, J.; Monje-Galvan, V.; Venable, R. M.; Klauda, J. B.; Im, W., CHARMM-GUI Membrane Builder toward realistic biological membrane simulations. *J. Comput. Chem.* **2014,** *35*, 1997-2004.
30. Kutzner, C.; Grubmuller, H.; de Groot, B. L.; Zachariae, U., Computational electrophysiology: the molecular dynamics of ion channel permeation and selectivity in atomistic detail. *Biophys. J.* **2011,** *101*, 809-17.
31. Le Grand, S.; Götz, A. W.; Walker, R. C., SPFP: Speed without compromise - A mixed precision model for GPU accelerated molecular dynamics simulations. *Comput. Phys. Commun.* **2013,** *184*, 374-380.
32. Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C., ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. *J. Chem. Theory Comput.* **2015,** *11*, 3696-3713.
33. Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L., Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983,** *79*, 926-935.
34. Joung, I. S.; Cheatham, T. E., 3rd, Determination of alkali and halide monovalent ion parameters for use in explicitly solvated biomolecular simulations. *J. Phys. Chem. B* **2008,** *112*, 9020-41.
35. Li, P.; Song, L. F.; Merz, K. M., Parameterization of highly charged metal ions using the 12-6-4 LJ-type nonbonded model in explicit water. *Journal of Physical Chemistry B* **2015,** *119*, 883-895.
36. Murzyn, K.; Róg, T.; Pasenkiewicz-Gierula, M., Phosphatidylethanolamine-phosphatidylglycerol bilayer as a model of the inner bacterial membrane. *Biophys. J.* **2005,** *88*, 1091-1103.
37. Pluhackova, K.; Kirsch, S. A.; Han, J.; Sun, L.; Jiang, Z.; Unruh, T.; Bockmann, R. A., A Critical Comparison of Biomembrane Force Fields: Structure and Dynamics of Model DMPC, POPC, and POPE Bilayers. *Journal of Physical Chemistry B* **2016,** *120*, 3888-3903.
38. Hung, W. C.; Lee, M. T., The interaction of melittin with E. coli membrane: The role of cardiolipin. *Chinese J Phys* **2006,** *44*, 137-149.
39. Liu, Y.; Nagle, J. F., Diffuse scattering provides material parameters and electron density profiles of biomembranes. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.* **2004,** *69*, 1-4.
40. Doyle, D. A.; Morais Cabral, J.; Pfuetzner, R. A.; Kuo, A.; Gulbis, J. M.; Cohen, S. L.; Chait, B. T.; MacKinnon, R., The structure of the potassium channel: molecular basis of K+ conduction and selectivity. *Science* **1998,** *280*, 69-77.
41. Bhate, M. P.; McDermott, A. E., Protonation state of E71 in KcsA and its role for channel collapse and inactivation. *Proc. Natl. Acad. Sci. U.S.A.* **2012,** *109*, 15265-15270.
42. Fowler, P. W.; Tai, K.; Sansom, M. S., The selectivity of K+ ion channels: testing the hypotheses. *Biophys. J.* **2008,** *95*, 5062-5072.
43. Heer, F. T.; Posson, D. J.; Wojtas-Niziurski, W.; Nimigean, C. M.; Bernèche, S., Mechanism of activation at the selectivity filter of the KcsA K+ channel. *eLife* **2017,** *6*, 1-18.

44. Kopfer, D. A.; Song, C.; Gruene, T.; Sheldrick, G. M.; Zachariae, U.; de Groot, B. L., Ion permeation in K(+) channels occurs by direct Coulomb knock-on. *Science* **2014,** *346*, 352-355.
45. Boiteux, C.; Berneche, S., Absence of ion-binding affinity in the putatively inactivated low-[K+] structure of the KcsA potassium channel. *Structure* **2011,** *19*, 70-79.
46. Zhou, Y.; Morais-Cabral, J. H.; Kaufman, A.; MacKinnon, R., Chemistry of ion coordination and hydration revealed by a K+ channel-Fab complex at 2.0 A resolution. *Nature* **2001,** *414*, 43-48.
47. Lee, J.; Patel, D. S.; Stahle, J.; Park, S. J.; Kern, N. R.; Kim, S.; Lee, J.; Cheng, X.; Valvano, M. A.; Holst, O.; Knirel, Y. A.; Qi, Y.; Jo, S.; Klauda, J. B.; Widmalm, G.; Im, W., CHARMM-GUI Membrane Builder for Complex Biological Membrane Simulations with Glycolipids and Lipoglycans. *J. Chem. Theory Comput.* **2019,** *15*, 775-786.
48. Lee, J.; Cheng, X.; Swails, J. M.; Yeom, M. S.; Eastman, P. K.; Lemkul, J. A.; Wei, S.; Buckner, J.; Jeong, J. C.; Qi, Y.; Jo, S.; Pande, V. S.; Case, D. A.; Brooks, C. L.; MacKerell, A. D.; Klauda, J. B.; Im, W., CHARMM-GUI Input Generator for NAMD, GROMACS, AMBER, OpenMM, and CHARMM/OpenMM Simulations Using the CHARMM36 Additive Force Field. *J. Chem. Theory Comput.* **2016,** *12*, 405-413.
49. Jo, S.; Cheng, X.; Lee, J.; Kim, S.; Park, S. J.; Patel, D. S.; Beaven, A. H.; Lee, K. I.; Rui, H.; Park, S.; Lee, H. S.; Roux, B.; MacKerell, A. D., Jr.; Klauda, J. B.; Qi, Y.; Im, W., CHARMM-GUI 10 years for biomolecular modeling and simulation. *J. Comput. Chem.* **2017,** *38*, 1114-1124.

Table 1. Lipids and cholesterol available in PACKMOL-Memgen[a]

DAPA	DMPA	DPPA	PLPA	SDPA
DAPC	DMPC	DPPC	PLPC	SDPC
DAPE	DMPE	DPPE	PLPE	SDPE
DAPG	DMPG	DPPG	PLPG	SDPG
DAPS	DMPS	DPPS	PLPS	SDPS
DLPA	DOPA	DSPA	POPA	PSM
DLPC	DOPC	DSPC	POPC	SSM
DLPE	DOPE	DSPE	POPE	CHL1
DLPG	DOPG	DSPG	POPG	
DLPS	DOPS	DSPS	POPS	

[a] Abbreviations used are: **DA = diarachidonoyl; DL = dilauroyl; DM = dimyristoyl; DO = dioleoyl; DP = dipalmitoyl; DS = distearoyl; PL = palmitoyl lauroyl; PO = palmitoyl oleoyl; SD = stearoyl docosahexaenoyl; PSM = palmitoyl sphingomyelin; SSM = stearoyl sphingomyelin; CHL1 = cholesterol; PA = phosphatidic acid; PC = phosphatidylcholine; PE = phosphatidylethanolamine; PG = phosphatidylglycerol; PS = phosphatidylserine**

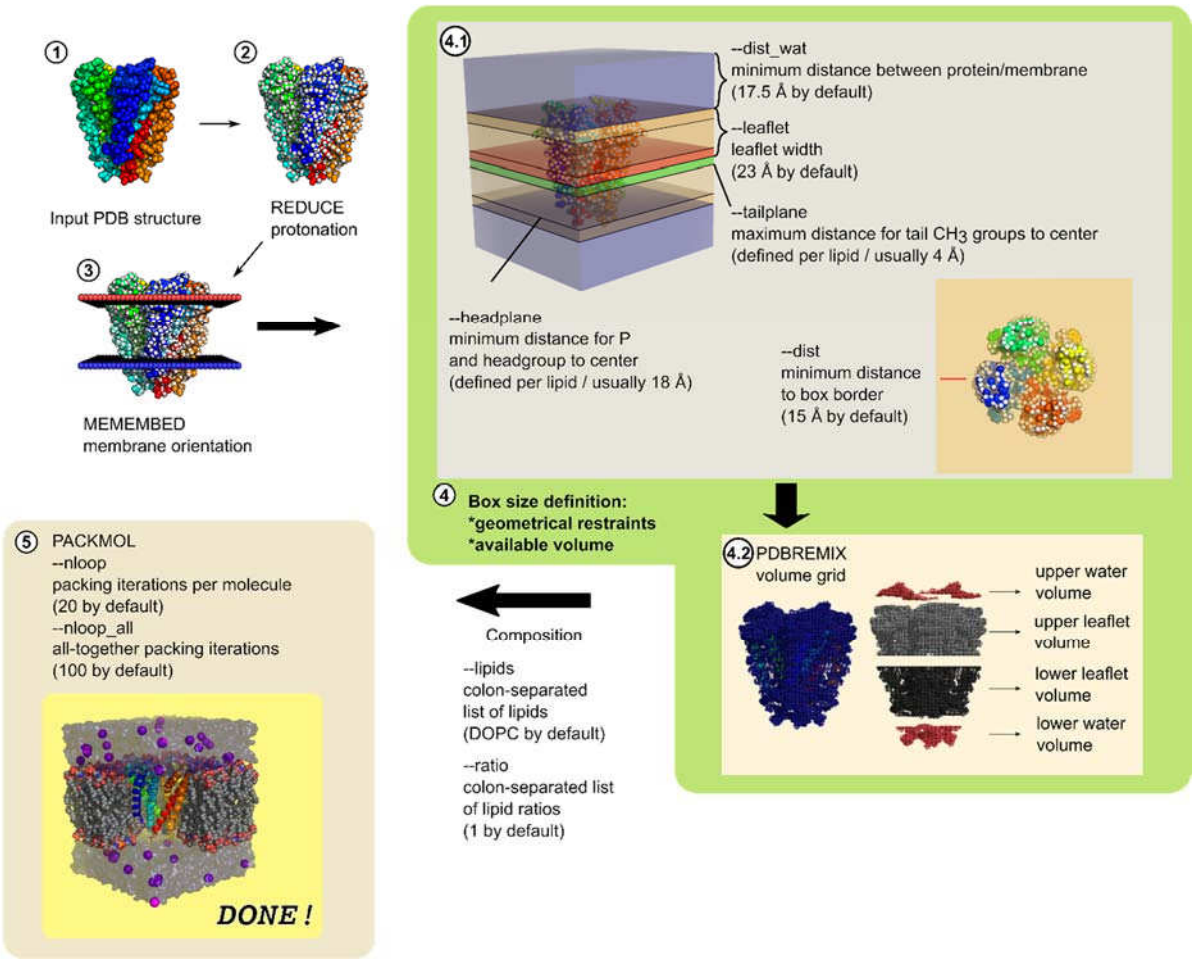

Figure 1. PACKMOL-Memgen workflow applied to the protein structure PDB ID 1BL8. The process comprises multiple, often optional, steps that are controlled by flags available in the software. The most important flags are mentioned in the figure. For a detailed description, see the workflow section.

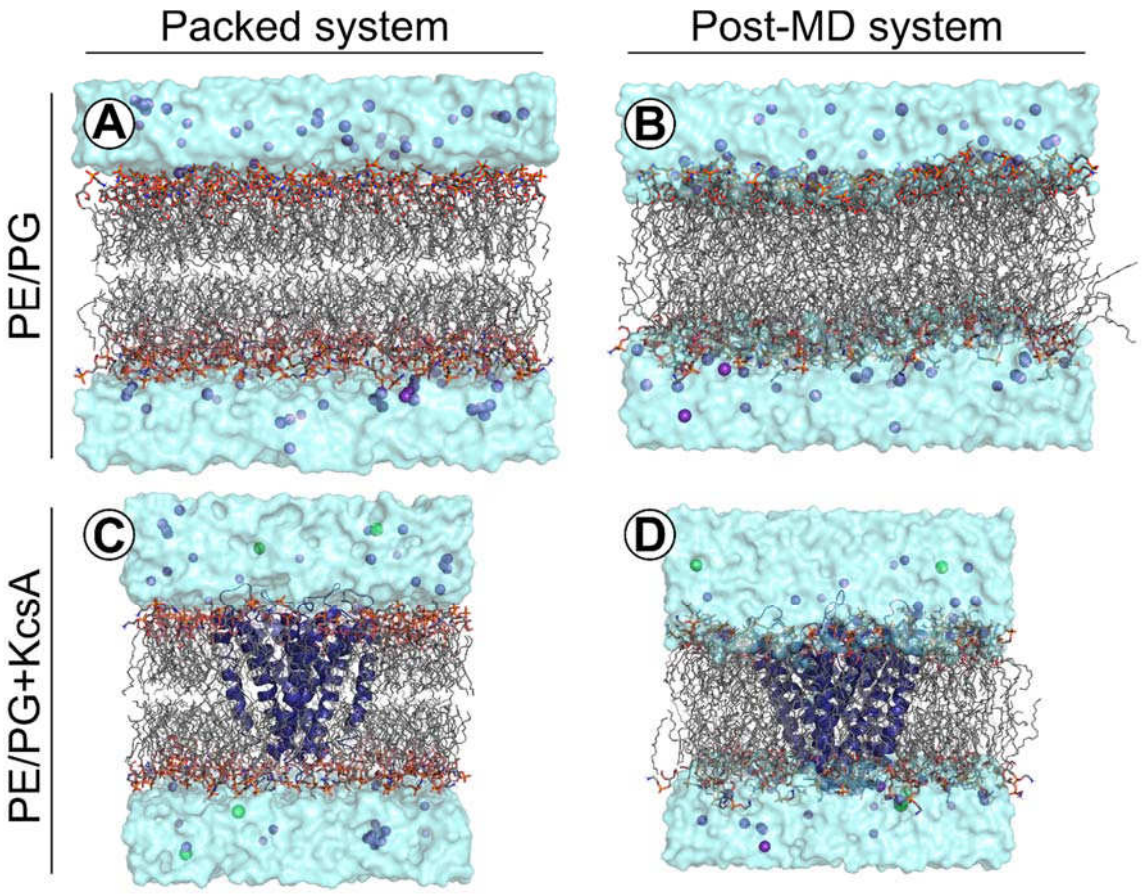

Figure 2. Representative structures for the membrane-only system (A, B) and the KcsA channel embedded in a membrane (C, D) generated as described in the main text, immediately after packing with PACKMOL-Memgen (A and C) and after 500 ns of MD simulations (B and D).