JCTC Journal of Chemical Theory and Computation

Article

pubs.acs.org/JCTC

# Computational Lipidomics with *insane*: A Versatile Tool for Generating Custom Membranes for Molecular Simulations

Tsjerk A. Wassenaar,*[,†,‡,∥] Helgi I. Ingólfsson,[†,∥] Rainer A. Böckmann,[‡] D. Peter Tieleman,[§] and Siewert J. Marrink[†]

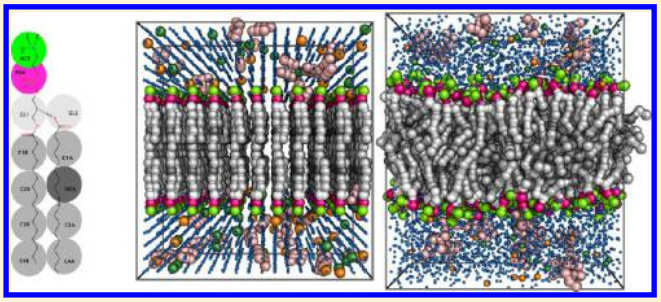
[†]Groningen Biomolecular Sciences and Biotechnology Institute and Zernike Institute for Advanced Materials, University of Groningen, Nijenborgh 7, 9747 AG Groningen, The Netherlands

[‡]Computational Biology, Department of Biology, Friedrich-Alexander University of Erlangen-Nürnberg, Staudtstrasse 5, 91052 Erlangen, Germany

[§]Centre for Molecular Simulation and Department of Biological Sciences, University of Calgary, 2500 University Drive NW, Calgary, Alberta T2N 1N4, Canada

Supporting Information

**ABSTRACT:** For simulations of membranes and membrane proteins, the generation of the lipid bilayer is a critical step in the setup of the system. Membranes comprising multiple components pose a particular challenge, because the relative abundances need to be controlled and the equilibration of the system may take several microseconds. Here we present a comprehensive method for building membrane containing systems, characterized by simplicity and versatility. The program uses preset, coarse-grain lipid templates to build the membrane, and also allows on-the-fly generation of simple lipid types by specifying the headgroup, linker, and lipid tails on the command line. The resulting models can be equilibrated, after which a relaxed atomistic model can be obtained by reverse transformation. For multicomponent membranes, this provides an efficient means for generating equilibrated atomistic models. The method is called *insane*, an acronym for INSert membrANE. The program has been made available, together with the complementary method for reverse transformation, at http://cgmartini.nl/. This work highlights the key features of *insane* and presents a survey of properties for a large range of lipids as a start of a computational lipidomics project.



## INTRODUCTION

In 1925, Gorter and Grendel[1] first described biological membranes as "a layer of lipoids just two molecules thick". The main role of the bilayer was considered to keep the biological molecules together, while allowing influx of nutrients and efflux of waste material. Now, almost a century of research later, this simplistic barrier model has grown to a mesmerizingly complex view of the cellular envelope as an active constituent of life. The cell surface, as well as organellar membranes, form a stage for lipids, proteins, and other compounds involved in a myriad of interactions and processes. The role of lipids therein is active, determining the physical and biological properties of the membrane and modulating protein activity both directly and indirectly.

Due to the importance of membranes and membrane proteins, many current day molecular simulations focus on systems containing lipids. Here, a difficulty arises in obtaining suitable starting structures, especially for bilayers consisting of multiple components or containing one or more proteins. Small and simple bilayers may be obtained by starting from random mixtures of lipids in water, through self-assembly.[2] Larger ones are commonly built from smaller patches of pre-equilibrated bilayers, but this puts restrictions on the composition. The construction of a protein−membrane system is a challenge in particular, and several approaches have been proposed, as described in reviews.[3,4]

A starting structure for a protein−membrane system can be built from scratch, e.g., by packing individual lipid structures around a protein, using lipid structures from a library. This is the basis of the PackMol[5] method, on which the currently popular CHARMM-GUI web interface[6,7] is also based, which assists users in generating input files for protein/membrane systems. A particular feature of the CHARMM-GUI method is that it starts with a 2D Monte Carlo simulation to determine the optimal arrangement of headgroups, prior to building the actual lipids. Another web interface was developed recently for generating Gromacs-based input files,[8] coupled to the Lipid-Book database.[9]

The most common approach for generating protein−membrane starting structures is merging the protein structure with an existing, equilibrated membrane patch. This approach is

Received: December 5, 2014
Published: April 10, 2015





DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

similar to the traditional method of solvating proteins using an equilibrated solvent box, in which the molecules overlapping with the solute are removed. Several tools have been developed over the years to assist with this approach. The first step in these procedures always consists of positioning the protein with respect to the bilayer. Subsequently, lipids overlapping with the protein can simply be removed or they can be driven out using a repulsive force. The latter can be accomplished with a cylindrical potential, or using a potential that takes on the shape of the protein in the membrane.[10] Two currently popular methods, InflateGro[4,11] and g_membed[12] are variations on these themes. InflateGro increases the lipid spacing in the membrane plane before deleting overlapping lipids, followed by deflation and equilibration. This allows optimal packing of the lipids around the protein. g_membed deflates the protein in the plane to needle-like dimensions, followed by a phase in which the protein is grown, pushing the lipids and solvent outward. Typically, these methods require several cycles and/or parameter adjustments to yield a stable system, which makes them difficult to use in high-throughput processing. Also proposed recently was a new method for merging a protein structure with an existing membrane, using high lateral pressure to push the two components together.[13] This procedure avoids the need to remove lipids, which may otherwise alter the membrane composition, especially for small, heterogeneous membrane patches.

For all methods that merge a protein structure with an existing membrane, an important objective is to avoid perturbing the lipid and protein configurations as much as possible. This is also true for related methods, such as the LipidWrapper,[14] which can take an existing, planar membrane patch and re-distribute the lipids over any complex triangulated surface, and for the LipidConverter,[15] which can be used to translate a given membrane to another force field description, or convert lipid types, provided that the lipid topologies match sufficiently.

The methods described previously work well if the membrane consists of a single lipid type and if equilibrated bilayers are available. However, with multicomponent membranes, removal of lipids will alter the stoichiometry, and the specific interactions of the protein inserted with the different constituents affect the partitioning. This will likely cause the resulting structure to be out of equilibrium, despite using a careful procedure. In such cases, a system needs to be re-equilibrated, which may require up to microseconds of simulation.[16−18] Such time scales are, generally speaking, still out of reach for atomistic molecular dynamic (MD) simulations, and, as a consequence, methods for building atomistic membranes directly may fail to yield sufficiently equilibrated structures, unless the bilayer consists of a single lipid type.

To overcome time scale and convergence problems in simulations, a large variety of coarse-grain (CG) models have been introduced.[19] Such low-resolution models allow running tens to hundreds of microseconds of simulation of membrane patches composed of tens of thousands of lipids. In particular, the popular Martini model[20−22] can be used to reach equilibration of membrane structures with complex composition, because of the system sizes and time scales that can be simulated and the large number of lipid types available. The Martini model has been used to characterize lipid mixing, domain formation, and partitioning due to specific lipid−protein interactions. A comprehensive review can be found elsewhere.[23]

The increased efficiency of CG models is not only due to the smaller number of particles. They also use softer potentials, which result in a smoother potential energy landscape, thus allowing larger time steps to be used. The smoother energy landscape results in increased robustness and causes CG simulations to be much more tolerant to distortions of the starting structures than atomistic simulations. This allows for constructing complex CG membranes from scratch by building lipids from simple templates, which are arranged on a grid and subsequently relaxed. Because of the intrinsic robustness, such templates need not even be correct structures and need only be grossly in line with the topological definition, allowing the resulting system to equilibrate properly. For most CG lipids, such abstract templates would be very simple and allow for a simple syntax to specify custom lipids on the command line.

Based on the foregoing considerations, the membrane-building tool *insane* was developed. The name *insane* is an acronym for "INSert membrANE" but also reflects the general philosophy, aiming for the simplest approach conceivable. The tool was specifically designed for building custom membranes for use in molecular simulations, with emphasis on flexibility, including full control over the lipid composition of both leaflets, and the ability to embed membrane proteins or include defects. Membrane defects/holes can be used to study membrane properties such as lipid pore stability, and they are a convenient way for allowing lipids to equilibrate between membrane leaflets, for example in combination with a cylindrical potential to maintain the pore.[24,25]

With *insane* it is possible to generate membranes (and solvents) of arbitrary composition. This implies that it can be used for the systematic exploration of properties and interactions of lipids *in silico*, or "computational lipidomics". The design follows the modular software model developed previously,[26] which facilitates the incorporation in automated workflows, supporting high-throughput membrane simulations.[27] In this work, we take a first step toward high-throughput lipid characterization, by setting up and simulating a large range of bilayers composed of diacylglycerol lipids with different head and tail groups.

Although *insane* was developed in relation to the Martini force field, we note that the underlying principles are not specific to that model and could easily be applied to build systems for other coarse-grain models.

The outline of this work is as follows. First, *insane* is described, providing the background and implementation of the different elements in the workflow. Next, the robustness, convergence, and membrane properties are explored for a number of systems, including a hundred, single component bilayers, obtained by combining five different headgroups with 20 different acyl tails of varying length and level of unsaturation. After that, several examples are given of projects that use this method for setting up molecular simulations with membranes and it is demonstrated how this method can be used in combination with a routine for reverse transformation, to obtain equilibrated atomistic membranes of complex composition, which is at present difficult to achieve with other methods.

## BACKGROUND AND IMPLEMENTATION

The basic approach used by *insane* is setting up a grid, marking the cells occupied by a protein or a hole in either leaflet, and


DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

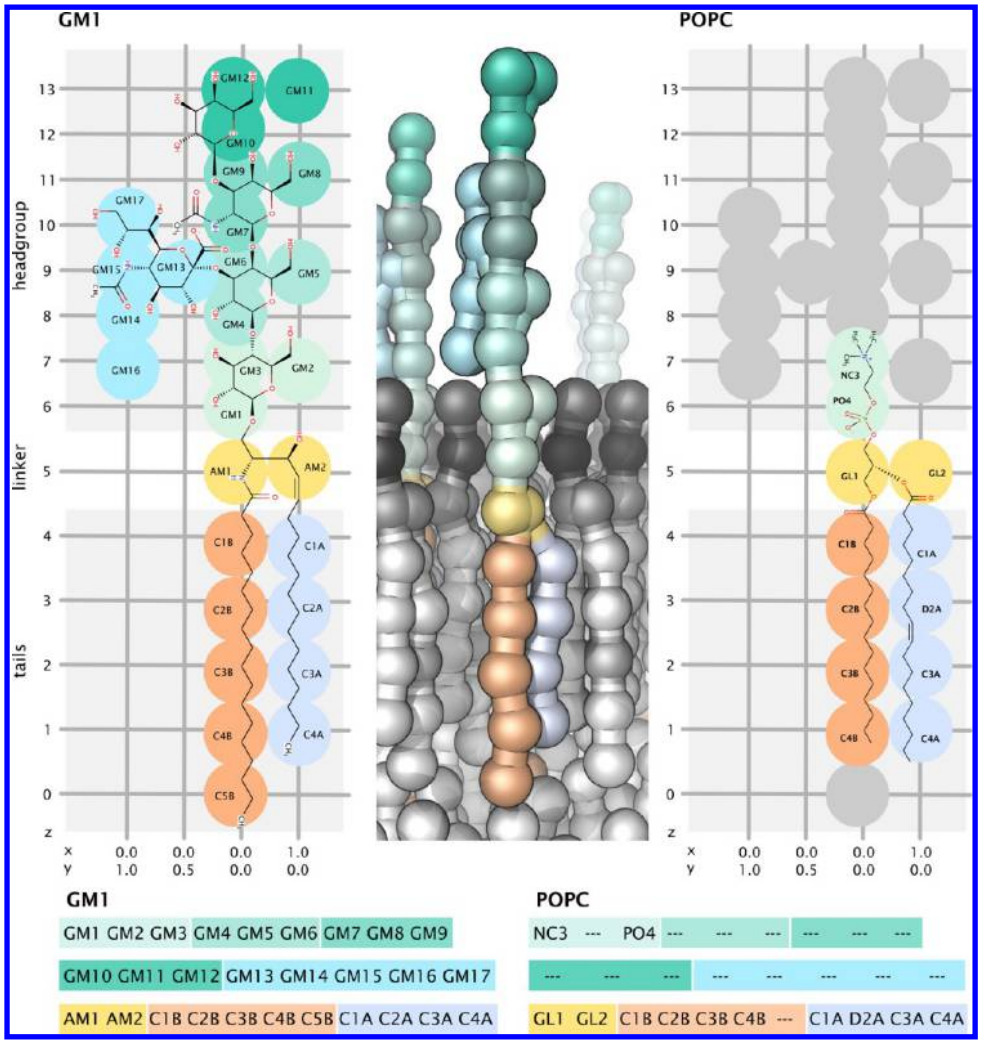



**Figure 1.** Lipid templates. Due to the smoother potential energy landscape employed by CG models, CG lipids can be built from lipid templates with approximate bead locations only, followed by relaxation to their equilibrated structures. Here we show the *insane* lipid template for monosialotetrahexosylganglioside (GM1, left) and 1-palmitoyl-2-oleoyl-*sn*-glycero-3-phosphocholine (POPC, right), indicating the *x,y,z* pseudocoordinates of the CG beads. The approximate atomistic structure is shown for comparison, and the resulting 3D lipid model produced by *insane* is shown in the middle panel.

then filling the remaining cells with simple lipid structures. What is specific to *insane* is that these structures are built from schematic definitions, which are written based on templates. This makes it trivial to generate structures, even for lipids for which none are available. A similar approach is used for generating a solvent configuration. The robustness makes the program particularly suited for incorporation in automated workflows, enabling high-throughput simulations.

In the following sections the different aspects of building a system with *insane* are described, together with the implementation. The first section describes the generation of (lipid) structures from templates, which is followed by an explanation of the syntax and mechanism for defining lipids on the command line. The subsequent sections give a description of the grids used for placement of lipids and solvent, in relation to the periodic boundary conditions (PBC) and the placement of a protein and/or a hole. Some additional background is given concerning the options for and use of PBC for membrane simulations.

**Molecular Templates.** A molecule in molecular simulations consists of a set of coordinates and a topological description. The latter lists the atom types, bonds, and angles, etc. Often it is easy to identify molecular classes or building blocks, which can be considered variations of a smaller set of common templates. For CG molecules, these classes are usually much simpler than their atomistic counterparts, and templates generally cover wider ranges of molecules. This is particularly true for most lipids, which have a very simple CG structure. As a consequence, a single template can be written that covers many different lipids. Figure 1 shows such a template for two example lipids, in which the relative positions of the CG beads are specified by *x,y,z* pseudocoordinates. This template was developed to match the complex glycolipid GM1 (Figure 1, left), but it is easy to see how the same template can be used for simpler lipids, such as POPC (Figure 1, right). Of course, this would not be sufficient, if the simplicity of the structural template did not match an equally simple topological template.

A topological description usually starts out with a listing of the particles in the molecule, and in most (coarse-grain) force fields the lipid topology has these ordered according to headgroup, linker, and tails, respectively. The order and names of particles in the Martini GM1 topology are GM[1...17]-AM1-AM2-C[1A...5A]-C[1B...4B] (Figure 1, bottom), and each particle has a corresponding position in the structural template.


DOI: 10.1021/acs.jctc.5b00209
*J. Chem. Theory Comput.* 2015, 11, 2144−2155

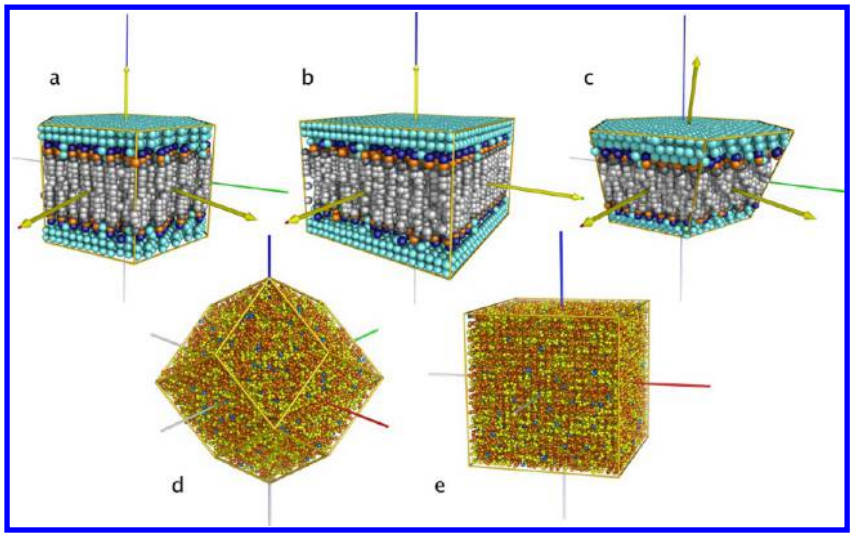



**Figure 2.** Periodic boundary conditions available in *insane*. The default for a membrane containing system is the hexagonal prism (a), which is 14% smaller than the corresponding rectangular cell (b) for the same distances between images. A skewed hexagonal prism (c) is possible, in which the layers are maximally decoupled by an $x/y$ shift of the third lattice vector. This unit cell is called "optimal" and corresponds to a rhombic dodecahedron if the lattice vectors have equal lengths. For systems without a membrane, the default unit cell of *insane* is the rhombic dodecahedron (d), here shown containing a fructose/glucose/water mixture, corresponding to honey. A cubic unit cell (e) is also available. In all panels, the yellow arrows correspond to the lattice vectors, while the red/green/blue cylinders depict the $x/y/z$ axes, respectively.

For POPC, the Martini topology has the bead order NC3-PO4-GL1-GL2-C1A-D2A-C3A-C4A-C1B−C2B-C3B−C4B (Figure 1, bottom). This can be easily matched with the GM1-based template by marking the positions occupied with the corresponding particle names.

Based on this general design, *insane* currently has templates for simple diacylglycerols, ceramides, glycolipids, sterols, and mycolic acids and contains definitions for the most common lipids, as well as a range of less abundant ones, often added *ad hoc* for specific studies. A listing of the available lipids and their definitions is given in the Supporting Information. We note that the lipids defined in *insane* form only a small subset of the entire lipidome. Given the large variety of natural and synthetic lipids, it is impractical to make a complete list. Therefore, the implementation of the templates is aimed explicitly at facilitating the addition of new lipids. Furthermore, a command line syntax and building block naming scheme were developed for on-the-fly generation of simple custom lipids from a specification of the headgroup(s), the linker, and the tails. These custom types are generated together with corresponding topological descriptions, using standard Martini building blocks and rules.[20,21] Details of this syntax are given in the Supporting Information, together with the block names for all supported headgroups, linkers, and tails, and several examples. Currently, only Martini lipids and templates are included and the command line syntax only supports simple lipids types. However, for most CG force fields, the same principles apply and the template and molecule definitions can be adapted to match other models as well as other more complex lipids.

The membranes generated by *insane* are characterized by the ordered arrangement of lipids on the grid and of particles within the lipids (Figure 1, middle). To ensure a rapid loss of the order, each particle is given a random displacement. These distortions add tension, which will dissipate, thereby giving a kick-start to the simulation, enhancing the mixing and equilibration. This is where benefit is drawn from the robustness of CG models. It is noted that this procedure typically requires usage of position restraints on certain simulation components, such as proteins, during the initial stages of equilibration.

A template-based generation is also used for solvent and small solutes, such as sugars, currently including monosaccharides and a number of disaccharides. In the case of larger solutes, such as disaccharides, the molecules are typically deflated to avoid overlaps between neighbors.

**Membrane Grid.** As mentioned earlier, the simplified structural models are placed on a grid to construct a starting configuration. First, the membrane is built using a 2D grid, after which the solvent is generated using a 3D grid. However, prior to placement, the grids need to be defined. These depend on the type of PBC and on the approximate area per lipid, but notably also on the presence of one or more proteins and/or the specification of a hole of a given radius. As such, the complete *insane* workflow can be summarized as follows:

1. Set up and orient macromolecule(s) and/or a hole
2. Set PBC
3. Build membrane
4. Build solvent

These different stages are explained in more detail as follows.

**Protein Handling, Orientation, and Placement.** The orientation of proteins in membranes is a nontrivial problem, but several solutions have been proposed and a database is available, which contains all membrane proteins available in the Protein Data Bank (PDB), aligned with respect to a lipid bilayer.[28] Therefore, the problem of orientation is kept apart from the building of the system, and *insane* assumes that a protein input structure is aligned with a membrane having a normal vector along $z$ and the center at $z = 0$. This means that a protein can also be positioned above/below a membrane or partly embedded in the membrane by shifting the input structure. Alternatively, this can be achieved by specifying a shift along $z$.

**Hole.** To allow equilibration of lipids between leaflets, it is possible to add a hole of specified radius, which can be maintained during the simulation using a cylindrical potential,


DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

which is available, e.g., for the Gromacs[25] and NAMD[24] simulation packages. A hole can also be useful for investigating membrane plasticity by looking at the thermodynamics and kinetics of pore closure. If only a hole is specified, it will be centered similar to a protein. If both a protein and a hole are specified, the protein is centered in the rectangular brick and the hole is positioned such that the distance between the protein and any periodic image of the hole is maximal given the settings for the periodic boundary conditions.

**Periodic Boundary Conditions.** Several types of PBC are available for membranes (Figure 2a−c) and for systems without membrane (Figure 2d,e). The default for membranes is the hexagonal prism, which gives an optimal in-plane distribution, although rectangular and square PBC can also be specified. For a given distance between periodic images, the hexagonal prism is 14% smaller as compared to the rectangular PBC, providing a significant computational advantage. Besides, a rectangular unit cell may give artifacts similar to those observed with fully solvated proteins.[29] It is also possible to choose an "optimal" simulation cell, in which the third vector is not aligned with the membrane normal, but is skewed to maximally decouple the bilayers across the PBC (Figure 2c).

For systems that do not contain a membrane, the default unit cell of choice is the rhombic dodecahedron (Figure 2d), although the rectangular and the cubic unit cell are also available (Figure 2e). The truncated octahedron is not available, as it has no particular advantage, being both nonoptimal and skewed.

In addition to these choices for the PBC, it is possible to specify the simulation unit cell on the command line or to retain the PBC defined in a protein input structure. Otherwise, the actual unit cell is determined from the dimensions of the solute and/or hole, the minimal distance required between periodic images, and the type of unit cell.

**Membrane Grid.** When the unit cell is determined, two grids are set up spanning the *xy*-plane, with the size of each grid cell approximately equal to the average area per lipid specified, but adjusted to fit in the periodic unit cell. These two grids correspond to the upper and lower leaflets. Per leaflet and per bin, the protein atoms falling in it are counted, and when the cell occupancy exceeds a specified value (the fudge factor), the bin is marked as unavailable for placement of a lipid. Similarly, bins are marked unavailable if they coincide with a hole. Subsequently, the available bins are shuffled and filled with lipid structures, derived from the templates, according to the relative abundances. In this way each lipid is placed in one cell, irrespective of the lipid size, relying on the robustness of the CG model to relax the membrane. The area per lipid can be set per leaflet, to allow building membranes with differently sized lipids, e.g., when one leaflet contains a larger fraction of lipids with more tails, such as cardiolipins. Additionally, the absolute number of lipids in the upper and lower leaflets can be adjusted by specifying a number of cells to keep unoccupied. When setting up large and complex asymmetrical membranes, it is advised to measure the equilibrated average area per lipid for each leaflet using symmetrical simulations and then adjust the total number of lipids or the area per lipid in each asymmetrical membrane leaflet accordingly.

By default, the interior of proteins is excluded for placement. However, this restriction can be overruled to correctly handle ring-forming proteins and to support setting up a bilayer around multiple proteins.

Currently, multilamellar systems and nonlamellar membranes, including vesicles, micelles, and cylindrical bilayers, are not supported. These will become available in a future version.

**Solvent Grid.** A similar approach as used for the membranes is used for generating solvent, where a 3D grid is placed over the unit cell, based on a preset solvent radius. Then the grid cells occupied by membrane and/or protein are flagged unavailable and the remaining cells are filled with solvent and small solutes, according to the relative abundances. To help avoid placement of solvent in the interior of the membrane or protein, each particle is replaced by 20 points distributed on a sphere with a radius of 0.1 nm, and these points are assigned to the grid cells and counted. This lowers the probability that interior grid cells are left unoccupied. In addition, an exclusion range can be specified, which makes all grid cells falling within that range of the bilayer center unavailable.

After addition of the solvent, the system is shifted to have the protein and/or membrane in the center of the rectangular brick representation of the unit cell.

## METHODS

**Simulation Setup.** All simulations were performed with the Gromacs molecular dynamics suite version 4.5.x,[30] using the standard Martini 2.0 simulation settings.[20,21] Unless stated otherwise, systems were energy-minimized (steepest descent, 500 steps) and simulated for 0.5 ns using a short time step of 10 fs. Next, the bilayer area was relaxed in a 30 ns long simulation using a 30 fs time step and the Berendsen barostat.[31] After equilibration, production simulations were run for at least 1 $\mu$s. These were run using semi-isotropic pressure coupling maintaining the system at 1 bar using the Parrinello−Rahman barostat[32] with a time constant $\tau_p$ = 4.0 ps and a compressibility of $3 \times 10^{-4}$ bar$^{-1}$. The temperature was controlled by weak coupling to an external heat bath at 310 K using the velocity rescaling thermostat of Bussi et al.[33] with a $\tau_T$ = 1.0 ps. In the final production simulation a time step of 30 fs was used, unless mentioned otherwise.

For all systems, the membrane and solvent were generated using *insane*. The Supporting Information lists the specific commands used for each system.

**Membrane Proteins.** Proteins were obtained from the OPM database[28] and simulated with the automated workflow *martinate*.[34] This program first converts the protein to Martini[20−22] using the program *martinize*.[35] Subsequently, *insane* is called to generate the membrane and solvent, after which the equilibration protocol and the production run are performed. For the protein−membrane systems presented in this work, the production run consisted of 10 ns relaxation.

The cytochrome bc1 complex has previously been simulated with Martini to study cardiolipin binding[16] and was used as the test system for development of the reverse transformation method *backward*.[36] Here, the cytochrome bc1 complex (with PDB ID 3CX5) was set up in a mitochondrial-like membrane containing 40% POPC, 24% POPE, 16% PI, 4% POPS, and 16% cardiolipin and solvated with a 0.1536 M NaCl solution. The resulting system contained 361 lipids in the upper leaflet and 351 in the lower leaflet. The solvent was comprised of 21,434 Martini water molecules, 264 $Na^+$ ions, and 220 $Cl^-$ ions. The total number of particles in the system was 41,702.

**Backmapping.** Conversion of CG models to GROMOS united atom representation[37] was performed using the method *backward* through the *initram* interface.[36] This method consists


DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

of a geometric reconstruction of the united atom structure, followed by energy minimization and a short equilibration simulation.

**Lipid Topology Generation.** All lipid topologies were constructed using the auxiliary lipid topology builder *lipid-martini-itp*, according to the standard Martini 2.0 rules and building blocks.[20,21] This companion to *insane* and the detailed topologies for all of the lipids used in this work have been made available at the Martini portal (http://cgmartini.nl). For *insane* and the topology builder, we developed a naming scheme, providing shortcuts for Martini beads and building blocks. Currently, six different CG headgroup beads are supported: choline (C), ethanolamine (E), glycerol (G), serine (S), phosphate (P), and double charged phosphate (O). Headgroups can be chained together, like "C P" to specify a phosphocholine headgroup, or can be skipped altogether, e.g., to give diacylglycerols or ceramides. For the linker, two different types are currently available: glycerol (G) and sphingosine (A) based. Finally, three different types of tail beads are available, which can be used to specify any length of unbranched acyl chains: saturated with no double bonds (C) or with 1−2 cis (D) or a trans (T) double bond. The syntax is consistent between the topology builder and *insane*.

The focus in this work is on phosphatidylcholines (PC), phosphatidylethanolamine (PE), phosphoglycerol (PG), phosphatidylserine (PS), and phosphatidic acid (PA) lipids. All of these have a glycerol linker with a phosphate comprising headgroup. For each headgroup/linker type, the bead names and corresponding Martini bead types are listed in Supporting Information Table S1. These headgroups were combined with 20 different acyl tails, with lengths in the range from 2 to 6 CG beads, both saturated and unsaturated. Each bead maps to approximately four carbon atoms, such that the tails included in this work correspond to atomistic tails ranging from 8 to 24 carbon atoms. The lipid tails used are listed in Table 1, and listing of fatty acid one letter names, bead assignment, and examples of corresponding atomistic molecules is given in Supporting Information Table S1.

**Table 1. List of Tested Lipid Tails**

| ID | acronym | sn1 tail | sn2 tail |
|---|---|---|---|
| 1 | DT | CC | CC |
| 2 | DL | CCC | CCC |
| 3 | DP | CCCC | CCCC |
| 4 | DB | CCCCC | CCCCC |
| 5 | DX | CCCCCC | CCCCCC |
| 6 | DY | CCD | CCD |
| 7 | DV | CCDC | CCDC |
| 8 | DO | CDCC | CDCC |
| 9 | DI | CDDC | CDDC |
| 10 | DF | CDDD | CDDD |
| 11 | DG | CCDCC | CCDCC |
| 12 | DA | DDDDC | DDDDC |
| 13 | DR | DDDDDD | DDDDDD |
| 14 | DN | CCCDCC | CCCDCC |
| 15 | LP | CCC | CCCC |
| 16 | PO | CCCC | CDCC |
| 17 | PG | CCCC | CCDCC |
| 18 | PI | CCCC | CDDC |
| 19 | PA | CCCC | DDDDC |
| 20 | PR | CCCC | DDDDDD |

**Analysis.** To compare the different lipid types, several properties were calculated for each simulation, including the following: area per lipid, area compressibility, average tail order, lipid diffusion, and bilayer thickness. The area per lipid was calculated from the average box area in the plane of the bilayer during the simulation divided by the number of lipids in each leaflet. The bilayer area compressibility modulus ($K_A$) was calculated from the amplitude of the box area fluctuations. $K_A = kT\langle A\rangle/(N\langle(A - A_0)^2\rangle)$, where $kT$ is the Boltzmann constant and temperature in Kelvin, $A$ the box area, $A_0$ the equilibrium area, and $N$ the number of lipids in a leaflet/monolayer. The lipid tail order parameter was calculated from the angle $\theta$ between the normal of the bilayer surface and the vector along each bond in the lipid tails, following $P_2 = (1/2)(3\cos^2(\theta - 1))$. The lateral diffusion of lipids was calculated from the mean square displacement (MSD) of their phosphate PO4 beads obtained with the g_msd tool in Gromacs. The center of mass motion of the system was removed and the diffusion coefficient obtained by fitting a line through the MSD curve vs time, using linear regression, omitting 10% of the data at both ends. The bilayer thickness was measured as the distance between the centers of mass of the PO4 beads of the lipids in the two leaflets. All reported errors, except for lipid diffusion, are standard errors of the mean obtained from block averaging. The number of blocks was increased from a single block to the point where each block had only five data points. The reported error is the maximum standard error found within the last 20% of the block sizes studied. For lipid diffusion the reported error is the difference between fits of the first and second halves of the MSD curve, as done by g_msd.

## RESULTS AND DISCUSSION

**Building Complex Bilayers.** The main purpose of *insane* is facilitating building complex membranes and solvents. Figure 3 gives three examples of systems generated with *insane*, including a simple, DOPC-only bilayer (Figure 3a), a complex asymmetric bilayer (Figure 3b), and a POPC bilayer in a 0.1 M trehalose solution (Figure 3c). In each case, the output structure from *insane* is shown, together with the configurations after 30 ns and after 1 $\mu$s of simulation. The figure panels illustrate the nonphysical, highly ordered starting structures generated, as well as the quick relaxation. For the DOPC only membrane (Figure 3a), the relaxation is complete within 30 ns of simulation. All in all, from the generation of the starting structure, this takes less than 15 min on a four core 2.30 GHz Intel Core i7 notebook.

The raft-forming mixture starts with a random distribution of the lipids in either leaflet, which can be seen from Figure 3b. The initial order in the system again rapidly dissipates, and in the DPPC/DIPC containing leaflet, segregation is already observed after 30 ns of simulation. Equilibrating lipid mixing in both leaflets, however, takes much longer, requiring hundreds of nanoseconds for this mixture, as can be seen from the neighbor counts in Supporting Information Figure S1. This 1 $\mu$s simulation takes approximately half a day on a 2.30 GHz i7 machine. For this system, the corresponding GROMOS united atom representation would have around 75,000 atoms, which, using a 4 fs time step, would run about 60 times slower and take a month on the same computer, exemplifying the problem of generating equilibrated complex bilayers atomistically.

The system shown in Figure 3c demonstrates the setup and equilibration of a bilayer with a specific solution, here of CG trehalose in water. The time required for equilibration for this


DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

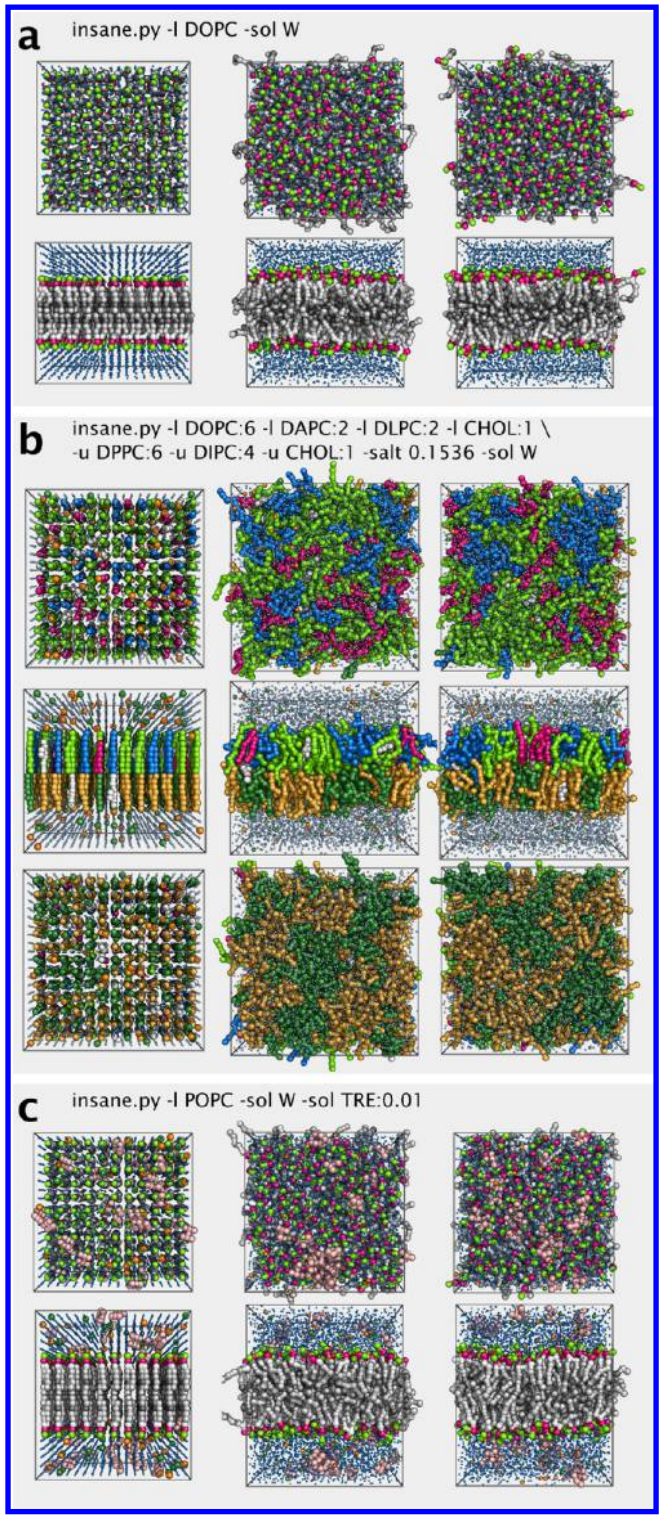



**Figure 3.** Generation and relaxation of bilayers and solvent. Left most panels show configurations generated by *insane*, middle panels show the systems after 30 ns of production simulation, and right most panels show the same systems after 1 $\mu$s. (a) Pure DOPC bilayer. (b) Raft-forming, asymmetric membrane, containing DOPC, DAPC, DLPC, and cholesterol in one leaflet and DPPC, DIPC (where DI represents a dilinoleic tail; see Supporting Information Table S1) and cholesterol in the other leaflet. Solvent contains 153.6 mM NaCl with compensation for the net charge of the membrane. (c) POPC bilayer in a 0.1 M trehalose solution.

system is mostly dependent on the bilayer, which only has a single component and thus relaxes within 30 ns.

**Martini Lipidomics.** The stability of the different systems tested and the convergence properties suggest that an *insane*-based protocol may be suitable for exploring equilibrium properties of bilayers, in particular for quick characterization of single-component membranes. Considering that a 1 $\mu$s simulation of a bilayer consisting of two leaflets of 11 by 11 lipids takes approximately half a day on a modern eight core machine, the setup with *insane* allows a true high-throughput approach to explore the coarse-grain lipidome.

In principle, one-component membranes can also be set up using self-assembly, see ref 23 and references therein, except for lipids that are highly charged or have high intrinsic curvature. In addition, self-assembly simulations still pose the problem of generating starting configurations, in particular for lipids for which no structures are available. Furthermore, this would add significant simulation time to account for the assembly process, as bilayers can take very long to form or may have long-lasting defects.

In contrast, the command-line interface offered by *insane* can be easily used to script generation of many lipid types by combining different headgroups, linkers, and tail groups, prearranged as a bilayer. Here, we simulated over a hundred different Martini diacylglycerol lipids using such an automated protocol. The study comprises six different headgroup types, including two zwitterionic (PC and PE), three negatively charged (PG, PS and PA), and one double negatively charged (PO; phosphatidic acid with a net −2 charge). For each headgroup type 20 different acyl chains, representing atomistic chain lengths from 8 to 24 carbons and with different levels of unsaturation, were explored (see Table 1 and Supporting Information Table S1 for details). To characterize and compare the different lipid types, several physicochemical properties were calculated. In Figure 4 the area per lipid is shown, and Supporting Information Figure S2 further shows in-plane compressibility, average tail order, lipid diffusion, and bilayer thickness for most of the tested lipids. The properties calculated are tabulated in Supporting Information Table S2 for all lipids tested.

Both lipid structures and topologies were built from templates, rather than constructed or edited manually. Yet they demonstrate consistent physicochemical properties and are in general agreement with atomistic simulations and experiments. We note that experimental properties for lipid bilayers can be tricky to measure, as they are heavily dependent on environmental factors, such as the level of solvation, the ionic strength, the temperature, and others. This causes reported values to vary greatly, depending on the experimental method used or their interpretation.[38]

In all cases, increasing the length of saturated tail acyl chains gives rise to a linear increase in bilayer thickness (Supporting Information Figure S2a). This result is in good agreement with experimental observations.[39] The area per lipid was consistently largest for PC, and following the order PC > PS > PG > PA > PE, with PE always yielding the smallest area for a given lipid tail type (see Figure 4). Again, this is in agreement with available experimental and (atomistic) simulation data.[40,41]

As expected, the area per lipid increased significantly with increasing levels of unsaturation (see Figure 4, where D beads mark unsaturated bonds).[41,42] Also as seen experimentally, increasing the length of the saturated acyl tails has little effect on the area per lipid (Figure 4). However, at a fixed


DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

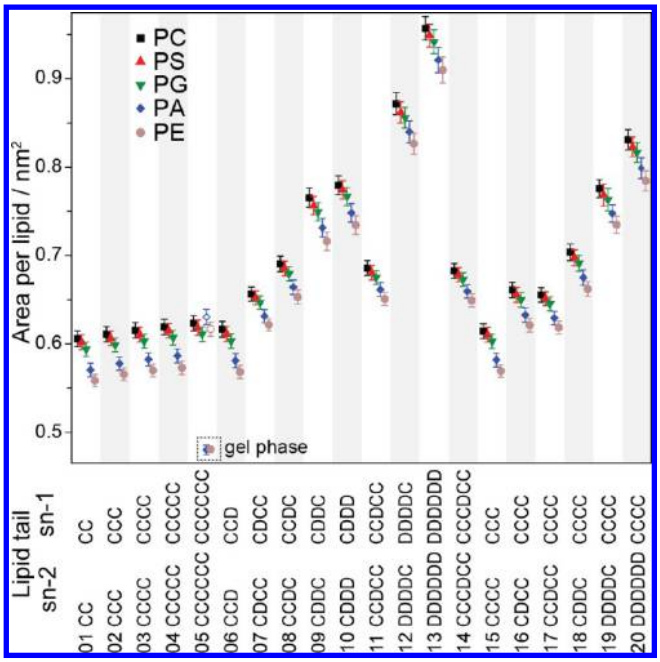



**Figure 4.** Martini diacylglycerol lipid properties. The area per lipid is shown for 100 different lipid types (mean ± sd). Five different headgroup types are shown, each with ×20 different acyl chains (see Table 1 and Supporting Information Table S1 for details). Other bilayer properties are shown in Supporting Information Figure S2, and all calculated properties are tabulated in Supporting Information Table S2. Two lipid types (PA and PE with tail no. 05) entered the gel phase at the 310 K regular test temperature and were repeated at 350 K (open symbols).

temperature experimental results do show a slight decrease in area per lipid with increasing length of saturated acyl tails[42,43] in contrast to the slight increase seen here. The discrepancy is probably the result of a changing balance between enthalpic and entropic contributions that is hard to capture in coarse-grain models, due to the removal of internal degrees of freedom. The exact mechanisms are, however, still elusive and would require a systematic investigation to solve that is beyond the scope of this work.

The results also highlight a known issue of a too low phase transition temperature in Martini lipids with long saturated tails.[44] None of the lipids with five or six saturated CG bead tails should be fluid at the temperature of 310 K used, but only the PE and PA lipids having six CG bead tails went into the gel phase. The simulations of these lipids were repeated at 350 K, where they were in the liquid phase (see open symbols in Figure 4 and Supporting Information Figure S2).

Starting from preformed bilayers and aided by both the stabilizing effect of PBC and by the tolerance to frustration provided by a CG force field, we can explore lipid types that do not form stable bilayers by themselves experimentally and for which only indirect experimental values exist. Diacylglycerol lipids (no headgroup bead above the linker) are still too unstable and were therefore not explored further. At pH 7 around half of the phosphatidic acid lipids should have a net negative charge of −2 (PO) and the remainder −1 (PA).[45,46] The high charge density of pure PO lipids triggers formation of a mesh/lattice with their counterions at the headgroup level and gives rise to abnormal bilayer properties; these bilayers were simulated and their properties calculated, see Supporting Information Table S2, but not analyzed further.

Being a CG model, there is no direct correspondence of a given Martini lipid to a specific atomistic one. Rather, each lipid corresponds to a range of atomistic structures. This makes a strict quantitative comparison difficult, but it also poses a challenge when choosing the optimal CG lipid to represent a certain atomistic lipid. This is of particular importance when setting up CG simulations of membrane proteins, because the bilayer properties not only influence protein insertion and orientation, but also protein function.[47,48] Therefore, it is crucial to select appropriate solvating lipids for proper characterization of the protein studied.

**Complex Membranes.** The foregoing results show that *insane* can be used in conjunction with Martini to set up and simulate membranes with a large range of lipid types. This large range, together with the options offered to control the composition per leaflet, suggests that *insane* could be used to set up simulations of membranes with near-biological complexity. Two recent studies exemplify the potential of *insane* in this respect. To illustrate this, Figure 5 shows the systems generated

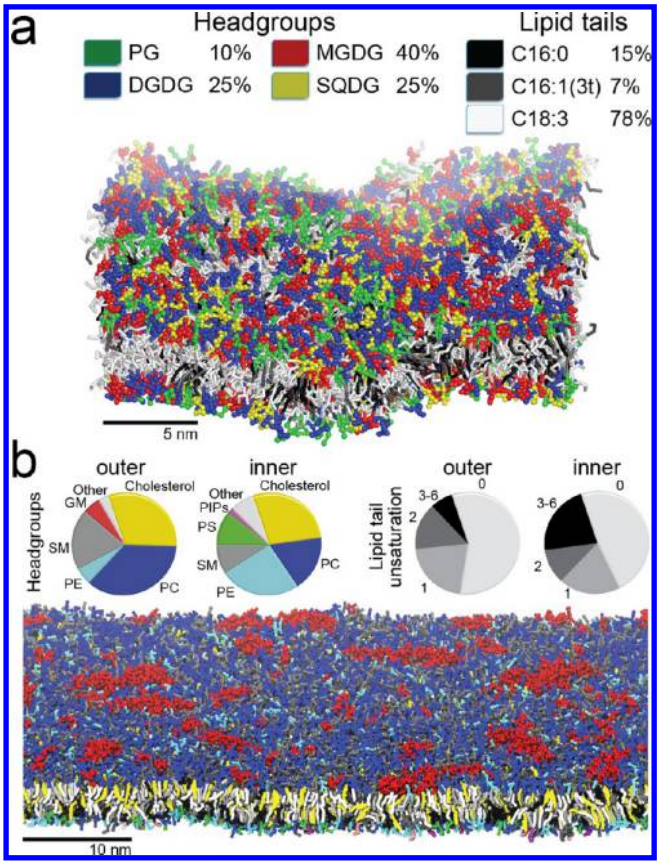



**Figure 5.** Complex, biological bilayer systems setup with *insane*. (a) Thylakoid membrane composed of seven different lipid types, of which one phosphatidylglycerol (PG), two types of digalactosyldiacylglycerol (DGDG), two monogalactosyldiacylglycerol (MGDG), and two sulfoquinovosyldiacylglycerol (SQDG). Relative lipid headgroup and tail composition are shown with a snapshot after 10 $\mu$s of simulation. Image produced using data from ref 49. (b) Idealized mammalian plasma membrane composed of 63 different lipids types, including cholesterol, phosphatidylcholines (PC), sphingomyelins (SM), phosphatidylethanolamines (PE), gangliosides (GM), and phosphatidylserines (PS). The overall headgroup composition and number of unsaturated bonds in the lipid tails are shown for the outer and inner leaflets together with a snapshot of the outer leaflet after 40 $\mu$s of simulation. Image produced using data from ref 50.



DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

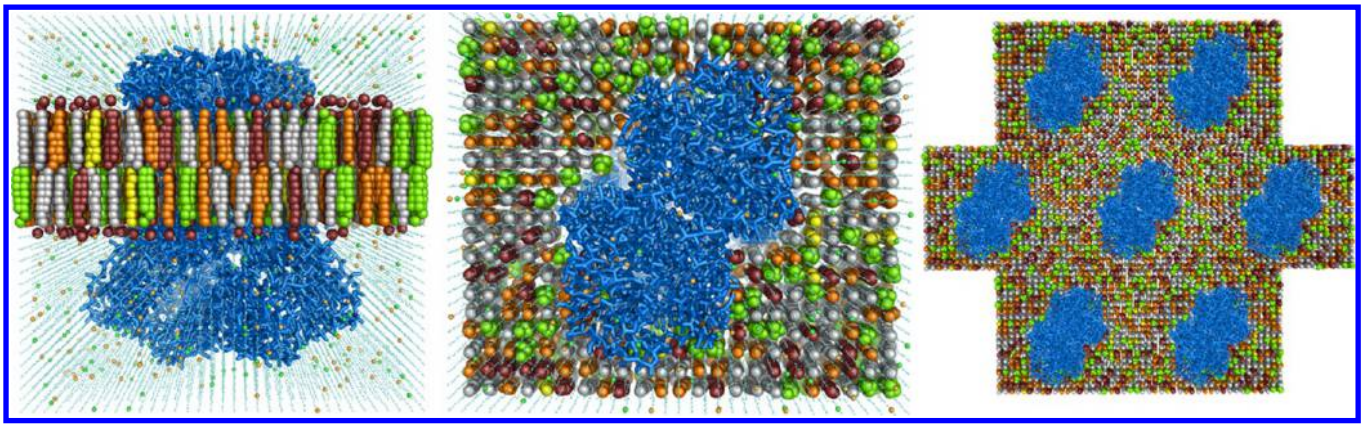


**Figure 6.** Protein−membrane system setup with *insane*. The cytochrome bc1 complex (blue) set up in a hexagonal prism with a bilayer consisting of 40% POPC (gray), 24% POPE (orange), 16.2% PI (ruby), 3.8% POPS (yellow), and 16% cardiolipin (green) and solvated with a 0.1536 M NaCl solution. Left, side view; middle, bottom view; right, seven periodic images, showing the hexagonal lattice structure.

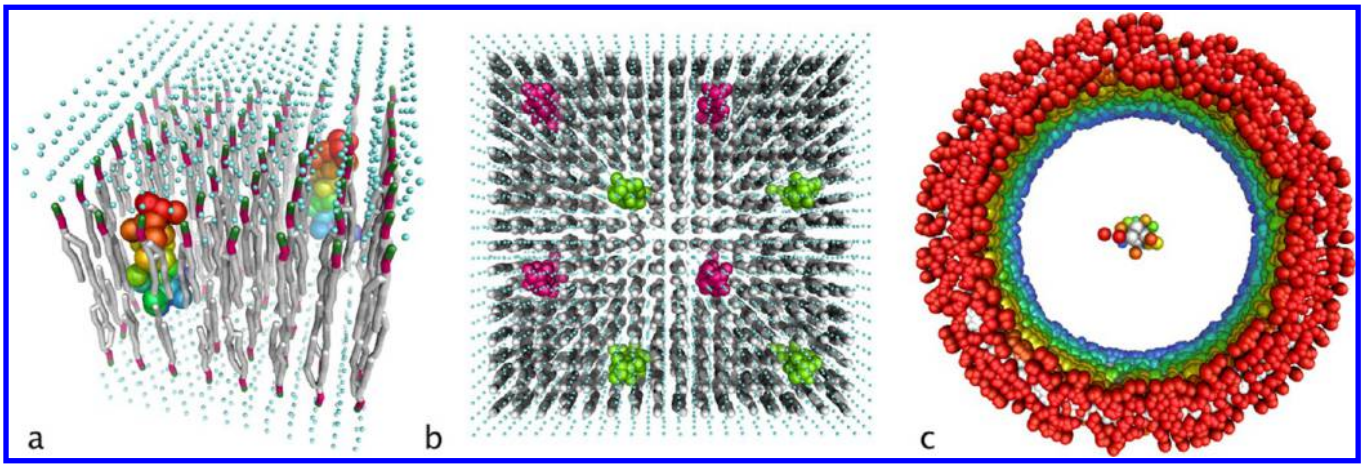



**Figure 7.** High-throughput protein−membrane association studies, using the DAFT approach. (a) Setup of two glycophorin A transmembrane helices in a POPC membrane at a specified distance, generated with *insane*. (b) Top view showing four copies of the system, revealing the DAFT layout for two components. (c) Ensemble of 500 glycophorin A dimer starting configurations, fitted on one helix, exemplifying the distribution of relative orientations at the starting states.

with *insane* for these studies, the details of which are given in refs 49 and 50. The first system is a thylakoid membrane (Figure 5a), comprising seven different lipid types, four of which are glycolipids. The membrane contains 2044 lipids and was simulated for 10 $\mu$s.[49] The second system is a plasma membrane (Figure 5b), consisting of 63 different lipids types (14 different headgroups and 11 different tails), asymmetrically distributed across the leaflets mimicking an idealized mammalian plasma membrane. The membrane contains ~20,000 lipids and was simulated for 40 $\mu$s.[50]

**Inserting Proteins in Bilayers.** One of the original objectives of *insane* was to build combined protein−membrane systems with arbitrary lipid and solvent composition. An example of such a system is shown in Figure 6, which shows the mitochondrial cytochrome bc1 complex embedded in a complex membrane with headgroup composition mimicking the mitochondrial outer membrane. The images show the system as generated by *insane*, exemplifying the regular arrangement of lipids and solvent molecules due to the use of grids. The generation of this structure, containing 42,102 particles, took a few seconds, starting from the atomistic crystal structure. Simulating this system on a 2.30 Ghz Intel Core i7 yields 270 ns/day.

**High-Throughput Simulations of Protein−Membrane Systems.** The intrinsic robustness of *insane* in generating protein−membrane−solvent systems allows for high-throughput simulations of coarse-grained protein structures. This has led us to develop a method for studying protein−protein interactions from many short simulations. The method is called DAFT for "Docking Assay For Transmembrane components"[27] and is built around a set of layouts for simulations containing 0−9 protein components. In Figure 7 an example is shown of a DAFT setup for two glycophorin A transmembrane (TM) helices, consisting of 500 simulations. Figure 7a shows a single simulation system, with the two TM helices embedded in a POPC membrane. Figure 7b shows the same, but copied four times, exemplifying the layout for two components, which is developed such that each protein is surrounded by the partner and not by its own periodic image. Finally, Figure 7c shows the ensemble of 500 starting configurations, fitted on one of the helices, exemplifying the random distribution of relative orientations, ensuring that the set of simulations is unbiased.

**Building Atomistic Systems.** The appeal on the robustness of coarse-grain models makes the *insane* approach intrinsically unsuitable for directly building atomistic systems. However, as has been mentioned before, any method for generating atomistic membranes of complex composition poses


DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

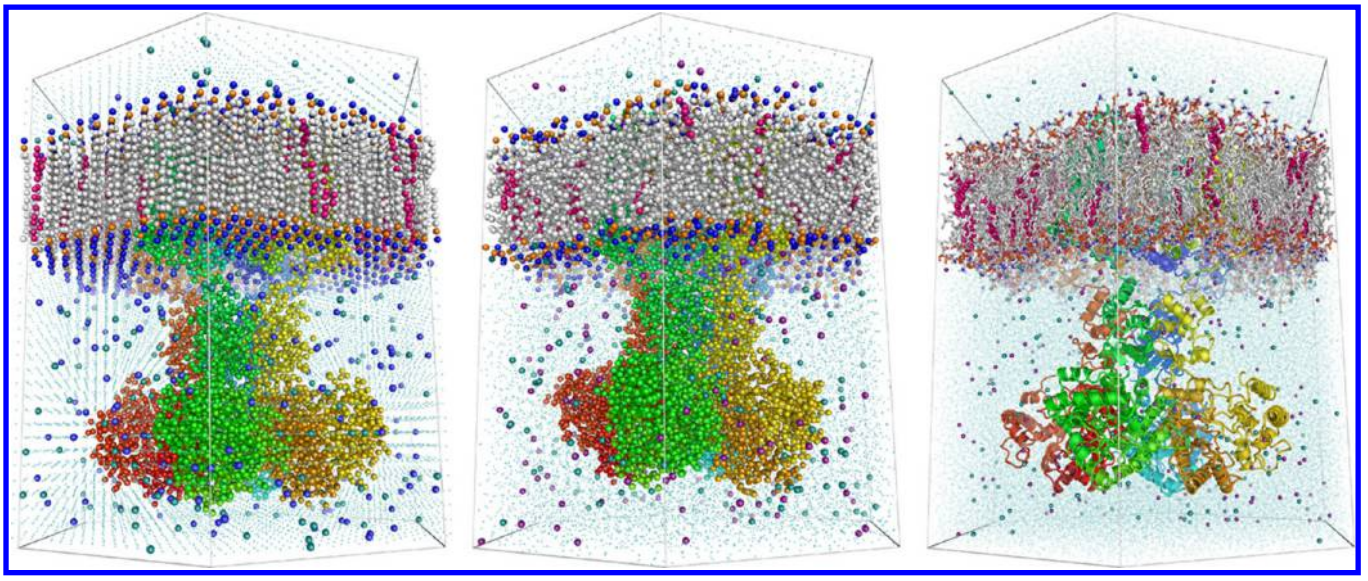


**Figure 8.** Building atomistic protein−membrane systems using *insane*. *insane* can be used in conjunction with *martinate*[34] and *backward*[36] to generate an atomistic equilibrated membrane of complex composition around a protein, as shown here for the Kv1.2−2.1 paddle chimera channel (Protein Data Bank ID 4JTC). The protein was first coarse-grained with *martinize* and subsequently, embedded in a POPC/cholesterol membrane and solvated with a NaCl solution (left panel). The whole system was then relaxed (middle panel) and converted to atomistic resolution (right panel).

the problem of equilibration. In fact, probably the best approach to obtain an equilibrated complex bilayer, with or without protein, is using a sequential multiscale approach, similar to the approaches used by Aryal et al.[51] and Rzepiela et al.[52] The former used CG simulations to condense a bilayer around the protein, while the latter used a CG model to simulate membrane pore formation by antimicrobial peptides. Both then converted their systems to atomistic resolution using a tool for reverse transformation.

Recently, we introduced the *backward*[36] method for converting CG models to atomistic resolution. This method was shown to be accurate and robust and can be integrated in a workflow with *martinate*/*insane* to yield equilibrated membrane and protein−membrane systems. Here, we illustrate the procedure in Figure 8. Figure 8 shows the Kv1.2−2.1 paddle chimera channel (PDB ID 4JTC). The protein was obtained from the OPM database and processed first to obtain an atomistic structure and topology, using the GROMACS tool pdb 2gmx. The structure was then coarse-grained using *martinize* and, with *insane*, embedded in a POPC/cholesterol membrane surrounded by an isotonic NaCl solution. The CG system contained a total of 31,000 particles and was briefly relaxed in a 10 ns simulation, taking 1.6 h on a 2.30 GHz Intel Core i7. Clearly, for full equilibration the simulation should be run longer, and this stage is the computationally most expensive part of the protocol. Subsequently, the whole system was converted to atomistic resolution, using the atomistic topology as the backmapping target. This process yielded a total of 274,885 atoms.

With *backward* it is currently possible to convert from Martini to GROMOS, AMBER, and CHARMM. Like *insane*, the method *backward* is in principle force field independent, and both can be adapted to allow sequential multiscale workflows for setting up atomistic membrane simulations using other CG/AA force field combinations. Alternatively, *insane* and *backward* can be joined with the LipidConverter,[15] using Martini and, e.g., GROMOS to yield an intermediate atomistic system, which is then converted to another force field definition.

## CONCLUSION

The *insane* approach is shown to enable high-throughput simulations of lipid bilayers and of protein−membrane systems, as well as the generation of some of the most complex membranes simulated to date, such as the thylakoid[49] and plasma membranes.[50] The incorporation in the CG simulation protocol *martinate* makes setting up CG protein−membrane simulations a single-step protocol, and the coupling with the reverse transformation tool *backward* extends the scope of the program to set up atomistic simulations. Due to the long equilibration times of multicomponent membranes, such a sequential multiscale approach currently appears to be the most optimal route to generate equilibrated systems with complex lipid constitution.

Our work also demonstrates the use of *insane* for computational lipidomics, characterizing the properties of a range of diacylglycerol-based lipids with varying headgroup and tail constitution. The list of lipid types explored here demonstrates the scope and flexibility of the Martini lipidome and hopefully will guide researchers in selecting appropriate CG lipids in future studies.

Planned future extensions of *insane* include features such as the building of multilamellar and nonlamellar membranes, including vesicles, as well as biased placement of lipids, based on preferred interactions, which should allow decreasing the equilibration time for large systems. In addition, new molecular templates and building blocks for on-the-fly generation will be added to extend the range of systems that can be generated.

*insane* and related programs, including the lipid topology builder, *martinate*, and *backward* have been made available at the Martini portal (http://cgmartini.nl/).



DOI: 10.1021/acs.jctc.5b00209
J. Chem. Theory Comput. 2015, 11, 2144−2155

## ■ ASSOCIATED CONTENT

### Supporting Information

Text describing default templates present in *insane* and examples of *insane* building commands, figures showing calculated lipid properties, and tables listing Martini lipid naming conventions and all calculated lipid properties. The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jctc.5b00209.

## ■ AUTHOR INFORMATION


### Corresponding Author

*E-mail: tsjerkw@gmail.com.

### Author Contributions

‖T.A.W. and H.I.I. contributed equally to this work.



### Funding

We acknowledge support by the emerging field initiative "Synthetic Biology" of the Friedrich-Alexander University of Erlangen-Nürnberg (T.A.W. and R.A.B.) and by the Research Training Group "Dynamic Interactions at Biological Membranes—From Single Molecules to Tissue" (RTG 1962, R.A.B.) funded by the German Science Foundation (DFG). This work was also supported in part by the Canadian Institutes for Health Research (D.P.T.). D.P.T. is an Alberta Innovates Health Solutions Scientist and Alberta Innovates Technology Futures Strategic Chair in (Bio)Molecular Simulation. H.I.I. was supported by a Rubicon grant from The Netherlands Organization for Scientific Research (NWO). S.J.M. is supported by a TOP and ECHO grant from NWO. Computer access was granted from the National Supercomputing Facilities through NWO.


### Notes

The authors declare no competing financial interest.

## ■ REFERENCES


(1) Gorter, E.; Grendel, F. On Bimolecular Layers of Lipoids on the Chromocytes of the Blood. *J. Exp. Med.* **1925**, *41*, 439−443.

(2) Marrink, S. J.; Lindahl, E.; Edholm, O.; Mark, A. E. Simulation of the Spontaneous Aggregation of Phospholipids Into Bilayers. *J. Am. Chem. Soc.* **2001**, *123*, 8638−8639.

(3) Sperotto, M. M.; May, S.; Baumgaertner, A. Modelling of Proteins in Membranes. *Chem. Phys. Lipids* **2006**, *141*, 2−29.

(4) Kandt, C.; Ash, W. L.; Tieleman, D. P. Setting Up and Running Molecular Dynamics Simulations of Membrane Proteins. *Methods* **2007**, *41*, 475−488.

(5) 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.

(6) 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.

(7) Wu, E. L.; Cheng, X.; Jo, S.; Rui, H.; Song, K. C.; Dávila-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.

(8) Ghahremanpour, M. M.; Arab, S. S.; Aghazadeh, S. B.; Zhang, J.; van der Spoel, D. MemBuilder: A Web-Based Graphical Interface to Build Heterogeneously Mixed Membrane Bilayers for the GROMACS Biomolecular Simulation Program. *Bioinformatics* **2014**, *30*, 439−441.

(9) Domański, J.; Stansfeld, P. J.; Sansom, M. S. P.; Beckstein, O. Lipidbook: A Public Repository for Force-Field Parameters Used in Membrane Simulations. *J. Membr. Biol.* **2010**, *236*, 255−258.

(10) Faraldo-Gómez, J. D.; Smith, G.; Sansom, M. S. Setting Up and Optimization of Membrane Protein Simulations. *Eur. Biophys. J.* **2002**, *31*, 217−227.

(11) Schmidt, T. H.; Kandt, C. LAMBADA and InflateGRO2: Efficient Membrane Alignment and Insertion of Membrane Proteins for Molecular Dynamics Simulations. *J. Chem. Inf. Model.* **2012**, *52*, 2657−2669.

(12) Wolf, M. G.; Hoefling, M.; Aponte-Santamaria, C.; Grubmueller, H.; Groenhof, G. G_Membed: Efficient Insertion of a Membrane Protein Into an Equilibrated Lipid Bilayer with Minimal Perturbation. *J. Comput. Chem.* **2010**, *31*, 2169−2174.

(13) Javanainen, M. Universal Method for Embedding Proteins Into Complex Lipid Bilayers for Molecular Dynamics Simulations. *J. Chem. Theory Comput.* **2014**, *10*, 2577−2582.

(14) Durrant, J. D.; Amaro, R. E. LipidWrapper: An Algorithm for Generating Large-Scale Membrane Models of Arbitrary Geometry. *PLoS Comput. Biol.* **2014**, *10*, No. e1003720.

(15) Larsson, P.; Kasson, P. M. Lipid Converter, a Framework for Lipid Manipulations in Molecular Dynamics Simulations. *J. Membr. Biol.* **2014**, *247*, 1137−1140.

(16) Arnarez, C.; Mazat, J.-P.; Elezgaray, J.; Marrink, S. J.; Periole, X. Evidence for Cardiolipin Binding Sites on the Membrane-Exposed Surface of the Cytochrome Bc1. *J. Am. Chem. Soc.* **2013**, *135*, 3112−3120.

(17) Arnarez, C.; Marrink, S. J.; Periole, X. Identification of Cardiolipin Binding Sites on Cytochrome C Oxidase at the Entrance of Proton Channels. *Sci. Rep.* **2013**, *3*, No. 1263.

(18) Periole, X.; Knepp, A. M.; Sakmar, T. P.; Marrink, S. J.; Huber, T. Structural Determinants of the Supramolecular Organization of G Protein-Coupled Receptors in Bilayers. *J. Am. Chem. Soc.* **2012**, *134*, 10959−10965.

(19) Ingólfsson, H. I.; Lopez, C. A.; Uusitalo, J. J.; de Jong, D. H.; Gopal, S. M.; Periole, X.; Marrink, S. J. The Power of Coarse Graining in Biomolecular Simulations. *WIREs Comput. Mol. Sci.* **2014**, *4*, 225−248.

(20) Marrink, S. J.; De Vries, A. H.; Mark, A. E. Coarse Grained Model for Semiquantitative Lipid Simulations. *J. Phys. Chem. B* **2004**, *108*, 750−760.

(21) 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*, 7812−7824.

(22) 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*, 819−834.

(23) Marrink, S. J.; Tieleman, D. P. Perspective on the Martini Model. *Chem. Soc. Rev.* **2013**, *42*, 6801−6822.

(24) Phillips, J. C.; Braun, R.; Wang, W.; Gumbart, J.; Tajkhorshid, E.; Villa, E.; Chipot, C.; Skeel, R. D.; Kale, L.; Schulten, K. Scalable Molecular Dynamics with NAMD. *J. Comput. Chem.* **2005**, *26*, 1781−1802.

(25) Risselada, H. J.; Mark, A. E.; Marrink, S. J. Application of Mean Field Boundary Potentials in Simulations of Lipid Vesicles. *J. Phys. Chem. B* **2008**, *112*, 7438−7447.

(26) Wassenaar, T. A.; van Dijk, M.; Loureiro-Ferreira, N.; van der Schot, G.; de Vries, S. J.; Schmitz, C.; van der Zwan, J.; Boelens, R.; Giachetti, A.; Ferella, L.; Rosato, A.; Bertini, I.; Herrmann, T.; Jonker, H. R. A.; Bagaria, A.; Jaravine, V.; Güntert, P.; Schwalbe, H.; Vranken, W. F.; Doreleijers, J. F.; Vriend, G.; Vuister, G. W.; Franke, D.; Kikhney, A.; Svergun, D. I.; Fogh, R. H.; Ionides, J.; Laue, E. D.; Spronk, C.; Jurkša, S.; Verlato, M.; Badoer, S.; Dal Pra, S.; Mazzucato, M.; Frizziero, E.; Bonvin, A. M. J. J. WeNMR: Structural Biology on the Grid. *J. Grid Comput.* **2012**, *10*, 743−767.

(27) Wassenaar, T. A.; Pluhackova, K.; Moussatova, A.; Sengupta, D.; Marrink, S. J.; Tieleman, D. P.; Bockmann, R. A. High-Throughput Simulations of Dimer and Trimer Assembly of Membrane Proteins. The DAFT Approach. *J. Chem. Theory Comput.* **2015**, DOI: 10.1021/ct5010092.




DOI: 10.1021/acs.jctc.5b00209
*J. Chem. Theory Comput.* 2015, 11, 2144−2155

(28) Lomize, M. A.; Lomize, A. L.; Pogozheva, I. D.; Mosberg, H. I. OPM: Orientations of Proteins in Membranes Database. *Bioinformatics* **2006**, *22*, 623−625.
(29) Wassenaar, T. A.; Mark, A. E. The Effect of Box Shape on the Dynamic Properties of Proteins Simulated Under Periodic Boundary Conditions. *J. Comput. Chem.* **2006**, *27*, 316−325.
(30) Pronk, S.; Pall, S.; Schulz, R.; Larsson, P.; Bjelkmar, P.; Apostolov, R.; Shirts, M. R.; Smith, J. C.; Kasson, P. M.; van der Spoel, D.; Hess, B.; Lindahl, E. GROMACS 4.5: A High-Throughput and Highly Parallel Open Source Molecular Simulation Toolkit. *Bioinformatics* **2013**, *29*, 845−854.
(31) Berendsen, H. J. C.; Postma, J. P. M. Molecular Dynamics with Coupling to an External Bath. *J. Chem. Phys.* **1984**, *81*, 3684−3690.
(32) Parrinello, M.; Rahman, A. Polymorphic Transitions in Single Crystals: A New Molecular Dynamics Method. *J. Appl. Phys.* **1981**, *52*, 7182−7190.
(33) Bussi, G.; Donadio, D.; Parrinello, M. Canonical Sampling Through Velocity Rescaling. *J. Chem. Phys.* **2007**, *126*, No. 014101.
(34) Wassenaar, T. A.; Ingólfsson, H. I.; Prieß, M.; Marrink, S. J.; Schäfer, L. V. Mixing MARTINI: Electrostatic Coupling in Hybrid Atomistic−Coarse-Grained Biomolecular Simulations. *J. Phys. Chem. B* **2013**, *117*, 3516−3530.
(35) 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*, 687−697.
(36) 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*, 676−690.
(37) Oostenbrink, C.; Villa, A.; Mark, A. E.; van Gunsteren, W. F. A Biomolecular Force Field Based on the Free Enthalpy of Hydration and Solvation: The GROMOS Force-Field Parameter Sets 53A5 and 53A6. *J. Comput. Chem.* **2004**, *25*, 1656−1676.
(38) Nagle, J. F.; Tristram-Nagle, S. Structure of Lipid Bilayers. *Biochim. Biophys. Acta* **2000**, *1469*, 159−195.
(39) Lewis, B. A.; Engelman, D. M. Lipid Bilayer Thickness Varies Linearly with Acyl Chain Length in Fluid Phosphatidylcholine Vesicles. *J. Mol. Biol.* **1983**, *166*, 211−217.
(40) Dickey, A.; Faller, R. Examining the Contributions of Lipid Shape and Headgroup Charge on Bilayer Behavior. *Biophys. J.* **2008**, *95*, 2636−2646.
(41) Marsh, D. *Handbook of Lipid Bilayers*, 2nd ed.; CRC Press: Boca Raton, FL, USA, 2013.
(42) Kučerka, N.; Nieh, M.-P.; Katsaras, J. Fluid Phase Lipid Areas and Bilayer Thicknesses of Commonly Used Phosphatidylcholines as a Function of Temperature. *Biochim. Biophys. Acta* **2011**, *1808*, 2761−2771.
(43) Petrache, H. I.; Dodd, S. W.; Brown, M. F. Area Per Lipid and Acyl Length Distributions in Fluid Phosphatidylcholines Determined by (2)H NMR Spectroscopy. *Biophys. J.* **2000**, *79*, 3172−3192.
(44) Marrink, S. J.; Risselada, J.; Mark, A. E. Simulation of Gel Phase Formation and Melting in Lipid Bilayers Using a Coarse Grained Model. *Chem. Phys. Lipids* **2005**, *135*, 223−244.
(45) Kooijman, E. E.; Carter, K. M.; van Laar, E. G.; Chupin, V.; Burger, K. N.; de Kruijff, B. What Makes the Bioactive Lipids Phosphatidic Acid and Lysophosphatidic Acid So Special? *Biochemistry* **2005**, *44*, 17007−17015.
(46) Shin, J. J.; Loewen, C. J. Putting the pH Into Phosphatidic Acid Signaling. *BMC Biol.* **2011**, *9*, 85.
(47) Sackmann, E.; Kotulla, R.; Heiszler, F. J. On the Role of Lipid-Bilayer Elasticity for the Lipid-Protein Interactions and the Indirect Protein-Protein Coupling. *Can. J. Biochem. Cell Biol.* **1984**, *62*, 778−788.
(48) Lundbæk, J. A.; Collingwood, S. A.; Ingólfsson, H. I.; Kapoor, R.; Andersen, O. S. Lipid Bilayer Regulation of Membrane Protein Function: Gramicidin Channels as Molecular Force Probes. *J. R. Soc. Interface* **2010**, *7*, 373−395.
(49) van Eerden, F. J.; de Jong, D. H.; de Vries, A. H.; Wassenaar, T. A.; Marrink, S. J. Characterization of Thylakoid Lipid Membranes from Cyanobacteria and Higher Plants by Molecular Dynamics Simulations. *Biochem. Biophys. Acta, Biomembr.* **2015**, *848*, 1319−1330.
(50) 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*, 14554−14559.
(51) Aryal, P.; Abd-Wahab, F.; Bucci, G.; Sansom, M. S. P.; Tucker, S. J. A Hydrophobic Barrier Deep within the Inner Pore of the TWIK-1 K2P Potassium Channel. *Nat. Commun.* **2014**, *5*, 1−9.
(52) Rzepiela, A. J.; Sengupta, D.; Goga, N.; Marrink, S. J. Membrane Poration by Antimicrobial Peptides Combining Atomistic and Coarse-Grained Descriptions. *Faraday Discuss.* **2009**, *144*, 431.


DOI: 10.1021/acs.jctc.5b00209
*J. Chem. Theory Comput.* 2015, 11, 2144−2155