Chapter 1

Multiscale Modeling and Simulation Approaches to Lipid–Protein Interactions

Roland G. Huber, Timothy S. Carpenter, Namita Dube, Daniel A. Holdbrook, Helgi I. Ingólfsson, William A. Irvine, Jan K. Marzinek, Firdaus Samsudin, Jane R. Allison, Syma Khalid, and Peter J. Bond

Abstract

Lipid membranes play a crucial role in living systems by compartmentalizing biological processes and forming a barrier between these processes and the environment. Naturally, a large apparatus of biomolecules is responsible for construction, maintenance, transport, and degradation of these lipid barriers. Additional classes of biomolecules are tasked with transport of specific substances or transduction of signals from the environment across lipid membranes. In this article, we intend to describe a set of techniques that enable one to build accurate models of lipid systems and their associated proteins, and to simulate their dynamics over a variety of time and length scales. We discuss the methods and challenges that allow us to derive structural, mechanistic, and thermodynamic information from these models. We also show how these models have recently been applied in research to study some of the most complex lipid–protein systems to date, including bacterial and viral envelopes, neuronal membranes, and mammalian signaling systems.

Key words Molecular dynamics (MD) simulation, Molecular modeling, Protein–lipid interactions, Lipid-binding protein, Membrane proteins, Membrane peptides, Multiscale, Coarse-grained (CG) models

1 Introduction

Lipid membranes are ubiquitously used by nearly all life forms to separate themselves from their environment, as well as compartmentalize different functional areas internally. This compartmentalization necessitates on the one hand a mechanism for construction and maintenance of the lipid barriers, and on the other hand a means to transport substances and transmit signals across these membranes to allow for the uptake of nutrients, reaction to changes in environmental conditions, cell–cell communication, etc. [1, 2]. These functions are usually provided by various proteins. In the context of lipid homeostasis, the low solubility of

Jörg H. Kleinschmidt (ed.), *Lipid-Protein Interactions: Methods and Protocols*, Methods in Molecular Biology, vol. 2003, https://doi.org/10.1007/978-1-4939-9512-7_1,

lipid species in water necessitates an array of transport proteins to shuttle the lipids from their sites of synthesis to their terminal locations. In the context of signaling and cross-membrane transport, membrane-associated and membrane-embedded proteins play a crucial role. These can be roughly divided into peripheral membrane-associated and transmembrane peptides and proteins. Major classes of transmembrane proteins are cellular receptors and a variety of pores and channels. It is estimated that 20–30% of genes encode membrane proteins in most species [3] and their crucial role in cellular signaling makes them highly attractive drug targets [4]. Unfortunately, the study of this interesting class of proteins is made more challenging by a variety of biophysical phenomena. Membrane proteins are notoriously hard to crystallize, and hence it is difficult to obtain reliable structural information in many cases [5]. Moreover, membrane proteins may not easily adopt their native fold in the absence of a lipid bilayer environment [6]. Their exposed hydrophobic surfaces can also interfere with expression and purification [5].

Accurate modeling of transmembrane and membrane-associated peptides and proteins can therefore provide an attractive pathway to the investigation of these important biomolecules. Computational modeling of proteins in aqueous solution is a well-established research tool and is used ubiquitously in biophysical research and drug discovery [7–13]. The special attraction for computational modeling for protein–lipid interactions lies in the fact that a suite of highly optimized modeling techniques can be applied to research problems that are difficult to tackle with conventional experimental techniques. Molecular dynamics (MD) simulations represent the biological systems in atomistic or near-atomistic resolution, and apply physics-based interaction potentials among the interaction sites [7]. In conjunction with application of appropriate thermostats/barostats for maintenance of the system ensemble (*see* **Note 1**), the equations of motions can be propagated forward in time to obtain a realistic trajectory of the particles in the system. A large amount of effort has gone into the derivation of appropriate interaction potentials over recent decades, yielding a variety of different force fields that can be applied to a system of interest (*see* **Notes 2–4**). Most force fields consist of a similar set of additive functions, which may be separated into bonded and non-bonded terms. The former broadly consist of harmonic bond and angle terms, and typically sinusoidal functions for torsion angles. The latter consist of pairwise potentials for charge–charge interactions, described using Coulomb's law, and van der Waals interactions, described via the Lennard–Jones potential.

The MD simulation technique is in principle relatively independent of scale; interaction sites may consist either of all atoms (fully atomistic), heavy atoms grouped together with nonpolar hydrogens (united atom description, in which only the polar hydrogens

are treated explicitly), or of even larger groups of atoms (coarse-grained description). Extremely coarse-grained (CG) models can group hundreds of atoms into single interaction sites (e.g., to investigate the dynamics of large viruses) [14–16]. The key challenge in generating accurate models is the derivation of model parameters. Whereas there exists a well-established set of fully atomistic and united-atom force fields for both proteins (mainly CHARMM [17, 18], Amber [19, 20], GROMOS [21]) and lipids (Slipids [22–25], Lipid14 and 17 [26] CHARMM [83], GROMOS [27, 28]), only a limited number of coarser, larger-scale force fields readily exist, the most prominent of which is MARTINI [29–32] which offers parameters for proteins, lipids, and other biomolecules. However, a multiscale approach can be used to validate a customized, coarser model to ensure that it faithfully reproduces the desired observables of a more detailed approach. Although a CG model will omit details that could be discerned in atomistic simulations, these models offer access to time and length scales inaccessible at higher detail [29, 33–35]. Moreover, mapping between different representations is possible, and mapping procedures are an active area of research [29, 35–38]. Although GPU acceleration has allowed for a considerable speedup of molecular simulations [20, 39], atomistic resolution simulations are currently limited to the hundreds of nanoseconds to microsecond time scale for a typical membrane-embedded protein system, unless specialized hardware is used [40, 41]. CG simulation approaches at the scale of the MARTINI force field, which maps approximately four heavy atoms to one interaction site, typically allow for simulations of the same systems over tens to hundreds of microseconds, or more commonly, for studies of much larger systems, for example, to study aggregation or oligomerization phenomena [42, 43].

We will now outline the basic prerequisites, key tools and techniques, and common analysis approaches to the computational modeling of protein–lipid systems. We will also present an exposé of the successful application of the presented techniques to biological research problems.

2 Materials

2.1 Survey of Structural Information

In order to construct a computational model of a protein–lipid system, structural information is crucial. Most known structures of biomolecules are deposited in the Protein Data Bank [44, 45] (http://www.rcsb.org/) and are freely accessible for download. The contents of the PDB predominantly comprise of protein structures determined by X-ray crystallography, but in recent years, the quantity of solution NMR structures [46], and cryo-electron microscopy (EM) structures [47] has been increasing steadily.

Available structures range from small peptides and single domain proteins to large-scale multiprotein or heterogeneous complexes such as the human ribosome or complete virus particles. A good starting structure for a molecular simulation may contain proteins, nucleic acids, or ligands of all sorts, but it is necessary that the structure has a sufficient resolution and that the key areas of interest are resolved [48, 49]. Resolution is mainly a limiting factor in the case of cryo-EM structures, but recent technological advances [47, 50] now allow for structure determination at near-atomistic resolution. NMR and X-ray structures may have missing dynamic or disordered regions, as these properties preclude the collection of high-quality experimental data in these regions. This is of special concern for membrane proteins [51]. Such missing regions must be carefully considered during the modeling process (*see* **Note 5**). At this stage, it is also crucial to familiarize oneself with the particular systems for which the structural measurements were undertaken, as sometimes the coordinates may have been determined for truncated subdomains, mutant or hybrid variants, or obtained under specific pH/solvent conditions that may or may not correspond to the conditions that are to be modeled.

2.2 Choice of Representation

Following a survey of available structural information, one needs to determine how the system shall be represented in the model. A selection of different force fields and resolution scales is readily available at the atomistic and near-atomistic scales (*see* **Notes 2** and **3**). Whereas some force fields try to cover as broad a chemical space as possible (a property highly valued for modeling specific ligands), other force fields specialize in a particular classes of system (e.g., lipids, proteins, or nucleic acids) and are not necessarily easily combined with others. Moreover, the choice of representation is also influenced by the research objectives: modeling the specific interactions of small ligands or individual lipids with a protein may call for a fully atomistic representation, whereas in studying, for example, large-scale aggregation behavior, such a choice may be detrimental, as the resulting loss in accessible time and length scales outweighs the more accurate description of interactions and hence a coarser representation would be appropriate.

2.3 Structure Preparation

Before a specific protein–lipid system can be modeled, several preparatory steps are necessary to ensure that the modeled system approximates the state of the biological system as close as possible. Common tasks in this pipeline include: The modeling of missing loops [52–54] and the reversal or introduction of specific mutations [52, 53] (*see* **Note 5**); deciding on the protonation state of acidic and basic side chains appropriate to the pH conditions that are to be modeled; inclusion of physiologically appropriate disulfide bonds; capping of protein chains at the N- and/or C-termini if a truncated protein is to be modeled; and addition of solvent,

counterions, and/or structural ions to achieve a biologically realistic system (*see* **Note 6**). In addition, modeling a lipid environment of appropriate composition and phase should be a key aim for studying any biologically relevant membrane phenomena. The ultimate goal is to obtain a model of the protein–lipid system that faithfully reproduces as many properties of the biological system as possible. It is therefore highly recommended that this step is undertaken with the utmost diligence, and that all desired inputs and properties are double-checked and verified, as at this point no significant computational resources have been expended, and later discovery of mistakes at this stage may invalidate weeks or months of simulation calculations.

2.4 Membrane Embedding

One of the key tasks for the simulation of membrane-associated and membrane-embedded proteins is their placement inside a representative membrane, for which existing coordinates and topologies are available (*see* **Note 2**). Several tools and methods are available that assist in the construction of lipid bilayers as well as the embedding [55–58] of proteins within them (*see* **Note 4** and Subheading 3 for further details). A key facet of membrane protein simulations is that there is a fundamental anisotropy within the system, as the surface tension of the membrane needs to be modeled independently from the global pressure scaling of the full system (*see* **Note 1**).

2.5 Multiscale Models

It is often desirable to conduct simulations at a CG level, as the spatial and temporal scales accessible are usually 1–2 orders of magnitude increased. However, it is necessary to ensure that the CG representations are able to accurately reflect the behavior observed at a higher level of accuracy in the key properties of interest. Hence, a multiscale approach [36, 38, 59] is often pursued, where high-resolution atomic-scale models are used to derive and validate coarser models, which can then in turn be used to sample slow processes in large systems, while having confidence that the individual model components retain appropriate behavior. Conversely, parts of CG simulations can be extracted and can be back-mapped to higher-resolution representations, where after some refinement these structures are either used as the starting point for further simulations or are analyzed with respect to specific interactions.

2.6 The Simulation Loop

The basic idea behind MD simulations is to start at a specific structure close to the system equilibrium, assign velocities corresponding to a specific temperature, and then propagate the motions of all particles forward for a small time step. At each time step, the forces that act on each particle are calculated from the gradients of the pair potentials that constitute the force field. These forces are then applied to each individual particle, changing their velocity, which is then again propagated forward in time yielding

new particle positions. At these new positions the forces are again evaluated, thus concluding the simulation loop. This simulation loop is performed for as many steps as desired to obtain a trajectory for each individual particle at each point in time. The key limiting factor in this procedure is the magnitude of the time step, which has to be small enough to prevent even high-velocity particles from deviating too far outside equilibrium bond lengths, angles, etc. Such deviation is unphysical and leads to very large reaction forces, which introduces numerical instability that may cause the simulation to crash. Advanced integration methods, such as a Verlet-type integrator, offer added numerical stability and energy conservation. Typical time step durations for atomistic simulations are 1–2 fs, subject to the use of constraints to stabilize high-frequency bonds or angles such as those associated with hydrogen atoms, whereas even moderately CG simulations typically allow time steps upward of 15 fs. As a consequence, obtaining a 1 μs trajectory of an atomistic simulation requires the simulation loop to be evaluated 0.5–1 billion times, which requires considerable computational resources. A variety of software packages is available to perform these calculations, the most broadly used of which are GROMACS [60–62], NAMD [63, 64], CHARMM [17], and Amber [19, 20, 65].

2.7 Simulation Protocol

Following careful system preparation, a simulation run is usually preceded by an energy minimization protocol. The main purpose of this minimization is to reduce or eliminate close contacts, overlaps, or otherwise energetically highly unfavorable situations that would introduce a steep energy gradient and hence produce large forces in the initial steps of a simulation. A thorough minimization may be conducted in several steps, keeping different parts of the system restrained (e.g., one may initially only allow the solvent to move). Following minimization, an equilibration procedure is generally performed. This may take the form of a "thermalization" protocol to ensure that the system is equilibrated at the temperature of interest, ideally along with one or more simulations in which initially strong harmonic restraints are placed on protein and/or lipid components, allowing for temperature and pressure to stabilize and the solvent to adjust to the solute. These restraints are gradually relaxed during equilibration until a stable system is obtained without restraints. A proper equilibration protocol should retain all key structural characteristics of the investigated proteins. Production simulations of membrane–protein systems are usually conducted as unrestrained NPT calculations using semiisotropic pressure scaling to account for membrane surface tension, at a temperature above that of the gel-to-liquid phase transition. A low root-mean-square deviation (RMSD) of the protein backbone with respect to its initial/experimental coordinates is usually a good metric to judge stability of a protein during equilibration, while it is advisable to check for stability of key bilayer properties over time, such as mean

area per lipid (APL), and density of individual lipid components with respect to the membrane normal. It is difficult to propose a specific equilibration protocol for all systems, as the amount and style of equilibration required will differ from system to system, depending amongst other things upon the quality of the initial structure, the number and character of changes introduced during structure preparation, removal or addition of ligands and ions, and the insertion protocol within the lipid membrane.

2.8 Basic Analysis

All simulations will yield a trajectory, but not all simulation trajectories are meaningful. Hence, a simulation result needs to be carefully inspected to ensure that the data produced is robust. Key facets of successful simulations are that the proteins largely retain their equilibrium conformation, lipid membranes are stable, cofactors and ligands remain bound, and temperature, pressure, and energy of the system remain stable. Most of these parameters can be assessed by plotting simple graphs (RMSD, energy vs. time plots, etc.), and by visual inspection of the trajectory using a molecular graphics viewer such as VMD [66]. It is not unusual to observe a certain amount of drift in some or all of these parameters in the early stages of the simulation, as the components find their equilibrium state under the influence of the chosen force field. However, during further analysis, such periods of drift should be excluded and the processing should focus on the converged parts of the trajectory, where system equilibrium states are sampled. If the system does not reach an equilibrium state, it might be helpful to extend the duration of the simulation, or to reconsider earlier choices in how the system model was constructed.

2.9 Computational Considerations

As previously suggested, the computational demand of molecular simulations is considerable whenever large systems and/or long time scales are of interest for a particular research problem. The computational cost of the MD technique is in large part due to the calculation of the nonbonded interactions, in particular due to electrostatics (*see* **Note 7**). Whereas simple calculations can be carried out on a desktop computer, it is usually desirable to have some sort of high-performance computing (HPC) facility available. Modern-day simulation packages are designed to make efficient use of parallel processing, and can utilize a variety of different computer architectures and coprocessors [61]. In recent years, GPU computing has contributed to a significant speedup of simulation runs, as the massively parallel nature of GPUs is well suited to the force computations characteristic of MD [20, 65]. Most simulation packages are distributed in source code form and can hence be compiled and/or modified to take advantage of specific architectures. In addition to general-purpose HPC hardware, some work has gone into the design of application-specific integrated circuits (ASIC) dedicated to MD simulations. The Anton machines created

by DE Shaw Research [40, 41] make use of such ASICs and have enabled access to millisecond time scales for biologically relevant systems in full atomic resolution.

3 Methods

3.1 Setting Up Membrane Protein and Membrane Peptide Systems

Once the coordinates for the protein model have been prepared, it is necessary to embed them in an appropriate lipid environment prior to solvation (*see* **Note 4**). The slow dynamics associated with the exchange of lipids means that care must be taken to correctly position a protein within a model membrane or mimetic environment, to ensure that biologically meaningful results are obtained from the subsequent production simulation. Several approaches are outlined below for building biologically or experimentally relevant systems.

The starting configurations for transmembrane protein simulations are typically created by using one, or a combination of, two strategies: (a) the lipid membrane is constructed around the protein; or (b) the protein is inserted directly into a preexisting, equilibrated lipid membrane (*see* **Note 2**). Both require some relaxation of the system before production data can be collected. The first of these strategies is utilized by the popular CHARMM-GUI online server [56] and "*insane*" script for CG simulations [58]. In certain circumstances, however, the second of these strategies is preferred, particularly if the insertion process minimally perturbs the membrane environment and thereby reduces the time required for the system to relax.

3.1.1 Inserting a Protein Within a Lipid Bilayer Environment

1. The coordinates of some preequilibrated lipid bilayer mixtures are available freely online (*see* **Note 2**), but in cases where a complex mixture is required the bilayer will need be constructed. This may be carried out via a number of available online tools (*see* **Note 4**), or may be performed "in house" using techniques similar to those that construct the lipid bilayer with the protein in situ. The speed of construction of an isolated lipid membrane can be increased by first forming a small, equilibrated bilayer "patch" containing ~16 or so lipids in each leaflet. The initial configuration of the lipids in the patch is biased so that they closely resemble a lipid bilayer: by placing the individual lipids on grid points or by algorithmically packing them into a bilayer configuration (*see* also Subheading 3.3). The assembled bilayer patch is then relaxed for approximately 100–1000 ns, or until an observable property such as the APL has plateaued. After the lipid bilayer has stabilized, the final frame is taken and the patch is replicated in the x and y dimensions (assuming z to be the normal to the bilayer) until the membrane is large enough in size to accommodate the transmembrane protein of interest.

2. The correct initial position of the protein relative to the plane of the bilayer can be estimated by eye, using the position of the amphipathic aromatic "girdles" that anchor themselves in the lipid–head group interface [67]. Alternatively, empirical approaches can be used to facilitate positioning, such as the Orientations of Proteins in Membranes (OPM) database [68] and associated server (http://opm.phar.umich.edu/server.php) which predicts the protein location based on the transfer energy of accessible amino acids between water and the membrane core.
3. A straightforward approach for membrane embedding termed g_membed developed by Wolf et al. [55], related to another method described by Yesylevskyy [69], involves growing an initially "contracted" protein back to its correct size in the membrane plane while pushing lipids away during a short MD simulation, minimally disrupting the lipid bilayer. Similarly, the method of Faraldo-Gomez et al. [70], uses an implicit protein grid-based force field for specificity at the protein–lipid interface and applies weak repulsive forces to nearby lipid molecules perpendicular to the solvent-accessible protein surface during multiple short MD simulations, resulting in a volume adapted to the protein surface with only minimal perturbation of the existing bilayer structure.
4. Another approach, based on the LAMBADA and inflateGRO2 [71] tools, works sequentially to perform the positioning and insertion of the protein. LAMBADA decides the appropriate orientation for insertion by calculating a hydrophilicity score along the protein's axis as it is tilted at angles relative to the bilayer plane. Once positioned, inflateGRO2 differentiates lipids based on their overlap with the protein, removing those with high overlap and relaxing the lipids that exist in the inner most annular layers surrounding the protein.
5. Alternatively, one of the simplest strategies to insert a transmembrane peptide or protein in an equilibrated lipid bilayer is to carefully superimpose the coordinates of both and then to remove any protein-overlapping lipid molecules. This insertion method can result in excessively large holes in the bilayer with the lipids suboptimally packed around the protein, especially for nonuniform proteins with irregular surface shapes. However, with increasing computational power and faster algorithms, it is now often possible to simply use an extended "equilibration period" to fix these issues. In this approach, positional restraints are applied to the protein coordinates to prevent structural drift during an MD simulation, during which local lipid and solvent molecules can relax freely around the protein surface. Subsequently, protein restraints may be gradually released, prior to production MD.

6. The above methodologies may fail in the case of proteins whose surfaces are particularly irregular or asymmetric, potentially leading to large gaps within the lipid bilayer that are not easily equilibrated. A possible solution to this problem is to use alchemical transformation to "grow" the peptide within the membrane phase. Alchemical methods have traditionally been used to calculate the free energy (ΔG) between two thermodynamic states [72]; in practice, this takes the form of a series of transformations between nonphysical intermediates defined as a function of the coupling parameter λ. Such an approach has proven useful for peptides that exist parallel to the membrane surface, but buried within the bilayer. For example, immunoreceptor tyrosine-based activation motifs (ITAMs) containing conserved YxxL/I sequences have been proposed to be embedded in this manner. Phosphorylation of ITAM tyrosines present in the cytoplasmic tails of T-cell receptor (TCR) associated chains serve to propagate antigen-induced activation, and it has been proposed that this may be regulated by the extent of peptide sequestration within the lipid bilayer [73]. To investigate this further, the CD3ε ITAM peptide [73] was alchemically "grown" at different positions within a membrane model. This was achieved via a series of short MD simulations, with the protein coordinates weakly restrained, starting from $\lambda = 0$ (peptide absent) and increasing in successive $\lambda = 0.05$ increments until $\lambda = 1$ (peptide fully inserted). To avoid sudden atomic overlaps due to nascent peptide-lipid interactions, which may cause simulation instabilities, best results were achieved by: (a) independently "switching on" the Lennard–Jones interactions first, followed next by introducing the point charges of the peptide atoms; and (b) using a soft-core potential to avoid endpoint errors [74]. Jefferys et al. coined the term "alchembed" for a variant of this strategy, and have made available a tutorial for the GROMACS simulation package [75]. The ITAM simulations suggested that burial of the peptide within the bilayer hydrophobic core would likely lead to significant lipid deformation and membrane instability (Fig. 1). Instead, in the resting state, the ITAM peptide likely sits at the membrane interface, where the tyrosine residues can still interact with lipids.
7. Finally, a "brute force" simulation strategy entails placing a protein in a random mixture of lipid and solvent, prior to an extended simulation, leading to spontaneous self-assembly into a membrane-inserted state, thereby avoiding the need for user intervention [76]. With appropriate lipid–solvent ratios, micelles or other nonlamellar phases may also be achieved. Unfortunately, the compute times for performing such simulations in full-atomic resolution are costly, and this method can

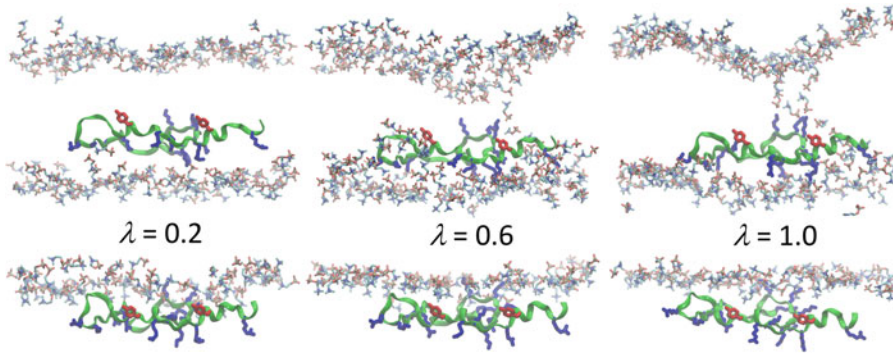

Fig. 1 Alchemical insertion of ITAM peptide into a lipid bilayer within the hydrophobic core (top row) or at the membrane interface (bottom row). Representative snapshots are shown at three different values of coupling parameter λ, indicated inset. Lipid head groups are indicated in CPK wireframe format; the ITAM peptide is shown in green ribbons format, with basic side chains (blue) and tyrosines (red) represented in licorice format. Lipid tails and water molecules are omitted for clarity

be somewhat unpredictable in terms of the final system configuration that is obtained. Fortunately, CG approaches can help to solve this problem [77], yielding significant speed-up and enabling rapid testing of multiple starting conditions, such as system component concentrations and lipid composition. Such a strategy can be useful for complex, multicomponent systems; for example, it has been applied to understand why some thrombin-derived C-terminal peptides in wound fluids form antibacterial amyloid-like particles [43], while others exert antiseptic activity by binding to endotoxin aggregates or immune receptors [78]. Furthermore, once a desired CG protein–lipid assembly has been generated, it is possible to "backmap" or "reverse transform" the coordinates to all-atom representation for detailed refinement. This may be achieved via alignment against libraries of molecular fragments combined with homology modeling approaches [79], or using a tool based on geometric projection and subsequent cycles of relaxation based on energy minimization and position-restrained MD [80], which enables straightforward conversion of MARTINI systems to their atomic counterparts for a variety of common force fields. Such a strategy can, for example, prove useful in predicting both the global conformation and detailed atomic interactions of membrane protein oligomers in a lipid bilayer environment [81].

3.1.2 Setting Up Peripheral Membrane Protein Systems

Peripheral membrane-bound proteins adhere temporarily to the surface of the lipid bilayer, and are important in a wide variety of cellular functions, including, for example, regulatory roles in channels and receptors, enzyme targeting, and signaling in

protein–protein complexes. Stable binding of such proteins often involves electrostatic interactions, which act over long distances between charged amino acids and lipids. While approximate methods are available to predict association of peripheral proteins with membranes, these tend to lack details of specific interactions which may be biologically important. Thus, a new method named Rotational Interaction Energy Profiling (RIEP) has recently been developed to rapidly evaluate the electrostatically optimal orientation of a protein with a lipid bilayer of specific composition, yielding configurations for subsequent seeding of MD simulations [82]:

1. The aim is to rapidly evaluate and identify optimal protein orientation(s) with respect to the membrane on the basis of electrostatics. The procedure allows the characterization of membrane–protein association, the identification of important residues, and initiation of MD simulation of the binding process from predetermined ideal orientation(s).
2. The requirements for this method are: (a) Python wrapper script for calling GROMACS [61] tools, available from https://github.com/allison-group/riep. This can run across multiple nodes of a compute cluster. (b) GROMACS software, version 3 onward. (c) Separate coordinate, topology, and index files for the equilibrated protein and membrane components.
3. The procedure, as outlined in Fig. 2, is as follows:
   (a) Rotate protein coordinates around pitch, roll, and yaw in user-determined degree increments (rotation increments of 30° are recommended).
   (b) Place the rotated protein at a user-determined minimum distance from the membrane. A minimum protein–membrane distance of 5 Å is recommended. Note that this distance may change during energy minimization, typically becoming closer for favorable orientations.
   (c) Combine the rotated protein coordinates with the non-rotated membrane coordinates, and subsequently solvate and energy minimize the system.
   (d) Initiate a short MD simulation of the protein–membrane system for a user-determined number of steps. Fewer than 25 integration steps are recommended; the goal of this and the energy minimization is to relax the system at minimal computational expense.
   (e) Calculate protein–membrane Coulombic and Lennard–-Jones potential energies for each set of protein–membrane coordinates. Optimal orientations of the protein for membrane association are those with the lowest energy. Many extrinsic membrane proteins associate with the membrane via attractive electrostatic interactions; thus, the

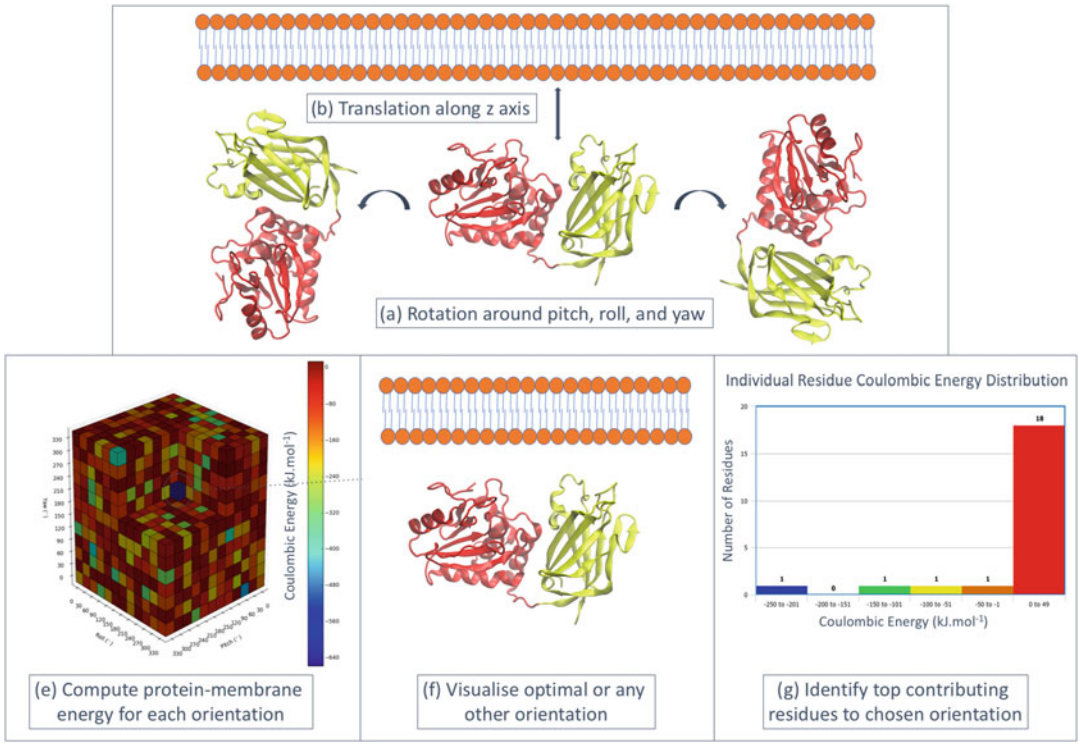

Fig. 2 An efficient protocol for setting up optimal peripheral membrane protein systems. Note that each of the labeled steps correspond to those in the text

Coulombic potential energy is typically of higher magnitude and most informative.

(f) Visualize the optimal coordinates, for example, with VMD [66].

(g) Finally, to calculate per-residue decomposition of protein–membrane energy for a given orientation, residues are selected within the Coulombic cutoff distance of the membrane, and the short MD simulation rerun, using the identified residues of interest as energy groups.

3.2 Preparation of Complex Membrane Protein Systems

3.2.1 Lamellar and Nonlamellar Starting Structures with CHARMM-GUI

It is often desirable to investigate the properties of a specific combination of lipids organized in a nonplanar configuration. When a CG self-assembly-based strategy (*see* Subheading 3.1.1) is insufficient, the CHARMM-GUI server (http://www.charmm-gui.org) may be used to create such systems [56]. Recently, the membrane builder has expanded its repertoire of lipids to include over 180 different variants [83], and it additionally supports the use of CG lipids from the MARTINI force field. The chosen lipids can be assembled into the lamellar and nonlamellar lipid structures of vesicles, micelles, and hexagonal phase membranes [57, 84].

3.2.2 A Protocol for Simulating Enveloped Virus Particles

Beyond the challenge of predicting the orientation of membrane peptides and proteins within a lipid membrane, MD simulations can also be used in refining large multicomponent protein–lipid complexes, such as enveloped viruses, in which a lipid vesicle derived from the cell is "coated" by embedded viral proteins [14]. This was recently demonstrated for the entire dengue virus envelope particle, with a diameter of ~50 nm, based on a combination of data from atomic-resolution and CG MARTINI-based simulations along with cryo-EM [42]. The steps required to achieve this are outlined below, and illustrated schematically in Fig. 3.

1. Initially, atomic-resolution simulations should be performed for isolated viral envelope protein subunits and/or small assemblies thereof, either for the ectodomains in solution, or for the full-length constructs containing transmembrane regions embedded within a small lipid bilayer patch. This provides a measure of the dynamics and stable structural properties of the proteins, enabling subsequent calibration of the CG model. A straightforward way to map such dynamics between resolutions is to utilize an elastic network within the CG MARTINI model [29], and to iteratively tune the associated parameters (i.e., cutoff distances and harmonic potential force constants) until comparable protein flexibility is achieved at both resolutions.
2. At the same time, regions such as flexible loops or transmembrane helices missing from the cryo-EM structure should be reconstructed (*see* Subheading 2). Subsequently, mapping of the entire viral atomic protein coordinates into MARTINI representation should be conducted, along with an energy minimization protocol to ensure the absence of steric clashes (Fig. 3b).
3. A CG viral vesicle may now be built, for example, using the CHARMM-GUI Martini Maker [57], in accordance with the diameter estimated from cryo-EM maps or other experimental measurements, and with a lipid composition guided by available lipidomics data. Due to the tightly packed mesh of proteins typical of (pseudo)icosahedral viral envelopes, a simple process of overlay of protein coordinates and deletion of overlapping lipids is typically insufficient, since most lipid molecules in the vesicle will at least partially coincide with the protein. To overcome this, a procedure involving shrinking of lipids along their principal axes, deletion of remaining overlapping lipids, followed by multiple iterative rounds of energy minimization and protein position-restrained equilibration (Fig. 3a) should result in a reasonable model of the viral protein embedded vesicle [42].

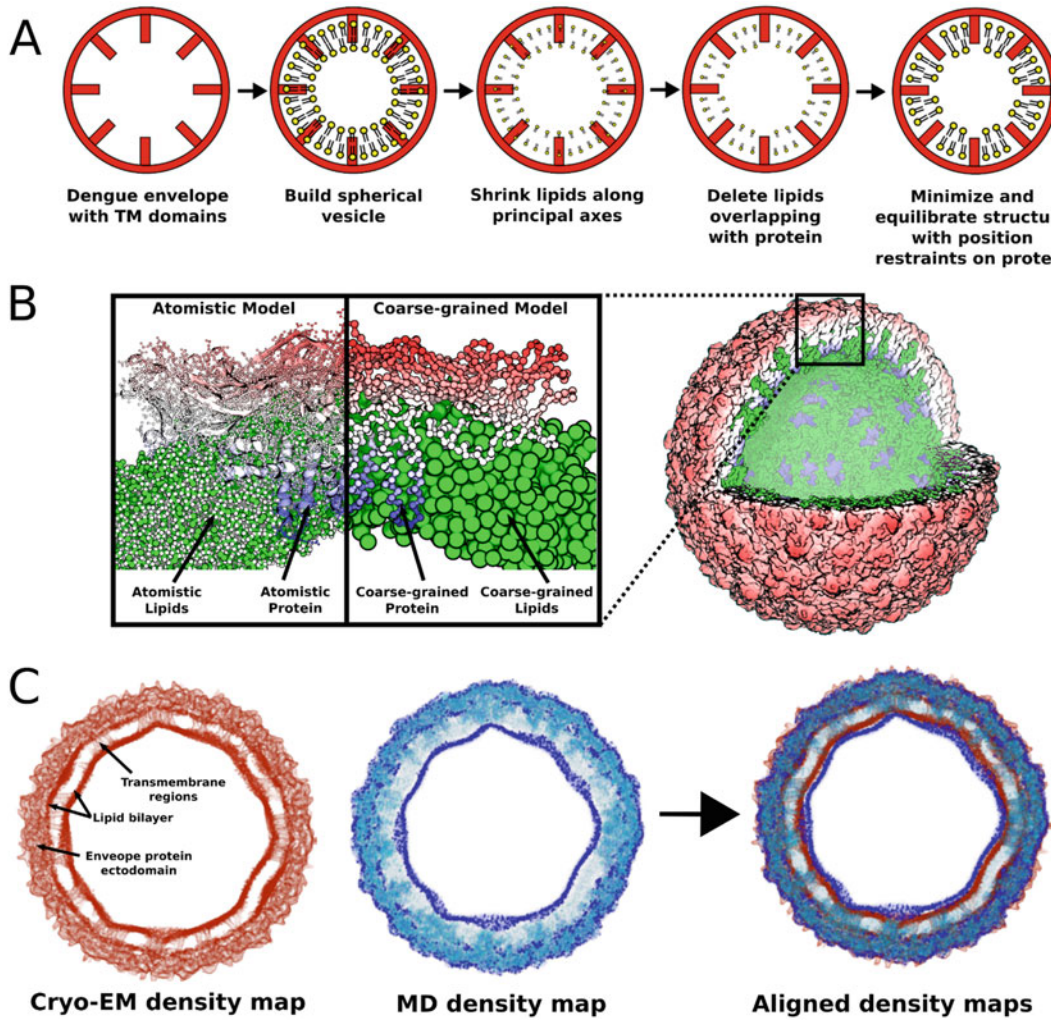

Fig. 3 Protocol for multiscale modeling and refinement of viral envelope against cryo-EM data. (**a**) Schematic for stepwise method to embed viral proteins within a lipid vesicle. (**b**) Multiscale illustration of dengue virus envelope. The simulation construct is shown for the entire viral envelope complex in surface representation on the right. A "zoomed" snapshot on the left shows the dimeric envelope proteins in both atomistic and CG representations (left). Protein is colored according to radial distance from the center of the virus, and lipids are colored green. (**c**) Alignment of density maps determined by cryo-EM (red) and refined during simulation (blue)

4. The CG enveloped viral protein–lipid complex should next be solvated within a simulation box (typically a dodecahedron or truncated octahedron is useful in minimizing excessive solvation of the spherical virus particle) and progressively equilibrated. Initially, sets of ~10,000–100,000 step protein backbone-restrained equilibration simulations in the NVT ensemble should be run with a short integration time step, incrementally increasing this by ~2–4 fs for each successive simulation. Subsequently, once the maximum integration

time step has been reached, longer equilibration runs (e.g., ~10–100 ns) should be run in the NPT ensemble, during which the position restraints on protein coordinates are gradually reduced. This may finally be followed by an unrestrained NPT production run.

5. MDanalysis [85] is an object-oriented Python library which enables analysis of trajectories derived from many MD packages (https://www.mdanalysis.org), and includes tools to generate theoretical density maps from the underlying simulation frames. This may be used to generate a simulation-averaged density map, using a grid spacing in accordance with the resolution of the experimental cryo-EM map (Fig. 3c). The Chimera [86] software package (https://www.cgl.ucsf.edu/chimera/) enables alignment of the experimental and theoretical maps, by minimizing the mean cosine angle between vectors obtained via trilinear interpolation; this also yields a correlation, providing a measure of agreement between simulation and experiment.
6. Finally, refined viral envelope coordinates may be back-mapped and simulated at atomic resolution (*see* Subheading 3.1.1), depending on availability of computational resources, enabling fine-grained analysis of lipid–protein interactions (Fig. 3b).

3.3 Setting Up Biologically Realistic Membrane Systems

Building good starting structures for lipid bilayer simulations can be quite involved due to the interwoven nature and long relaxation timescales of lipids. The complexity increases further with bilayer size and number of lipids types due to slow equilibrium of lipid mixing, lipid flip-flop, and bilayer undulation dynamics. Improved bilayer building tools (*see* **Note 4**), use of coarser, more forgiving force fields, and faster computation have made bilayer construction easier, but interest in larger membranes (up to ~500 × 500 nm), complex bilayer geometries (various capsids and organelles), and more complex lipid mixtures (>60 different lipid types) have complicated matters—*see* for example [87–90]. As simulation complexity begins to approach that of a physiologically relevant membrane composition, one must begin to consider membrane asymmetry and the associated technical difficulties related to its setup and simulation. The primary concern when generating starting configurations for asymmetric membranes is that the average APL varies based on the lipid type and the particular lipid mixture the lipid is in; therefore, for a periotic system, the number of lipids in each membrane leaflet should likely differ. If the aggregated area occupancy in the two leaflets is not the same, then an artificial membrane "frustration" will arise that affects the lateral pressure profile, membrane curvature, and leaflet surface tension.

Adjusting the number of lipids in each leaflet based on their projected APL reduces some of the artifacts but does not fully

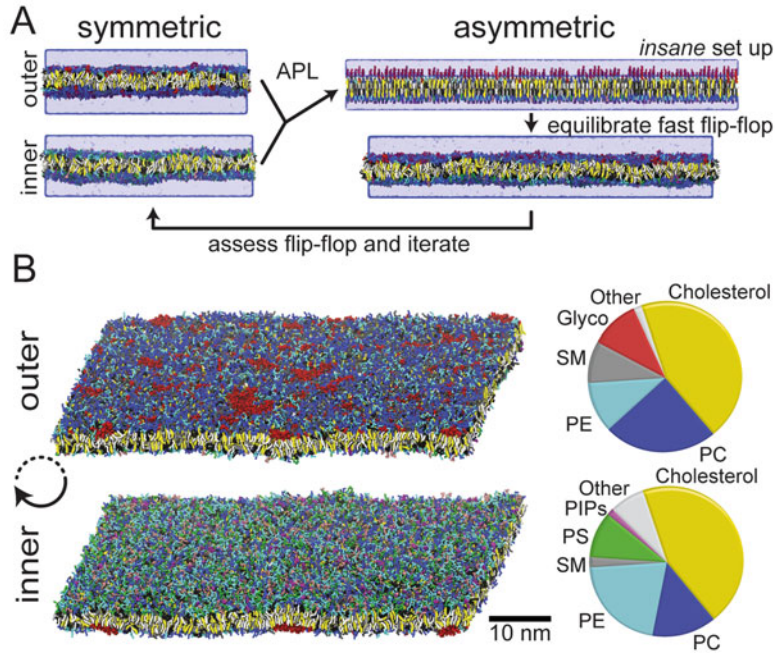

Fig. 4 Setup of an asymmetric neuronal plasma membrane. (**a**) Two simulations of the symmetric outer and inner leaflet mixtures are run to estimate the APL. After setup with the *insane* tool the asymmetrical mixture is simulated and lipid flip-flop is monitored. If flip-flop is "fast" and asymmetric the setup needs to be iterated with an adjusted lipid mixture. (**b**) Snapshot of the asymmetric neuronal plasma membrane simulation described in [89] is shown from the top, showing the outer leaflet, and the bottom, showing the inner leaflet. Pie charts with the overall distribution of lipids head group types in the outer/inner leaflet are also shown

resolve the asymmetry. APLs cannot simply be reverse-engineered from measured APLs, as these (a) are usually identified at a specific temperature that may not be appropriate for the simulation in question, (b) are dependent on the local lipid environment, and/or (c) may not even exist for a specific lipid in isolation. Another approach is to artificially induce lipid flip-flop, allowing lipids to equilibrate between the leaflet; this will relieve any area asymmetry, but will also alter the lipid mixtures of the two leaflets. Here, a third method is described based on separate APL measurements for the two leaflets (Fig. 4), which has been used in, for example, [87, 89]:

1. The outer/inner leaflet lipid mixtures are used to instigate two individual simulations; one with the outer leaflet composition as a symmetrical membrane, and another with the inner leaflet composition as a symmetrical membrane. These simulations are run concurrently, and the global lateral areas (in terms of box size) of these membranes are monitored until they reach an

equilibrium value. Note that this can take several microseconds for complex membranes. These simulations produce the equilibrated average area for both the inner and outer leaflets. The ratio between these values provides a scaling function to calculate the number of lipids required for each leaflet. This data can be utilized by tools such as *insane* [58], which provides an "offset" flag to indicate the asymmetry of lipid numbers between the leaflets.

2. Bilayers undulate and their tendency to undulate is lipid mixture dependent. Therefore, to get a better estimate of the average APL of the two different mixtures, either the bilayer surface should be fitted, or the bilayers should be restrained to be similarly flat. This can be done by applying a weak position restraint potential on the head group particles of a major constituent of *one* of the leaflets. This restraint potential is applied only in the direction of the normal of the bilayer, and only on a single leaflet so as not to affect the bilayer thickness.
3. Having determined the appropriate number of lipids present in each leaflet, one must also then consider the effects of membrane components that have the ability to flip-flop (such as cholesterol) within the timeframe of the simulation. This is especially important for CG simulations that easily reach timescales in which cholesterol can equilibrate between the two leaflets. Deviation of this cholesterol distribution from its original leaflet fractions causes buildup of cholesterol in one of the leaflets and will again lead to artificial membrane "frustration."
4. In order to adjust for cholesterol flip-flop (and other fast flip-flopping lipids), the original, asymmetric lipid system (with corrected leaflet-dependent densities) is simulated until the cholesterol distribution has equilibrated. Again, this may take many microseconds of simulation. If cholesterol displays significant deviation from the starting configuration then these new cholesterol inner/outer leaflet fractions are incorporated back into the original leaflet compositions (and any minor adjustments made accordingly). These updated leaflet compositions with adjusted cholesterol content again need to have their APL/densities calculated via simulation of a pair of symmetric bilayers (repeating **step 1**). The revised asymmetric bilayer (adjusted for cholesterol equilibration and leaflet offset) is simulated again (repeating **steps 2** and **3**) to monitor for any further drift in the between leaflet lipid distributions. This process is iteratively repeated until the cholesterol leaflet distributions do not drift from their starting values during simulation.

It should be noted that this method is not without its faults. By definition, any bilayer that has an asymmetric leaflet lipid

composition is not at equilibrium, therefore, what constitutes as fast lipid flip-flop will depend on the application in question. Furthermore, there is the assumption that the APL of the lipids in a restrained, flat membrane is representative of how the lipid would behave in a freely undulating bilayer.

3.4 Beyond Membranes: Bacterial Envelopes and Cell Walls

The cell wall is an often-forgotten component of the bacterial cell envelope. Sandwiched between the inner and outer membranes in an aqueous compartment called periplasm, the cell wall is made of a network of peptide and sugar molecules commonly known as peptidoglycan [91, 92]. The peptidoglycan mesh is linked covalently to the outer membrane via Braun's lipoproteins [93] and noncovalently to both membranes via integral membrane proteins like OmpA and TolR [94–96]. Earlier simulation studies of the cell wall by Gumbart et al. focused on elucidating its physical properties such as elasticity, pore size, and thickness [97]. However, molecular details of how the cell wall is positioned and interact with other numerous components of the cell envelope remained sparse. We therefore developed atomistic parameters for simulation of peptidoglycan network (Fig. 5) in the presence of Braun's lipoprotein and OmpA [98, 99].

The Braun's lipoprotein is anchored to the outer membrane via a lipidated N-terminus and binds the peptide chain of the cell wall on its C-terminus. The length of the Braun's lipoprotein, therefore, has a direct influence on the distance between the cell wall layer and the outer membrane [100]. Our simulations, however, showed that this is not quite as simple as often suggested. The Braun's lipoprotein was able to tilt and bend significantly with respect to the outer membrane, effectively shifting the cell wall closer to the latter, during simulations. This smaller gap in turn facilitated the initial binding of OmpA periplasmic C-terminal domain to the cell wall.

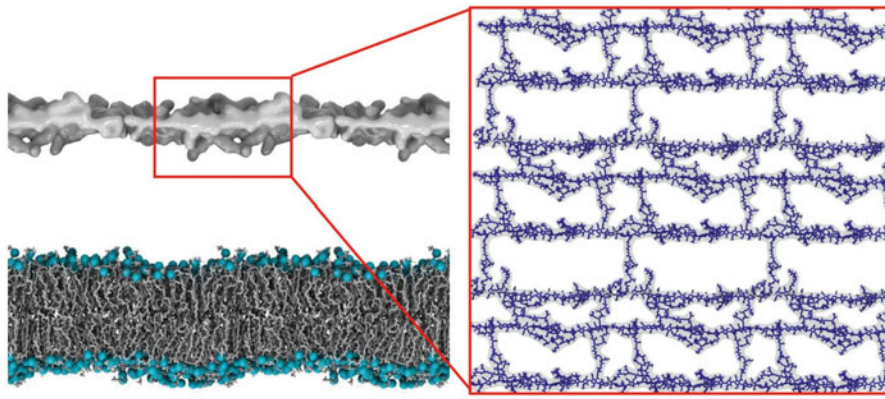

Fig. 5 Model of the gram-negative bacterial *E. coli* inner membrane and peptidoglycan layer. In the inner membrane model below, phospholipid head group phosphorus atoms are shown in cyan. Above, the molecular surface of the peptidoglycan model is shown, with the inset depicting that the strands are 50% cross-linked

In fact, in the absence of Braun's lipoprotein the C-terminal domain of OmpA monomer showed a high propensity to bind to the inner leaflet of the outer membrane instead of the cell wall, further corroborating the instrumental role of Braun's lipoprotein. Dimerization of OmpA, interestingly, eliminated this dependency as the C-terminal domain bound to the cell wall even without Braun's lipoprotein, perhaps due to the stronger electrostatic interaction formed in the OmpA dimer. While the functional relevance of OmpA dimerization is still unknown, in vivo cross-linking studies and mass spectrometry showed that the dimeric interface is largely localized in the C-terminal domain [101, 102], suggesting a likely role in improving cell wall binding. Our simulations also indicated that, once bound, the linker connecting the membrane embedded N-terminal beta barrel domain and the peptidoglycan bound C-terminal domain of OmpA to be highly adaptable, which is potentially important to provide a flexible mechanical support for the underlying cell wall network. In essence, these simulations have uncovered important molecular details of the dynamic interplay between the cell wall and some of the components of the gram-negative bacterial cell envelope.

3.5 Thermodynamics of Lipid–Protein Complex Interactions

Free energy calculations allow the assessment of thermodynamics of a given biological process. A convenient approach for binding and transfer processes is the calculation of a potential of mean force (PMF) [103]. The PMF may be obtained by defining a reaction coordinate (e.g., the distance between two binding partners), and then sampling the forces acting on the respective solutes when restrained at a specific distance. Using the weighted histogram analysis method (WHAM) allows to reconstruct the PMF from these forces [104]. In the case of no interaction, the average force should be zero, whereas attractive interactions are indicated by forces pulling the particles to one another. Sampling the PMF along the entire reaction coordinate allows one to obtain the free energy difference between the bound minimum energy and the unbound energy at larger solute distances, which is equivalent to the difference in free energy between these states. The technique of using discrete restrained sampling points to obtain the PMF is called umbrella sampling [105], which is implemented in most major simulation codes.

Using this approach, we recently demonstrated how an extensive series of free energy calculations in combination with multiscale models allow us to trace the thermodynamics of transfer of a ligand between multiple protein partners and membranes along a "biological relay" [106] (Fig. 6). The relay in question is the Toll-like receptor 4 (TLR4) system [107, 108]; TLR4 is part of the mammalian innate immune pathway which is associated with the plasma membrane and signals the presence of pathogen invasion by detecting lipopolysaccharide (LPS), the dominant

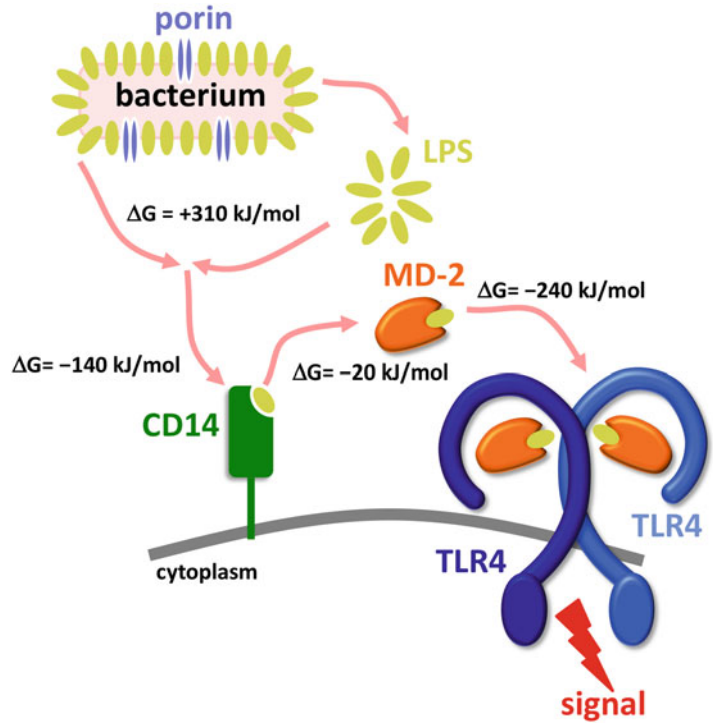

Fig. 6 Thermodynamics of LPS transfer from the bacterial membrane to the terminal TLR4/MD-2 receptor complex. In gram-negative bacteria, the outer leaflet of the outer membrane of is largely constituted of LPS, while the inner leaflet contains a mixture of phospholipid species, and the membrane is also interspersed with a variety of outer membrane proteins such as the modeled OmpF porin. Extracting LPS from this membrane requires considerable energetic effort. LPS is subsequently handed from protein to protein in a cascade, following a path of increasing affinity until ultimately bound in the TLR4/MD-2 complex. The energetic cost estimated from PMF calculations of each stepwise transfer process in this cascade is indicated inset. Dimerization of the terminal receptor activates downstream signaling, thus enabling an innate immune response

component of the outer leaflet of gram-negative bacterial outer membranes [109]. Due to its high hydrophobicity, the energy required to extract LPS from bacterial membranes is likely to be considerable. Thus, a cascade transport system exists, including the essential membrane-associated GPI-anchored CD14 receptor [110–113]. We applied PMF calculations to derive energetic information for the stepwise processes, from the extraction of LPS from biologically realistic bacterial membrane models, LPS binding to CD14 and to the lipid-binding coreceptor of TLR4, MD-2 [114, 115], and finally LPS/MD-2 transfer to the terminal TLR4 receptor. This technique enabled us to demonstrate that LPS follows a thermodynamic funnel, leading to a favorable net change in

free energy upon traversing the complete pathway. Moreover, subsequent CG simulations enabled us to observe the formation of a hydrophobic bridge as part of a transient complex between CD14, MD-2, and TLR4 during LPS transfer [106], thus circumventing the energetic penalty of exposing the highly hydrophobic acyl tails to aqueous solvent.

4 Notes

1. Usually, biological systems should be modeled in the isothermal–isobaric ensemble, with maintenance of a constant number of particles, pressure, and temperature (NPT), as this reflects experimental and in vivo conditions. Whereas globular proteins can be modeled with isotropic pressure scaling, this is generally not appropriate for membrane systems. A semiisotropic pressure coupling that uses distinct barostats for the membrane plane and the normal direction may be employed to allow for independent changes in lipid area. Alternatively, simulations can be conducted in the canonical ensemble (NVT) at preset box dimensions or with a constant area in the membrane plane to constrain the system to a preset area per lipid, or alternatively, with an additional surface tension term.
2. Topologies with example coordinates for specific lipids and/or lipid bilayers can now be obtained from many websites, as exemplified by the following nonexhaustive list: http://wcm.ucalgary.ca/tieleman/downloads (Tieleman group); https://lipidbook.bioch.ox.ac.uk (Sansom group); http://terpconnect.umd.edu/~jbklauda/research/download.html (Klauda group); http://www.softsimu.net/downloads.shtml (Karttunen group); http://www.dsimb.inserm.fr/~luca/downloads/ (Monticelli group); http://www.charmm-gui.org/?doc=archive&lib=lipid_pure (CHARMM-GUI site); http://mackerell.umaryland.edu/charmm_ff.shtml (MacKerell group); https://biophys.uni-saarland.de/downloads.html (Hub group); http://cgmartini.nl/index.php/force-field-parameters/ (Marrink group); and https://atb.uq.edu.au/index.py?tab=existing_molecules (ATB and Repository).
3. Parameters for nonstandard (i.e., nonprotein, nonlipid, non-nucleic acid) molecules, such as for drug-like molecules, need separate topology files describing their connectivity, bond, angle, and dihedral parameters and charges. These can be constructed by combining preexisting topologies of smaller fragments, generated manually, or generated in an automated fashion using one of several web servers or local tools such as "antechamber" [116]. Popular servers include: ParamChem (www.paramchem.org); SwissParam (http://www.swissparam.

ch), the GlycoBioChem PRODRG2 Server (http://davapc1.bioch.dundee.ac.uk/cgi-bin/prodrg); the Automated Topology Builder (ATB) and Repository (https://atb.uq.edu.au); and the CHARMM General Force Field (CGenFF) program (https://cgenff.paramchem.org). Parameters obtained from an automated tool always need to be carefully examined to ensure that connectivity, atom types, and bond orders have been properly applied, and where possible, tested for reproduction of accurate experimentally validated properties during simulations.

4. Common tools for membrane building include: *cellPACK*—A specialized version of the *autoPACK* tool for large-scale packing of biological macromolecules into arbitrary geometries, also supports packing of lipids [117] (autopack.org); CmME—CELLmicrocosmos Membrane Editor is a Java-based tool to generate heterogeneous membranes based on lipid shape [118] (www.cellmicrocosmos.org); CHARMM-GUI—A versatile web-based Graphical User Interface that supports building both atomistic (CHARMM) and CG (Martini) bilayers from a large list of supported lipid types [57, 83, 119] (www.charmm-gui.org); *insane*—"INSert membrANE" is a flexible python-based command-line tool for building Martini bilayers and inserting proteins in bilayers; it supports numerous lipids and uses extendable lipid templates that allow for easy additions of new lipid types [58] (cgmartini.nl/index.php/insane); LipidBuilder—A webserver that can build bilayers of CHARMM lipids created from a library of head group and hydrocarbon tails building blocks [120] (lipidbuilder.epfl.ch); LipidWrapper—A tool to curve bilayers into an arbitrary shape [121] (nbcr.ucsd.edu/lipidwrapper); MemBuilder—A webserver to build heterogeneous atomistic lipid bilayers [122] (www.membuilder.org); MemGen—A force field in dependent webserver for setting up heterogenous bilayers based on uploaded lipid structures [123] (memgen.uni-goettingen.de); PACKMOL—A program to generate initial coordinates for molecular dynamics simulations, including bilayers [124] (m3g.iqm.unicamp.br/packmol); VMD—The Visual Molecular Dynamics, molecular modeling and visualization, program has a membrane plugin that can replicate and trim preequilibrated bilayer patches to the desired size [66] (www.ks.uiuc.edu/Research/vmd).
5. It may be necessary to "repair" parts of an experimentally solved structure, for example, in the case of dynamic and hence unresolved loops. For small modifications such as missing amino acid side chain atoms, the simulation package itself will likely be sufficient. For more complicated problems, homology modeling approaches may be necessary. Homology

modeling enables the derivation of unknown "target" protein structures from homologous "template" proteins of known structure. This is based on the observation that structures tend to be more conserved than sequences. Useful structural homologues typically have sequence identities of at least ~30% to generate models with confidence. Known structural elements (domains, TM helices, etc.) can be incorporated as constraints, which improves reliability. Conserved features, such as inverted topological repeats, may be used to model alternative conformational states of membrane proteins [125]. Homology modeling is usually performed by advanced stand-alone programs such as Modeller [53]. Additionally, online servers may be used for homology modeling (e.g., SWISS-PROT (http://swissmodel.expasy.org)) and/or for ab initio protein structure prediction (e.g., Rosetta (https://www.rosettacommons.org/software/servers), I-TASSER (https://zhanglab.ccmb.med.umich.edu/I-TASSER/)).

6. Solvation of simulation systems is usually performed by superimposing preequilibrated solvent boxes with other system components and deleting overlapping water molecules. Additionally, water may be manually removed from the inner hydrophobic core of lipid bilayer models, or a restrained equilibration could be performed to ensure any water molecules spontaneously exit the membrane. Usually, a physiological salt concentration of 0.1–0.15 M NaCl is established by replacing water molecules with the respective ions, either via random replacement or based on the electrostatic potential of the system. These steps have generally been at least semi-automated in most modern simulation software packages.
7. In contrast to bonded and van der Waals interactions, electrostatic interactions do not become negligible with distance. While the Coulomb potential decays with $1/r$, the number of interaction partners increases with r^3, and hence applying a cutoff for electrostatic interactions is not appropriate. Two approaches are generally used to deal with this effect: at distances larger than a specific cutoff (usually ~1.5 nm), electrostatics are modeled by a response from a uniform background dielectric, a so-called reaction field. Alternatively, in periodic boundary conditions, Ewald summation may be used to calculate exact electrostatic interactions, usually in the form of FFT-based Particle Mesh Ewald (PME) [126]. This can however introduce periodicity artifacts into the system [127]. In the absence of additional information, the choice of treatment or electrostatics should be guided by the current standards of the force field.

Acknowledgments

T.S.C. and H.I.I. acknowledge that this work has been supported in part by the Joint Design of Advanced Computing Solutions for Cancer (JDACS4C) program established by the U.S. Department of Energy (DOE) and the National Cancer Institute (NCI) of the National Institutes of Health. T.S.C. and H.I.I. note that this work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-JRNL-752805. J.R.A. acknowledges the following sources of funding: Rutherford Discovery Fellowship (15-MAU-001); Marsden grant (15-UOA-105); New Zealand Ministry of Business, Innovation and Employment (MBIE) Endeavour Smart Ideas grant (UOCX1706); Maurice Wilkins Centre for Molecular Biodiscovery Flagship Project grant (MWC 3716850). W.A.I. was supported by the following sources of funding: Massey University Doctoral Scholarship; Massey University Doctoral Dissemination grant. N.D. thanks the Nehru trust for Cambridge University and Rajiv Gandhi (UK) foundation for funding. P.J.B. and J.K.M. acknowledge funding from the Ministry of Education in Singapore (MOE AcRF Tier 3 Grant Number MOE2012-T3-1-008).

References

1. Bernlohr DA, Simpson MA, Hertzel AV, Banaszak LJ (1997) Intracellular lipid-binding proteins and their genes. Annu Rev Nutr 17:277–303
2. De Libero G, Mori L (2005) Recognition of lipid antigens by T cells. Nat Rev Immunol 5:485–496
3. Russ AP, Lampel S (2005) The druggable genome: an update. Drug Discov Today 10:1607–1610
4. Overington JP, Al-Lazikani B, Hopkins AL (2006) How many drug targets are there? Nat Rev Drug Discov 5:993–996
5. Arora A, Tamm LK (2001) Biophysical approaches to membrane protein structure determination. Curr Opin Struct Biol 11:540–547
6. Phillips R, Ursell T, Wiggins P, Sens P (2009) Emerging roles for lipids in shaping membrane-protein function. Nature 459:379–385
7. Karplus M, McCammon JA (2002) Molecular dynamics simulations of biomolecules. Nat Struct Biol 9:646–652
8. Karplus M, Kuriyan J (2005) Molecular dynamics and protein function. Proc Natl Acad Sci U S A 102:6679–6685
9. Durrant JD, McCammon JA (2011) Molecular dynamics simulations and drug discovery. BMC Biol 9:71
10. Ash WL, Zlomislic MR, Oloo EO, Tieleman DP (2004) Computer simulations of membrane proteins. Biochim Biophys Acta 1666:158–189
11. Domene C, Bond PJ, Sansom MSP (2003) Membrane protein simulations: ion channels and bacterial outer membrane proteins. In: Protein simulations. Elsevier, Amsterdam, pp 159–193
12. Bond PJ, Sansom MSP (2006) Insertion and assembly of membrane proteins via simulation. J Am Chem Soc 128:2697–2704
13. Sansom MSP, Bond P, Beckstein O et al (2008) Water in ion channels and pores-simulation studies. In: Ion channels: from atomic resolution physiology to functional genomics. John Wiley & Sons, Ltd, New York, NY, pp 66–83

14. Huber RG, Marzinek JK, Holdbrook DA, Bond PJ (2017) Multiscale molecular dynamics simulation approaches to the structure and dynamics of viruses. Prog Biophys Mol Biol 128:121–132
15. Angelescu DG, Linse P (2008) Viruses as supramolecular self-assemblies: modelling of capsid formation and genome packaging. Soft Matter 4:1981
16. Reddy T, Sansom MSP (2016) Computational virology: from the inside out. Biochim Biophys Acta 1858:1610–1618
17. Brooks BR, Brooks CL, Mackerell AD et al (2009) CHARMM: the biomolecular simulation program. J Comput Chem 30:1545–1614
18. Huang J, Mackerell AD (2013) CHARMM36 all-atom additive protein force field: validation based on comparison to NMR data. J Comput Chem 34:2135–2145
19. Case DA, Cheatham TE, Darden T et al (2005) The Amber biomolecular simulation programs. J Comput Chem 26:1668–1688
20. Salomon-Ferrer R, Götz AW, Poole D et al (2013) Routine microsecond molecular dynamics simulations with AMBER on GPUs. 2. Explicit solvent particle mesh ewald. J Chem Theory Comput 9:3878–3888
21. Schmid N, Eichenberger AP, Choutko A et al (2011) Definition and testing of the GROMOS force-field versions 54A7 and 54B7. Eur Biophys J 40:843
22. Jämbeck JPM, Lyubartsev AP (2012) Derivation and systematic validation of a refined all-atom force field for phosphatidylcholine lipids. J Phys Chem B 116:3164–3179
23. Jämbeck JPM, Lyubartsev AP (2012) An extension and further validation of an all-atomistic force field for biological membranes. J Chem Theory Comput 8:2938–2948
24. Jämbeck JPM, Lyubartsev AP (2012) Another piece of the membrane puzzle: extending slipids further. J Chem Theory Comput 9:774–784
25. Ermilova I, Lyubartsev AP (2016) Extension of the slipids force field to polyunsaturated lipids. J Phys Chem B 120:12826–12842
26. Dickson CJ, Madej BD, Skjevik ÅA et al (2014) Lipid14: the amber lipid force field. J Chem Theory Comput 10:865–879
27. Schuler LD, Daura X, van Gunsteren WF (2001) An improved GROMOS96 force field for aliphatic hydrocarbons in the condensed phase. J Comput Chem 22:1205–1218
28. Berger O, Edholm O, Jähnig F (1997) Molecular dynamics simulations of a fluid bilayer of dipalmitoylphosphatidylcholine at full hydration, constant pressure, and constant temperature. Biophys J 72:2002–2013
29. Marrink SJ, Risselada HJ, Yefimov S et al (2007) The MARTINI force field: coarse grained model for biomolecular simulations. J Phys Chem B 111:7812–7824
30. Monticelli L, Kandasamy SK, Periole X et al (2008) The MARTINI coarse-grained force field: extension to proteins. J Chem Theory Comput 4:819–834
31. De Jong DH, Singh G, Bennett WFD et al (2013) Improved parameters for the martini coarse-grained protein force field. J Chem Theory Comput 9:687–697
32. Uusitalo JJ, Ingólfsson HI, Akhshi P et al (2015) Martini coarse-grained force field: extension to DNA. J Chem Theory Comput 11:3932–3945
33. Nielsen SO, Lopez CF, Srinivas G, Klein ML (2004) Coarse grain models and the computer simulation of soft materials. J Phys Condens Matter 16:R481–R512
34. Saunders MG, Voth GA (2013) Coarse-graining methods for computational biology. Annu Rev Biophys 42:73–93
35. Zhang Z, Pfaendtner J, Grafmüller A, Voth GA (2009) Defining coarse-grained representations of large biomolecules and biomolecular complexes from elastic network models. Biophys J 97:2327–2337
36. Noid WG, Chu JW, Ayton GS et al (2008) The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models. J Chem Phys 128:244114
37. Noid WG, Liu P, Wang Y et al (2008) The multiscale coarse-graining method. II. Numerical implementation for coarse-grained molecular models. J Chem Phys 128:244115
38. Izvekov S, Voth GA (2005) A multiscale coarse-graining method for biomolecular systems. J Phys Chem B 109:2469–2473
39. Götz AW, Williamson MJ, Xu D et al (2012) Routine microsecond molecular dynamics simulations with AMBER on GPUs. 1. generalized born. J Chem Theory Comput 8:1542–1555
40. Shaw DE, Bowers KJ, Chow E et al (2009) Millisecond-scale molecular dynamics simulations on Anton. In: Proc Conf High Perform Comput Netw Storage Anal SC 09 1. ACM, New York, NY
41. Shaw DE, Grossman JP, Bank JA et al (2014) Anton 2: raising the bar for performance and

programmability in a special-purpose molecular dynamics supercomputer. In: Proceedings of the International Conference for High Performance Computing, Networking. Storage and Analysis. IEEE Press, Piscataway, NJ, USA, pp 41–53

42. Marzinek JK, Holdbrook DA, Huber RG et al (2016) Pushing the envelope: dengue viral membrane coaxed into shape by molecular simulations. Structure 24:1410–1420
43. Petrlova J, Hansen FC, van der Plas MJA et al (2017) Aggregation of thrombin-derived C-terminal fragments as a previously undisclosed host defense mechanism. Proc Natl Acad Sci U S A 114:E4213–E4222
44. Berman HM, Westbrook J, Feng Z et al (2000) The protein data bank. Nucleic Acids Res 28:235–242
45. Kleywegt GJ, Jones TA (1998) Databases in protein crystallography. Acta Crystallogr Sect D Biol Crystallogr 54:1119–1131
46. Wuthrich K (1986) NMR of proteins and nucleic acids. Wiley, New York, NY
47. Cheng Y (2015) Single-particle Cryo-EM at crystallographic resolution. Cell 161:450–457
48. Wlodawer A, Minor W, Dauter Z, Jaskolski M (2008) Protein crystallography for non-crystallographers, or how to get the best (but not more) from published macromolecular structures. FEBS J 275:1–21
49. Brown EN, Ramaswamy S (2007) Quality of protein crystal structures. Acta Crystallogr Sect D Biol Crystallogr 63:941–950
50. Hryc CF, Chen DH, Chiu W (2011) Near-atomic resolution cryo-EM for molecular virology. Curr Opin Virol 1:110–117
51. Carpenter EP, Beis K, Cameron AD, Iwata S (2008) Overcoming the challenges of membrane protein crystallography. Curr Opin Struct Biol 18:581–586
52. Fiser A, Šali A (2003) Modeller: generation and refinement of homology-based protein structure models. In: Methods in enzymology. Elsevier, Amsterdam, pp 461–491
53. Sali A (2008) MODELLER a program for protein structure modeling release 9v4, r6262. Structure:779–815
54. Shen M, Sali A (2006) Statistical potential for assessment and prediction of protein structures. Protein Sci 15:2507–2524
55. Wolf MG, Hoefling M, Aponte-Santamaría C et al (2010) g_membed: efficient insertion of a membrane protein into an equilibrated lipid bilayer with minimal perturbation. J Comput Chem 31:2169–2174
56. Jo S, Lim JB, Klauda JB, Im W (2009) CHARMM-GUI membrane builder for mixed bilayers and its application to yeast membranes. Biophys J 97:50–58
57. Qi Y, Ingólfsson HI, Cheng X et al (2015) CHARMM-GUI martini maker for coarse-grained simulations with the martini force field. J Chem Theory Comput 11:4486–4494
58. Wassenaar TA, Ingólfsson HI, Böckmann RA et al (2015) Computational lipidomics with insane: a versatile tool for generating custom membranes for molecular simulations. J Chem Theory Comput 11:2144–2155
59. Chang R, Ayton GS, Voth GA (2005) Multiscale coupling of mesoscopic- and atomistic-level lipid bilayer simulations. J Chem Phys 122:244716
60. Abraham MJ, Murtola T, Schulz R et al (2015) GROMACS: high performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX 1–2:19–25
61. Berendsen HJC, van der Spoel D, van Drunen R (1995) GROMACS: a message-passing parallel molecular dynamics implementation. Comput Phys Commun 91:43–56
62. Van Der Spoel D, Lindahl E, Hess B et al (2005) GROMACS: fast, flexible, and free. J Comput Chem 26:1701–1718
63. Phillips JC, Braun R, Wang W et al (2005) Scalable molecular dynamics with NAMD. J Comput Chem 26:1781–1802
64. Sanbonmatsu KY, Tung CS (2007) High performance computing in biology: multimillion atom simulations of nanoscale systems. J Struct Biol 157:470–480
65. Götz AW, Williamson MJ, Xu D et al (2012) Routine microsecond molecular dynamics simulations with amber - Part I: Generalized born. J Chem Theory Comput 8:1542–1555
66. Humphrey W, Dalke A, Schulten K (1996) VMD: visual molecular dynamics. J Mol Graph 14(27-28):33–38
67. Ulmschneider MB, Sansom MSP (2001) Amino acid distributions in integral membrane protein structures. Biochim Biophys Acta 1512:1–14
68. Lomize MA, Lomize AL, Pogozheva ID, Mosberg HI (2006) OPM: orientations of proteins in membranes database. Bioinformatics 22:623–625
69. Yesylevskyy SO (2007) ProtSqueeze: simple and effective automated tool for setting up membrane protein simulations. J Chem Inf Model 47:1986–1994
70. Faraldo-Gómez J, Smith G, Sansom M (2002) Setting up and optimization of

membrane protein simulations. Eur Biophys J 31:217–227

71. Schmidt TH, Kandt C (2012) LAMBADA and InflateGRO2: efficient membrane alignment and insertion of membrane proteins for molecular dynamics simulations. J Chem Inf Model 52:2657–2669
72. Shirts MR, Mobley DL, Chodera JD (2007) Chapter 4 Alchemical free energy calculations: ready for prime time? Annu Rep Comput Chem 3:41–59
73. Xu C, Gagnon E, Call ME et al (2008) Regulation of T cell receptor activation by dynamic membrane binding of the CD3ε cytoplasmic tyrosine-based motif. Cell 135:702–713
74. Shirts MR, Pande VS (2005) Solvation free energies of amino acid side chain analogs for common molecular mechanics water models. J Chem Phys 122:134508
75. Jefferys E, Sands ZA, Shi J et al (2015) Alchembed: a computational method for incorporating multiple proteins into complex lipid geometries. J Chem Theory Comput 11:2743–2754
76. Bond PJ, Cuthbertson JM, Deol SS, Sansom MSP (2004) MD simulations of spontaneous membrane protein/detergent micelle formation. J Am Chem Soc 126:15948–15949
77. Scott KA, Bond PJ, Ivetac A et al (2008) Coarse-grained MD simulations of membrane protein-bilayer self-assembly. Structure 16:621–630
78. Saravanan R, Holdbrook DA, Petrlova J et al (2018) Structural basis for endotoxin neutralization and anti-inflammatory activity of thrombin-derived C-terminal peptides. Nat Commun 9:2762
79. Stansfeld PJ, Hopkinson R, Ashcroft FM, Sansom MSP (2009) PIP2-binding site in kir channels: definition by multiscale biomolecular simulations. Biochemistry 48:10926–10933
80. Wassenaar TA, Pluhackova K, Bockmann RA et al (2014) Going backward: a flexible geometric approach to reverse transformation from coarse grained to atomistic models. J Chem Theory Comput 10:676–690
81. Kargas V, Marzinek JK, Holdbrook DA et al (2017) A polar SxxS motif drives assembly of the transmembrane domains of Toll-like receptor 4. Biochim Biophys Acta 1859:2086–2095
82. Irvine WA, Flanagan JU, Allison JR (2018) Computational prediction of amino acids governing protein-membrane interaction for the PIP3 cell signalling system. Structure 27:371
83. Wu EL, Cheng X, Jo S et al (2014) CHARMM-GUI membrane builder toward realistic biological membrane simulations. J Comput Chem 35:1997–2004
84. Cheng X, Jo S, Lee HS et al (2013) CHARMM-GUI micelle builder for pure/mixed micelle and protein/micelle complex systems. J Chem Inf Model 53:2171–2180
85. Michaud-Agrawal N, Denning EJ, Woolf TB, Beckstein O (2011) MDAnalysis: a toolkit for the analysis of molecular dynamics simulations. J Comput Chem 32:2319–2327
86. Huang CC, Couch GS, Pettersen EF, Ferrin TE (1996) Chimera: an extensible molecular modeling application constructed using standard components. In: Pac. Symp. Biocomput. World Scientific, Hackensack, NJ, p 724
87. Ingólfsson HI, Melo MN, van Eerden FJ et al (2014) Lipid organization of the plasma membrane. J Am Chem Soc 136:14554–14559
88. Reddy T, Sansom MSP (2016) The role of the membrane in the structure and biophysical robustness of the dengue virion envelope. Structure 24(3):375–382
89. Ingólfsson HI, Carpenter TS, Bhatia H et al (2017) Computational lipidomics of the neuronal plasma membrane. Biophys J 113:2271–2280
90. Koldsø H, Shorthouse D, Hélie J, Sansom MSP (2014) Lipid clustering correlates with membrane curvature as revealed by molecular simulations of complex lipid bilayers. PLoS Comput Biol 10:e1003911
91. Vollmer W, Blanot D, De Pedro MA (2008) Peptidoglycan structure and architecture. FEMS Microbiol Rev 32:149–167
92. Vollmer W, Seligman SJ (2010) Architecture of peptidoglycan: more data and more models. Trends Microbiol 18:59–66
93. Braun V (1975) Covalent lipoprotein from the outer membrane of Escherichia coli. Biochim Biophys Acta 415:335–377
94. Koebnik R (1995) Proposal for a peptidoglycan-associating alpha-helical motif in the C-terminal regions of some bacterial cell-surface proteins. Mol Microbiol 16:1269–1270
95. Parsons LM, Lin F, Orban J (2006) Peptidoglycan recognition by Pal, an outer membrane lipoprotein. Biochemistry 45:2122–2128
96. Roujeinikova A (2008) Crystal structure of the cell wall anchor domain of MotB, a stator component of the bacterial flagellar motor: implications for peptidoglycan recognition. Proc Natl Acad Sci 105:10348–10353
97. Gumbart JC, Beeby M, Jensen GJ, Roux B (2014) Escherichia coli peptidoglycan

structure and mechanics as predicted by atomic-scale simulations. PLoS Comput Biol 10:e1003475

98. Samsudin F, Ortiz-Suarez ML, Piggot TJ et al (2016) OmpA: a flexible clamp for bacterial cell wall attachment. Structure 24:2227–2235
99. Samsudin F, Boags A, Piggot TJ, Khalid S (2017) Braun's lipoprotein facilitates OmpA interaction with the escherichia coli cell wall. Biophys J 113:1496–1504
100. Cohen EJ, Ferreira JL, Ladinsky MS et al (2017) Nanoscale-length control of the flagellar driveshaft requires hitting the tethered outer membrane. Science 356:197–200
101. Zheng C, Yang L, Hoopmann MR et al (2011) Cross-linking measurements of in vivo protein complex topologies. Mol Cell Proteomics 10:M110-006841
102. Marcoux J, Politis A, Rinehart D et al (2014) Mass spectrometry defines the C-terminal dimerization domain and enables modeling of the structure of full-length OmpA. Structure 22:781–790
103. Doudou S, Burton NA, Henchman RH (2009) Standard free energy of binding from a one-dimensional potential of mean force. J Chem Theory Comput 5:909–918
104. Kumar S, Rosenberg JM, Bouzida D et al (1992) THE weighted histogram analysis method for free-energy calculations on biomolecules. I. The method. J Comput Chem 13:1011–1021
105. Kästner J (2011) Umbrella sampling. Wiley Interdiscip Rev Comput Mol Sci 1:932–942
106. Huber RG, Berglund NA, Kargas V et al (2018) A thermodynamic funnel drives bacterial lipopolysaccharide transfer in the TLR4 pathway. Structure 26:1151–1161
107. Lu Y-C, Yeh W-C, Ohashi PS (2008) LPS/TLR4 signal transduction pathway. Cytokine 42:145–151
108. Kim HM, Park BS, Kim J-I et al (2007) Crystal structure of the TLR4-MD-2 complex with bound endotoxin antagonist eritoran. Cell 130:906–917
109. Brandenburg K, Seydel U (1984) Physical aspects of structure and function of membranes made from lipopolysaccharides and free lipid A. Biochim Biophys Acta 775:225–238
110. Ryu J-K, Kim SJ, Rah S-H et al (2017) Reconstruction of LPS transfer cascade reveals structural determinants within LBP, CD14, and TLR4-MD2 for efficient LPS recognition and transfer. Immunity 46:38. https://doi.org/10.1016/j.immuni.2016.11.007
111. Juan TS-C, Kelley MJ, Johnson DA et al (1995) Soluble CD14 truncated at amino acid 152 binds lipopolysaccharide (LPS) and enables cellular response to LPS. J Biol Chem 270:1382–1387
112. Kelley SL, Lukk T, Nair SK, Tapping RI (2013) The crystal structure of human soluble CD14 reveals a bent solenoid with a hydrophobic amino-terminal pocket. J Immunol 190:1304–1311
113. Kim J-I, Lee CJ, Jin MS et al (2005) Crystal structure of CD14 and its implications for lipopolysaccharide signaling. J Biol Chem 280:11347–11351
114. Zimmer SM, Liu J, Clayton JL et al (2008) Paclitaxel binding to human and murine MD-2. J Biol Chem 283:27916–27926
115. Ohto U, Fukase K, Miyake K, Shimizu T (2012) Structural basis of species-specific endotoxin sensing by innate immune receptor TLR4/MD-2. Proc Natl Acad Sci U S A 109:7421–7426
116. Wang J, Wang W, Kollman PA, Case DA (2001) Antechamber, an accessory software package for molecular mechanical calculations. J Am Chem Soc 222:U403
117. Johnson GT, Autin L, Al-Alusi M et al (2014) cellPACK: a virtual mesoscope to model and visualize structural systems biology. Nat Methods 12:85–91
118. Sommer B, Dingersen T, Gamroth C et al (2011) CELLmicrocosmos 2.2 MembraneEditor: a modular interactive shape-based software approach to solve heterogeneous membrane packing problems. J Chem Inf Model 51:1165
119. Jo S, Kim T, Iyer VG, Im W (2008) CHARMM-GUI: a web-based graphical user interface for CHARMM. J Comput Chem 29:1859–1865
120. Bovigny C, Tamò G, Lemmin T et al (2015) LipidBuilder: a framework to build realistic models for biological membranes. J Chem Inf Model 55:2491–2499
121. Durrant JD, Amaro RE (2014) LipidWrapper: an algorithm for generating large-scale membrane models of arbitrary geometry. PLoS Comput Biol 10:e1003720
122. Ghahremanpour MM, Arab SS, Aghazadeh SB et al (2013) MemBuilder: a web-based graphical interface to build heterogeneously mixed membrane bilayers for the GROMACS biomolecular simulation program. Bioinformatics 30:439–441

123. Knight CJ, Hub JS (2015) MemGen: a general web server for the setup of lipid membrane simulation systems: Fig. 1. Bioinformatics 31:2897–2899
124. Martinez L, Andrade R, Birgin EG, Martínez JM (2009) PACKMOL: a package for building initial configurations for molecular dynamics simulations. J Comput Chem 30:2157–2164
125. Vergara-Jaque A, Fenollar-Ferrer C, Kaufmann D, Forrest LR (2015) Repeat-swap homology modeling of secondary active transporters: updated protocol and prediction of elevator-type mechanisms. Front Pharmacol 6:183
126. Darden T, York D, Pedersen L (1993) Particle mesh Ewald: an N·log(N) method for Ewald sums in large systems. J Chem Phys 98:10089–10092
127. Patra M, Karttunen M, Hyvönen MT et al (2003) Molecular dynamics simulations of lipid bilayers: major artifacts due to truncating electrostatic interactions. Biophys J 84:3636–3645