JCTC Journal of Chemical Theory and Computation

Article

pubs.acs.org/JCTC

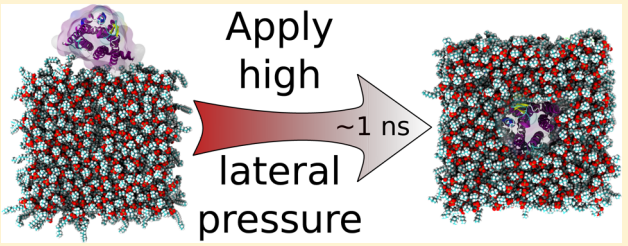
Universal Method for Embedding Proteins into Complex Lipid Bilayers for Molecular Dynamics Simulations

Matti Javanainen*

Tampere University of Technology, Tampere, Finland

Supporting Information

ABSTRACT: The growing interest toward membrane protein simulations calls for a universal and efficient protocol for embedding multiple proteins simultaneously into asymmetric many-component lipid membranes. To this end, here, we present a new and simple technique based on pushing proteins into a lipid membrane from its side by applying a high lateral pressure on the system. This approach is compatible with most (if not all) simulation software packages, and it is independent of external program codes. Most importantly, our protocol does not alter the lipid composition or the transmembrane lipid distribution of the host bilayer membrane. It can be employed with both atomistic and coarse-grained models, and it allows multiple proteins to be embedded into a membrane at the same time. It is shown that the protein structure is unaffected by the pressure applied to the system during the procedure, and the simulation resources required for protein insertion are shown to be modest, regarding both atomistic and coarse-grained simulation models.



1. INTRODUCTION

During the last two decades, it has become more and more clear that a large fraction of the functions of cells is based on the interplay of membrane proteins and lipids. In many cases, lipids influence the stability of membrane proteins or even modulate their conformation and function.[1−4] This, in part, has become evident from the three-dimensional (3D) crystal structures of membrane proteins that have revealed lipids to be an integral component of protein structure,[5,6] suggesting that there are binding sites for specific lipids. There are also proteins whose function is known to be modulated by specific lipids,[7] even if lipid binding sites seem to be absent. Furthermore, membrane-mediated interactions can also have a role to play in protein activation, as has been highlighted for proteins that are sensitive to membrane elasticity, lateral stress, and membrane curvature.[8]

The current view for membrane protein function is captured by the lipid raft paradigm,[9] which states that there are nanoscale functional units composed of membrane proteins and lipids, and when these match, the function may emerge. The condition for matching the protein(s) with specific lipids relates to both the lateral distribution of lipids around the protein(s), as well as the asymmetric transmembrane lipid distributions,[10−12] since they both can contribute to protein activation. Since they both deal with nanoscale phenomena, the understanding of the structure of lipid raftlike units has remained quite limited.

Molecular simulations are largely the method of choice for unraveling nanoscale phenomena, since they can provide one with almost unlimited accuracy to understand the properties of lipid membranes and lipid−protein complexes. In this context, the progress during the last 2−3 decades has been substantial, as the focus of molecular simulations in membrane systems has shifted from pure lipid systems[13−17] to complex lipid−protein units.[18−25] Meanwhile, studies of multicomponent lipid membranes and their effects on membrane proteins[26] have become increasingly popular, and particular attention has been paid to proteins partitioning into rafts.[27,28] In addition, consideration of asymmetric transmembrane lipid distributions has also become more accurate in simulation studies.[29−33] Important to the progress in the field is also the increase in available computational power and the increasing pace at which new structures for membrane proteins are being determined.[34] Overall, one can summarize that the chances to describe biological membranes in a realistic manner have improved quite dramatically.

However, while computational simulations of membrane proteins have become one of the mainstream areas of membrane biophysics, the methodology for constructing initial structures for these simulations is not well-established. Numerous membrane protein insertion methods have been suggested, but they all suffer from shortcomings (see below for details). Most importantly, many methods involve the removal of lipids from a lipid membrane, therefore altering its lipid composition and lipids' transmembrane distributions upon protein insertion. This can be a serious problem, since, for practical reasons, the size of the studied system should be optimized to the smallest reasonable level in order to access longest possible simulation times. This implies that the removal of even a few lipids might result in significant deviations from the desired lipid distributions, which might alter the desired

Received: January 19, 2014
Published: May 1, 2014

transmembrane symmetry/asymmetry. This, in turn, as explained above, may change the behavior of proteins, since the specific interplay of proteins with their neighboring lipids may be compromised.

In this context, we briefly discuss the previously suggested methods for membrane protein insertion into lipid membranes. For a more thorough report of the available approaches, see the review by Kandt et al.[35]

In the two very earliest attempts to study membrane proteins embedded in a lipid bilayer, the membrane was built around the protein.[36,37] This approach is currently implemented, to the best of our knowledge, only in the CHARMM-GUI,[38,39] which provides an intuitive and highly automatized tool for building heterogeneous membranes around proteins. However, it is restricted to be employed with the CHARMM force field and with the lipid types available in the CHARMM library.

Another method involves the insertion of a protein in a void generated by the removal of overlapping lipid molecules. This is the most straightforward approach available. It can be employed by either deleting all conflicting lipids upon protein insertion, or by allowing some overlap, which is then eliminated during energy minimization.[40] This procedure alters the lipid composition because of the removal of lipids that collide with the protein. In addition, because of the large number of lipid conformations available, the void space resulting from lipid removal might be quite substantial, therefore requiring long re-equilibration. An example of this method can be achieved in GROMACS[41] using the tool Genbox, which will "solvate" any membrane protein in a provided membrane with the removal of overlapping lipid molecules.

This conceptually simple method has evolved to include various more advanced modifications, too. In general they involve some sort of external forces that facilitate the formation of the void into which the protein is embedded. The simplest approach includes an additional simulation during which a cylindrical repulsive force is applied around the void created by the removal of lipids, thereby enlarging it to a suitable size.[42] In a yet more advanced method, the repulsive forces are not exerted in a cylindrical shape, yet they follow the shape of the protein.[43] First, colliding lipids are removed based on their headgroup locations. The repulsive forces on other surrounding lipid atoms are applied based on the shape of the solvent accessible surface (SAS) of the protein. This approach is available as a GROMACS-compatible tool Make_hole (also called Mdrun_hole). In addition to the issues related to the removal of lipids, this approach relies heavily on the MD engine resulting in code portability problems, and it requires the SAS calculations to be performed usually by third-party software. So far, it has only been implemented in some outdated GROMACS versions. The CHARMM-GUI[38] offers pre-equilibrated membrane patches with cylindrical holes of multiple sizes in them. However, the available bilayers are currently limited to two sizes and only homogeneous bilayers with a few lipid types are available.

A much more refined method is provided by GRIFFIN (GRId-based Force Field INput).[44] In this approach, a void is again generated by the removal of lipids. This is followed by an optimization of the protein/lipid interface based on forces provided by the GRIFFIN software aiming to mimic the presence of the protein. The strength of this method is clearly in being compatible with hollow (i.e., doughnut-like) proteins that can encapsulate lipids in the void at their center. However, GRIFFIN is an external software with multiple dependencies and it needs to be compiled prior to execution. In addition, it is currently limited to being used together with GROMACS or NAMD software packages.

The GROMACS simulation package currently includes a G_membed tool[45] for protein insertion into lipid membranes. In this approach, a small void is generated by lipid removal, followed by inserting a squeezed version of the protein into the membrane. This is followed by inflating the protein back to its real size during which the surrounding lipid molecules are pushed back from the volume occupied eventually by the protein. The largest shortcoming of this approach, as in all methods listed above, is the necessary lipid removal step, which changes the lipid composition of the host bilayer. In addition, the method is currently available exclusively in the GROMACS package, preventing its universal adaptation.

Another clever approach available, InflateGRO,[35] begins with the expansion of the host bilayer by scaling its coordinates. This is followed by insertion of the protein by removal of the colliding lipids and the subsequent stepwise compression and energy minimization steps. This method was recently updated to InflateGRO2, a fully automatized and improved version of the tool.[46] Even though it is commended because of its minimal requirement for manual intervention, the approach also involves lipid removal and it requires external, file-format-dependent tools, making it currently compatible only with the GROMACS package. In the InflateGRO tool, the compression step is supposed to be terminated as the area per lipid matches that of the protein-free bilayer, a quantity quite tedious to estimate for a protein−lipid system. This is overcome in InflateGRO2. However, whereas the previous version allows for embedding of doughnut-shaped proteins encapsulating lipids, this is no longer supported in the updated one. It was also noted that employing the tool results in a loss of membrane hydration.[45]

Allowing the lipid bilayer to spontaneously form around the protein has also been suggested.[35] This is well within the reach of current computational resources, yet again the requirement of desired symmetric or asymmetric lipid composition of the leaflets renders it obsolete.

Here, we present a new approach to membrane protein embedding that is based on pushing the protein into the lipid membrane from one side, followed by relaxation of the membrane area back to equilibrium. Our method does not involve the removal of lipid molecules from the host membrane and it can be employed with both atomistic and coarse-grained approaches, as described below. Most importantly, it can be used with any simulation packages without further modifications or need of external tools, provided that they support semi-isotropic pressure coupling and absolute position restraints, which are standards in all commonly employed simulation packages. In addition, multiple proteins can be embedded at once, which is important since studies involving protein oligomerization and protein crowding effects are becoming increasingly common.[23]

2. EMBEDDING PROTOCOL

The new suggested protein embedding protocol consists of the following steps: (1) placement of the lipid membrane and the protein in the same simulation box; (2) inclusion of required restraints; (3) simulation with applied lateral pressure ("push simulation"); and (4) recovery of the system from the applied lateral pressure ("relaxation simulation").

2.1. Placement of the Lipid Membrane and the Protein in the Same Simulation Box. The protein is positioned next to the membrane in its plane and at such a height that its hydrophobic residues match the hydrophobic core of the membrane (see Figure 1). This manual preparation

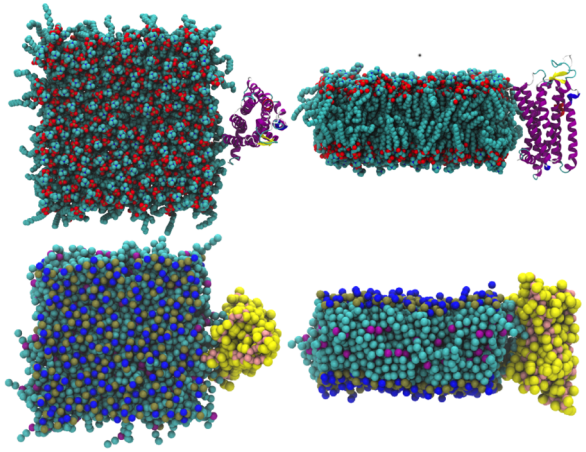


Figure 1. Positioning of a protein next to a lipid bilayer prior to inclusion of lateral pressure in step 1 of the embedding protocol. The positioning is shown for an atomistic system (top row) as well as a coarse-grained system (bottom row). For details of the systems, see discussion in the text. Protein in the coarse-grained system is shown as yellow and pink spheres, whereas in the atomistic system it is depicted with the colors corresponding to its secondary structure. Nitrogen and phosphorus atoms (choline and phosphate groups in the coarse-grained representation) are shown in blue and brown, respectively. Oxygen atoms in the atomistic system are shown in red. All carbon atoms in atomistic system as well as hydrocarbon tails in coarse-grained system are shown in cyan with the coarse-grained beads including the double bond colored purple. Hydrogen atoms and water are not shown for clarity. Snapshots were rendered with the aid of the Tachyon ray tracer[51] supplied with the VMD software.[47]

task can be easily performed by either using graphical tools such as VMD[47] or PyMol,[48] or by employing command line tools supplied with the MD package, e.g., EDITCONF in GROMACS[41] or TRANSLATE in AmberTools.[49] At the same time, the box is enlarged in such a way that the protein fits in it without collisions with the membrane. If the box is made a square in the membrane plane, it will maintain its shape through the process as a semi-isotropic pressure bath is employed. However, the embedding process is accelerated if the length of the simulation box is only increased in the dimension necessary for including the protein. Note also for more-complex purposes that multiple proteins can be embedded at once and their intermolecular distance can be adjusted by altering their initial positioning in the vacuum space next to the bilayer patch. In the case of identical proteins, this is also achieved by single insertion followed by replicating the obtained system. The bilayer employed here can either be hydrated or not. In addition, a pre-equilibrated bilayer as well as its rough approximation from any membrane builder can be employed.

2.2. Inclusion of Required Restraints. During the pushing of the protein into the bilayer, its secondary and ternary structures should remain intact. Therefore, all the atoms of the protein are kept in place by position restraints, which, in many cases, are automatically generated as the protein topology is created. Based on our tests (see below), a force constant of 10 000 kJ mol^{-1} nm^{-2} is adequate for this purpose when using a harmonic potential for the restraints. Other restraining methods are equally acceptable, provided that no structural changes in the protein are observed. In addition, the lipid molecules should not be allowed to escape from the membrane plane to the vacuum created in step 1. Accordingly, one atom of their headgroup (such as a nitrogen atom in the case of a phospholipid with a choline headgroup) is restrained in the direction normal to the membrane plane. In rare occasions when the lipid tails flip toward the vacuum space, the terminal groups of these tails can also be restrained in the direction normal to the membrane plane. This also prevents the tails, in a pre-equilibrated bilayer, from straightening up and beginning to form a gel phase upon the application of pressure. If solvent is present in the system, restraining it (i.e., oxygen atom for water) in the normal direction will accelerate equilibration, since the solvent molecules are not required to escape the hydrophobic membrane core into which they might get trapped without applying these restraints.

2.3. Simulation with Applied Lateral Pressure ("Push Simulation"). The energy of the constructed system is minimized. The system is then simulated with a large lateral pressure with all restraints active. The Berendsen barostat[50] is the natural choice for pressure coupling, because it takes the system to equilibrium rapidly and monotonously. Based on our sample cases (see below), 1000 bar should be adequate for both atomistic and coarse-grained systems. In our example systems, the protein is swallowed by the bilayer in approximately a nanosecond in both atomistic and coarse-grained systems. At this moment, the vacuum space has disappeared around the bilayer and the decrease of the box edge length slows down substantially. These two criteria can be employed to select the end state of the push simulation. The effects of the extended push simulation on the subsequent system relaxation are discussed in the Supporting Information (SI).

2.4. Recovery of the System from the Applied Lateral Pressure ("Relaxation Simulation"). After the protein is surrounded by lipids, another simulation is performed in which the system is allowed to relax to recover the tensionless state of the membrane. In this simulation, only the protein restraints are active. The lateral pressure is simply set to its standard value and the system is simulated until its lateral area is stabilized. In our test cases (see below), this took a few nanoseconds for both atomistic and coarse-grained systems. The validity of these short relaxation simulations is verified in the SI. However, the length of this simulation is expected to be dependent on the system. The equilibration of the area should be checked prior to advancing to the following steps (see below).

The purpose of the relaxation step is to compensate for the effects of the lateral pressure on the membrane. This does not include the relaxation of the protein structure, relative to its environment, which must be performed afterward. Since the removal of protein restraints (often performed during multiple steps) is a standard protocol necessary in all protein insertion methodologies, it will not be discussed here. However, an example on the effect of the relaxation simulation duration on protein structure upon removal of the restraints in given in the SI.

3. EXAMPLE SYSTEMS USED TO DEMONSTRATE THE PRINCIPLE

The proposed approach is applied to three model systems. The embedding of an adenosine A_{2A} receptor (Protein Data Bank[52]

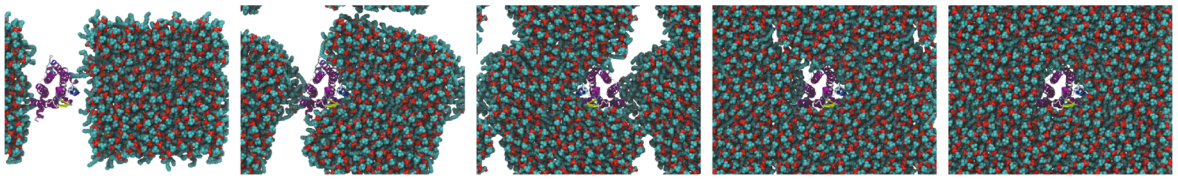


Figure 2. Snapshots presenting the atomistic single-protein insertion process at time steps of 0, 250, 500, 750, and 1000 ps. Coloring is as described in Figure 1.

record 3EML) into a lipid bilayer formed of 288 POPC (1-palmitoyl,2-oleoyl-*sn*-glycero-3-phosphocholine) lipid molecules is considered, employing both atomistic OPLS[53,54] and coarse-grained MARTINI[55,56] approaches using GROMACS. In the atomistic case, we also consider the embedding of two A_{2A} receptors at once. The simulation parameters of these sample cases are made as identical as possible. Water molecules were included in all simulations, even though, in most situations, the speed boost obtained by initially removing them might be greatly beneficial. In addition, the considered host bilayers were adequately pre-equilibrated, even though this is by no means necessary for the insertion process. Upon placing the protein next to the host membrane, the box was made square in both cases.

3.1. Atomistic System. Prior to protein insertion, the atomistic lipid bilayer was constructed and adequately hydrated with either 45 or 60 water molecules per lipid to be used in embedding one and two proteins, respectively. The membranes were then sufficiently equilibrated. Ten (10) Cl^- ions were included per protein to neutralize protein charges. The atomistic insertion and relaxation simulations were performed for 1 and 2 ns, respectively. The simulation parameters for the insertion of one and two proteins were identical. A time step of 2 fs was employed. The nitrogen atom and the terminal carbon atoms of the lipid tails were restrained along the normal of the membrane with a force constant of 10 000 kJ mol^{-1} nm^{-2}. Heavy atoms (non-hydrogen) of the protein were restrained in all three dimensions with a force constant of 10 000 kJ mol^{-1} nm^{-2}. The cutoff radii for both Lennard-Jones and electrostatic interactions were set to 1 nm, beyond which the Particle-Mesh Ewald[57] approach was employed for the latter. A neighbor list of 1 nm was updated every 10 steps. The temperature was kept constant at 310 K with the Berendsen thermostat[50] with a coupling time constant of 0.5 ps. The entire system was coupled together. The pressure was controlled semi-isotropically through the Berendsen barostat,[50] with a coupling time of 20 ps. During protein insertion, the lateral pressure (in the membrane plane) was set to 1000 bar, whereas a value of 1 bar was employed during relaxation simulation. The pressure in the direction of the bilayer normal was always set to 1 bar. A standard workstation computer was able to run the two simulations (without GPU acceleration) within 2 and 4 h, respectively.

3.2. Coarse-Grained System. The protein employed in the atomistic simulations was coarse-grained with the Martinize script. The secondary structure was obtained from the DSSP[58] tool and supplied as an input for the script. A POPC bilayer was constructed and equilibrated with adequate hydration of 45 water molecules per lipid, resulting in a total of 3240 water beads. After including the protein, 10 chlorides were added as counterions for the protein charges.

A simulation time step of 10 fs was employed and the insertion and relaxation simulations were run for 1 and 2 ns, respectively. The choline bead as well as the last beads of the hydrocarbon tails were restrained in the direction normal to the membrane plane with a force constant of 10 000 kJ mol^{-1} nm^{-2}. All protein beads were restrained by a force constant of 10 000 kJ mol^{-1} nm^{-2}. A standard shift approach was employed for nonbonded interactions. Electrostatic and Lennard-Jones interactions were shifted to zero between 0 to 1.2 and 0.9 to 1.2 nm, respectively. A neighbor list with a radius of 1.2 nm was updated every 10 steps. The Berendsen thermostat and barostat[50] were employed for maximal stability and quick relaxation with coupling times of 0.5 and 20 ps, respectively. The reference temperature was set to 310 K, and the entire system was coupled together. Semi-isotropic scaling was employed for the barostat. The reference pressure in the membrane plane was set to 1000 and 1 bar in the insertion and relaxation simulations, respectively, whereas normal to the membrane, a value of 1 bar was employed in both simulations. With this standard setup, the simulated time scales (1−2 ns) were reachable within ∼1 min on a regular desktop workstation.

Based on our tests, the key parameter affecting the stability of the insertion simulation is the coupling time constant of the barostat. Whereas employing 20 ps resulted in a stable simulation in all the cases considered here, increasing this value might be required in some rare cases. This will naturally slightly increase the time needed for the vacuum space to disappear from the simulation box.

3.3. Results Show That the Principle Works. The push simulation is visualized in Figure 2. Additional data are given as a video presentation in the SI. The time evolution of the box edge in both push and relaxation simulations is shown in Figure 3 for both sample cases. Solid lines show data from the push step, whereas the dashed lines refer to relaxation simulations. Black lines stand for the coarse-grained system, whereas blue and red curves show data for the atomistic system in the case of embedding one or two proteins, respectively. The data clearly indicate that very short simulations of ∼1 ns are adequate in

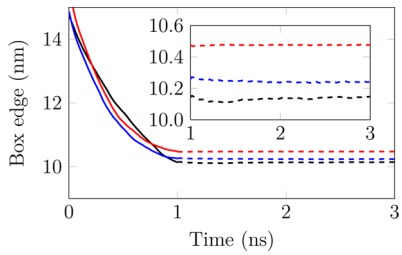


Figure 3. Time evolution of the simulation box edge during protein insertion (solid lines) and relaxation (dashed lines) steps in coarse-grained (black) and atomistic (red and blue) systems. Data for insertion of one and two proteins in an atomistic system are shown in blue and red, respectively. The box edge time evolution during the relaxation simulation is shown in the inset with more detail.

pushing a protein into the bilayer in both atomistic and coarse-grained descriptions of this considered system. This is also clear from the videos provided in the SI. The determination of the suitable end state is trivial either by visually observing the system or by considering the saturation of the box size.

The bilayer itself is not compressed to a large extent, which is indicated by the rapid recovery of the membrane size as the excessive lateral pressure is removed (dashed lines in Figure 3). This is also evident from the time evolution of the box size during much longer relaxation simulations (see the SI). No lipid flip-flops or gel domain formation were observed during the simulations, because of the restraints.

It must be emphasized that the lengths of the simulations employed in the example cases presented in this paper are only guidelines. The proper disappearance of the vacuum space during the push simulation and the area equilibration during the relaxation simulation should always be verified as the required simulation length might vary from one system to another.

The protein structure was basically unaffected by the lateral pressure during the insertion simulations, as can be seen from the root-mean-square deviation (RMSD) of the proteins presented in Figure 4. Dashed lines show the RMSD of proteins under vacuum with the same restraints applied. The small values obtained and the minor increase caused by the lateral pressure in comparison to the vacuum simulations suggest that the chosen force constant of 10 000 kJ mol^{-1} nm^{-2} is sufficient to preserve the secondary structure of the protein during insertion.

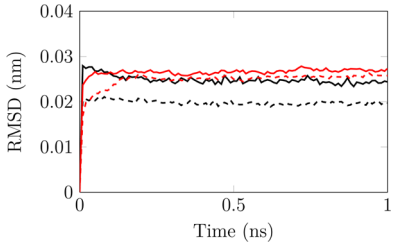


Figure 4. Root-mean-square deviation (RMSD) of protein atoms as a function of time during insertion simulations (solid lines). The RMSD under vacuum is also shown for comparison (dashed lines). Red and black lines show data for atomistic (insertion of one protein) and coarse-grained systems, respectively.

4. CONCLUSIONS

In this article, we have presented a new approach for embedding proteins into lipid membranes. The approach is, in multiple ways, superior to various other methods suggested and employed earlier. It is independent of the employed simulation software and does not rely heavily on software or scripts. It can be employed with all force fields by simply running two fairly short simulations. First, the protein is pushed into the host lipid bilayer by applying a large lateral pressure to the system. This is followed by a quick relaxation simulation, which is essentially required in all protein insertion simulations. No removal of lipids is involved and, therefore, the lipid composition and the transmembrane lipid distribution of the host bilayer are preserved during the insertion process. The suggested approach also allows the simultaneous insertion of multiple proteins. In addition to the examples presented here, the method has been successfully adapted in numerous ongoing studies on various GPCRs, using both atomistic and coarse-grained models.

There are also some downsides of the suggested methodology. First, the required computation time is slightly increased, yet this is compensated for by the lower demand for human effort or input. Based on our tests on atomistic and coarse-grained models, the required equilibration times involved with our method are of the order of a few nanoseconds. Moreover, the area of the host membrane is adequately stabilized during this short period. Therefore, the time scale required by the suggested method is comparable to other existing methods when the required equilibration after protein insertion is considered. In addition, the time requirements of our method pose no challenge for a modern workstation, even in the atomistic system, and the situation is even better with coarse-grained models as simulations in the nanosecond regime require ∼1 min of real time on a standard workstation computer. In addition, considering that the microsecond regime for membrane protein simulations has been available for quite some time[59] and that this time scale is required for proper convergence of these simulations,[60] the imposed requirement is considered acceptable already with the computing power presently available. Second, for the time being, the insertion of conical proteins or doughnut-shaped proteins encapsulating lipids requires some additional manual input. These complicated cases are discussed in the SI.

■ ASSOCIATED CONTENT

Supporting Information

Videos showing the embedding process in the atomistic system. Examination on the effect of the durations of the pushing and relaxation simulations on the system equilibration. The capability of the membrane to host an unrestrained protein after a short relaxation simulation is verified. Discussion on the possible ways to insert conical and doughnut-shaped proteins. This material is available free of charge via the Internet at http://pubs.acs.org/.

■ AUTHOR INFORMATION

Corresponding Author

*Tel.: +358 (0)40 1981185. Fax: +358 (0)3 31153015. E-mail: matti.javanainen@tut.fi.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The author is grateful to Ilpo Vattulainen, Hector Martinez-Seara Monné, and Tomasz Róg for fruitful discussions and support and for the colleagues for eagerly employing the methodology on various systems. The author also thanks CSC−IT Center for Science for computational resources and the Finnish Doctoral Programme in Computational Sciences (FICS) for funding.

■ REFERENCES

(1) Lee, A. G. *Biochim. Biophys. Acta, Biomembr.* **2004**, *1666*, 62−87.
(2) Oates, J.; Watts, A. *Curr. Opin. Struct. Biol.* **2011**, *21*, 802−807.
(3) Jafurulla, M.; Chattopadhyay, A. *Curr. Med. Chem.* **2013**, *20*, 47−55.
(4) Phillips, R.; Ursell, T.; Wiggins, P.; Sens, P. *Nature* **2009**, *459*, 379−385.
(5) Solmaz, S. R. N.; Hunte, C. *J. Biol. Chem.* **2008**, *283*, 17542−17549.

(6) Hanson, M. A.; Cherezov, V.; Griffith, M. T.; Roth, C. B.; Jaakola, V.-P.; Chien, E. Y. T.; Velasquez, J.; Kuhn, P.; Stevens, R. C. *Structure* **2008**, *16*, 897−905.
(7) Coskun, U.; Simons, K. *Structure* **2011**, *19*, 1543−1548.
(8) McIntosh, T. J.; Simon, S. A. *Annu. Rev. Biophys. Biomol. Struct.* **2006**, *35*, 177.
(9) Lingwood, D.; Simons, K. *Science* **2010**, *327*, 46−50.
(10) Bretscher, M. S. *Nature (London), New Biol.* **1972**, *236*, 11.
(11) Verhoven, B.; Schlegel, R.; Williamson, P. *J. Exp. Med.* **1995**, *182*, 1597−1601.
(12) van Meer, G. *Cold Spring Harbor Perspect. Biol.* **2011**, *3*, a004671.
(13) Feller, S. E. *Curr. Opin. Colloid Interface Sci.* **2000**, *5*, 217−223.
(14) Pastor, R. W. *Curr. Opin. Struct. Biol.* **1994**, *4*, 486−492.
(15) Tieleman, D. P.; Marrink, S.-J.; Berendsen, H. J. C. *Biochim. Biophys. Acta, Rev. Biomembr.* **1997**, *1331*, 235−270.
(16) Merz, K. M., Jr. *Curr. Opin. Struct. Biol.* **1997**, *7*, 511−517.
(17) Pandit, S. A.; Scott, H. L. *Biochim. Biophys. Acta, Biomembr.* **2009**, *1788*, 136−148.
(18) Lindahl, E.; Sansom, M. S. *Curr. Opin. Struct. Biol.* **2008**, *18*, 425−431.
(19) Gumbart, J.; Wang, Y.; Aksimentiev, A.; Tajkhorshid, E.; Schulten, K. *Curr. Opin. Struct. Biol.* **2005**, *15*, 423−431.
(20) Bond, P. J.; Holyoake, J.; Ivetac, A.; Khalid, S.; Sansom, M. S. *J. Struct. Biol.* **2007**, *157*, 593−605.
(21) Roux, B.; Schulten, K. *Structure* **2004**, *12*, 1343−1351.
(22) Ash, W. L.; Zlomislic, M. R.; Oloo, E. O.; Tieleman, D. P. *Biochim. Biophys. Acta, Biomembr.* **2004**, *1666*, 158−189.
(23) Grossfield, A. *Biochim. Biophys. Acta, Biomembr.* **2011**, *1808*, 1868−1878.
(24) Vattulainen, I.; Rog, T. *Cold Spring Harbor Perspect. Biol.* **2011**, *3*, a004655.
(25) Niemela, P. S.; Miettinen, M.; Monticelli, L.; Hammaren, H.; Bjelkmar, P.; Murtola, T.; Lindahl, E.; Vattulainen, I. *J. Am. Chem. Soc.* **2010**, *132*, 7574−7575.
(26) Sadiq, S. K.; Guixà-González, R.; Dainese, E.; Pastor, M.; De Fabritiis, G.; Selent, J. *Curr. Med. Chem.* **2013**, *20*, 22−38.
(27) Berkowitz, M. L. *Biochim. Biophys. Acta, Biomembr.* **2009**, *1788*, 86−96.
(28) Chini, B.; Parenti, M. *J. Mol. Endocrinol.* **2004**, *32*, 325−338.
(29) Polley, A.; Vemparala, S.; Rao, M. *J. Phys. Chem. B* **2012**, *116*, 13403−13410.
(30) López Cascales, J. J.; Otero, T. F.; Smith, B. D.; Gonzalez, C.; Marquez, M. *J. Phys. Chem. B* **2006**, *110*, 2358−2363.
(31) Gurtovenko, A. A.; Vattulainen, I. *J. Phys. Chem. B* **2008**, *112*, 4629−4634.
(32) Gurtovenko, A. A.; Vattulainen, I. *J. Phys. Chem. B* **2008**, *112*, 1953−1962.
(33) Gurtovenko, A. A.; Vattulainen, I. *J. Am. Chem. Soc.* **2007**, *129*, 5358−5359.
(34) Bill, R. M.; Henderson, P. J.; Iwata, S.; Kunji, E. R.; Michel, H.; Neutze, R.; Newstead, S.; Poolman, B.; Tate, C. G.; Vogel, H. *Nat. Biotechnol.* **2011**, *29*, 335−340.
(35) Kandt, C.; Ash, W. L.; Peter Tieleman, D. *Methods* **2007**, *41*, 475−488.
(36) Edholm, O.; Berger, O.; Jähnig, F. *J. Mol. Biol.* **1995**, *250*, 94−111.
(37) Woolf, T. B.; Roux, B. *Proc. Natl. Acad. Sci. U.S.A.* **1994**, *91*, 11631−11635.
(38) Jo, S.; Kim, T.; Im, W. *PLoS One* **2007**, *2*, e880.
(39) Jo, S.; Lim, J. B.; Klauda, J. B.; Im, W. *Biophys. J.* **2009**, *97*, 50−58.
(40) Tieleman, D.; Berendsen, H. *Biophys. J.* **1998**, *74*, 2786−2801.
(41) Hess, B.; Kutzner, C.; van der Spoel, D.; Lindahl, E. *J. Chem. Theory Comput.* **2008**, *4*, 435−447.
(42) Shen, L.; Bassolino, D.; Stouch, T. *Biophys. J.* **1997**, *73*, 3−20.
(43) Faraldo-Gómez, J. D.; Smith, G. R.; Sansom, M. S. *Eur. Biophys. J.* **2002**, *31*, 217−227.
(44) Staritzbichler, R.; Anselmi, C.; Forrest, L. R.; Faraldo-Gómez, J. D. *J. Chem. Theory Comput.* **2011**, *7*, 1167−1176.
(45) Wolf, M. G.; Hoefling, M.; Aponte-Santamara, C.; Grubmüller, H.; Groenhof, G. *J. Comput. Chem.* **2010**, *31*, 2169−2174.
(46) Schmidt, T. H.; Kandt, C. *J. Chem. Inf. Model.* **2012**, *52*, 2657−2669.
(47) Humphrey, W.; Dalke, A.; Schulten, K. *J. Mol. Graphics* **1996**, *14*, 33−38.
(48) Schrödinger, LLC. *The PyMOL Molecular Graphics System, Version 1.3r1*; unpublished work, 2010.
(49) Case, D. A.; Cheatham, T. E.; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. *J. Comput. Chem.* **2005**, *26*, 1668−1688.
(50) Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. *J. Chem. Phys.* **1984**, *81*, 3684.
(51) Stone, J. *Tachyon Parallel/Multiprocessor Ray Tracing System*, Available via the Internet at http://jedi.ks.uiuc.edu/~johns/raytracer/, 1998.
(52) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. *Nucleic Acids Res.* **2000**, *28*, 235−242.
(53) Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. *J. Am. Chem. Soc.* **1996**, *118*, 11225−11236.
(54) Kaminski, G. A.; Friesner, R. A.; Tirado-Rives, J.; Jorgensen, W. L. *J. Phys. Chem. B* **2001**, *105*, 6474−6487.
(55) Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; de Vries, A. H. *J. Phys. Chem. B* **2007**, *111*, 7812−7824.
(56) Monticelli, L.; Kandasamy, S. K.; Periole, X.; Larson, R. G.; Tieleman, D. P.; Marrink, S.-J. *J. Chem. Theory Comput.* **2008**, *4*, 819−834.
(57) Darden, T.; York, D.; Pedersen, L. *J. Chem. Phys.* **1993**, *98*, 10089.
(58) Kabsch, W.; Sander, C. *Biopolymers* **1983**, *22*, 2577−2637.
(59) Khelashvili, G.; Grossfield, A.; Feller, S. E.; Pitman, M. C.; Weinstein, H. *Proteins: Struct., Funct., Bioinf.* **2009**, *76*, 403−417.
(60) Grossfield, A.; Feller, S. E.; Pitman, M. C. *Proteins: Struct., Funct., Bioinf.* **2007**, *67*, 31−40.