

CELLmicrocosmos 2.2 MembraneEditor: A Modular Interactive Shape-Based Software Approach To Solve Heterogeneous Membrane Packing Problems

Björn Sommer,^{*†} Tim Dingersen,[†] Christian Gamroth,[†] Sebastian E. Schneider,[‡] Sebastian Rubert,[†] Jens Krüger,[§] and Karl-Josef Dietz^{||}

[†]Bio-/Medical Informatics Department, Bielefeld University, D-33615 Bielefeld, Germany

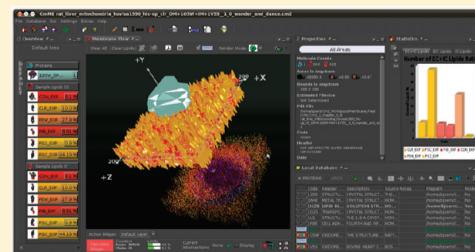
[‡]Physik-Department, Technische Universität München, D-85748 Garching, Germany

[§]Organic Chemistry Department, University of Paderborn, D-33098 Paderborn, Germany

^{||}Department of Plant Biochemistry and Physiology, Bielefeld University, D-33615 Bielefeld, Germany

 Supporting Information

ABSTRACT: New perspectives have been developed to understand the processes of modeling heterogeneous membranes. These are crucial steps prior to applying advanced techniques like molecular dynamic simulations of whole membrane systems. Lipid, protein, and membrane packing problems are addressed based on biochemical properties in combination with computational optimization techniques. The CELLmicrocosmos 2.2 MembraneEditor (CmME) is introduced as an appropriate framework to handle such problems by offering diverse algorithmic approaches. Its algorithm plug-in-interface enables modelers to generate problem-specific algorithms. Good solutions concerning runtime and lipid density are realized by focusing on the outer shapes of the PDB-based molecules. Application cases are presented like the publication-based modeling of inner and outer mitochondrial membrane-fragments, semiautomatic incorporation of proteins, and the assembly of rafts. Concerning geometrical aspects of the lipids, the achieved results are consistent with experimental observations related to lipid densities and distributions. Finally, two membranes simulated with GROMACS are analyzed and compared: the first is generated with conventional scripting techniques, the second with the CmME Distributor algorithm. The examples prove that CmME is a valuable and versatile tool for a broad set of applications in analysis and visualization of biomembranes.



1. INTRODUCTION

1.1. Membrane Modeling. Since the original mosaic model was introduced,¹ many studies have advanced our understanding of (bio)membrane assembly, organization, and function. Nowadays, the heterogeneity of biological membranes is a very important topic with broad implications.² On the experimental side, microscopic studies have proven to be powerful enough to determine the biophysical and chemical features of model membranes. To extend the thusly gained knowledge and overcome the limitations of today's microscopy solutions and resolutions, different computational simulations like Molecular Dynamic (MD) or Monte Carlo Simulations (MC) have been developed.³ Among the very large theoretical number of 180,000 different molecular lipid types 9600 belong to the group of phospholipids.⁴ However most studies address lipid mixtures of only two or three different lipid types. Moreover, lipid rafts play an important role in the lateral heterogeneity of bilayer membranes.² Surprisingly, only a few publications aim at modeling and/or simulating more realistic raft assemblies.

The increase of computational power steadily improves the ability to simulate large membrane systems. A number of preassembled membranes exist on the Internet.⁵ These membrane models are normally based on the Protein Data Bank format (PDB),⁶ which is also often used to obtain coordinate files compatible with simulation environments. Few programs offer solutions to the problem of generating membranes with customized compositions: ChemSW Chemsite Pro,⁷ CHARMM-GUI Membrane Builder,⁸ or the Membrane Plug-in⁹ for the VMD (Visual Molecular Dynamics).¹⁰ Alternatives are script-based approaches like molecule-replacement methods¹¹ or the generation of molecule-grids with MOE.^{12,13} These partially commercial programs may be well-suited for particular purposes, but they are unable to create large-scale, multilayered, raft- and protein-containing membrane structures with different compositions.

1.2. CELLmicrocosmos MembraneEditor. CmME realizes a freely available and modular PDB-based membrane modeling

Received: October 1, 2010

Published: April 19, 2011

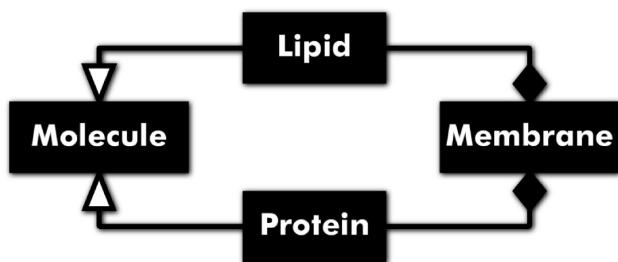


Figure 1. The class diagram shows the relationship between molecule, lipid, protein, and membrane.

approach which overcomes the addressed restrictions by giving the modeler the opportunity to generate scalable membrane patches in either a fast and easy or a more complex and accurate modeling mode. A number of shape-based algorithms support different kinds of assemblies, and a plug-in-interface helps the user program customized algorithms if needed. In addition, proteins may be (semi-) automatically positioned into the membrane by using theoretical structural data retrieved from two databases.^{14,15} PDB structures can be downloaded directly from the PDB database or imported from the local file system. In addition, the membrane models are exportable to external all-atom based simulations.

This work discusses different options provided by CmME to generate membranes. The modeled membranes were qualitatively and quantitatively evaluated by comparison with experimental data on lipid distributions, lipid densities, and lipid behavior. In addition, we introduce and analyze a first approach of a DPPC bilayer modeled by CmME and simulated with GROMACS.¹⁶

1.3. Membrane Packing Problems. Packing problems constitute a principal challenge in computer science¹⁷ with economical and ecological implications, e.g. in the logistics sector, where the optimization of transport space is handled.^{18–20} They belong to the NP-complete problems, which means that efficient and exhaustive solutions are unavailable. The appliance of packing problems to membrane modeling is, as mentioned above, not new. However, an adequate and accurate theoretical classification from a computational point of view, focusing especially on geometry-based membrane modeling, appears to be overdue.

A membrane mainly consists of lipids and proteins. Of course both can be interpreted as subclasses of molecules (Figure 1). While the size and shape of different lipid types is relatively similar, these properties are highly variable in the case of proteins. Therefore, the distinct and adequate treatment of lipids and proteins was part of the CmME concept from the beginning. Three related optimization features had to be examined in order to adequately address the membrane packing problem (MPP): (i) the lipid (LPP) and (ii) protein packing (PPP) problems and (iii) the combination of both subproblems. Questions concerning the LPP are: How could one or multiple lipid models be optimally packed onto a monolayer or bilayer? For PPP: How could one or multiple protein models be optimally packed onto a mono- or bilayer? And for MPP: How could one or multiple lipid and protein models be optimally packed together under consideration of LPP and PPP onto a mono- or bilayer?

Until now the term MPP was only used by a few authors for describing the mathematical packing problem that occurs when an elastic membrane contacts a rigid obstacle.^{21–23} Here the definition is extended to the molecular level. One of the first

mentions of the term LPP was found in conjunction with a method by Israelachvili et al. which describes the lipids as the principal source of the intrinsic curvature important for the formation of vesicles.^{24,25} Feng et al. investigated a PPP by using an icosahedron as a model to describe the orientational distribution of contacts in protein clusters derived from a high-resolution protein data set.²⁶

After classifying the optimization requirement, the optimization features needed to be defined and restricted in light of the large number of physicochemical properties which could be taken into account. The main aim of CmME is a shape-based approach in order to keep the computation time within the limits of desktop applications. The optima (i to vi) relevant for this work were defined as follows:

- (i) OPT_LPP_LOW_AREA: The optimum is defined by the minimal possible area per lipid (\AA^2) (section 3.1, 3.4).
- (ii) OPT_LPP_DEF_AREA: The optimum is defined by the user-defined area per lipid (\AA^2) (section 3.1).
- (iii) OPT_LPP_2D_AREA: The optimum is defined by the highest possible membrane area occupied by lipids (%) (section 3.2).
- (iv) OPT_LPP_RATIO: The optimum is defined by reaching the correct lipid type ratios (%) (section 3.1, 3.2, 3.4, 3.5).
- (v) OPT PPP POS: The optimum is defined by the correct protein positioning in relation to the bilayer (section 3.3).
- (vi) OPT_LPP_ENERGY: The optimum is defined by achieving the energetic equilibrium state (section 3.6).

Finally, terms for the solution of problems LPP, PPP, and MPP had to be found. At this point the authors would like to introduce the following terms: LPA: Lipid Packing Algorithms, PPA: Protein Packing Algorithms, and MPA: Membrane Packing Algorithms.

This terminology is especially important to differentiate MPA from the term “membrane algorithms”. These kinds of algorithms are biologically motivated theoretical models of parallel and distributed computing.²⁷

In this work, the focus was placed on six algorithms and their solution approaches to LPP (section 2.5. and 3.). In addition, PPP and MPP are discussed in section 3.3.

1.4. The Knapsack Problem. This section will introduce the appliance of a computer science related packing problem, namely the Knapsack Problem to LPP.²⁸ The Knapsack Problem (KP) is formally defined as follows:

$$\text{maximize } \sum_{i=1}^n v_i x_i \text{ subject to constrain } \sum_{i=1}^n w_i x_i \leq W, x_i \in \{0, 1, \dots, c_i\}.$$

There exist n kinds of items with values of v_i and a weight of w_i . The total weight of the given container is restricted to W . In regular packing problems, the container is normally defined as the object. In the following text, the authors prefer the word “container”, which is more intuitive for the KP. The number of copies x_i of each item is restricted to the maximum number c_i . The latter criterion defines a special case of the Knapsack Problem, the Bounded KP (BKP). If c_i is infinity and therefore the upper bound is removed, the problem is regarded to as the Unbounded KP (UKP). For the application cases discussed here, only positive integer values for v_i and w_i are of relevance.

Already the original mosaic membrane model proposed the bilayer as a two-dimensional liquid arrangement.¹ The weight w_i of the KP is a virtual value which is also often used as a dimensional value. In addition, there exist knapsack approaches for multiple dimensions, called the multidimensional KP.²⁸

Because this work focuses on rectangular bilayers, the two-dimensional^{17,19} or geometric²⁹ KP class is appropriate (2D-KP). Although the six algorithms discussed in this work are working on two dimensions, the CmME plug-in-interface also supports the development of future algorithms aiming at three-dimensional packing. In this case, the adequate definition would be the three-dimensional KP (3D-KP), relevant for example for vesicle modeling.

KP is restricted to one container. In case a problem contains multiple *related* containers, the KP would be extended to the multiple KP (MKP), which is similar to the bin packing problem.¹⁸ It is intuitive to define a monolayer patch as one container. A bilayer patch can be seen as two containers. Because all six algorithms discussed here focus on the geometric properties of the molecules and refuse physicochemical interactions as well as periodic boundary conditions, both layers are regarded as being strictly divided by the hydrophobic core (although in nature both layers steadily interact). In addition, the percent lipid distribution is defined separately for each layer to enable membrane asymmetry. For both reasons, in this work no MKP is assumed for a bilayer but rather two separated KPs. If a CmME application would require interactions beyond the separate lipid distribution settings for each layer, e.g. when computing flip flops or other interactions between the layers,³⁰ the approaches could be regarded as MKP. In addition, if multiple membrane stacks are layered on top of each other without a hydrophobic core in between, it could be considered a 3D-KP.

After defining KP as the appropriate problem class for the approaches discussed here, the different variables needed to be assigned. The container represents exactly one membrane layer, and the items are the molecules. Many items of relatively few different figures or shapes exist¹⁸ for the regular LPP. The frequency is defined by the lipid ratio and the weight w_i by the width and length of the molecule; therefore, it is a 2D-KP. The algorithms discussed here count each molecule as one and add it to the total number of molecules, thus the value v_i is one. In case a layer contains only one lipid type, c_i is infinity and the UKP is appropriate. Conversely, if a layer contains different lipid types (for example 90% phospholipid and 10% cholesterol), c_i is restricted to the frequency of the items: the user-defined lipid percentages. Even if there is vacant space on the layer for a lipid type to fit in, it will not be inserted when the actual real percentage of this lipid type is equal to or larger than the target percentage. The according problem class is the BKP. Another important feature is the shape of the items, which will be discussed in section 2.5.2. It will be shown that the definition of the 2D-KP is insufficient, depending on the algorithm applied.

Classical KP depends on one or two dimensions. The orientation of the objects is regularly fixed or rotated in 90° steps. This restriction is inevitable for a number of algorithms for orthogonal packing problems using this feature as a base for runtime optimization,^{17,19} but in the case of a membrane, the degree of rotation around the Y-axis is not restricted. Therefore many KP approaches are not applicable to the LPP. Because KP is an NP-complete problem, there exist various alternative nonexhaustive solutions: greedy algorithms, approximation algorithms, heuristic algorithms, etc.²⁸ For many application cases a single good solution is sufficient. In section 3.2 the implication of this theoretic background on the CmME algorithms in conjunction with a special case of a rotation-restricted KP will be discussed as well as the question, if the finding of an optimal solution is inevitable for LPP.

1.5. Lipid Densities. An important factor to judge the geometric quality of membrane models is the lipid density of the resulting structures. Densities of mono- or bilayer membranes may vary, depending on variables like lipid types, pressure, and temperature: For phosphatidylcholine (PC), a value of 42.5 Å² space per lipid molecule was derived from fluorescence microphotolysis³¹ and theoretically confirmed for DLPC monolayers³² and by electron diffraction studies made for DPPC bilayers (40.7 Å²).³³ Extreme values like 26 Å² for DLPC³¹ or 38.3 Å² for cholesterol and as the upper limit, 134 Å² for PC derived by thin-layer chromatography, are also sometimes found under certain experimental conditions but are not representative.³⁴ Established average values are described by Peitzsch et al.: Lipids with unsaturated chains range from 65 to 70 Å².³⁵ In agreement with this data, Hulbert and Else suggest cross-section areas of 60 to 70 Å² for membranes at a natural pressure of 40 dyn/cm.^{34,36} For liquid-crystalline POPC membranes, pressure-dependent values of 62 to 68 Å² have been reported.³⁷ In addition, there seem to be no³³ or only very small lipid density differences of up to ±1.6 Å² for different PC between lipid mono- and bilayers.³⁸ Shipley³⁹ provided a good overview on this topic describing different lipid types derived by X-ray diffraction studies. Similar values were obtained by MD simulations. For example, a value of 64.5 Å² was derived from a simulated DPPC⁴⁰ and ~54.8 Å² from a simulated POPC bilayer.³⁷ Different coarse-grained simulation approaches are also able to compete with previously mentioned all-atom models in terms of the area per lipid.^{41–44} The MARTINI model for example, achieves results similar to experimental values ranging from ~59 to ~69 Å² for different PC lipids.⁴⁵ Nagle and Tristram-Nagle⁴⁶ gave a resume on lipid MD relevant lipid densities, which will be taken into account in section 3.6 for the MD simulation of a DPPC bilayer. There exist various computational methods to measure the area per lipid.⁴⁷ In this work, it is computed by dividing the layer area by the total number of lipids (average lipid density).

1.6. The Application Cases. Six application cases were used to evaluate the CmME methods introduced in section 2 to generate membranes with regard to the criteria established above:

Section 3.1. deals with the modeling of outer and inner mitochondrial membranes. The application of the classical 2D-KP to LPP is discussed in section 3.2. In section 3.3., the placement of a membrane-spanning and a single α-helix-anchored protein into the two previously generated membranes is evaluated. The computation of extreme densities, testing the ability of the algorithms to achieve a very dense packing, is done in section 3.4. The generation of a cholesterol sphingomyelin lipid raft-containing plasma membrane is discussed in section 3.5. And, in section 3.6., a DPPC bilayer is generated with CmME and simulated with GROMACS.

2. METHODS

During the development of CmME, different methods were implemented and verified, which were necessary for the reliable modeling of the membrane:

2.1. PDB Integration. CmME is able to download protein files directly from the PDB database. For lipid files, different alternative resources exist on the Internet: KloTho⁴⁸ offers a number of modeled lipid files. The databases HIC-Up⁴⁹ and Ligand Expo⁵⁰ and Web sites like Chemistry Molecular Models⁵¹ contain artificially structured as well as experimentally derived

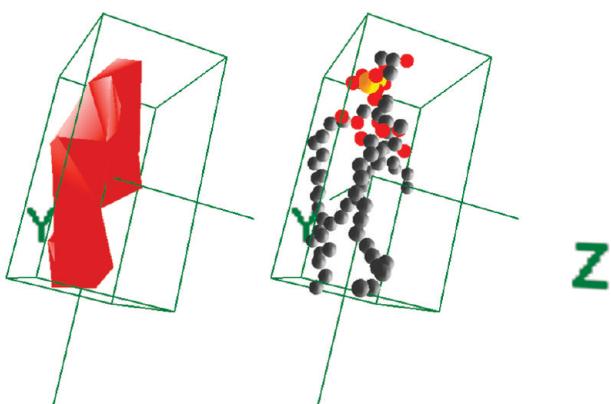


Figure 2. The two collision modes used in CELLmicrocosmos 2.2 MembraneEditor are shown on cardiolipin (cdn_exp). Shape (left) and atom collision (right).

PDB compounds. Preassembled membranes can be used to extract lipid models⁵ which sometimes additionally contain topological information important for MD simulations.⁵² Every valid PDB file can be directly imported into CmME. To ensure compatibility between the exported PDB format of the membrane and different structure viewers or simulation environments like GROMACS, the PDB export properties can be adjusted. The settings affect, among others, the serial numbers and chain IDs important for the secondary structure, the coordinate transformation and the integration of reverse parsing information. This may be important if the exported membrane is altered with other software and should thereafter be reimported for further editing.

2.2. Lipid and Protein Handling. As mentioned in section 1.3, the program needs to distinguish between two types of molecules: lipids and proteins (Figure 1). The lipid distribution is defined by percent ratios per membrane-layer as input and has a precision of up to three decimal places. The computation is done using the algorithms described below (section 2.5). The position of proteins is defined manually via the Drag and Drop mechanism. Justification and placement of all molecules can be altered directly in the Alignment Mode (see below in Figure 9), which shows only the neighbors of the selected molecule. In addition, a number of proteins can be automatically justified to the membrane by using the Protein Data Bank of Transmembrane Proteins (PDBTM)¹⁵ and/or the Orientations of Proteins in Membranes database (OPM)¹⁴ (OPT_PPP_POS). PDBTM employs the TMDET algorithm to detect trans-membrane regions by using the 3D structure of integral proteins only.⁵³ OPM uses an approach to minimize the transfer energies from water to the lipid bilayer to compute the spatial arrangement of the proteins. Using this method, it distinguishes transmembrane and membrane-integral monotopic proteins from water-soluble ones.⁵⁴ CmME is not only capable of placing a protein into an existing membrane, it can also generate a membrane around any preplaced protein.

2.3. Shape-Based Computation and Visualization. The handling of molecules, especially the computation of lipid bilayers requires collision detection. In order to reduce computational costs, the application includes a geometry-based collision detection which exploits two methods:

- (1) For the shape collision method (Figure 2 left), a three-dimensional outer shape of the molecule is computed which provides a closed surface for collision detection.

The closed surface avoids the possibility of small molecules extending into thin areas of other members (e.g., the pore of a channel). The detail level is adjustable and affects the number of triangle points and hence the shape visualization as well as the computational complexity.

- (2) The atom collision method (Figure 2 right) takes relevant atoms of the molecules into account. Those atoms which are considered collision candidates are tested for spherical intersections. Using this method fully avoids intersections, even if low resolution molecule surfaces are used, but molecules may overlap. Both methods can be combined.

The shape-based visualization of the membrane with adjustable detail-levels ensures the interactive handling of large-scale membrane models. In addition, standard visualization modes like covalent or van der Waals radii and Balls and Sticks are possible. The preview option shows the assembled membranes using the external Jmol viewer.^{55,56}

2.4. Membrane Model. The size of the membrane patch is defined in angstrom (\AA). Multiple single- and double layer stacks with different lipid compositions can be added for comparison or modeling of interacting layers. Lipid raft microdomains can be defined, which are small cohesive areas in membranes featuring special lipid compositions.⁵⁷ Except for Simulated Annealing, the lipid mobility is restricted to the defined raft area in all CmME algorithms discussed in this publication.

2.5. Lipid Packing Algorithms. A number of algorithms have been implemented in CmME to solve the LPP. All of them are geared to the outer shapes of the molecules, intramolecular atom positions and these conformations are invariable thus far. Each algorithm implements different options which can be changed by the user. The runtime estimation denotes the number of elementary operations. In the worst case, a computational step may require manipulation of the CmME data pool plus the 3D environment and in the case of Simulated Annealing, the computation of distances to all molecules inside the cutoff range. Not taken into account are the repetitions of correlated single computation steps, like the combination of movement, rotation, and the collision detection of molecules, which are regarded as constants. An overview of Lipid Packing Algorithms follows:

2.5.1. Linear Placing (LP). This algorithm adds lipids row by row. This is the fastest and most simple algorithm. It operates online, which means that molecules are added successively without reallocation and without knowing the succeeding items *a priori*.¹⁸ The lipid area relevant for the placing process is related to the width and depth of its boundary box - a similar characteristic like the 2D-KP. But it is apparent that LP is no full-value optimization algorithm: If a lipid cannot be placed at a certain position during the placing process, this position is skipped, and the next lipid is tried for placement at the following position. For this reason, LP provides the best runtime of all algorithms discussed here, with a linear complexity of

$$O_{LP}(n \times L) = n \quad (1)$$

where n is the number of placing attempts, and L is the lipid placing process. Best packing is achieved if a uniform lipid orientation is used.

2.5.2. Random Placing (RP). Random placing as well as the random orienting of lipids quickly creates a well-filled bilayer. In order to achieve a high density, a cyclic process of shaking and approaching the nearest protein or membrane center is applied

to all lipids, constantly attempting to create space for the addition of new lipids. This algorithm needs to be stopped manually as soon as a satisfactory result is achieved. The runtime computes as follows

$$O_{RP}(r \times (l \times A + n \times P + l \times S)) = (r \times (2l + n)) \quad (2)$$

The number of repetitions until the algorithm is stopped by the user is r . The limit l is a constant value, the sum of the actual membrane or microdomain width and depth. A random lipid placing attempt is represented by A . The variable n is the number of previously added lipids, and S is the lipid shaking and twisting function. In contrast to LP, RP is an optimization algorithm. The optimum is defined by OPT_LPP_LOW_AREA, which is also the case for most of the following algorithms.

Thus, the algorithm combines online, first placing of the lipid, with the off-line method of approaching the neighboring molecules based on the surfaces of the already added lipids and shows parallels to annealing methods discussed in section 2.5.6.

Another important difference to LP is the usage of the complete three-dimensional shape for the collision detection, instead of taking only the lipid bounding box into account. While the movement of the lipid is still restricted to two-dimensions, the collision detection is not. In addition, the three-dimensional shape is irregular. Therefore, RP and the following algorithms have to deal with geometrical as well as combinatorial problems of heterogeneity. Many packing algorithms take advantage of assigning many different items to a few types of differing shapes.¹⁸ This is also the case for lipids of a single lipid type. But since the shape is irregular and extended into the third dimension, each collision has to be computed separately. For this reason, the RP algorithm goes beyond pure 2D-KP approaches. On the contrary, as was previously mentioned, the term 3D-KP describes Knapsack Packing Problems in different interrelated layers. In summary, RP and the following algorithms should be considered as 2.5D-KP for the following three reasons: (i) the movement is restricted to one two-dimensional container, (ii) the shape of the items is three-dimensional, and (iii) irregular. The definition of a two-and-a-half-dimensional problem follows also the definition of Dyckhoff, who used the term one-and-a-half-dimensional for the combination of a one-dimensional problem with continuous measurements.⁵⁸ In other packing contexts the term two-and-a-half-dimensional is used to describe the usage of different consistent layers, where the items of different layers are not packed on top of each other.⁵⁹ In the knapsack context, this type of packing is already subsumed in the definition of the MKP (section 1.4).

2.5.3. Distributor (DI). The DI algorithm should be used to define a fixed number of lipids or an area per lipid value for each membrane-layer. The basic lipid filling process is the same as in the RP algorithm but takes the requested lipid counts into consideration. DI creates new membranes or alters already computed membranes. Additionally, the entire membrane gets shaken by setting the lipids to a high rate of movement, which causes a permanent reorganization of the membrane. The time calculates as follows

$$O_{DI}(r \times ((n \times M \vee O_{RP}) + n \times D)) = (r \times (2n)) \quad (3)$$

In contrast to RP, r , the number of repetitions, may be limited after attaining the user-defined lipid number. Therefore, the optimization of DI is defined primarily by OPT_LPP_DEF_AREA. Only if the user-defined values are not achievable, the

optimum is defined by OPT_LPP_LOW_AREA. Once again, n is the number of lipids in the membrane. For DI there are two options: It may be used to remove redundant, formerly added lipids, represented by the variables $n \times M$, where M is a mixing attempt. This method is useful, if, for example a bilayer is too densely packed and should be relaxed. The O_{RP} supports the opposite process, the addition of lipids. Finally, the $n \times D$ is the eponymous method for DI: This method distributes all lipids along the membrane, a relaxing process without removing lipids. The distances to the nearest neighbors are computed to find an optimal position for each lipid.

2.5.4. Advanced Random Placing (AP). This finite trial-and-error placing algorithm subdivides the membrane into smaller sections and tries to randomly fill those areas with lipids – a classical greedy algorithm. The overall density of lipids can be roughly defined as well as the range for the random orientation, shifting, and tilting. Each parameter affects the runtime

$$O_{AP}(g \times (s \times (A + b \times B))) = (g \times s \times n) \quad (4)$$

The number of different grid segments is g . The salary s is computed by the user-defined lipid density and an empirical derived value. The first lipid placing attempt A randomly tries to place a lipid in the defined boundaries of the actual grid segment. Another lipid is placed in dependence on the previously used position onto the opposite membrane side. If this insertion was unsuccessful, the placing attempt B is repeated b times by randomly tilting and rotating the lipid inside the user-defined limits. The ultimately realized value also depends on the user-defined density. The method of reusing the position of the previously placed lipid is a typical performance-improving heuristic step. Because the oppositely placed lipid gets a new random rotation, the problem of a perfectly mirrored layer does not occur.

2.5.5. The Wanderer (TW). This plug-in-algorithm (section 2.6) has been developed to achieve high lipid densities. It is a greedy algorithm placing molecules row by row combined with a position correction function. After placing the first molecule, the next is placed east of it. The position is computed by using the bounding radius values of both molecules. Now, the newly added lipid wanders toward the west until it collides with the formerly positioned lipid. Then it continues wandering to the north, until reaching the membrane border or another lipid. After colliding again, the lipid is rotated in 1° steps around its Y-axis (it “dances”), trying to move even closer toward the opposing lipid by wandering northwest again. This is done for each lipid n consecutively

$$O_{TW}(n \times (F + D + C)) = (n) \quad (5)$$

The function F places only the first lipid in each row, D is the regular placing attempt, and C is the correction attempt of D . Each of these functions causes the n^{th} inserted lipid to wander into different directions. For example, the wandering process D causes the lipid to first move east and south, in order to find a free place. Then it moves west and north until another lipid is reached, and finally it tries to find the optimal rotation by approaching the other lipid and rotating around its Y-axis. The number of movement steps is defined by the distance between the actual lipid to be placed and the molecule approached.

2.5.6. Simulated Annealing (SA). SA is the only heuristic approximation algorithm introduced here. The application of this MC-related method⁶⁰ to combinatorial problems has been

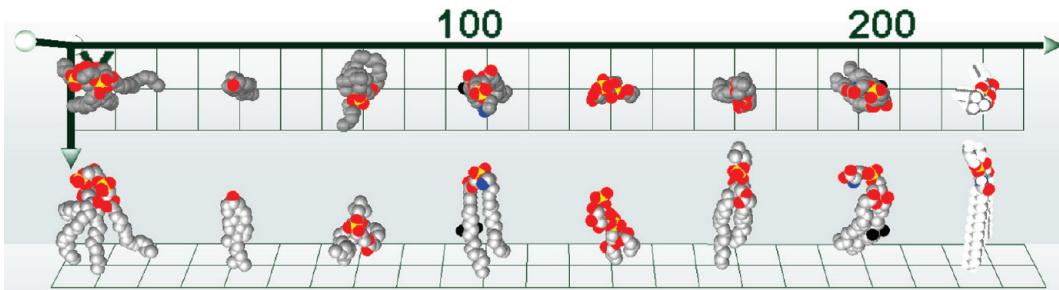


Figure 3. Shape of mitochondrial lipids. The mitochondrial membranes were constructed by use of eight different lipid types: cardiolipin (CL): cdn_exp; phosphatidylethanolamine (PE): pew_exp; phosphatidylinositol (PI): pib_exp; phosphatidylserine (PS): ps2_exp; phosphatidylcholine (PC): psc_exp; cholesterol (Chol): clr_exp; DLPC, dlp_exp; SM, sphingomyelin.^{66,68}

introduced by Kirckpatrick et al. and Černý.^{61,62} This kind of algorithm is applied to the one-dimensional one-zero KP.^{63,64} Cagan also introduced a shape-based annealing method, but it was based on simple two-dimensional shapes and their orientation was restricted by a defined grammar.²⁹ In the case of atomic-structure based shape generation, these constraints are not applicable. Here a geometrical approach is introduced, which simulates the tendency of the membrane to form a consistent bilayer, by focusing on the outer molecular shape only.

The initial state of the membrane is computed by a simple finite random online algorithm which creates an organized layer (similar to AP). The SA process works off-line: the positions of all molecules are known to the algorithm and their reallocation is intended.¹⁸ The workflow of the algorithm follows a regular cooling plan with exceptions: The global state is computed directly after the position change of each lipid so that the scoring function has to be evaluated only for local changes. The global optimum, which is defined by the minimization of free space in the membrane, evolves from several local optima. The computation of the molecule position is done randomly, restricting the mobility to 1 Å. The energy function ε calculates the medial distance between the actual molecule m_0 and the molecules m_i in a defined cutoff range k

$$\varepsilon = \sum_{i=1}^k \text{dist}(m_0 - m_i)/k \quad (6)$$

For the runtime evaluation a closer look at the algorithm is inevitable. The computation steps are summarized as follows

$$O_{SA}(p \times (m \times (2 \times k + n))) = (p \times m \times k) \quad (7)$$

where p is the number of plateau steps or more precisely

$$p = \left(\frac{|\text{initial temperature}-\text{final temperature}|}{\text{cooling rate}} \times \text{plateau length} \right) \quad (8)$$

where m is the number of different molecule positions and/or rotation variations, k is the number of molecules inside the cutoff range, and n is the number of attempts to add new molecules to the membrane. The energy function ε has to be computed twice for all molecules inside the cutoff k , one time prior to and once after positioning. The runtime O_{SA} summarizes to polynomial complexity. This is in agreement with established runtime estimations for SA algorithms generating acceptable results.^{29,61,65}

2.6. Modularity. CmME offers a JAVA based plug-in-interface to create algorithms without the need to know, understand, or edit the original program source code. All required methods for manipulating lipids and proteins including their identifiers, positions, orientations, membrane-layers, microdomains, and single atoms, etc. are implemented.

The source code of a very short exemplary algorithm can be found in the Supporting Information. The documentation of the plug-in-interface and the plug-in-algorithms “The Wanderer” (section 2.5.5) and “Simulated Annealing 90 Degree” as well as the plug-in-tools “Molecule Boxifier” and “Dimension Lister” (all section 3.2) are available on our Web site.

2.7. Reproducibility. Running a membrane algorithm, the generation of random numbers is needed for the computation of lipid positioning, justification, etc. CmME supports user-definable seeds assuring the reproducibility of conformations.

2.8. Availability. CELLMicrocosmos 2.2 MembraneEditor (CmME) is released on GNU General Public License/GPL. Based on Java 6 and Java3D 1.5.2, CmME is Webstart-ready. This program can be directly installed and executed from its Web resource: <http://Cm2.CELLMicrocosmos.org>

3. RESULTS AND DISCUSSION

3.1. Modeling of Outer and Inner Mitochondrial Membranes. The species- and organelle-specific percent lipid ratios within given membranes can be derived from many references. The first application case discusses the modeling of a membrane based on the publication by Hovius et al.⁶⁶ This publication introduces a method to separately compare the lipid composition of the inner and outer mitochondrial membrane. The mitochondrial lipid composition is a good challenge for CmME as the membrane contains cardiolipin, which is predominantly found in the mitochondrial inner membrane, where it is reported to be synthesized.⁶⁷ This phospholipid has a space-consuming structure because it connects two phosphatidylglycerols with a glycerol. The lipid models in PDB format needed for the computation were taken from the HIC-Up database:⁴⁹ cardiolipin (CL): cdn_exp, phosphatidylethanolamine (PE): pew_exp, phosphatidylinositol (PI): pib_exp, phosphatidylserine (PS): ps2_exp and phosphatidylcholine (PC): psc_exp (Figure 3). Each one represents an experimental lipid model extracted from different PDB files. Using these models, the question was addressed, whether CmME supports the computation of realistic lipid ratios and densities operating on simple geometry-based algorithms.

Table 1. Lipid Composition of Outer Mitochondrial Membrane of Rat Hepatocyte^{66a} and Values Obtained by Different Algorithms of CmME

	CmME initial experim.% ⁶⁶	CmME initial values (%)	Linear Placing	Random Placing ^b	Adv. Rand. placing	Adv. Rand. Placing + ^c	Simulated Annealing ^d	Distributor + ^e	Distributor ^f	The Wanderer ^g
CL	9 ± 2	9	9.012	8.805	8.886	8.929	8.9	8.9	8.805	8.91
PC	48 ± 5	49.1	48.7	49.161	49.171	49.074	49.132	49.132	49.161	49.121
PE	31 ± 1.6	31	31.196	31.027	31.043	30.952	31.042	31.042	31.027	30.95
PI	9.9 ± 1.7	9.9	10.052	9.958	9.953	9.987	9.913	9.913	9.958	9.965
PS	1 ± 0.3	1	1.04	1.048	0.984	1.058	1.013	1.013	1.048	1.055
deviation sum ^h		0.8	0.389	0.297	0.288	0.2	0.2	0.2	0.389	0.281
number of lipids		282/295	464/490	425/419	761/751	686/696	686/696	686/696	464/490	851/855
average lipid density (Å ²)		141.84/135.59	86.20/81.63	94.11/95.46	52.56/53.26	58.30/57.47	58.30/57.47	58.30/57.47	86.20/81.63	47/46.78
computation time ⁱ		4s [4078]	1 m [60672]	13s [13343]	39 m 37s [2377344]	3d 20 h 50 m 48s [334248703]	11 m 32s [692750]	47s [47516]	27 m 24s [1644250]	

^a The CmME membrane compositions were tested on a 200 × 200 Å² membrane with standard algorithm values and atom collision only. The values represent the average of both layers. The seed for the random number generation was 22. Special algorithm settings were as follows: ^b Stopped after 1 min, run with visualization. ^c With superb density distribution. ^d Advanced initial state, automatic adding of lipids activated. ^e The absolute lipid numbers were taken from the resulting membrane of the Simulated Annealing Algorithm. ^f The absolute lipid numbers were set to 400 from the beginning. ^g Wander for fellow and dance option, 50 possible missteps, this algorithm is not included in CmME and can be downloaded separately. ^h Based on the CmME initial values. ⁱ Test configuration: Intel Pentium 4; 3 GHz (no HyperThreading); 2 GB Ram (Java Virtual Memory: 1 GB); Windows XP SP3; Java 1.6.0_21; Java3D 1.5.2 fcs (build4); computation without visualization (except ^b).

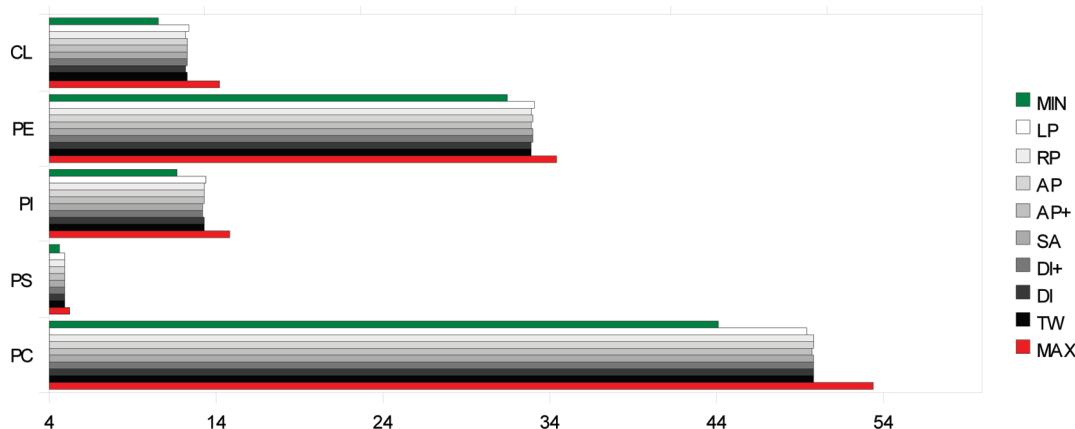


Figure 4. Comparison of percent values and values from CmME algorithms. MIN: minimal %, MAX: maximal %, LP: Linear Placing, RP: Random Placing, AP: Advanced Random Placing, AP+: Advanced Random Placing Superb; SA: Simulated Annealing, DI+: Distributor with SA values; DI: Distributor; TW: The Wanderer.

3.1.1. Modeling an Outer Mitochondrial Membrane. Before defining the initial values, an average value of 100% for the lipid composition has to be reached. Normally, CmME can adjust the percent values of the different lipid types automatically, but this option was avoided here, in order to compute values corresponding to the standard deviation (SD) limits defined by the publication:⁶⁶ For the outer membrane, the sum of the percent values derived from the publication is 98.9%; 1.1% less than 100%. Since PC has a SD value of 5%, the missing 1.1% was added only to PC. The results presented in Table 1 were computed using the atomic outer shape of the lipids, providing a higher accuracy (section 2.3).

Comparing the model with empirical data shows that the different computed values range within limits defined by the maximal and minimal values based on the original report (Figure 4). Focusing on phosphatidylserine, which has the lowest SD value, the original publication gives a value of 1 ± 0.3% derived from three experiments. The lowest value was computed by the AP algorithm with 0.984%, the highest by the AP+ with 1.058%. Even the simplest and fastest algorithm, LP, does not exceed the limits with 1.04%.

Focusing on the sum of deviations from the initial values, it is not surprising in light of the long runtime and lipid redistribution functions that best results were achieved with the DI+ and the SA algorithm with a deviation sum of only 0.2. Closest to this result were the algorithms providing the highest density, the AP+ with 0.288 and TW with 0.281.

The difference in packing quality of the CmME algorithms becomes apparent when comparing the lipid space values derived from the results shown in Table 1. LP as the simplest algorithm was unable to achieve realistic lipid densities. The algorithms DI, RP, and AP with regular settings produced membranes with a low density of ~85 to 95 Å². This is closer to realistic values but still not satisfactory. The SA algorithm with ~58 Å² produced results close to experimental values. This is an interesting fact, because it is the only CmME algorithm tentatively utilizing a natural paradigm, albeit in a very simplified manner, namely by considering the outer atomic shapes of the lipid models to compute the simulated annealing process. As a trade off and in contrast to all other CmME algorithms, the program needed nearly four days to compute the model membrane. The DI was used to reproduce

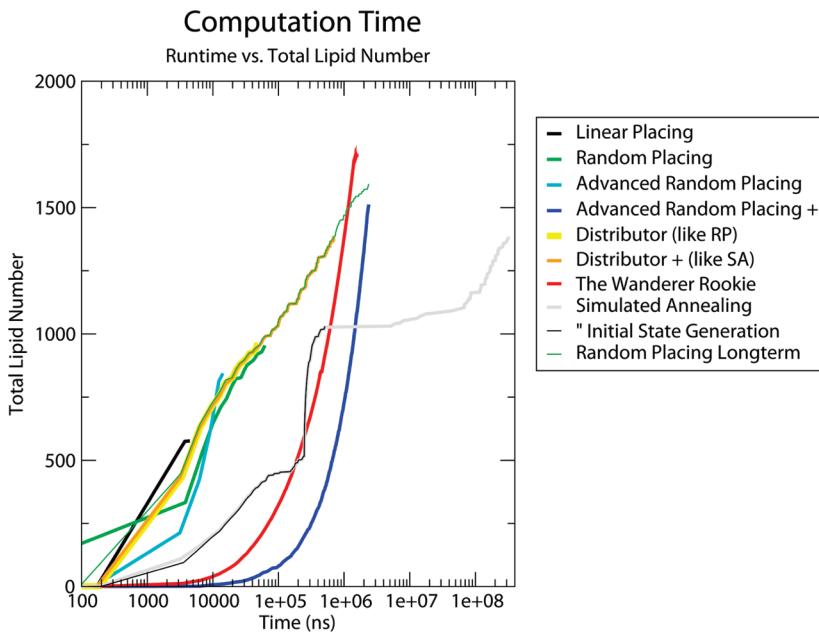


Figure 5. The computation time of the algorithms vs the total lipid number. The time is presented on a logarithmic axis. See Table 1 for the precise values.

the lipid density of SA algorithm. By setting the absolute lipid number to 686 and 696 vice versa, it needed about 12 min to reproduce the lipid packing (Table 1, DI+). The results from the two other algorithms showed that it is possible to generate an even denser packing in acceptable time: AP+ with ~ 53 and TW with $\sim 47 \text{ \AA}^2$ each lipid. To achieve a specified input, the DI can be used to define the exact lipid area desired. The algorithm automatically deletes excessive lipids until the lipid/space ratio is reached.

The computation times of the CmME algorithms are compared in Figure 5. Additionally, the Initial State Generation time of SA is shown, and a RP (Longterm) was run to be comparable to the runtime of AP+. The RP Longterm shows a slightly better area per lipid result than AP+, which divides the membrane into different subproblems. In addition, RP Longterm shows similar runtime behavior as DI. The reason is that DI mainly uses the methods of RP for the membrane computation (section 2.5.3.). Because the regular RP was run using visualizations, it starts earlier than all other algorithms and becomes slower than DI and RP Longterm. Obviously the time minimizing CmME when starting the computation takes 100–200 ns. The two curved segments in the beginning of SA show the generation of the inner and outer membrane layer. LP is the fastest algorithm with the poorest results, while TW generates a dense membrane in appropriate time. As discussed above (section 1.5), a lipid density value of ~ 60 – 70 \AA^2 is representative for a phospholipid bilayer. Referring to Nagle and Tristram-Nagle,⁴⁶ for example, a DPPC lipid at 20 °C in the gel phase would occupy $\sim 47 \text{ \AA}^2$ area per lipid. This was realized here by using TW. The same lipid would occupy 56–72 \AA^2 at 50 °C in the fluid phase, which was achieved by most algorithms discussed here, including DI+ (Figure 5).

3.1.2. Incorporation of Cholesterol. In addition to the lipids used above, sterols needed to be incorporated to achieve a membrane packing close to reality. Based on the publication by Levy and Saudner, 10% of the lipids in a rat hepatocyte

mitochondrion are sterols.⁶⁸ The values defined in Table 1 were reduced by 10% each, resulting in the following percent distribution: CL: 8.1, PE: 27.9, PI: 8.91, PS: 0.9, PC: 44.19, and cholesterol (Chol): 10%. The experimental PDB file clr_exp was retrieved from the HIC-Up database (Figure 3). The membrane was generated using TW (see Table 1 for settings, Table 2 and Figure 6 for results) and showed a molecule packing similar to the “umbrella model”.⁶⁹ Theoretical considerations predict that cholesterol tends to be shielded by the larger head groups of phospholipids to avoid energetically disadvantageous contact with water molecules surrounding the membrane leaflets. Therefore, the surface area of a cholesterol-containing membrane is less than the sum of the individual lipid areas (cholesterol condensing effect).⁴⁷ In contrast to this aspect, the average area per lipid ratio decreases now to 43.5 \AA^2 due to the small size of cholesterol. From the 914 lipids in the outer leaflet, 92 were sterols. Quantification of the headgroups in the van der Waals visualization (Figure 7.4) showed that 17 out of 92 (ca. 15%) were completely hidden beneath the larger molecules. The yellow areas represent the cholesterol in the upper membrane leaflet of covalent radii mode (Figure 7.3) and the blue areas the cholesterol using the van der Waals radii (Figure 7.2). The pure yellow areas in Figure 7.4 are those cholesterol molecules completely shielded by headgroups of larger phospholipids. This aspect can also be observed by focusing on a single cholesterol (marked in Figure 7.4) and its surrounding lipids from top and west perspectives (Figure 7.6). Comparing the simple shape-based (Figure 7.1) with the complex and time-consuming van der Waals radii visualization (Figure 7.2) reveals that the amount of visible lipids is nearly the same. Therefore, the shapes represent an adequate preview option to visualize the umbrella model.

3.1.3. Addition of an Inner Mitochondrial Membrane. For the inner mitochondrial membrane, 97.3% of the lipids are defined by the original average values given in the publication with CL 13.7 ± 2.3 , PC 39.2 ± 0.9 , PE 43.4 ± 1.1 , and PI 1.0 ± 0.2 . The

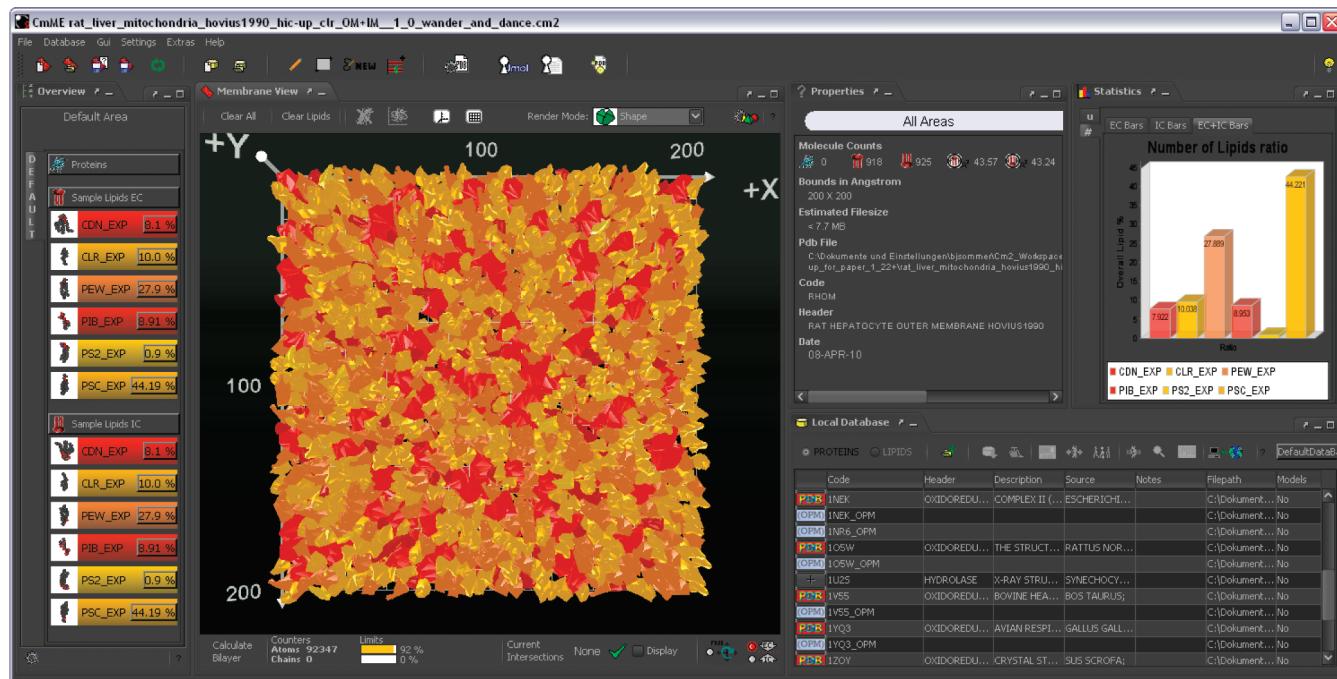


Figure 6. The outer mitochondrial membrane created from experimental lipid data from the HIC-Up database (Table 2). The six lipid types are indicated by different colors. The initial percent ratios are shown on the left side, the resulting percent ratios by the diagram on the right side. Left of this diagram, information about the actual membrane model is indicated, as are data on the average lipid density and the molecule numbers. The bottom shows the local database for proteins and the center view the $200 \times 200 \text{ \AA}^2$ membrane model in shape mode. TW was used for the membrane generation.

Table 2. Lipid Compositions with Cholesterol in Outer and Inner Mitochondrial Membrane of Rat Hepatocyte^{66,68a}

	initial values outer membrane (%) ^{66,68}	The Wanderer	initial values inner membrane (%) ^{66,68}	The Wanderer
Chol	10	10.038	10	10.044
CL	8.1	7.922	12.681	12.599
PC	44.19	44.221	36.252	36.3
PE	27.9	27.889	40.131	40.088
PI	8.91	8.953	0.936	0.969
PS	0.9	0.977	-	-
number of lipids		918/925		568/567
average lipid density (\AA^2)		43.57/43.24		70.42/70.54

^a The CmME membrane compositions were tested on a $200 \times 200 \text{ \AA}^2$ membrane with standard algorithm values and atom collision mode only. The values represent the average of both layers. The seed for the random number generation was 22. Special algorithm settings: wander for fellow and dance option.

remaining 2.7% were proportionally added to the other values, in order to prevent the exceeding of SD values, resulting in CL 14.09, PC 40.28, PE 44.59, and PI 1.04.⁶⁶ Finally, the cholesterol was added as discussed in 3.1.2 (see Table 2 for the results). The distance between the inner and outer membrane of a mouse liver mitochondrion was found to be around $\sim 100 \text{ \AA}$, based on tomographic data downloaded from the Cell Centered Database (ID:3864).^{70,71}

3.2. Application to the Classical 2D-Knapsack-Problem. In section 1.4, the 2D-KP was defined as an appropriate definition for the LPP with regular three-dimensional shapes. The former section showed that the lipid packing algorithms of CmME performed well regarding the runtime and the achieved area per lipid values. Here, a set of CmME algorithms was used to examine the application to orientation-restricted 2D packing approaches. In this case, the orientation of the rectangular

boxes was restricted to 90° steps. This is an important restriction for many of the classical two-dimensional packing and/or KP solutions.^{17–19} In addition, the optimal solution can be easily defined with this approach since the optimum is the ideal membrane area ratio of 100%.

The lipids CL, PC, PE, PI, and PS were “boxed” by the optional CmME Molecule Boxifier plug-in-tool (section 2.6), which shifts all atoms to the edges of the molecules’ bounding box. Rectangular boxes were created as a result. In contrast to the former section, only shape collision was used in this mode, because the size of the outer atoms appeared irrelevant and a closed surface was needed. The algorithms LP, AP, and TW provide the option to rotate lipids only in 90° steps along the Y-axis. The original SA algorithm does not support this option. Therefore, a special version was created for this problem (Simulated Annealing 90 Degree: SA(90°)). The optional

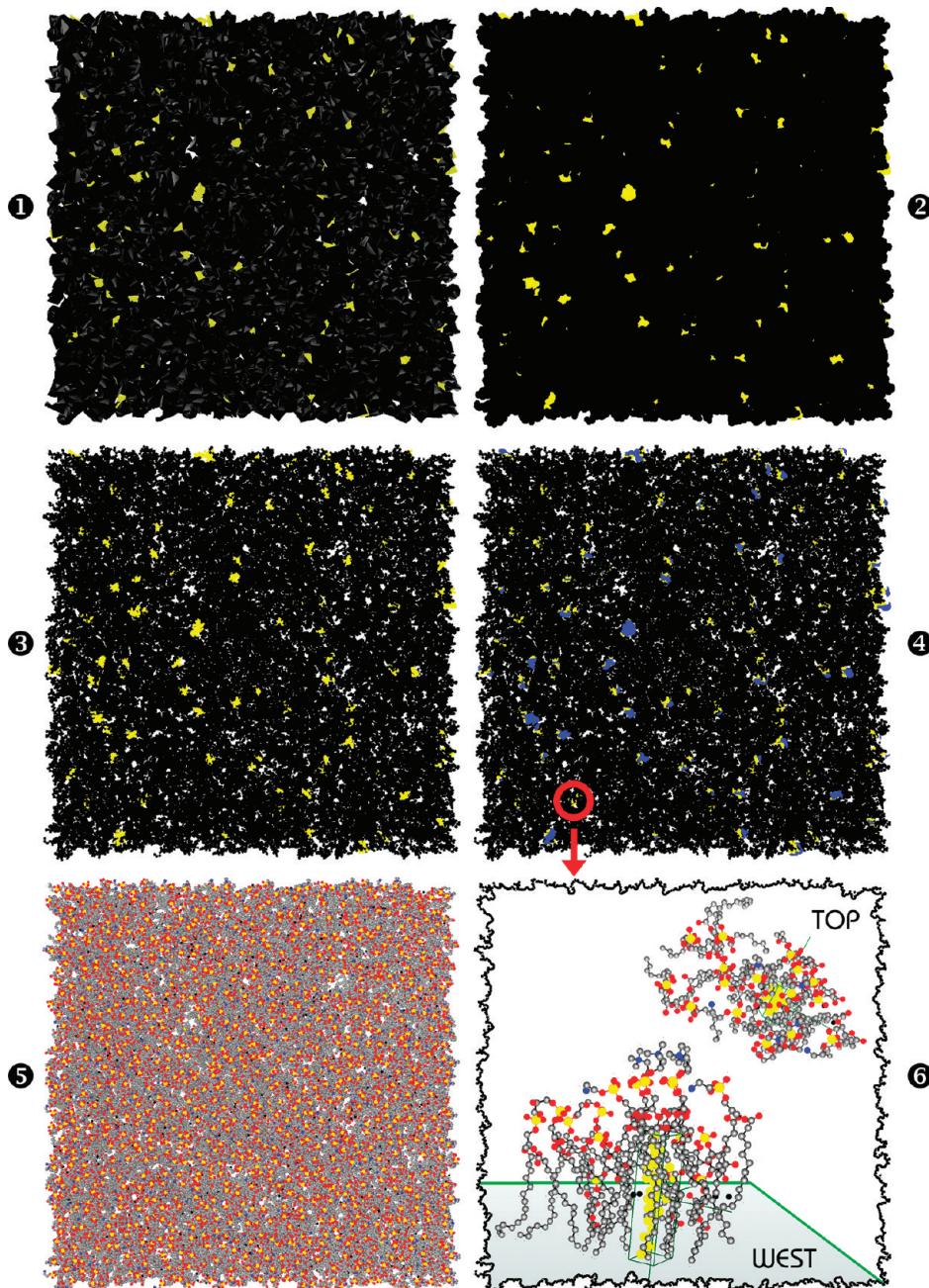


Figure 7. Different visualization modes of a membrane containing 10% cholesterol (colored yellow): (1) shapes; (2) van der Waals radii; (3) covalent radii; (4) combination of 2 (blue areas) and 3 (yellow areas): the yellow headgroups are completely shielded by the larger headgroups of other lipid types; (5) atom type dependent coloring; (6) the molecular environment of the cholesterol marked in (4) shown from top and west perspective.

Dimension Lister plug-in-tool was used to determine the membrane area ratio. For this purpose, the rectangular areas of every lipid were summed up and compared to the overall size of the membrane. Table 3 shows the results of the six different computations. The gray column indicates an AP run in superb mode without rotation restriction to 90°. The second worst membrane area ratio of 51.6% shows that the optimized packing of rectangular boxes needs an orientation restriction, although this algorithm generates the second highest density applied to the LPP (Table 1). The standard AP(90°) achieved the least density, while the LP with 55.26% performed better. This indicates that the placement of boxes row by row outperforms

regular random placing approaches, also in opposition to the results from Table 1. The heuristic approximation algorithm SA(90°) achieved a much better value with nearly 75%. The best solution was achieved by TW(90°) with nearly 80% (Figure 8). The resulting lipid distribution percentages for TW are CL: 8.705%, PC: 49.107%, PE: 31.027%, PI: 10.045%, and PS: 1.116% which are inside the SD values stated in Table 1. Obviously, good solutions were found for OPT_LP_RATIO and OPT_LPP_2D_AREA.

Summing up the results from Table 3, the quality of the LPAs was verified by using a classical 2D-KP approach. It shows that CmME is able to produce good solutions for this as well as the

Table 3. Lipid Density Values in a Boxified Monolayer Based on the Outer Mitochondrial Membrane of Rat Hepatocyte^{66a}

	Adv. Placing (90°) ^b	Rand. Placing (90°) ^{c,d}	Linear Placing	Adv. Placing+ (90°) ^{b,c}	Simulated Annealing 90 Degree ^e	The Wanderer (90°) ^f
Number of Lipids	256	289	311	358	419	448
Average Lipid Density (\AA^2)	156.25	138.40	128.61	111.73	95.46	89.28
Membrane Area Ratio (%) ^g	45.37	51.60	55.28	63.57	74.35	79.65
Computation Time ^h	2s [2078]	8m 49s [529703]	1s [1546]	6m 54s [414953]	3h 14m 14s [11654046]	25s [25485]

^a For algorithmic and computational configurations see Table 1. Each lipid was boxified by the optional CmME Molecule Boxifier Tool, and a $200 \times 200 \text{\AA}^2$ monolayer was created by using the shape collision mode. Special algorithm settings were as follows: ^b Rotations along the Y-axis in 90° steps, with no tilts or shifts. ^c With superb density distribution. ^d The gray column indicates that this is not a classical 2D-KP application. ^e Special plug-in-algorithm with rotation restricted to 90° steps along the Y-axis and no rotations along the X-/Z-axis, advanced initial state, automatic addition of lipids activated. ^f Wander for fellow and dance option, rotation along the Y-axis in 90° steps, 50 possible missteps. ^g Percentages from total membrane area of 40000 \AA^2 . ^h See Table 1 for details.

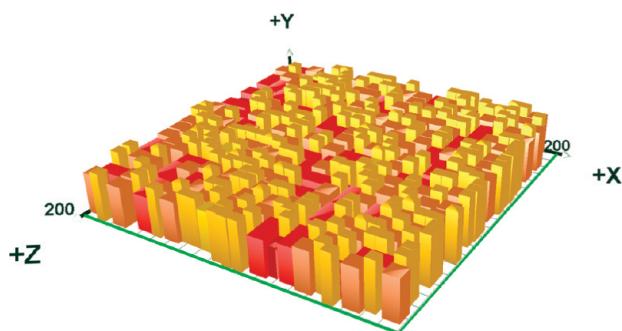


Figure 8. A classical 2D-KP membrane generated with The Wanderer (see Table 3 for details). The lipid height is not taken into account during computation, only the width and length are relevant.

much more complex LPP of section 3.1. While the classical 2D-KP approach aims at maximization of the membrane area ratio, LPP tries to minimize the area per lipid. It is shown that CmME algorithms are able to achieve higher areas per lipid than for regular application cases needed. Therefore, the optimal solution of a 2D-KP is not needed to solve the regular LPP discussed in the previous section, because the realistic average area per lipid would be exceeded. Using DI, publication-based lipid distribution values can be defined by the user for most application cases.

3.3. Placement of a Membrane-Spanning and a Single α -Helix-Anchored Protein of Mitochondria. Protein placement prediction (OPT_PPP_POS) is implemented in different approaches such as ProtSqueeze,⁷² PDB_TM, and OPM. While

ProtSqueeze is a MPA, combining pre-equilibrated layers with a protein, the latter two databases can be used to implement PPA and/or MPA, as it was done in CmME.

The reaction center of bacterial photosystem II was the first membrane structure for which a high-resolution structure became available in 1985.⁷³ In the mid 1990s the atomic structure of several membrane proteins were resolved including that of bovine cytochrome c oxidase.⁷⁴ These structures are available from databases such as PDB_TM which contain only transmembrane proteins and the OPM database which additionally contain some peripheral proteins and membrane-active peptides. Some PDB entries are found in both databases. PDB_TM contains 1336 and OPM 1023 entries; 329 PDB entries match. Thus, CmME can semiautomatically place 2030 of 71138 structures (~2.85%) into the membrane (reference date ninth February 2011).^{14,15,6}

Cytochrome c oxidase (CcO) (EC 1.9.3.1) is a component of the respiratory electron transport chain and is used as a marker protein specifically for the inner mitochondrial membrane of eukaryotes. CcO activity was also used to characterize the mitochondrial membrane fractions prior to lipid analysis in the study used for this work.⁶⁶ The PDB model 1 VS5 was obtained from the CcO of *Bos taurus*.^{43,74} Binding of CcO to CL, PC, and PE has been reported.⁷⁵ Therefore, it was concluded that contact to Chol, PS, or PI should not occur while placing the protein into the model membrane from section 3.1.1. (Figure 9 and 11 left). The high resolution structure of CcO resolved the position of 8 phospholipid molecules. This knowledge allows for defining the precise position of CcO in the membrane.⁷⁶ The transmembrane domain which consists of a bundle of amphiphilic α -helices is

highly conserved among species as revealed by hydropathy plots and amino acid analysis.⁷⁶ Lipophilic amino acid residues prevail in a distance of ± 1 nm from the bilayer center and still are overrepresented in a distance ± 2 nm. The ratio of the diameter of CcO (perpendicular to the membrane) and the helical bundles is 2.6 in the high resolution structure and 2.3 in the automatic placement, indicating a slightly higher membrane insertion in the automatic placement of CmME routine.

Monoamine oxidase (MAO) (EC 1.4.3.4) catalyzes the oxidative deamination of some neurotransmitters, e.g. serotonin and dopamin, and is a marker enzyme specifically for the outer mitochondrial membrane.⁶⁶ The PDB model 1OSW was obtained for monoamine oxidase from the outer mitochondrial membrane of *Rattus norvegicus*.⁷⁷ Daum reported the binding of PL to monoamine oxidase from rat liver.⁷⁵ Therefore the environment to place the protein into is known, only Chol in our model membrane had to be omitted (Figure 10 right). MAO

binds to the membrane through its hydrophobic C-terminal α -helix of six turns. The location of the last five amino acids Lys-Lys-Leu-Pro-Cys was structurally not resolved, but the positively charged Lys residues are suggested to interact with the negatively charged membrane surface on the trans-side of the bilayer.⁷⁷ Positively charged residues are also found on the globular protein surface next to the protruding α -helix stabilizing the interaction of the protein on the cis-side. When investigating the insertion upon automatic placement by CmME, the extension of the extra membrane region of MAO (perpendicular to the membrane) relative to the bilayer inserted portion is below 1, which is slightly less than determined from the domain assignment by the authors in the original publication.⁷⁷

3.4. Computation of Extreme Densities. The next investigations addressed the question, if CmME is able to realize extreme lipid densities, like those found in Ghosh et al.³⁴ All the following computations were carried out on a $100 \times 100 \text{ \AA}^2$ membrane. For the upper area limit for PC of 134 \AA^2 , the DI algorithm can be used. The smallest area is reported for cholesterol with 38.3 \AA^2 and may be taken as the lower limit. Again the experimental PDB structure clr_exp from the HIC-Up database was used. By using the AP algorithm, 38.46 \AA^2 , by using the AP+ algorithm, 22.42 \AA^2 , and by using TW, 16.42 \AA^2 was achieved (see Table 1 for algorithm properties). The results prove that DI can be used to generate a membrane nearly equivalent to the experimental value with an area per lipid of 38.31 \AA^2 .

For DLPC an area of 26 \AA^2 is reported under very extreme experimental conditions.³¹ This peculiar value was considered interesting. Using the structural model dlp_exp with AP+, an area of 119.04 \AA^2 was realized, using TW, an area of 131.57 \AA^2 . Obviously, the algorithms failed in this special case to compute an adequate placing. Examination of the 3D structure of the lipid model allowed identification of the problem: The retrievable structural model adopts a very space-consuming spatial structure, which does not allow a closer packing without changing the structure. Using ideal coordinates instead of the experimental derived values could alleviate this problem (Figure 3). In section 3.6 a lipid model having a similar structure like DLPC, namely DPPC, was used to achieve a dense packing compatible to MD simulations.

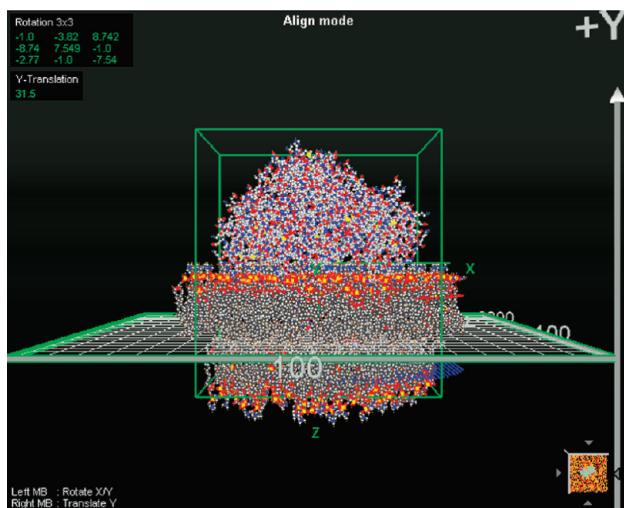


Figure 9. Automatic placement of cytochrome c oxidase. The Alignment mode showing 1OSW with the OPM layers marking the outer (red) and inner (blue) membrane leaflet and the semitransparent bilayer.

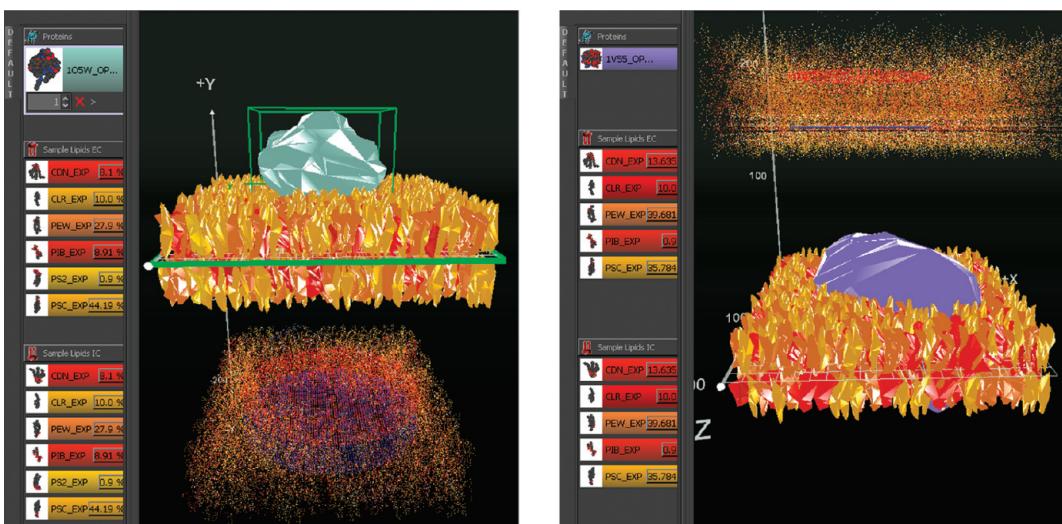


Figure 10. The outer membrane with 1OSW (left) and the inner membrane with 1VSS (right) assembled. OPM data has been used for the semiautomatic placement.

Table 4. Area-Dependent Constraints in Lipid Composition of Model Membrane^a

	experim.% ⁶⁶	initial values (%)	50 × 50 Å ²		100 × 100 Å ²		200 × 200 Å ²	
			Linear Placing	The Wanderer	Linear Placing	The Wanderer	Linear Placing	The Wanderer
CL	9 ± 2	9	6.458	7.705	8.725	8.696	9.009	8.91
PC	48 ± 5	49.1	45	49.444	49.273	49.104	48.705	49.121
PE	31 ± 1.6	31	29.167	30.773	29.684	30.946	31.196	30.949
PI	9.9 ± 1.7	9.9	12.917	9.879	10.864	10.231	10.049	9.965
PS	1 ± 0.3	1	6.458	2.198	1.454	1.023	1.04	1.055
number of lipids		16/15	46/45	65/73	197/194	287/301	826/845	
average lipid density (Å ²)		156.25/166.66	54.34/55.55	153.84/136.98	50.76/51.54	141.84/135.59	47/46.78	
computation time		<1s[218]	29s [29171]	<1s[641]	2 m 37s [157687]	2s [2937]	28 m 25s[1705766]	

^aLipid compositions in outer mitochondrial membrane of rat hepatocyte⁶⁶ computed on a 50 × 50, 100 × 100, and 200 × 200 Å² membrane patch. For further details see Table 1. The italicized entries indicate results that exceed the limits defined by %.

Table 5. Lipid Composition of Plasma Membrane with and without Cholesterol-Sphingomyelin-Raft of Mouse Hepatocyte^{66a}

	initial values surrounding Membrane (%) ^(81,79)	Adv. Random Placing+ ^b	Adv. Random Placing+ ^c	initial values Membrane Raft (%) ^(81,79)	Adv. Random Placing+ ^b	Adv. Random Placing+ ^d
Chol	21	21.055	21.123	46	46.013	46.029
PC	36.972	36.933	36.78	21.06	21.089	21.029
PE	18.012	17.998	18.021	11.34	11.285	11.328
PI	4.345	4.389	4.431	4.86	4.88	4.948
PS	3.002	2.909	2.806	5.4	5.359	5.339
SM	16.669	16.716	16.839	11.34	11.373	11.328
deviation sum		0.292	0.704		0.191	0.233
Number of Lipids		1012/ 1016	337/ 340		1152/ 1143	763/ 773
Average Lipid Density (Å ²)		39.52/ 39.37	43.18/ 42.80		34.72/ 34.99	33.35/ 32.91

^aThe CmME membrane compositions were tested on a 200 × 200 Å² membrane with Adv. Rand. Placing+ and atom collision only (Table 1). The values represent the average of both layers. ^bBilayer patch of 200 × 200 Å². ^cOuter Border of the 200 × 200 Å² bilayer patch without Ø180 Å raft area. ^dØ180 Å raft area.

Another critical limitation may be seen when investigating the data of Table 4: When using a very small membrane of 50 × 50 Å², the LP algorithm computed false values for PE and PS. On one side there were ten lipids and on the other side only 8. The very low number of lipids did not allow the realization of the specified compositions. By using TW, the results slightly exceeded the limits defined by SM only for PS. For a 100 × 100 Å² membrane the Linear Placing gave nearly acceptable results, except for PS.

It is concluded that there are two cases which need modified algorithms or other more compact and optimized structures to

realize adequate packing, i.e. when (i) a very high lipid density is needed (OPT_LPP_LOW_AREA) and/or (ii) the used lipid model features an unfavorable atomic conformation.

3.5. Modeling of a Cholesterol Sphingomyelin Lipid Raft-Containing Plasma Membrane. The existence of lipid rafts in mitochondria is still under controversial discussion.^{78,79} Therefore, a cholesterol-sphingomyelin raft was constructed based on data for plasma membrane of mouse hepatocytes.⁸⁰ The size of the raft was set to a diameter of 180 Å, which lies inside the observed size for lipid rafts.^{81,82} The experimental values were

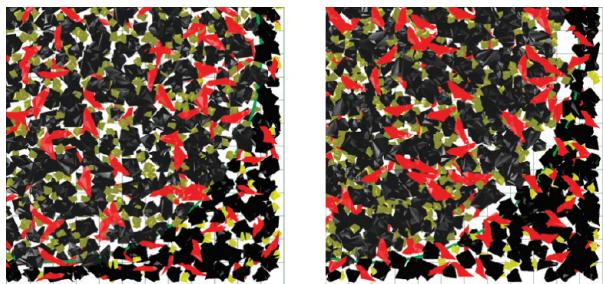


Figure 11. Comparison of a Chol/SM-Raft generated with DI (left) and TW (right). Chol: yellow, SM: red, PC: black.

taken from the mouse wild type and adjusted as follows for the surrounding membrane: Chol 21%, PC 36.972%, PS 3.002%, PI 4.345%, PE 18.012%, sphingomyelin (SM) 16.669% and for the inner lipid raft: Chol 46%, PC 21.06%, PE 11.34%, PI 4.86%, PS 5.4%, SM 11.34%. Because there is no experimental or ideal data available for the structure of sphingomyelin in the HIC-Up database, a constructed model from Chemistry Molecular Models was used (Figure 3).⁵¹ All hydrogen atoms were removed from the original SM model, since the HIC-UP models do not contain any hydrogen atoms.

Three $200 \times 200 \text{ \AA}^2$ membrane patches were computed by using AP+. The first patch contains only the plasma membrane composition, the second patch only the raft composition, and the third model combines the raft with the outer membrane. Table 5 compares the average lipid densities of the different membrane patches. Obviously, the difference between both raft compositions with ca. 1.7 \AA^2 is quite low. Comparing the values of the outer membrane, the difference with ca. 3.5 \AA^2 is higher. The perimeter of the outer patch borders has a negative impact on the area per lipid. The outer membrane surrounding the raft has an outer and an inner border; therefore, the area per lipid performance is inferior. However, all three membranes behave as expected concerning the dependence of the area per lipid on the cholesterol ratio.

Figure 11 shows this raft generated with AP+ (left) and with TW (right). TW (in this case with a step size of 0.1) again generates a higher lipid density, but obviously this northwest wandering approach (section 2.5.5) is outperformed by random placing in terms of a smooth lipid distribution along the south-east border of the raft.

3.6. Using CmME-Modeled Membranes for MD Simulations. As an additional application, and a quality-check for the PDB membrane generated with CmME, a bilayer membrane was modeled comprising 512 DPPC lipids (256 per monolayer) and then used as a starting structure for an MD simulation. Here, the GROMACS MD program¹⁶ was used together with lipid forcefield parameters consistent with the Gromos96 (45a3) forcefield.^{83,84} It should be noted that the CmME membrane could have been also used with any other MD engine and forcefield. The desired initial lateral lipid density of $\sim 68 \text{ \AA}^2$ per lipid within a bilayer patch of about $13 \times 13 \text{ nm}^2$ was achieved using the DI algorithm. The membrane was exported to a GROMACS compatible PDB format via the PDB Properties menu.

For additional comparison a second approach was used, which had been implemented and successfully used earlier.¹² In brief, a single lipid conformer was copied onto an equidistant grid with a spacing of 0.7 nm. The resulting monolayer is mirrored to create

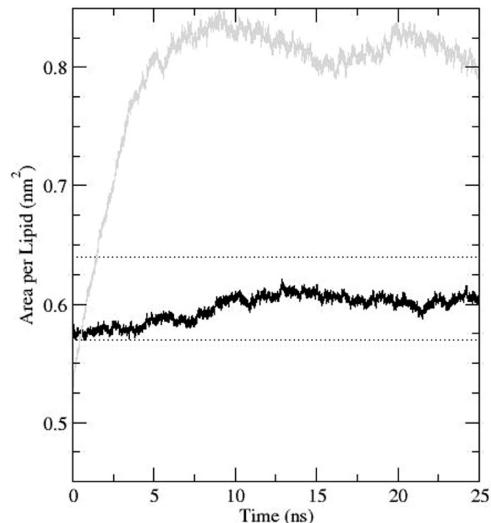


Figure 12. The area per lipid for the CmME DI based system (black line) in comparison to a system created via a simple grid-like placement (gray line). The bilayer systems consisting of 512 lipid molecules were simulated for 25 ns in explicit solvent using GROMACS-4.5.3. The range of experimental reference values is indicated by the dashed lines.⁴⁶ The later system is showing oscillations and is not reaching equilibrium within 25 ns, while the DI based system perfectly resembles experimental observations.

a symmetric bilayer. Chemical Computing Groups (CCG) Scientific Vector Language (SVL) was used with MOE-2010 to carry out these steps.¹³

The two identical systems of 20992 molecules (87040 atoms) consist of two layers of 256 DPPC lipids each forming a bilayer and 20480 SPC waters. The only difference lies in the method of how the bilayers were created, yielding a different initial conformation. All other settings were identical.

The dynamics of the membrane patch was simulated for 25 ns with a time step of 2 fs. Neighbor searching was performed every 10 steps. The PME algorithm was used for electrostatic interactions with a cutoff of 1.0 nm. A reciprocal grid of $42 \times 42 \times 28$ cells was used with fourth order B-spline interpolation. A twin-range van der Waals cutoff (1.0/1.4 nm) was used, where the long-range forces were updated during neighbor searching. Temperature coupling was done with the V-rescale algorithm, coupling lipid, and solvent separately to a temperature of 323 K with a coupling interval of 0.1 ps, enabling a direct comparison to experimental data.⁴⁶ Pressure coupling was done with the Parrinello-Rahman algorithm with a reference pressure of 1 bar and a coupling interval of 4 ps using a semi-isotropic approach decoupling the xy-lipid-plane from the z-direction.

The DI based system (Figure 12) remained with its area per lipid throughout the whole simulation within the range of the experimentally derived reference.⁴⁶ It took approximately 10 ns until the system could be considered fully equilibrated. The system based on the simple grid approach did not reach equilibrium within the given experimental range. Its highly ordered initial conformation (Figure 13) made it impossible to reach this state within the simulated 25 ns. The system also exhibited some undesired oscillation. Nevertheless, it was possible to equilibrate this system using a more sophisticated simulation protocol involving a series of minimization and constrained MD runs (data not shown).

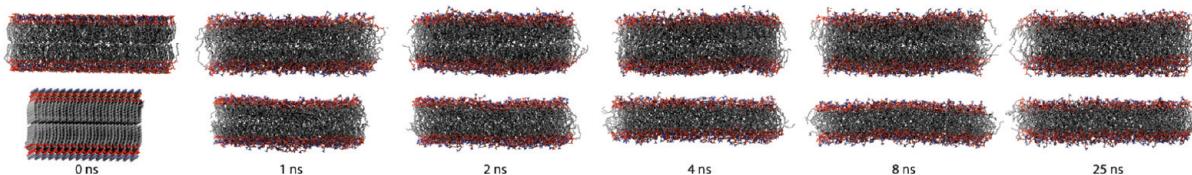


Figure 13. Snapshots along the trajectories of the DI-based system (upper row) and the simple grid-like lipid placement (lower row). The initially well-placed lipids in the Distributor system equilibrate fairly quickly, while the artificially overly organized lipids in the simple grid system cannot reach equilibrium within the simulated time.

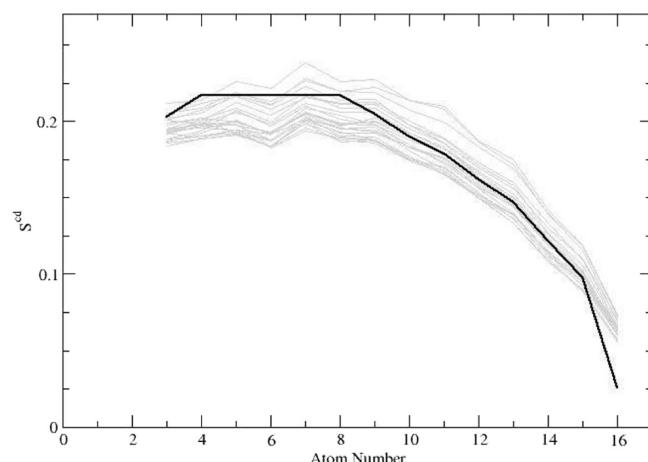


Figure 14. Comparison of the sn-2 alkyl tail order parameters of the DI-based system toward experimentally determined values (black line).⁸⁵ Each of the thin gray lines represents an average of 1 ns along the trajectory of totally 25 ns length. The upper curves correspond to the start of the simulation, the lower to its end.

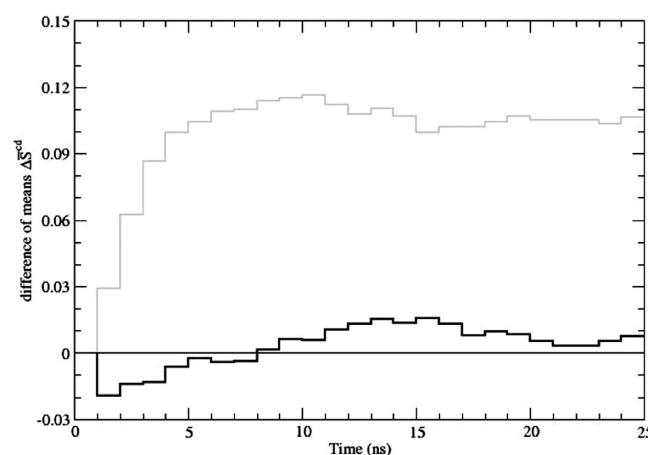


Figure 15. The order parameters for the sn-2 alkyl tails were measured and the difference of means compared to the experimental values.⁸⁵ The comparison is made for the DI-based system (black line) and the simple Grid-based system (gray line). The DI based system performs an order of magnitude better.

To ensure that the lipid system was in fact fully equilibrated, the order parameters for the sn-2 alkyl tails, as they can be measured with nuclear magnetic resonance spectrometry,⁸⁵ were averaged for each nanosecond of the 25 ns long trajectory. The order parameters reflect how well the average conformation of the lipid molecules was reproduced by the simulations.⁸⁴ The DI based system already

showed very good agreement with the experimental data during the early phase of the 25 ns simulation (Figures 14 and 15). The Grid-based system (data only partially shown) gave a much weaker performance and did not reach equilibrium.

4. CONCLUSIONS

CmME offers a set of shape-based strategies to solve membrane packing problems. The Bounded two-and-a-half-dimensional Knapsack Problem (2.5D-BKP) is proposed as an appropriate definition for this kind of problems. The lipid packing problem was solved by modeling heterogeneous membranes combining experimental and ideal lipid models with the knowledge of different publications. Six lipid packing algorithms were introduced, showing that efficient greedy algorithms can compete with long-term simulated annealing approaches in terms of lipid ratios and densities. Published average values were already obtained for relatively small membranes (for membranes $>50 \times 50 \text{ \AA}^2$ with The Wanderer and for membranes $>100 \times 100 \text{ \AA}^2$ with random placing algorithms) and lipid densities (for phospholipids with lipid densities of ca. $>50 \text{ \AA}^2$). Using cholesterol-containing membranes in conjunction with shape-based visualization, a packing behavior similar to the umbrella model was observed. By semiautomatically adopting data from the OPM and PDB_TM databases, the protein packing problem was addressed. A first approach was carried out to import a DPPC-containing bilayer membrane generated with the CmME Distributor algorithm into GROMACS. The equilibrium state was reached during the MD simulation within 10 ns. This was not possible by using a conventional simple grid-based generation approach as the starting membrane. Particularly the future incorporation of external MD simulations, e.g. with GROMACS, and vesicle-like structures with CmME bears high potential for advanced analysis methods.

The CmME program, its documentation, and additional information can be found at <http://Cm2.CELLmicrocosmos.org>.

■ ASSOCIATED CONTENT

Supporting Information. The source code of a very short algorithm compatible with the CmME Plug-in-interface and additional information. This material is available free of charge via the Internet at <http://pubs.acs.org>.

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: bjoern@CELLmicrocosmos.org.

■ ACKNOWLEDGMENT

This work was funded within the Graduate College Bioinformatics (GK635) of the DFG (German Research Foundation). We are grateful to Prof. Ralf Hofestädt for his continuous support, to the Bio-/Medical Informatics Group of Bielefeld University, where this work has been realized, and all other people supporting or participating in this project: <http://team.Cm2.CELLmicrocosmos.org>. Additional thanks goes to Dr. Lars Schäfer of the Department of Biophysical Chemistry in Groningen for his advice and for carrying out the first MD simulations with CmME generated membranes (his simulations are not discussed here). The Cell Centered Database, from which data were used in section 3.1.3., is supported by NIH grants from NCRR RR04050, RR RR08605 and the Human Brain Project DA016602 from the National Institute on Drug Abuse, the National Institute of Biomedical Imaging and Bioengineering, and the National Institute of Mental Health.

■ ABBREVIATIONS:

XD-KP, X-Dimensional KP; Å, angstrom; AP, Advanced Random Placing algorithm; BKP, Bounded KP; Chol, Cholesterol; CL, cardiolipin; Cm2/CmME, CELLmicrocosmos MembraneEditor Project; DI, Distributor algorithm; DPPC, dipalmitoyl phosphatidylcholine; DLPC, dilauroyl phosphatidylcholine; HIC-Up, Heterocompound Information Centre - Uppsala; KP, Knapsack Problem; LP, Linear Placing algorithm; LPA, Lipid Packing Algorithm; LPP, Lipid Packing Problem; MC, Monte Carlo Simulation; MD, Molecular Dynamic Simulation; MPA, Membrane Packing Algorithm; MPP, Membrane Packing Problem; nm, nanometer; ns, nanoseconds; PC, phosphatidylcholine; PDB, Protein Data Bank format; PE, phosphatidylethanolamine; PI, phosphatidylinositol; POPC, 1-palmitoyl-2-oleoyl phosphatidylcholine; PPA, Protein Packing Algorithm; PPP, Protein Packing Problem; PS, phosphatidylserine; RP, Random Placing algorithm; SA, Simulated Annealing algorithm; SD, Standard Deviation; SM, sphingomyelin; TW, The Wanderer algorithm; UKP, Unbounded KP

■ REFERENCES

- (1) Singer, S. J.; Nicolson, G. L. The fluid mosaic model of the structure of cell membranes. *Science (Washington, DC, U.S.)* **1972**, *175*, 720–731.
- (2) Lingwood, D.; Simons, K. Lipid rafts as a membrane-organizing principle. *Science (Washington, DC, U.S.)* **2010**, *327*, 46–50.
- (3) West, B.; Schmid, F. Fluctuations and Elastic Properties of Lipid Membranes in the Tilted Gel State: A Coarse-Grained Monte Carlo Study. *Soft Matter* **2010**, *6*, 1275–1280.
- (4) Yetukuri, L.; Ekeross, K.; Vidal-Puig, A.; Oresic, M. Informatics and computational strategies for the study of lipids. *Mol. Biosyst.* **2008**, *4*, 121–127.
- (5) Jo, S.; Kim, T.; Im, W. Automated builder and database of protein/membrane complexes for molecular dynamics simulations. *PLoS One* **2007**, *2*, e880.
- (6) Berman, H.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T.; Weissig, H.; Shindyalov, I.; Bourne, P. The Protein Data Bank. *Nucleic Acids Res.* **2000**, *28*, 235–242.
- (7) Chemsite Pro; ChemSW: Fairfield, USA, 2010.
- (8) Jo, S.; Lim, J.; Klauda, J.; Im, W. CHARMM-GUI Membrane Builder for Mixed Bilayers and Its Application to Yeast Membranes. *Biophys. J.* **2009**, *97*, 50–58.
- (9) Balabin, I. Membrane Plug-in, Version 1.1; 2010. <http://www.ks.uiuc.edu/Research/vmd/plugins/membrane/> (accessed October 1, 2010).
- (10) Humphrey, W.; Dalke, A.; Schulten, K. VMD: visual molecular dynamics. *J. Mol. Graph.* **1996**, *14*, 33–38.
- (11) Zidar, J.; Merzel, F.; Hodosec, M.; Reboli, K.; Sepic, K.; Maciek, P.; Janezic, D. Liquid-ordered phase formation in cholesterol/sphingomyelin bilayers: all-atom molecular dynamics simulations. *J. Phys. Chem. B* **2009**, *113*, 15795–15802.
- (12) Krüger, J.; Fischer, W. B. Exploring the conformational space of Vpu from HIV-1: a versatile adaptable protein. *J. Comput. Chem.* **2008**, *29*, 2416–2424.
- (13) MOE; Chemical Computing Group: Montreal, Canada, 2010.
- (14) Lomize, M. A.; Lomize, A. L.; Pogozheva, I. D.; Mosberg, H. I. OPM: orientations of proteins in membranes database. *Bioinformatics* **2006**, *22*, 623–625.
- (15) Tusnády, G. E.; Dosztányi, Z.; Simon, I. PDB_TM: selection and membrane localization of transmembrane proteins in the protein data bank. *Nucleic Acids Res.* **2005**, *33*, D275–D278.
- (16) Hess, B.; Kutzner, C.; van der Spoel, D.; Lindahl, E. Gromacs 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation. *J. Chem. Theory Comput.* **2008**, *4*, 435–447.
- (17) Lodi, A.; Martello, S.; Monaci, M. Two-dimensional packing problems: A survey. *Eur. J. Oper. Res.* **2002**, *141*, 241–252.
- (18) Dyckhoff, H. A typology of cutting and packing problems. *Eur. J. Oper. Res.* **1990**, *44*, 145–159.
- (19) Dyckhoff, H.; Finke, U. *Cutting and packing in production and distribution: A typology and bibliography*; Physica-Verlag: Heidelberg, Germany, 1992.
- (20) Naujoks, G. *Optimale Stauraumnutzung*; Gabler Verlag: Wiesbaden, Germany, 1995.
- (21) Hyde, S.; Blum, Z.; Landh, T.; Lidin, S.; Ninham, B. W.; Andersson, S.; Larsson, K. *The Language of shape: the role of curvature in condensed matter-physics, chemistry, and biology*; Elsevier Science Ltd.: Amsterdam, The Netherlands, 1997.
- (22) Outrata, J. V.; Kočvara, M.; Zowe, J. *Nonsmooth approach to optimization problems with equilibrium constraints: Theory, applications, and numerical results*; Springer: New York, USA, 1998.
- (23) López, M.; Still, G. Semi-infinite programming. *Eur. J. Oper. Res.* **2007**, *180*, 491–518.
- (24) Israelachvili, J. N.; Mitchell, D. J.; Ninham, B. W. Theory of self-assembly of hydrocarbon amphiphiles into micelles and bilayers. *J. Chem. Soc., Faraday Trans. 2* **1976**, *72*, 1525–1568.
- (25) Marsh, D. Intrinsic curvature in normal and inverted lipid structures and in membranes. *Biophys. J.* **1996**, *70*, 2248–2255.
- (26) Feng, Y.; Jernigan, R. L.; Kloczkowski, A. Orientational distributions of contact clusters in proteins closely resemble those of an icosahedron. *Proteins* **2008**, *73*, 730–741.
- (27) Pan, L.; Martin-Vide, C. Solving multidimensional 0–1 knapsack problem by P systems with input and active membranes. *J. Parallel Distr. Com.* **2005**, *65*, 1578–1584.
- (28) Kellerer, H.; Pferschy, U.; Pisinger, D. *Knapsack problems*; Springer Verlag: Berlin, Germany, 2004.
- (29) Cagan, J. Shape annealing solution to the constrained geometric knapsack problem. *Comput. Aided Des.* **1994**, *26*, 763–770.
- (30) Bennett, W. F.; MacCallum, J. L.; Hinne, M. J.; Marrink, S. J.; Tielemans, D. P. Molecular view of cholesterol flip-flop and chemical potential in different membrane environments. *J. Am. Chem. Soc.* **2009**, *131*, 12714–12720.
- (31) Peters, R.; Beck, K. Translational diffusion in phospholipid monolayers measured by fluorescence microphotolysis. *Proc. Natl. Acad. Sci. U.S.A.* **1983**, *80*, 7183–7187.
- (32) O'Leary, T. J. Lateral diffusion of lipids in complex biological membranes. *Proc. Natl. Acad. Sci. U.S.A.* **1987**, *84*, 429–433.
- (33) Hui, S. W.; Cowden, M.; Papahadjopoulos, D.; Parsons, D. F. Electron diffraction study of hydrated phospholipid single bilayers. Effects of temperature, hydration and surface pressure of the “precursor” monolayer. *Biochim. Biophys. Acta* **1975**, *382*, 265–275.
- (34) Ghosh, D.; Williams, M. A.; Tinoco, J. The influence of lecithin structure on their monolayer behavior and interactions with cholesterol. *Biochim. Biophys. Acta* **1973**, *291*, 351–362.

- (35) Peitzsch, R. M.; Eisenberg, M.; Sharp, K. A.; McLaughlin, S. Calculations of the electrostatic potential adjacent to model phospholipid bilayers. *Biophys. J.* **1995**, *68*, 729–738.
- (36) Hulbert, A. J.; Else, P. L. Membranes as possible pacemakers of metabolism. *J. Theor. Biol.* **1999**, *199*, 257–274.
- (37) Jójárt, B.; Martinek, T. A. Performance of the general amber force field in modeling aqueous POPC membrane bilayers. *J. Comput. Chem.* **2007**, *28*, 2051–2058.
- (38) Wassall, S. R.; Stillwell, W. Polyunsaturated fatty acid-cholesterol interactions: domain formation in membranes. *Biochim. Biophys. Acta* **2009**, *1788*, 24–32.
- (39) Shipley, G. G. In *Biological Membranes*; Chapman, D., Wallach, D. F. H., Eds.; Academic Press: New York, USA, 1973; pp 1–86.
- (40) Potamitis, C.; Zervou, M.; Katsiaras, V.; Zoumpoulakis, P.; Durdagı, S.; Papadopoulos, M. G.; Hayes, J. M.; Grdadolnik, S. G.; Kyrikou, I.; Argyropoulos, D. Antihypertensive Drug Valsartan in Solution and at the AT1 Receptor: Conformational Analysis, Dynamic NMR Spectroscopy, in Silico Docking, and Molecular Dynamics Simulations. *J. Chem. Inf. Model.* **2009**, *49*, 726–739.
- (41) Shelley, J. C.; Shelley, M. Y.; Reeder, R. C.; Bandyopadhyay, S.; Moore, P. B.; Klein, M. L. Simulations of phospholipids using a coarse grain model. *J. Phys. Chem. B* **2001**, *105*, 9785–9792.
- (42) Izvekov, S.; Voth, G. A. Multiscale coarse-graining of mixed phospholipid/cholesterol bilayers. *J. Chem. Theory Comput.* **2006**, *2*, 637–648.
- (43) Shinoda, W.; DeVane, R.; Klein, M. L. Coarse-grained molecular modeling of non-ionic surfactant self-assembly. *Soft Matter* **2008**, *4*, 2454–2462.
- (44) Lu, L.; Voth, G. A. Systematic coarse-graining of a multi-component lipid bilayer. *J. Phys. Chem. B* **2009**, *113*, 1501–1510.
- (45) Kučerka, N.; Gallová, J.; Uhlíková, D.; Balagváy, P.; Bulacu, M.; Marrink, S. J.; Katsaras, J. Areas of Monounsaturated Diacylphosphatidylcholines. *Biophys. J.* **2009**, *97*, 1926–1932.
- (46) Nagle, J. F.; Tristram-Nagle, S. Structure of lipid bilayers. *Biochim. Biophys. Acta* **2000**, *1469*, 159–195.
- (47) Alwarawrah, M.; Dai, J.; Huang, J. A Molecular View of the Cholesterol Condensing Effect in DOPC Lipid Bilayers. *J. Phys. Chem. B* **2010**, *114*, 7516–7523.
- (48) Dunford-Shore, B.; Fabrizio, F.; Holcomb, J.; Wise, W.; Feng, B.; Sulaman, W.; Sanghi, G.; Sadekar, S.; Cannon, B.; Taylor, D.; Kazic, T. Klobo: Biochemical compounds declarative database; 2002. <http://www.biocheminfo.org/klobo/> (accessed May 3, 2010).
- (49) Kleywegt, G. J.; Jones, T. A. Databases in protein crystallography. *Acta Crystallogr., Sect. D: Biol. Crystallogr.* **1998**, *S4*, 1119–1131.
- (50) Feng, Z.; Chen, L.; Maddula, H.; Akcan, O.; Oughtred, R.; Berman, H. M.; Westbrook, J. Ligand Depot: a data warehouse for ligands bound to macromolecules. *Bioinformatics* **2004**, *20*, 2153.
- (51) Zamis, T. M. *Chemistry Molecular Models*. <http://www.uwsp.edu/chemistry/pdbs/> (accessed May 3, 2010).
- (52) Kukol, A. Lipid Models for United-Atom Molecular Dynamics Simulations of Proteins. *J. Chem. Theory Comput.* **2009**, *5*, 615–626.
- (53) Tusnády, G. E.; Dosztányi, Z.; Simon, I. TMDET: web server for detecting transmembrane regions of proteins by using their 3D coordinates. *Bioinformatics* **2005**, *21*, 1276.
- (54) Lomize, A. L.; Pogozheva, I. D.; Lomize, M. A.; Mosberg, H. I. Positioning of proteins in membranes: a computational approach. *Protein Sci.* **2006**, *15*, 1318–1333.
- (55) Herrez, A. *How to use Jmol to study and present molecular structures*; Lulu Enterprises: Morrisville, USA, 2007.
- (56) Jmol: an open-source Java viewer for chemical structures in 3D. <http://jmol.sourceforge.net> (accessed May 3, 2010).
- (57) Allen, J. A.; Halverson-Tamboli, R. A.; Rasenick, M. M. Lipid raft microdomains and neurotransmitter signalling. *Nat. Rev. Neurosci.* **2006**, *8*, 128–140.
- (58) Dyckhoff, H.; Kruse, D. Trim loss and related problems. *Omega* **1985**, *13*, 59–72.
- (59) Cagan, J.; Shimada, K.; Yin, S. A survey of computational approaches to three-dimensional layout problems. *Comput. Aided Des.* **2002**, *34*, 597–611.
- (60) Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E. Equation of state calculations by fast computing machines. *J. Chem. Phys.* **1953**, *21*, 1087–1092.
- (61) Kirkpatrick, S.; Gelatt, C. D., Jr.; Vecchi, M. P. Optimization by simulated annealing. *Science (Washington, DC, U.S.)* **1983**, *220*, 671–680.
- (62) Černý, V. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *J. Optimiz. Theory App.* **1985**, *45*, 41–51.
- (63) Drexel, A. A simulated annealing approach to the multiconstraint zero-one knapsack problem. *Computing* **1988**, *40*, 1–8.
- (64) Liu, A.; Wang, J.; Han, G.; Wang, S.; Wen, J. In *Sixth International Conference on Intelligent Systems Design and Applications*, 2006 (ISDA'06); IEEE, 2006; Vol. 2, pp 1159–1164.
- (65) Sasaki, G. H.; Hajek, B. The time complexity of maximum matching by simulated annealing. *J. Assoc. Comput. Mach.* **1988**, *35*, 387–403.
- (66) Hovius, R.; Lambrechts, H.; Nicolay, K.; de Kruijff, B. Improved methods to isolate and subfractionate rat liver mitochondria. Lipid composition of the inner and outer membrane. *Biochim. Biophys. Acta* **1990**, *1021*, 217–226.
- (67) Schlamme, M.; Haldar, D. Cardiolipin is synthesized on the matrix side of the inner membrane in rat liver mitochondria. *J. Biol. Chem.* **1993**, *268*, 74–79.
- (68) Levy, M.; Sauner, M. T. Specificité de composition en phospholipides et en cholesterol des membranes mitochondriales. *Chem. Phys. Lipids* **1968**, *2*, 291–295.
- (69) Huang, J.; Feigenson, G. W. A microscopic interaction model of maximum solubility of cholesterol in lipid bilayers. *Biophys. J.* **1999**, *76*, 2142–2157.
- (70) Yamaguchi, R.; Lartigue, L.; Perkins, G.; Scott, R. T.; Dixit, A.; Kushnareva, Y.; Kuwana, T.; Ellisman, M. H.; Newmeyer, D. D. Opal-mediated cristae opening is Bax/Bak and BH3 dependent, required for apoptosis, and independent of Bak oligomerization. *Molecular Cell* **2008**, *31*, 557–569.
- (71) Martone, M. E.; Gupta, A.; Wong, M.; Qian, X.; Sosinsky, G.; Ludäschner, B.; Ellisman, M. H. A cell-centered database for electron tomographic data. *J. Struct. Biol.* **2002**, *138*, 145–155.
- (72) Yesylevskyy, S. O. ProtSqueeze: Simple and Effective Automated Tool for Setting up Membrane Protein Simulations. *J. Chem. Inf. Model.* **2007**, *47*, 1986–1994.
- (73) Deisenhofer, J.; Epp, O.; Miki, K.; Huber, R.; Michel, H. Structure of the protein subunits in the photosynthetic reaction centre of Rhodopseudomonas viridis at 3[angstrom] resolution. *Nature* **1985**, *318*, 618–624.
- (74) Tsukihara, T.; Aoyama, H.; Yamashita, E.; Tomizaki, T.; Yamaguchi, H.; Shinzawa-Itoh, K.; Nakashima, R.; Yaono, R.; Yoshikawa, S. Structures of metal sites of oxidized bovine heart cytochrome c oxidase at 2.8 Å. *Science (Washington, DC, U.S.)* **1995**, *269*, 1069–1074.
- (75) Daum, G. Lipids of mitochondria. *Biochim. Biophys. Acta* **1985**, *822*, 1–42.
- (76) Wallin, E.; Tsukihara, T.; Yoshikawa, S.; Heijne, G. V.; Elofsson, A. Architecture of helix bundle membrane proteins: an analysis of cytochrome c oxidase from bovine mitochondria. *Protein Sci.* **1997**, *6*, 808–815.
- (77) Ma, J.; Yoshimura, M.; Yamashita, E.; Nakagawa, A.; Ito, A.; Tsukihara, T. Structure of rat monoamine oxidase A and its specific recognitions for substrates and inhibitors. *J. Mol. Biol.* **2004**, *338*, 103–114.
- (78) McBride, H. M.; Neuspiel, M.; Wasik, S. Mitochondria: more than just a powerhouse. *Curr. Biol.* **2006**, *16*, R551–R560.
- (79) Zheng, Y. Z.; Berg, K. B.; Foster, L. J. Mitochondria do not contain lipid rafts, and lipid rafts do not contain mitochondrial proteins. *J. Lipid Res.* **2009**, *50*, 988–998.
- (80) Atshaves, B. P.; McIntosh, A. L.; Payne, H. R.; Gallegos, A. M.; Landrock, K.; Maeda, N.; Kier, A. B.; Schroeder, F. SCP-2/SCP-x gene ablation alters lipid raft domains in primary cultured mouse hepatocytes. *J. Lipid Res.* **2007**, *48*, 2193–2211.
- (81) Pralle, A.; Keller, P.; Florin, E. L.; Simons, K.; Horber, J. K. H. Sphingolipid-cholesterol rafts diffuse as small entities in the plasma membrane of mammalian cells. *J. Cell Biol.* **2000**, *148*, 997–1008.

- (82) Prior, I. A.; Muncke, C.; Parton, R. G.; Hancock, J. F. Direct visualization of Ras proteins in spatially distinct cell surface microdomains. *J. Cell Biol.* **2003**, *160*, 165–170.
- (83) Schuler, L. D.; Daura, X.; Van Gunsteren, W. F. An improved GROMOS96 force field for aliphatic hydrocarbons in the condensed phase. *J. Comput. Chem.* **2001**, *22*, 1205–1218.
- (84) Chandrasekhar, I.; Kastenholz, M.; Lins, R. D.; Oostenbrink, C.; Schuler, L. D.; Tielemans, D. P.; van Gunsteren, W. F. A consistent potential energy parameter set for lipids: dipalmitoylphosphatidylcholine as a benchmark of the GROMOS96 45A3 force field. *Eur. Biophys. J.* **2003**, *32*, 67–77.
- (85) Douliez, J. P.; Leonard, A.; Dufourc, E. J. Restatement of order parameters in biomembranes: calculation of CC bond order parameters from CD quadrupolar splittings. *Biophys. J.* **1995**, *68*, 1727–1739.