JCTC Journal of Chemical Theory and Computation

Article

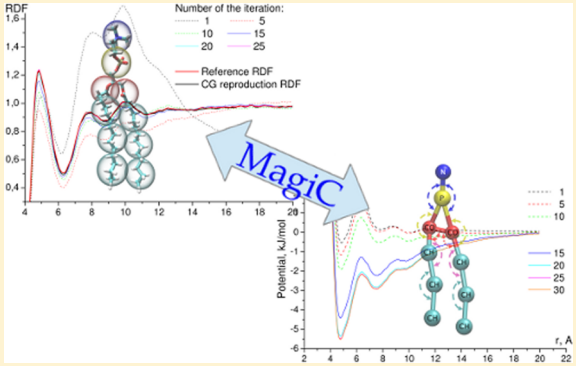
pubs.acs.org/JCTC

MagiC: Software Package for Multiscale Modeling

Alexander Mirzoev and Alexander P. Lyubartsev*

Division of Physical Chemistry, Department of Materials and Environmental Chemistry, Stockholm University, Stockholm, SE-10691, Sweden

ABSTRACT: We present software package MagiC, which is designed to perform systematic structure-based coarse graining of molecular models. The effective pairwise potentials between coarse-grained sites of low-resolution molecular models are constructed to reproduce structural distribution functions obtained from the modeling of the system in a high resolution (atomistic) description. The software supports coarse-grained tabulated intramolecular bond and angle interactions, as well as tabulated nonbonded interactions between different site types in the coarse-grained system, with the treatment of long-range electrostatic forces by the Ewald summation. Two methods of effective potential refinement are implemented: iterative Boltzmann inversion and inverse Monte Carlo, the latter accounting for cross-correlations between pair interactions. MagiC uses its own Metropolis Monte Carlo sampling engine, allowing parallel simulation of many copies of the system with subsequent averaging of the properties, which provides fast convergence of the method with nearly linear scaling at parallel execution.



1. INTRODUCTION

During the past decade, coarse-graining of molecular models has became an important technique used to expand the affordable size and time scale for simulations of various soft matter and biomolecular systems. The idea of uniting atoms into coarse-grain sites in order to reduce the number of degrees of freedom is straightforward, and there is a rich variety of practical approaches to do this.[1−5] As a result of coarse-graining, the potential energy landscape becomes smoother, which reduces the internal friction and allows one to access longer time scales. There exist simple empirically parametrized but numerically efficient bead−spring models which were used, for example, for qualitative description of amphiphilic lipid systems on a large scale[6] as well as for polymer simulations.[7,8] The Martini force field,[9] formulated as a standard molecular-mechanic type force field with Lennard-Jones and electrostatic terms for nonbonded interactions, has been parametrized by experimentally known thermodynamic data. Alternative approaches to parametrizing coarse-grained force fields are based on the so-called bottom-to-up concept, according to which parameters of a low-resolution, coarse grained model are deduced from a more detailed high resolution (fine-grained) model. This approach, also called systematic multiscale coarse graining, allows one in principle to start from *ab initio* quantum-chemical modeling, parametrize atom−atom potentials for atomistic class of models, and proceed further to different resolution coarse-grained models. The methods of deducing coarse-grained interaction potentials can be based on the force matching approach[10,11] or on reconstruction of the pair distribution function using the Inverse Monte Carlo (IMC)[12] and Newton inversion[13] or by Iterative Boltzmann Inversion (IBI).[14,15] The relative entropy minimization approach to deducing coarse-grained potentials has been also considered,[16] which in the case of pair potentials reproduces the pair distribution functions and thus in principle returns the same coarse-grained potentials as the IMC or IBI methods. The interconnection between force- and structure-based approaches to construct coarse-grained potentials has been discussed in terms of an information function.[17] As another recent development, we mention an approach aiming to compromise between the reproduction of structural and thermodynamic properties of the modeled system.[18]

While the above-mentioned methods have been known for a while and through the years have been applied to various molecular systems,[5,13,19−23] their practical use was concentrated mostly within the developer groups using in-house software. Only recently have two computational packages become available to a wider computational chemistry community: Versatile Object-oriented Toolkit for Coarse-graining Applications (VOTCA)[24] and IBIsCO.[25] The VOTCA package implements the iterative Boltzmann inversion, the inverse Monte Carlo, and the force matching algorithms. However, VOTCA does not have its own sampling engine, and it relies on a molecular dynamics algorithm in GROMACS[26] to sample the system while refining coarse-grained potentials by one of the methods. This may cause instabilities since GROMACS is not optimized for use of arbitrary tabulated potentials which may change unpredictably during the optimization procedure. The IBIsCO package implements only the IBI approach, which may

Received: November 20, 2012

   dx.doi.org/10.1021/ct301019v | *J. Chem. Theory Comput.* XXXX, XXX, XXX−XXX

have convergence problems in the case of complex multicomponent systems.

We have now developed a new software package under the name MagiC for the computation of effective coarse-grained potentials by the IMC and IBI methods. In designing this software, we have used our previous experience in applications of the IMC method to various systems. The package uses its own efficient Monte Carlo engine[27−29] for sampling of the configurational space. The use of the Monte Carlo algorithm for computation of the effective potentials has several advantages in comparison with molecular dynamics. First, the MC algorithm is more stable with respect to possible forms of the interaction potentials including possibilities for infinite values (which forbid the coarse grain system to visit configurations never observed in the atomistic simulations). Second, the Monte Carlo algorithm, implementing both trial moves of separate atoms and collective motions of the whole molecules or molecular fragments, can provide faster sampling of the configurational space which becomes especially noticeable for implicit-solvent coarse grained models. Third, the Monte Carlo algorithm allows very easy parallelization by starting many independent MC walks on different processors and then collecting averages from them.

In this paper, we present the MagiC package for systematic coarse-graining of molecular systems, describing how it implements IBI and IMC algorithms, the general work-flow, and principles of the software organization. We show also an example of practical use of the software for the computation of effective potentials in lipid systems.

2. METHODS

2.1. Structure-Based Systematic Coarse-Graining. The main idea of the structure-based systematic coarse-graining is to deduce effective potentials for a coarse-grained model from the structural properties obtained in detailed (atomistic) simulations of the same system, which we call below a "reference" simulation. It is supposed that initially simulation of the system of interest is run on a detailed level, which can be on a scale (in terms of system size and simulation time) which one can afford. Various structural properties, such as radial distribution functions, distributions of bond distances, angles, etc., are computed. Then the task is to find a set of effective coarse-grained potentials, which reproduces within the coarse-grained model the same structural properties as the reference simulation on a detailed level. The theoretical background of the structure-based coarse-graining is the Henderson theorem,[30] which stipulates that for a given RDF, there may exist only a single (with precision of an additive constant) pair interaction potential which returns this RDF. The Henderson theorem, initially proven for monocomponent systems, is straightforwardly generalized for the case of multiple RDF and pair effective potentials between different molecular sites, as well as to cases involving bond and angular interactions.[17] Two methods of practical solution of the inverse problem are briefly described below.

2.2. Iterative Boltzmann Inversion. Iterative Boltzmann inversion is based on the inversion of Boltzmann probability distribution for a system in a canonical ensemble. Let q be a configurational state of the coarse-grained system. Its probability distribution $P(q)$, which in principle is available from the reference simulation, is

$$P(q) = \frac{1}{Q}\exp\left(-\frac{U(q)}{k_B T}\right) \tag{1}$$

where $Q = \int \exp(-(U(q))/(k_B T))\,\mathrm{d}q$ is the partition function and $U(q)$ is the N-body potential of mean force:

$$U(q) = -k_B T \ln(P(q)) + \text{const} \tag{2}$$

Obtaining and further use of the N-body potential of mean force is however unrealistic; therefore at this point $P(q)$ is factorized, assuming independence of different degrees of freedom:

$$P(q) = \prod P_{\text{bond}}(r_{\text{bond}}) \prod P_{\text{ang}}(\phi) \prod P_{\text{dih}}(\theta) \prod P_{\text{NB}}(r) \tag{3}$$

where $P_{\text{bond}}(r_{\text{bond}})$, $P_{\text{ang}}(\phi)$, and $P_{\text{dih}}(\theta)$ are distributions of bond lengths, angles, and torsional degrees of freedom, respectively, and P_{NB} represents the intermolecular structure of the system given by a set of RDFs between different coarse-grained sites. Now, coarse-grained potentials can be extracted in the same fashion as in eq 2:

$$\begin{aligned} U(r, r_{\text{bond}}, \phi, \theta) &= U_{\text{bond}}(r_{\text{bond}}) + U_{\text{ang}}(\phi) + U_{\text{dih}}(\theta) \\ &\quad + U_{\text{NB}}(r) \\ &= -k_B T[\sum \ln(P_{\text{bond}}(r_{\text{bond}})) \\ &\quad + \sum \ln(P_{\text{ang}}(\phi)) + \sum \ln(P_{\text{dih}}(\theta)) \\ &\quad + \sum \ln(P_{\text{NB}}(r))] \end{aligned} \tag{4}$$

The key assumption behind eq 4 is the absence of correlations between different degrees of freedom. While for intramolecular degrees of freedom (bonds, angles) this assumption is usually valid to a certain degree, it is much less justified for degrees of freedom representing intermolecular correlations. This mean that effective potentials $U^{(0)}$ obtained by direct inversion according to eq 4 generally fail to reproduce distribution functions of the reference system. This problem can be partially overcome by iterative refinement of the coarse grained potentials. Assume a direct Boltzmann inversion, eq 4 resulted in potential $U^{(0)}$. The canonical average distributions $P^{(0)}$ for the system defined by $U^{(0)}$ can be straightforwardly computed in a standard MD or MC simulation. If they differ from reference distributions P_{ref}, the effective potential requires correction. A simple way to introduce such a correction was suggested by Schommers[31] and later reintroduced by Soper:[14]

$$U^{(i+1)} = U^{(i)} + k_B T \ln\frac{P^{(i)}}{P_{\text{ref}}} \tag{5}$$

Here, by index i we denote the iteration number of the potential refinement. If distribution $P^{(i)}$ approaches the reference distribution P_{ref}, the correction term becomes small and the process reaches a convergence. For nonbonded potentials, radial distribution function $g(r)$ is used instead of the distribution of nonbonded distances P_{NB}. This method, due to its principal similarity to eq 1, has become known as iterative Boltzmann inversion.[15] The IBI approach was successfully used to obtain both bonded and nonbonded effective potentials for a number of systems, but it does not account for cross-correlations between different degrees of freedom and the corresponding distributions, which causes convergence problems for multicomponent systems, for example, ion solutions.[32]

dx.doi.org/10.1021/ct301019v | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

That is why the use of the IBI method is mostly limited to simple liquids or polymers with no more than 1−2 different bead types, or to systems which can be split on fragments studied separately.[1]

2.3. Inverse Monte Carlo: Newton Inversion. The inverse Monte Carlo method was formulated for systems with a Hamiltonian presented in the form:[12]

$$H(\{q\}) = \sum_{\alpha} V_\alpha S_\alpha(\{q\}) \tag{6}$$

where basis functions $S_\alpha\{q\}$ define a space of Hamiltonians, while a set of parameters V_α defines a specific Hamiltonian in this space. For example, for a class of systems interacting by arbitrary tabulated pair interaction potentials, $S_\alpha\{q\}$ represents the number of particle pairs having the distance within the α-fragment of the range of considered distances, while V_α represents the values of pair potential at this distance. Canonical averages $\langle S_\alpha\rangle$ in this case are related to the radial distribution function by

$$\langle S_\alpha\rangle = 4\pi r_\alpha^2 \Delta r g(r_\alpha) \tag{7}$$

If a set of V_α is given, canonical averages $\langle S_\alpha\rangle$ can be evaluated in a standard MC or MD simulation. We however need to solve an inverse problem, that is to find a set V_α from a set of known averages $\langle S_\alpha\rangle = S_\alpha^*$ which were obtained from reference high resolution simulations. Since $\langle S_\alpha\rangle$ are functions of V_α, one can write

$$\Delta\langle S_\alpha\rangle = \sum_{\gamma} \frac{\partial\langle S_\alpha\rangle}{\partial V_\gamma}\Delta V_\gamma + O(\Delta V^2) \tag{8}$$

where derivatives $(\partial\langle S_\alpha\rangle)/(\partial V_\gamma)$ are given by

$$\frac{\partial\langle S_\alpha\rangle}{\partial V_\gamma} = \frac{\partial}{\partial V_\gamma}\left[\frac{1}{Q}\int \mathrm{d}q\; S_\alpha\, \mathrm{e}^{-(1/k_\mathrm{B}T)\sum_i V_i S_i}\right] = \frac{1}{k_\mathrm{B}T}(\langle S_\alpha\rangle\langle S_\gamma\rangle - \langle S_\alpha S_\gamma\rangle) \tag{9}$$

and can be also computed in a conventional MC or MD simulation. The set of eqs 8 and 9 makes it possible to solve the inverse problem iteratively. We start from a set of trial potential parameters $V_\alpha^{(i)}$ $(i = 0)$, run the simulation, and calculate averages $\langle S_\alpha\rangle^{(i)}$, as well as $\langle S_\alpha S_\gamma\rangle^{(i)}$. The differences between the computed and reference values $\Delta\langle S_\alpha\rangle^{(i)} = \langle S_\alpha\rangle^{(i)} - S_\alpha^*$ are substituted into the system of linear eqs 8, which returns (by neglecting terms of order $O(\Delta V^2)$) the corrections to the potential parameters ΔV_α. They are used to update the interaction potential:

$$V_\alpha^{(i+1)} = V_\alpha^{(i)} + \Delta V_\alpha^{(i)} \tag{10}$$

The procedure is repeated until convergence is reached and the reference averages S_α^* are reproduced with satisfactory accuracy. As an initial approximation of trial potential $V_\alpha^{(0)}$, the pair potential of mean force can be selected $V_\alpha^{(0)} = -k_\mathrm{B}T\ln(g(r_\alpha))$. For nonbonded atom pairs, the reference averages S_α^* are determined from the radial distribution functions (eq 7). For bonded interactions, S_α^* are distance distributions of covalent bonds or distributions of covalent angles θ_α: $S_\alpha^* = \Delta r\, g(r_\alpha)$ or $S_\alpha^* = \Delta\theta\, g(\theta_\alpha)$. The trial potential for these bonds can be constructed in a similar way to the nonbonded one $V_\alpha^{(0)} = -k_\mathrm{B}T\ln(g(r_\alpha))$, or a harmonic potential can be used.

3. SOFTWARE IMPLEMENTATION

In this section, we discuss the general outline of MagiC as well as implementation details of the main blocks. The package consists of the main kernel, which solves the inverse problem, and a set of supplementary utilities. The kernel, consuming the most computational resources, is written in Fortran 90 with MPI-based parallelization. The utilities are written partially in Fortran and partially in Python. Python utilities form an object-oriented module which serves as an interface to the main kernel. The utilities allow the user to build a coarse-grained trajectory from a high resolution trajectory, calculate reference distribution functions, analyze and visualize the kernel's output, and convert MagiC's internal data format to/from external formats (currently available GROMACS[26] and MDynaMix[33]). An object-oriented design makes it easy to extend the set of utilities for particular needs. Below, we will briefly discuss the work-flow of the whole package and give some details of implementation of the algorithms in the kernel.

3.1. General Work-Flow of the Software. In general, the systematic coarse graining can be considered as a multistage process which leads from a high resolution system description to a low resolution one. A work-flow of this process is depicted in Figure 1. Each stage uses an output of the preceding stage as an input, and additional input can be provided by the user (right-most blue frames).

The first stage is simulation of the system at high resolution, for example using molecular dynamics with an all-atom force field. This simulation creates a high resolution (atomistic) trajectory which is supposed to provide adequate sampling of the configuration space of the system of interest. This stage can be performed with any suitable simulation software.

At the second stage, a user has to perform a mapping between high resolution structure and low resolution structure. MagiC provides a utility (CGtraj) converting atomistic trajectory into CG trajectory, using a mapping scheme from the input file (provided by the user) stating correspondence between ith CG bead and set of atoms $\{ij\}$ in the atomistic model which will be presented by this CG bead. The bead's mass and charge are defined as the total mass and charge of its atomistic components $M_i = \Sigma_j m_{ij}$ and $Q_i = \Sigma_j q_{ij}$, and the bead is placed onto the center of masses of the representing atoms $\vec{R}_i = 1/M_i \sum_j \vec{r}_{ij} m_{ij}$. This stage results in a coarse-grained trajectory (saved in xmol/xyz format) and in the assignment of mass and charge to the CG beads which are stored in a designated file.

At the third stage, the structural reference distribution functions are calculated from the coarse-grained trajectory. This is performed by internal utility RDF. At this stage, the user has to specify which CG sites can be considered as equivalent and which sites are different (this determines the total number of pair RDFs and nonbonded potentials in the system), which CG sites are bound to each other, and which bonds can be considered as equivalent. The utility computes RDFs between all pairs of different CG sites, as well as the distribution of lengths of intramolecular bonds and distribution of angles defined in the CG model. Also, it creates a CG molecular topology file (with file extension *.mcm*) which is used by MagiC's kernel.

The next stage is the key point of the whole systematic coarse-graining process: solution of the inverse problem. This is performed by the kernel of the package which implements the inverse Monte Carlo and iterative Boltzmann inversion method described in section 2. It results in a set of effective potentials,

dx.doi.org/10.1021/ct301019v | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

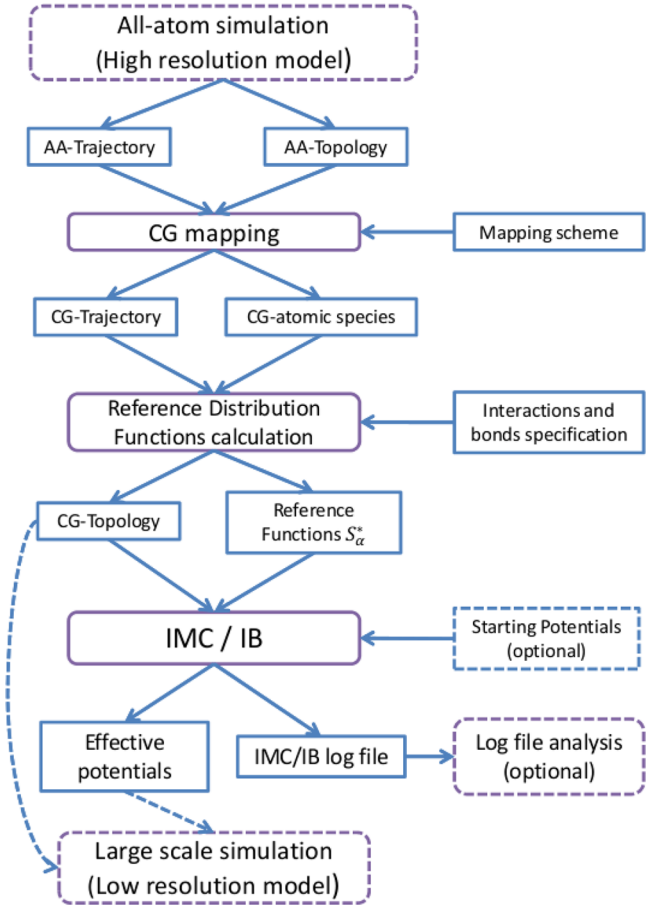


Figure 1. General work-flow of systematic coarse-graining with MagiC. Purple rectangles denote data processing procedures. Blue rectangles denote input/output data; right-most blue blocks show the user provided input. Optional input data and external software are indicated by dashed frames.

which reproduce reference distribution functions with desirable precision, and an extended log-file which reports details of each IMC/IBI iteration. The log-file can be analyzed by a set of postprocessing tools, which allows one to plot the convergence rate, effective potentials, potential corrections, intermediate RDFs, etc.

Once the effective potentials are obtained, they can be exported to an external MD or other type of mesoscale simulation software and used for further simulations of the coarse-grained system on a larger scale. At the moment, the postprocessing tools provide export of the tabulated potentials to the GROMACS topology format. It is straightforward in a similar fashion to create utilities for export of the effective potential to a format understandable by other packages such as ESPResSo[34] or LAMMPS.[35]

3.2. The Kernel. The kernel performing computation of effective potentials is the most important part of the software. Here, we give a description of basic principles of its organization. A general scheme of MagiC's kernel is shown in Figure 2.

The program starts from initialization, when it reads the input files—calculation settings, molecular topologies, reference distribution functions, and optionally initial potentials—as well as prepares necessary data structures. If initial potentials were not supplied, the program initiates a trial potential as a mean force potential from the corresponding distribution functions

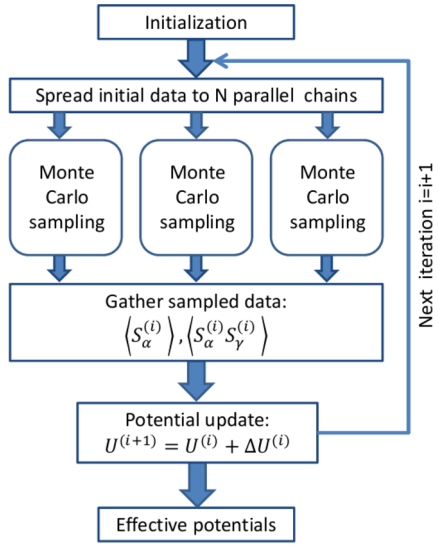


Figure 2. Block scheme of the kernel.

(eq 2). Then, the program starts an iterative procedure in order to refine the potentials and fit them to the reference RDFs and bond length/angle distributions. Each iteration step represents a conventional MC simulation with actual values of the potentials.

The MC simulation is the most time-consuming part of the program, and in order to perform this part faster and provide better sampling of the configurational space, this part is parallelized using a rather straightforward but efficient strategy. It relies on the stochastic nature of the MC method, which allows one to sample the system in many simultaneous (parallel) threads. Each thread runs its own local copy of the system, and when a sufficient number of MC steps has been performed, the sampled data are collected and averaged over all the threads. Such a parallelization scheme requires just two procedures: spreading the initial data to a set of parallel MC processes and gathering of the sampled data from the processes when they have finished. Each parallel thread starts with a different starting configuration and uses random numbers generated with different seeds. Though the time of the equilibration stage does not depend on the number of threads, after finishing the equilibration the systems appear in very distinct areas of the configuration space which substantially improve the overall sampling during the production part of the run. When all the threads have made the specified number of MC steps, the computed results (RDFs, cross-correlation matrix) are averaged and transferred to the program block calculating correction to the current potential.

To run a conventional MC simulation, we have implemented the classical Metropolis Monte Carlo algorithm[27−29] with three possible MC steps: random atom displacement, which is a default MC step, random molecule displacement (translation), and random molecule rotation.[29] The latter two steps are considered supplementary steps improving convergence of the method.

All interactions between atoms of the system (see Figure 3) are arranged into two groups: bonded and nonbonded. The bonded interactions are represented by covalent bonds $U^{AB}_{bond}(r)$ and angle bending bonds $U^{ABC}_{ang}(\theta)$. A torsion angle interaction is currently not implemented (having in mind that 1−4 pairs are usually weakly correlated in CG models), but they are planned to appear in further releases of the code.

dx.doi.org/10.1021/ct301019v | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

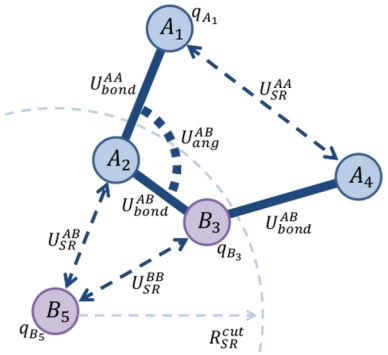


Figure 3. Typical interactions present in the system. Bold lines, covalent bonds; dotted line, angle bending bonds; dashed lines, nonbonded repulsion-dispersion short-range interactions. Symbols A and B denote bead type, and q_i denotes the charge of atom i.

The nonbonded interactions are presented as a sum of long-range electrostatics Coulomb interactions and a short-range part $U^{AB}_{SR}(r)$, which includes all other contributions (overlapping repulsion, van-der-Waals dispersion, effect of removed solvent, etc). All atom pairs are subject to nonbonded interactions (even if they belong to the same molecule), except the bonded pairs which are excluded from all nonbonded interactions. The electrostatic interactions are treated by the Ewald summation method,[28] and the Coulombic contribution coming from charged bonded pairs is subtracted from the Ewald energy. The short-range part of nonbonded interactions is treated by direct summation of contributions coming from pairs being within a certain cutoff radius R_{cutoff} which by default is set to the cutoff distance of the reference RDF.

When all threads finished MC simulations with current interaction potential, the collected and averaged data are transferred to the module, which performs the inverse procedure. Here, the program calculates an update for the potentials, which is based on the deviation between the calculated distribution functions and the reference ones. The update can be calculated either by the iterative Boltzmann inversion expression (eq 5) or by the inverse Monte Carlo (eq 8), described in the previous section. The system's partial charges are kept constant; thus the long-range electrostatic potentials are taken out of the update. If a satisfactory agreement between the averaged and reference distribution functions was reached, the program moves to the last (output) block; otherwise a new iteration with updated potential is started.

The output block produces the program's output in a human-readable form, with the basic data of the simulated system and the simulation's parameters, as well as the resulting effective potentials and distribution functions. The computed potentials are also written as a separate file in a format suitable for their further optimization and analysis in MagiC.

4. EXAMPLE: DERIVATION OF POTENTIALS FOR A COARSE GRAINED LIPID MODEL

During development of the MagiC software, the package was repeatedly tested on a number of systems of different complexity. The core of the present package comes from earlier works on the inverse MC method[12,36] in which effective solvent-mediated interactions of Na^+-Cl^- ions in a water solution have been computed. In a later extended study,[37] temperature dependence of the Na^+-Cl^- effective potentials

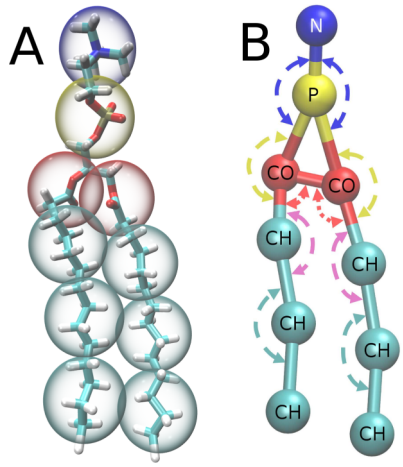


Figure 4. (A) Bead mapping used for coarse-graining of a DMPC lipid molecule. Blue bead, choline; yellow bead, phosphate group; red beads, ester groups; cyan beads, hydrocarbon tails. Beads of the same color belongs to same bead type. (B) Intramolecular bond introduced in the CG model. Solid lines denote covalent bonds; dashed arrows denote angle bending bonds. Bonds of the same color are assumed to be identical.

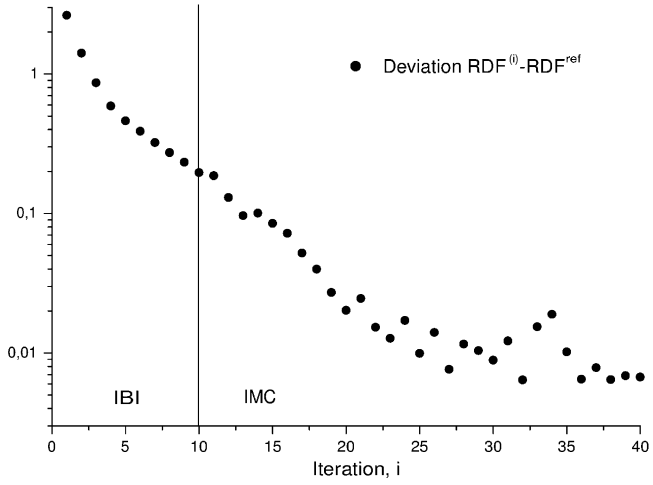


Figure 5. The deviation between the reference RDF and the RDF sampled in the IBI/IMC calculation. The first 10 iterations were performed with IBI, and the following iterations were performed with IMC.

has been investigated. Among other simple single-site systems, we can mention the derivation of effective potential for a united atom water model[13] and effective potentials between charged colloid particles.[38] In more complex systems, the coarse-grained structures were presented by more than one site per molecule. Thus, the package was used to obtain effective solvent-mediated ion−ion and ion−DNA potentials[39] and for a coarse-grained model of proline molecules dissolved in DMSO solvent.[13]

Here, we consider in more specific detail how MagiC software can be used for derivations of effective potentials taking a coarse-grained lipid model as an example, discuss using this example capabilities of MagiC, and highlight a number of methodological issues related to the use of the method. Various coarse-grained lipid models are widely used now to describe mesoscale behavior of biomembranes and other lipid assemblies such as micelles and vesicles.[6,9,40−42] Additional references can be found in recent reviews.[43−45] The outline of the approach,

dx.doi.org/10.1021/ct301019v | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

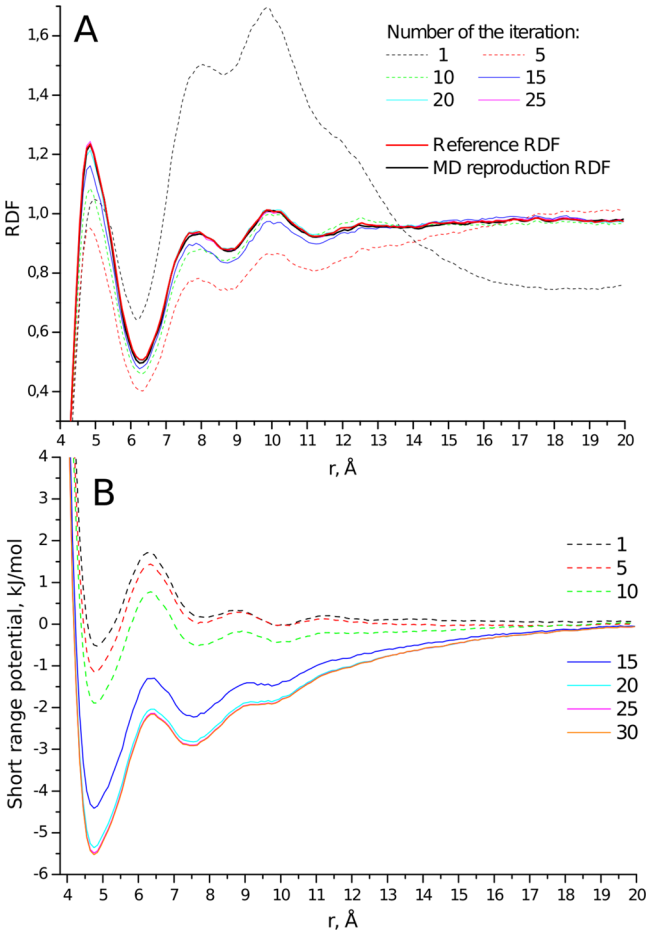


Figure 6. Radial distribution functions (A) and effective potentials (B) calculated for the N−P pair of beads during iterative solution of the inverse problem. Dashed lines refer to iterations made using IBI, and solid lines refer to the use of IMC. Thick red and black curves on plot A represent the reference RDF calculated in atomistic MD simulation and RDFs calculated in coarse grained MD simulations with final effective potentials, respectively.

used also in the previous work on the coarse-grained lipid model,[41] is depicted in Figure 1, where the MD-related studies (the first and the last stages) were performed by GROMACS, and MagiC took care of the rest of the work. The visualization of the trajectory was made using VMD.[46] All relevant setup files of MagiC which were used to perform calculations of the present example are given in the Examples subdirectory of the MagiC distribution.

First, an all-atom MD simulation was performed with 16 dimyristoylphosphatidylcholine (DMPC) lipids dissolved in 1600 water molecules. The temperature and pressure were fixed to be 303 K and 1 atm, respectively, by use of a Nose-Hoover thermostat[47,48] and a Parrinello−Rahman barostat.[49,50] A modified CHARMM force filed[51] was used for lipid description together with a rigid TIP3P[52,53] water model. All the bonds with hydrogen atoms were constrained using the LINCS algorithm,[54] and the integration time step was set to 1 fs. The electrostatic interactions were treated with the particle-mesh Ewald summation method (PME) with a real-space cutoff of 13 Å. The same cutoff was used for the Lennard-Jones short-range interactions. The all-atom MD simulation resulted in a 400 ns trajectory of which first 100 ns were disregarded. This atomistic trajectory was converted, using the CGtraj utility of MagiC, into a coarse grained trajectory. At this stage, the water molecules presented in the system were removed, and each DMPC lipid consisting of 118 atoms was mapped onto a CG model consisting of 10 beads, as it is shown in Figure 4A. The polar head of the lipid is represented by two charged groups: choline (blue) and phosphate (yellow). The phosphate group is connected to two ester groups (red), which are connected to a lipid tail represented by two hydrocarbon chains made of three beads (cyan) each. The position of each bead is defined as a center of mass of all the atoms included in the bead, and the total charge of the atomic group is assigned to the charge of the bead. Thus, the most charge belongs to N and P groups, leaving CO groups slightly charged and CH groups completely neutral. Since we had defined four bead types, 10 different intermolecular short-range pairwise interactions need to be included into the CG model. Besides that, we introduced covalent bonds and angle bending bonds shown in Figure 4B with solid and dashed lines, respectively. We assume also that bonds between the same types of species can be considered identical, so five different covalent bonds were defined: N−P, P−CO, CO−CO, CH−CO, CH−CH; and 5 angle bending bonds: N−P−CO, P-CO−CH, CO−CH−CH, CO−CO−CH, and CH−CH−CH. All together, it results in 20 different effective potentials, of which 10 are bonded and 10 are nonbonded. In order to obtain them from the inverse MC simulations, a reference distribution function for each of the 20 interactions is given: 10 radial distribution functions for nonbonded interactions and 10 distributions of bond lengths and angles for intramolecular interactions have been calculated using the RDF utility of MagiC. These distribution functions were sent as an input to MagiC's kernel, which employs IBI or IMC methods to obtain a set of effective pair potentials. The Monte Carlo sampling was carried out under the NVT ensemble conditions, using the same number of molecules, temperature, and average periodic box size as in the reference atomistic MD simulation. Thus, possible size effects on the RDFs should be avoided. The electrostatic interactions were treated by the Ewald method[28] using partial charges of the coarse-grained sites. Since the water was removed at this stage, a relative dielectric permittivity $\varepsilon = 70$ was used to describe effective dielectric media. The short-range interactions were set to have the same cutoff distance of 20 Å as the reference RDF. Ranges of distances with $g(r) = 0$ were considered prohibited for MC transitions. At each iteration, a new starting configuration of the coarse-grained system was randomly generated in order to avoid memory effects in the sampling. Note that MagiC also allows one to proceed to each next iteration starting from the last configuration of the completed iteration.

The iteration procedure began from the potentials of mean force. The first 10 iterations were run using the IBI method. It provides initial optimization of the trial potentials. After that, the IMC algorithm was switched on. The IMC algorithm takes the cross correlation terms into account, and for this reason it has a higher demand for the sampling quality compared to the IBI method. Also, a higher accuracy of sampling is needed at the final stages of the potential optimization when the agreement with the reference RDF became better, and the difference between simulated and reference RDFs needs to be distinguished from the statistical noise. We therefore were increasing (doubling) a number of MC sampling steps every 10 iterations while making 40 iterations in total. The convergence of the IMC algorithm is illustrated by Figure 5. It shows the

dx.doi.org/10.1021/ct301019v | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

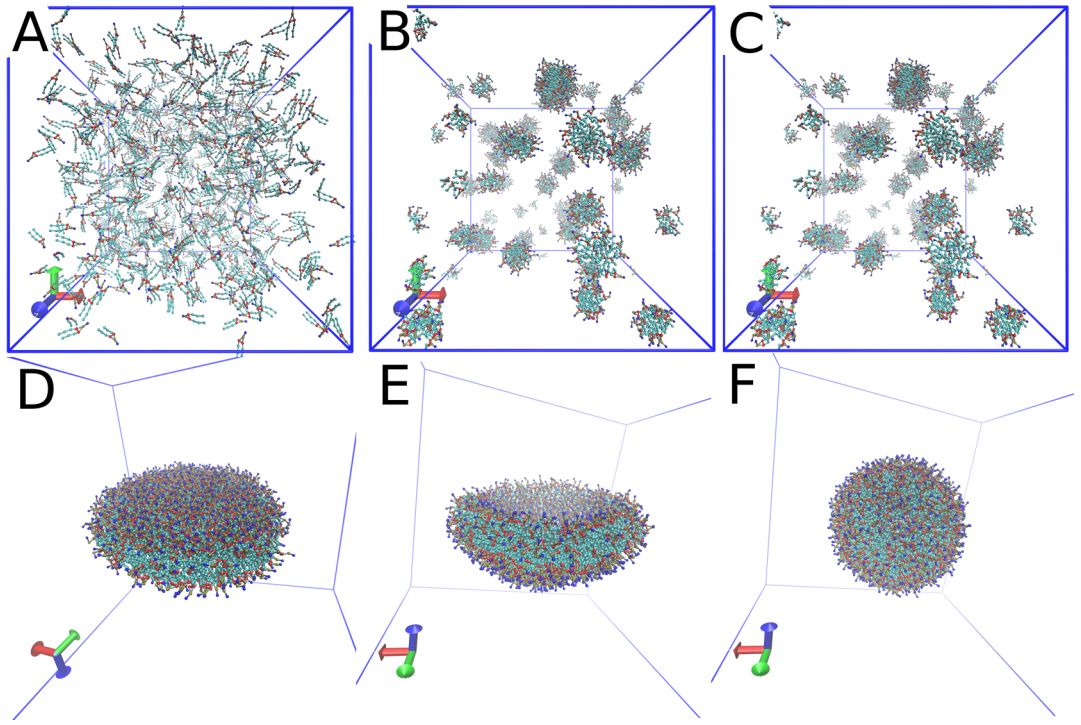


Figure 7. Vesicle formation. (A) Initial uniform distribution of lipids. (B) Formation of small spherical clusters. (C) Formation of small round-flat bicells. (D) Formation of large flat-round bicells. (E) Collapse of the flat-round bilayer (bicell) into a vesicle. (F) Stable spherical vesicle.

total deviation $\delta^{(i)}$ between the set of the reference distribution functions and the set obtained during each iteration i:

$$\delta^{(i)} = \sum_{N_{\mathrm{RDF}}} \frac{1}{r_{\max}} \int_0^{r_{\max}} (g^{\mathrm{ref}}(r) - g^{(i)}(r))^2 \, \mathrm{d}r$$

The deviation shows rather fast decay during the first 20 iterations. For the next 20 iterations, the deviation has significant fluctuations, still keeping the decaying trend on average. For another illustration of the convergence process, we consider evolution of the N−P nonbonded potential and the corresponding RDF, which are displayed in Figure 6A and B. The trial potential of the mean force used at the first iteration (black dashed line) does not provide a significant agreement with the reference RDF (bold solid red line), but after a few IBI iterations, the sampled RDF (red and green dashed lines) looks much closer to the reference one. When the potential update algorithm was switched to IMC (solid lines), the sampled RDFs came to a very close agreement with the reference RDF during the next 10 iterations (solid blue and cyan lines), and after that they became nearly indistinguishable by the eye (purple line). The effective potentials show a similar behavior. During first 10−15 iterations, the potential was subject to noticeable changes: the first and second minima became deeper, and the whole potential was shifted to an area of negative values. After that, there were no significant changes in the potential, and after iteration 25, all the curves become indistinguishable. This is a clear indication that the convergence has been reached, and the resulting set of the potentials reproduces the reference set of RDFs successfully.

In order to use the computed coarse-grained potentials in large scale MD simulations using GROMACS software, the potentials were smoothed to have a continuous first derivative, and soft harmonic-like walls were added to covalent bond potentials keeping the bonds in the region where bonding potential is defined. Nonbonded potentials were also extended with a repulsive quadratic potential in the core region. After exporting the potentials to the GROMACS format, a test on the reproducibility of RDFs was performed. The same system of 16 coarse grained lipids was simulated in the NVT ensemble at a temperature of 303 K using a Nose-Hoover thermostat. The size of the periodic box was kept the same as in the IMC calculations. A 0.5 μs trajectory was generated with a MD step of 3 fs. The resulting RDF for the N−P pair (Figure 6A, bold black line) shows perfect agreement with the reference curve; the same quality agreement was obtained for RDFs between other pairs.

Finally, a large scale MD simulation has been performed. The system consisted of 1000 coarse-grained DMPC lipids randomly distributed in a cubic box with a side of 30 nm at temperature T = 303 K. Stochastic Langevin dynamics was employed mimicking the friction and random forces arising from the solvent. At that moment, we did not fit the friction parameter (value of 1.0 ps^{-1} was used) in order to match dynamics of atomistic simulations, which may have resulted in a faster dynamics. The periodic cell size was kept constant. In total, a 1.9 μs trajectory was generated with an integration step of 5 fs. During this time, the system self-assembled from a completely random distribution of lipid molecules to a single spherical vesicle. This process passed through a few phases, which are presented in Figure 7. First, a fast clustering of lipids into small semispherical droplets (Figure 7B) was observed. These droplets merged soon into small disk-like structures, so-called bicells (Figure 7C), which finally formed one large "pancake" (Figure 7D). This pancake finally had collapsed into a spherical vesicle (Figure 7E,F), which remained stable for the rest of the simulation.

dx.doi.org/10.1021/ct301019v | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

5. CONCLUSIONS

We have developed a software package MagiC which enables systematic structure based coarse graining for arbitrary molecular models without the use of empirical parameters. As an input, the MagiC software uses trajectories obtained by detailed (high resolution) modeling of the studied system. The effective potentials between coarse-grained sites (low resolution description) are constructed to reproduce structural distribution functions obtained from the detailed high resolution description. Two methods of effective potential refinement are implemented: iterative Boltzmann inversion having a simplified refinement scheme and the inverse Monte Carlo method, which accounts for cross-correlations between pair interactions and thus has a faster convergence for a cost of more extensive sampling. An important feature of MagiC is the use of its own Metropolis Monte Carlo sampling engine, which supports also parallel execution of any number of threads simultaneously. Finally, MagiC can also be used to run standard Monte Carlo canonical ensemble simulations for arbitrary molecular systems interacting by a given pair interaction potential in a tabulated form.

The source code of MagiC is available at http://code.google.com/p/magic/ in open access under the terms of GNU public license.

AUTHOR INFORMATION

Corresponding Author

*E-mail: alexander.lyubartsev@mmk.su.se.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work has been supported by the Swedish Research Council (Vetenskå psradet), grant 70525601. The authors are thankful to the Swedish National Infrastructure for Computing (SNIC) for granting access to high performance computing facilities at the PDC Center for High Performance Computing, Stockholm, and High Performance Computing Center North (HPC2N), Umeå.

REFERENCES

(1) Peter, C.; Kremer, K. Multiscale simulation of soft matter systems - from the atomistic to the coarse-grained level and back. *Soft Matter* **2009**, *5*, 4357−4366.

(2) Tóth, G. Interactions from diffraction data: historical and comprehensive overview of simulation assisted methods. *J. Phys.: Condens. Matter* **2007**, *19*, 335220.

(3) Murtola, T.; Bunker, A.; Vattulainen, I.; Deserno, M.; Karttunen, M. Multiscale modeling of emergent materials: biological and soft matter. *Phys. Chem. Chem. Phys.* **2009**, *11*, 1869−1892.

(4) Ayton, G. S.; Noid, W. G.; Voth, G. A. Multiscale modeling of biomolecular systems: in serial and in parallel. *Curr. Opin. Struct. Biol.* **2007**, *17*, 192−198.

(5) Karimi-Varzaneh, H. A.; Müller-Plathe, F. In *Multiscale Molecular Models in Applied Chemsitry*; Kirchner, B., Vrabec, J., Eds.; Springer-Verlag: Berlin, 2012; Topics in Current Chemistry, Vol. *307*, pp 295−321.

(6) Cooke, I. R.; Kremer, K.; Deserno, M. Tunable generic model for fluid bilayer membranes. *Phys. Rev. E* **2005**, *72*, 011506.

(7) Baschnagel, J.; Binder, K.; Doruker, P.; Gusev, A.; Hahn, O.; Kremer, K.; Mattice, W.; Müller-Plathe, F.; Murat, M.; Paul, W.; Santos, S.; Suter, U.; Tries, V. *Viscoelasticity, Atomistic Models, Statistical Chemistry*; Springer: Berlin/Heidelberg, 2000; Advances in Polymer Science, Vol. *152*, pp 41−156.

(8) Kremer, K.; Grest, G. S. Dynamics of entangled linear polymer melts: A molecular-dynamics simulation. *J. Chem. Phys.* **1990**, *92*, 5057−5086.

(9) Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; de Vries, A. H. The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations. *J. Phys. Chem. B* **2007**, *111*, 7812−7824.

(10) Izvekov, S.; Parrinello, M.; Burnham, C. J.; Voth, G. A. Effective force fields for condensed phase systems from ab-initio molecular dynamics simulations: A new method for force-matching. *J. Chem. Phys.* **2004**, *120*, 10896−10913.

(11) Noid, W. G.; Chu, J.-W.; Ayton, G. S.; Krishna, V.; Izvekov, S.; Voth, G. A.; Das, A.; Andersen, H. C. The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models. *J. Chem. Phys.* **2008**, *128*, 244114.

(12) Lyubartsev, A. P.; Laaksonen, A. Calculation of effective interaction potentials from radial distribution functions: A reverse Monte Carlo approach. *Phys. Rev. E* **1995**, *52*, 3730−3737.

(13) Lyubartsev, A.; Mirzoev, A.; Chen, L.; Laaksonen, A. Systematic coarse-graining of molecular models by the Newton inversion method. *Faraday Discuss.* **2010**, *144*, 43−56.

(14) Soper, A. K. Empirical potential Monte Carlo simulation of fluid structure. *Chem. Phys.* **1996**, *202*, 295−306.

(15) Reith, D.; Pütz, M.; Müller-Plathe, F. Deriving effective mesoscale potentials from atomistic simulations. *J. Comput. Chem.* **2003**, *24*, 1624−1636.

(16) Chaimovich, A.; Shell, M. S. Coarse-graining errors and numerical optimization using a relative entropy framework. *J. Chem. Phys.* **2011**, *134*, 094112.

(17) Rudzinski, J. F.; Noid, W. G. Coarse-graining entropy, forces, and structures. *J. Chem. Phys.* **2011**, *135*, 214101.

(18) Jochum, M.; Andrienko, D.; Kremer, K.; Peter, C. Structure-based coarse-graining in liquid slabs. *J. Chem. Phys.* **2012**, *137*, 064102.

(19) Izvekov, S.; Voth, G. A. A Multiscale Coarse-Graining Method for Biomolecular Systems. *J. Phys. Chem. B* **2005**, *109*, 2469−2473.

(20) Izvekov, S.; Voth, G. A. Multiscale Coarse-Graining of Mixed Phospholipid/Cholesterol Bilayers. *J. Chem. Theory Comput.* **2006**, *2*, 637−648.

(21) Carmichael, S. P.; Shell, M. S. A New Multiscale Algorithm and Its Application to Coarse-Grained Peptide Models for Self-Assembly. *J. Phys. Chem. B* **2012**, *116*, 8383−8393.

(22) Murtola, T.; Karttunen, M.; Vattulainen, I. Systematic coarse graining from structure using internal states: Application to phospholipid/cholesterol bilayer. *J. Chem. Phys.* **2009**, *131*, 055101.

(23) Villa, A.; Peter, C.; van der Vegt, N. F. A. Transferability of Nonbonded Interaction Potentials for Coarse-Grained Simulations: Benzene in Water. *J. Chem. Theory Comput.* **2010**, *6*, 2434−2444.

(24) Rühle, V.; Junghans, C.; Lukyanov, A.; Kremer, K.; Andrienko, D. Versatile Object-Oriented Toolkit for Coarse-Graining Applications. *J. Chem. Theory Comput.* **2009**, *5*, 3211−3223.

(25) Karimi-Varzaneh, H. A.; Qian, H.-J.; Chen, X.; Carbone, P.; Müller-Plathe, F. IBIsCO: A molecular dynamics simulation package for coarse-grained simulation. *J. Comput. Chem.* **2011**, *32*, 1475−1487.

(26) 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.

(27) 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.

(28) Allen, M. P.; Tildesley, D. J. *Computer Simulation of Liquids*; Oxford science publications, Oxford University Press: New York, 1989.

(29) Frenkel, D.; Smit, B. *Understanding Molecular Simulation, Second ed.: From Algorithms to Applications (Computational Science)*; Academic Press: New York, 2001.

(30) Henderson, R. A uniqueness theorem for fluid pair correlation functions. *Phys. Lett. A* **1974**, *49*, 197−198.

(31) Schommers, W. Pair potentials in disordered many-particle systems: A study for liquid gallium. *Phys. Rev. A* **1983**, *28*, 3599−3605.

(32) Hess, B.; Holm, C.; Vegt, N. V. D. Modeling multibody effects in ionic solutions with a concentration dependent dielectric permittivity. *Phys. Rev. Lett.* **2006**, *96*, 1−4.
(33) Lyubartsev, A. P.; Laaksonen, A. MDynaMix - a scalable portable parallel MD simulation package for arbitrary molecular mixtures. *Comput. Phys. Commun.* **2000**, *128*, 565−589.
(34) Limbach, H.; Arnold, A.; Mann, B.; Holm, C. ESPResSo - an extensible simulation package for research on soft matter systems. *Comput. Phys. Commun.* **2006**, *174*, 704−727.
(35) Plimpton, S. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *J. Comput. Phys.* **1995**, *117*, 1−19.
(36) Lyubartsev, A. P.; Marcelja, S. Evaluation of effective ion-ion potentials in aqueous electrolytes. *Phys. Rev. E* **2002**, *65*, 041202.
(37) Mirzoev, A.; Lyubartsev, A. P. Effective solvent mediated potentials of Na+ and Cl- ions in aqueous solution: temperature dependence. *Phys. Chem. Chem. Phys.* **2011**, *13*, 5722−5727.
(38) Lobaskin, V.; Lyubartsev, A.; Linse, P. Effective macroion-macroion potentials in asymmetric electrolytes. *Phys. Rev. E* **2001**, *63*, 020401.
(39) Lyubartsev, A. P.; Laaksonen, A. Effective potentials for ion−DNA interactions. *J. Chem. Phys.* **1999**, *111*, 11207−11215.
(40) Shelley, J. C.; Shelley, M. Y.; Reeder, R. C.; Bandyopadhyay, S.; Klein, M. L. A Coarse Grain Model for Phospholipid Simulations. *J. Phys. Chem. B* **2001**, *105*, 4464−4470.
(41) Lyubartsev, A. P. Multiscale modeling of lipids and lipid bilayers. *Eur. Biophys. J.* **2005**, *35*, 53−61.
(42) Wang, Z.-J.; Deserno, M. A Systematically Coarse-Grained Solvent-Free Model for Quantitative Phospholipid Bilayer Simulations. *J. Phys. Chem. B* **2010**, *114*, 11207−11220.
(43) Lyubartsev, A. P.; Rabinovich, A. L. Recent development in computer simulations of lipid bilayers. *Soft Matter* **2011**, *7*, 25−39.
(44) Marrink, S. J.; de Vries, A. H.; Tieleman, D. P. Lipids on the move: Simulations of membrane pores, domains, stalks and curves. *Biochim. Biophys. Acta, Biomembr.* **2009**, *1788*, 149−168.
(45) Bennun, S. V.; Hoopes, M. I.; Xing, C.; Faller, R. Coarse-grained modeling of lipids. *Chem. Phys. Lipids* **2009**, *159*, 59−66.
(46) Humphrey, W.; Dalke, A.; Schulten, K. VMD − Visual Molecular Dynamics. *J. Mol. Graphics* **1996**, *14*, 33−38.
(47) Nosé, S. A molecular dynamics method for simulations in the canonical ensemble. *Mol. Phys.* **1984**, *52*, 255−268.
(48) Hoover, W. G. Canonical dynamics: Equilibrium phase-space distributions. *Phys. Rev. A* **1985**, *31*, 1695−1697.
(49) Parrinello, M.; Rahman, A. Crystal Structure and Pair Potentials: A Molecular-Dynamics Study. *Phys. Rev. Lett.* **1980**, *45*, 1196−1199.
(50) Nosé, S.; Klein, M. Constant pressure molecular dynamics for molecular systems. *Mol. Phys.* **1983**, *50*, 1055−1076.
(51) Högberg, C.-J.; Nikitin, A. M.; Lyubartsev, A. P. Modification of the CHARMM force field for DMPC lipid bilayer. *J. Comput. Chem.* **2008**, *29*, 2359−2369.
(52) Miyamoto, S.; Kollman, P. A. Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *J. Comput. Chem.* **1992**, *13*, 952−962.
(53) Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983**, *79*, 926−935.
(54) Hess, B.; Bekker, H.; Berendsen, H. J. C.; Fraaije, J. G. E. M. LINCS: A linear constraint solver for molecular simulations. *J. Comput. Chem.* **1997**, *18*, 1463−1472.