Molecular simulations of biomolecules: parametrisation and environment

Irene Marzuoli[1] and Franca Fraternali[1,*]

[1]Randall Centre of Cell and Molecular Biophysics, King's College London, London, UK
[*]franca.fraternali@kcl.ac.uk

Abstract

gffffffffffffffffff

1 Introduction

Molecular Dynamics simulations is a computational method which has gained popularity and significance in the past few decades to help the understanding of biophysical systems. The accumulation of data from experiments on biomolecular processes, and the increasing computational power available, have made possible to implement theoretical models of these systems which can be simulated on a computer, to access molecular dynamical properties inaccessible to experiments.

Modelling and simulating a biological system consists in describing its components and their mutual interactions, and implementing the laws of physics to reproduce its natural evolution. In principle, a quantum mechanic description of the system is needed but, to facilitate the task, several simplified models have been devised, each most suitable to investigate particular cases. In particular, a large class of models focuses on a classical mechanics description of the dynamics: for increasing sizes of the systems and longer time spans described, the classical approximations will be more and more accurate, and also the only possible representation computationally affordable.

Understanding the methodology of classical Molecular Dynamics (MD) provides the interpretative key with which simulations must be designed, run and interpreted in each specific case.

2 Materials

2.1 The evolution algorithm

In a classical MD framework, Newton's second law of motion rules the dynamics, stating that the acceleration $\mathbf{a}$ that a particle is subject to at time t, depend on the total force $\mathbf{F}$ acting on the particle itself and on its mass m (bold denotes vectorial quantities):

$$\mathbf{F}(t) = m \cdot \mathbf{a}(t). \tag{1}$$

As the acceleration $\mathbf{a}(t)$ is the second derivative of the position $\mathbf{r}(t)$ with respect to time, given the initial position and velocity of the particle $(\mathbf{r}(t_0), \mathbf{v}(t_0))$, their temporal evolution can be computed integrating $\mathbf{a}(t) = \mathbf{F}(t)/m$ as follow:

$$\mathbf{v}(t) = \mathbf{v}(t_0) + \int_{t_0}^{t} \frac{\mathbf{F}(t')}{m} dt'; \tag{2}$$

$$\mathbf{r}(t) = \mathbf{r}(t_0) + \int_{t_0}^{t} \mathbf{v}(t') dt' + \int_{t_0}^{t} \int_{t_0}^{t'} \frac{\mathbf{F}(t')}{m} dt'' dt'. \tag{3}$$

In the case of complex biomolecular systems with many particles and multiple interactions acting between them, it is impossible to integrate analytically Equation 2, while a different and feasible approach consists in discretising the equation of motion 1. The idea is to consider very short time steps of length Δt, so that in such intervals the forces are (almost) constant, and the integration of Equation 2 becomes trivial. A careful choice of the values to integrate allows to reduce the approximations derived from such approach. For example, choosing the velocity value at time $t_0 + \Delta t/2$ (and not at t_0) decreases the error down to orders of $(\Delta t)^4$ (rather than $(\Delta t)^2$), producing the so-called leap-frog algorithm, which is used in the vast majority of MD engines:

$$\mathbf{v}\left(t_0 + \frac{\Delta t}{2}\right) = \mathbf{v}\left(t - \frac{\Delta t}{2}\right) + \frac{\mathbf{F}(t)}{m} \Delta t; \tag{4}$$

$$\mathbf{r}(t_0 + \Delta t) = \mathbf{r}(t_0) + \mathbf{v}\left(t_0 + \frac{\Delta t}{2}\right) \Delta t. \tag{5}$$

This algorithm can thus "solve" every possible Newton equation, at the expenses of some precision.

2.2 Thermostats and barostats: rescuing the approximation limit

As the integration procedure is not exact, specific algorithms have been developed to alleviate the influence of the approximations and reproduce the simulation conditions of choice.

[REMOVE?] First, some bonds in the molecules are constrained to a particular length or angle. To ensure this property, after the update of all the atoms positions, a constraint algorithm is applied to bring the positions of mutually constrained atoms back to a value which satisfies the constraint. Several algorithms have been developed to do so, among which is LINCS (Linear Constraint Solver) [1]. As the problem of constraints is hardly solvable analytically, LINCS proceeds iteratively, finding the best solution within an approximation tolerance. On the contrary, the SETTLE algorithm [2] implements an exact solution to the constraint problem for rigid bodies of three elements, as the one of water in its atomistic description, for which it is usually employed.

Most experiments are conducted under constant temperature, so that it is desirable to reproduce this condition in simulations. To set up a temperature, at the beginning of the simulations particles can be given random initial velocities distributed according to the Maxwell-Boltzmann curve, which describe velocities of atoms of a noble gas at temperature T. The velocity of each of them will be influenced by the specific interactions occurring in the system but, in a constant temperature environment, the total average kinetic energy (proportional to T) must remain constant. Even in absence of any dissipative term in the dynamics, the approximations performed in the MD algorithm makes this quantity drift away from its initial value, therefore to ensure temperature is maintained throughout the simulation, thermostat algorithms have been devised.

The principle behind a thermostat consists in rescaling the velocity of one or few selected particles at fixed interval of times, to restore the correct average kinetic energy. This fixed

interval of time must not correspond to the timestep itself, as the goal of a thermostat is to maintain the average temperature, and not its value at all times, as fluctuations are allowed in natural systems. Moreover, it is strongly recommended to couple solute and solvent to two separate baths, to ensure that both maintain the correct temperature. Indeed, it is possible that biases in the parameterisation favour an energy flow from one component to the other. In this case, as the thermostat chooses at random a few particles to rescale their velocity, it is much more likely to select solvent ones, leaving the solute in its state incorrect temperature, if it is thermally coupled together with the solvent. The most used thermostat algorithms are the Berendsen [3], Nosé-Hoover [4, 5], Andersen [6] and velocity rescale [7], which differ in the quantities selected for such rescaling.

Another macroscopic condition one wish to maintain is either the volume or the pressure of the system. While maintaining the volume constant is easy (and, combined with constant temperature, gives the NVT ensemble), pressure regulation (i.e. maintaining a NPT ensemble) requires a barostat. Pressure is directly proportional to the average quantity of motion exchanged between the particles and the walls of the box they are confined to, which depends on the frequency of collision and thus on the extent of the box. Barostats change the box size to regulate the pressure. To be noticed that most MD simulations are run under periodic boundary conditions, i.e. a particle which exits from the simulation box during a move is brought back on the opposite side. This mimics the presence of an infinite number of equivalent boxes one next to the other, to alleviate the finite size effects that arise when simulating small systems. In this scenario, particles are not bouncing on the box walls, rather a virtual pressure is computed from the velocities of the ones trespassing the box boundaries during a move. Similarly as for thermostats, the coupling frequency of barostats must be larger than the time step. Moreover, usually all the box dimensions are rescales by the same amount. However, in the case of anisotropic systems like lipids patches, the directions parallel to the membrane plane can be rescaled separately with respect to the one perpendicular, to maintain the correct pressure in each of them. Also for pressure coupling several algorithms can used: the Berendsen [3], Parrinello-Rahman [8], or Martyna-Tuckerman-Tobias-Klein

(MTTK) [9] (to be used in conjunction with the Nosé-Hoover thermostat), which differ in the way they approach the desired pressure (e.g. exponentially, in an oscillatory way, etc.).

2.3 Force fields

2.3.1 Force field definition

Once the equations of motion and the control algorithms are set up, the next challenge is represented by modelling the forces and thus the potential energy function. Force fields for classical MD simulations usually rely on the breakdown of interactions into several, independent terms, identified on an empirical physical basis. We report here the functional form adopted for the GROMOS force field [10, 11] as implemented in the GROMACS MD engine [12–14], as an example of the structure of a classical force field.

Covalent (bonded) interactions Covalent interactions are modelled with potential energy terms representing bond stretching, angle bending, improper and proper dihedral angles torsion. The functional form of the potential energy function for each of them aims at a simplified, semi-classical description of the sub atomic motion of molecules. Often, it is modelled as a harmonic-like vibration around the equilibrium position, regulated by a constant with suitable energy units.

Type	Eq. pos.	Const.	[Const.]	Functional form
Bond	b_{ij}	k_{ij}^b	$\frac{\text{kJ}}{\text{mol}\,\text{m}^2}$	$V_b(\mathbf{r}_{ij}) = \frac{1}{4}\,k_{ij}^b\left(\lvert\mathbf{r}_{ij}\rvert^2 - b_{ij}^2\right)^2$
Angle	θ_{ijk}^0	k_{ijk}^θ	$\frac{\text{kJ}}{\text{mol}}$	$V_a(\theta_{ijk}) = \frac{1}{2}\,k_{ijk}^\theta\left(\cos(\theta_{ijk}) - cos\left(\theta_{ijk}^0\right)\right)^2$
Dihedral	ϕ_{ijkl}^0	k_{ijkl}^ϕ	$\frac{\text{kJ}}{\text{mol}\,\text{rad}^2}$	$V_d(\phi_{ijkl}) = k_{ijkl}^\phi\left(1 + \cos\left(n\,\phi_{ijkl} - \phi_{ijkl}^0\right)\right)$
Improper	ξ_{ijkl}^0	k_{ijkl}^ξ	$\frac{\text{kJ}}{\text{mol}}$	$V_{id}(\xi_{ijkl}) = \frac{1}{2}\,k_{ijkl}^\xi\left(\xi_{ijkl} - \xi_{ijkl}^0\right)^2$

$$(6)$$

In the GROMOS force field, this translates in the equations displayed in Table 6, where for proper dihedrals, the convention states that ϕ_{ijkl} is the angle between the (i, j, k) and (j, k, l) planes; with i, j, k, and l four subsequent atoms, for example along a protein backbone. A value of zero for ϕ_{ijkl} corresponds to a *cis* configuration and π to a *trans*. The integer

n denotes the number of equally spaced energy minima available in a $360°$ turn. The same conventions hold for improper dihedrals ξ_{ijkl}, which are used to ensure ring planarity and control the chirality of tetrahedric centres.

It must be noticed that these types of potentials can not model the rupture of a bond: for this, more sophisticated descriptions are needed.

Non bonded interactions Non bonded interactions include the short range Pauli repulsion, the "mid"-range van der Waals attraction, and the long range electrostatic term.

The first two can be modelled together by a Lennard-Jones potential. Its functional form, describing the interaction between two neutral atoms at distance r, models the long range dispersion with a r^6 behaviour typical of the dipole-dipole interactions found in noble gases (London dispersion forces), while the Pauli term is represented by a r^{12} behaviour to ease the computation in relation with the previous one:

$$V_{LJ}(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right]. \tag{7}$$

Two parameters, ϵ and σ, tune the interaction strength and the equilibrium distance. They are fitted against experimental data and are specific of each pair of atoms species.

The Coulomb energy between two charges q_1 and q_2 at distance r is represented by the Coulomb law:

$$V_C(r) = \frac{1}{4\pi\epsilon_0} \frac{q_i q_j}{\epsilon_r r_{ij}} \tag{8}$$

with ϵ_0 the dielectric constant of vacuum and ϵ_r the relative dielectric constant, introduced to properly take into account the screening provided by the material surrounding the object, as polarisability is not included in this description.

The treatment of non-bonded interactions requires particular care because of their long range nature: in every point of the simulation box many forces from distant atoms are acting at the same time, making the prediction of the outcome difficult. The van der Waals forces decay fast, therefore the tail of their functional can be cut after a threshold distance with

little impact on the outcome; while Coulomb interactions, with their slower decay, must be taken into account throughout the whole simulation box. Many algorithms have been devised to efficiently compute them, like the Particle Mesh Ewald [15] or the Reaction Field [16] approaches.

Finally, all biomolecular force field, and in particular their van der Waals interactions, are parametrised to describe systems at room temperature, therefore simulations performed at substantially different temperatures must be interpreted carefully.

2.3.2 Force fields: classifications

Many force fields for classical MD simulations adopt a functional form equal or similar to the one described above. Their difference lies in the number of degrees of freedom modelled, in a hierarchy of descriptions proceeding from detailed to coarse (Figure **??**). Three possible classes of descriptions are:

- all-atoms force fields, where all the atoms are presented in the description, and represented as spheres of variable size according to their van der Waals radius (e.g. proportional to σ in a Lennard-Jones model). Examples of all-atoms force fields are AMBER [17] and CHARMM [18, 19].

- united atoms force fields, similar to the previous ones but where non-polar hydrogens are incorporated in the heavy atom they are bonded to. The "united atom" is given a new σ parameter and increased mass according to how many hydrogen it includes. The GROMOS force field [10, 11] follows this philosophy.

- coarse grain force fields, which group together in one unique bead few atoms, to reduce the number of variables to compute. The clustered atoms are such that their mutual distances are expected to vary little with respect to the ample movements of components of the system far away from each other (which will be grouped in different beads). The MARTINI [20–22] and SIRAH [23, 24] force field belong to this category.

We now give a more detailed insight in the characteristics and parametrisation strategies an

atomistic and a coarse grain force field among the ones mentioned.

2.3.3   The GROMOS force field

All-atoms and united atoms force field are parametrised against experimental values. While for the all-atom force fields AMBER and CHARMM the parametrisation is based on quantum mechanics calculations, the united atom GROMOS force field relies on the reproduction of free enthalpies of solvation and heat of vaporization of small molecules at physiological temperatures and pressures. This procedure sets not only the constant of the bonded interactions, but also the partial charges of the atoms inside a molecule: as no electrons are included for the sake of efficiency, their redistribution across atoms which are bonded is modelled through fractional charges assigned to each atom (while the total charge of a molecule must sum to an integer). Moreover, it is assumed that the parametrisation performed for small moieties can be transferred to a larger compound including these moieties. This limits the number of chemical groups to be described in order to simulate biomolecules.

In every MD simulation, the description of water is crucial. Out of the many water model proposed, the GROMOS parametrisation has been performed with a flexible simple point charge (SPC [25]) model. This description represents water as a three atoms molecule, with a negative charge on the oxygen and a positive complementary one on the two hydrogen atoms, and allowing flexible hydrogen-oxygen bonds. This model reproduces correctly the density and dielectric permittivity of water.

The improvement of computational techniques and reparametrisation strategies prompts the periodical release of newer versions of force fields. Accordingly, the latest version of the GROMOS force field, version 54a8 [11], has been released in 2011.

2.3.4   The MARTINI force field

The MARTINI force field is a popular coarse grain description of biological molecules [20–22]: developed originally with a focus on lipids, it has been then extended to include proteins, small ligands and DNA/RNA molecules.

MARTINI opts for a four-to-one approach, i.e. four heavy atoms are grouped in one bead. The number of bead types has been kept to the minimum necessary to represent biological molecules. They are organised systematically in polar, non-polar, apolar, or charged, and each type has a number of subtypes with increasing polarity to differentiate the chemical nature of the underlying atomistic structures. This systematic approach can be easily transferred to new compounds, without the need of introducing new bead types. The only exception is represented by rings molecules, where a two-to-one approach is needed to maintain the circular topology.

Two approaches are possible develop a coarse grain description: parameters can be fit directly to global quantities derived from experiments, in a top-down approach as the one followed in GROMOS; or fitting the coarse-grain simulations results to the ones from atomistic simulations, in a bottom-up approach. The MARTINI force field chooses a top-down approach to parametrise non-bonded interactions, tuning them against experimental partitioning free energies between polar and apolar phases, while bonded interactions are derived from reference all-atom informations, in a bottom-up approach.

The four-to-one mapping implies that the amino acid backbone is represented by one bead only, preventing the description of directional bonds which are key to reproduce the secondary structure. The bonded parameters partially account for this, favouring for each residue type the backbone conformation in which it is most likely found (as computed from the Protein Data Bank - PDB [26]). When this is not sufficient, the protein can be constrained around a given structure through an elastic network model approach (ElNeDyn [27]). However, both the backbone parametrisation and the use of ElNeDyn imply that conformational changes in the structure are penalised and therefore not well sampled in MARTINI simulations.

The MARTINI force field provides two water model. The standard one groups four water molecules in one bead only, loosing the polarisability typical of water molecules, the effect of which is partially restored with the use of a high dielectric constant. The polarisable water model [28] maps instead four water molecule to a single "inflated" water, i.e. a three-beads molecule with the same shape of a single molecule, but expanded, and a charge splitting

which can account for the water dipole.

Backmapping techniques Coarse-grain descriptions are very effective in reproducing long time scales; however, to retrieve finer details after such extensive exploration, backmapping techniques have been designed to obtain atomistic configurations from the coarse-grain ones. The easy transfer between the two resolution, gave rise to many multi scale studies applied to biomolecular systems [29].

2.4   Beyond a classical atomistic framework

Without entering into the details, we want to bring to the reader attention two possible refinements of the aforementioned models, and two computational strategies which on the contrary speed up the calculations at the expenses of the loss of some details.

Regarding the accuracy of simulations, it must be noticed that none of the force field mentioned above takes into account polarisability, i.e. the displacement of electrons with respect to the nucleus, as a consequence of the surrounding electrostatic environment, because electrons and nucleus of an atom are modelled as a unique object. Specific force fields have been modelled to include this effect, on top of atomistic descriptions, as in the AMOEBA [30, 31], Drude polarisable CHARMM [32] or AMBERff02 [33], or on coarse grain descriptions, as in the ELBA force field [34]. Polarisability does improve the accuracy of simulations, but it can significantly slow down simulations. **[REMOVE?] Further beyond the classical approximation, for biological processes governed by quantum mechanics - such as photosynthesis, DNA mutation processes or some enzymatic activities - many semi classical hybrid techniques have been developed, to gain the accuracy of a quantum description in the region of interest and the speed up of a classical one in the surrounding areas [35].**

Tackling instead efficiency issues, an implicit solvent model can be used to speed up simulations. The solvent is represented as a continuous medium, as opposed to explicit models which include all its particles. Models of implicit solvent can be based on different assumptions: for example the solute-solvent interactions can be taken as proportional to

the solvent-accessible surface area (SASA) of every particle of solute [36], or instead can be derived from a solution of the Poisson-Boltzmann equation governing the charge density in a material, for example in the form of the Generalised Born equation [37] which is valid under particularly simple conditions. Another speed up technique is constituted by hybrid particle-field algorithms. The idea is to treat non-bonded interactions through a mean field approach, where atoms/beads move in the field generated by the others. The field does not need to be updated at every time step, as it is a collective and thus slowly evolving variable; moreover, for each particle only the interaction with the field, and not with all the neighbour particles needs to be computed, reducing the computations effort further. This approach has been employed with a coarse grain description of polymers and biological molecules in the OCCAM software [38].

Finally, further strategies are possible to enhance the sampling performed by a simulation in the case that the one obtained by the natural evolution of the system would exceed the computational time available. As a non comprehensive list of these techniques, we mention replica-exchange algorithms [39] which combine together multiple simulations held at different conditions, local potential-energy elevation (or metadynamics) [40, 41] which avoids the re-sampling of already visited conformations adding an energy penalty to them, umbrella sampling [42] which reconstructs free-energy barriers from simulations held at specific values of the coordinate along which the barrier exists, or finally simply the use of higher temperature to overcome energy barriers [43].

2.5 System setup

To set up a simulation, the desired resolution, i.e. force field, must be chosen according to the system to simulate. A few examples on this will be given in Section 3; however, for any choice of parameters, some common "ingredients" and steps are necessary to prepare the system:

- **structural file**: a pdb file (or any structural file format suitable with the MD engine of choice) which contains only the elements to be simulated, with a correct format of

11

names and positions. The choice of the initial conformation is particularly important, especially for atomistic resolution, which is likely to sample conformations in the vicinity of the initial position if it possible to simulate only a short time scale.

- **topology file**: given a structural file, every MD engine has dedicated commands to retrieve the list of parameters for bonded atoms. The presence of exotic residues or non standard network of bonds might need case-to-case manual parametrisation.

- **simulation box**: for simulations of a protein, it is good practice to place it within a box large enough to avoid interference of the protein with its periodic images, for example allowing a minimum distance between protein and box of at least the cut-off chosen for non-bonded interactions. However, it must be considered that the protein might extend during the simulation. In the case of lipids instead, to reproduce an infinite membrane and account for lateral tension, the box can be chosen exactly as big as the membrane patch (in the directions parallel to its plane), so that periodic images merge together.

- **solvation**: the prepared box is filled with molecules of the water model of choice, except in the spaces already occupied by the solute.

- **ions**: ions are added in replacement of a suitable number of water molecules to neutralise the charge of the system. Additional ones can be added to reach the experimental ion concentration (though there will be a slight imbalance between the positive and negative species due to the counter ions already added).

- **energy minimisation**: the system prepared according to the steps above might have several atom clashes. This can derive from the initial protein/membrane structure, or from imperfect packing of the solvent around the solute. To alleviate this, an energy minimisation is performed prior to the dynamical run. Usually, one iteration is performed restraining the solute, to relax solvent positions, and then a second one with free solute.

- **equilibration**: even after energy minimisation, the system prepared is far away from the ideal equilibrium configuration. For this reason, several rounds of simulations with specific conditions are run before the final production. The specific protocol depends on the system and the force field, however for proteins in solution it is quite common to start with NVT ensemble runs with position restraints at increasing temperatures, followed by NPT runs at the same temperatures but without restraints, and only then from the production. For lipid system, the ensemble is preferably always NPT to prevent penetration of water in the bilayer by shrinking the box if needed, thus only a temperature increase protocol is followed. In some cases, for example for the MARTINI force field with standard water, it is not possible to simulate low temperatures, thus the equilibration consists in a short production employing different thermostat and barostat with respect to the production. Indeed, some are more suitable for approaching efficiently the correct temperature or pressure value from a configuration far from equilibrium, while others for maintaining the correct ensemble. Generally, every force field provides validated equilibration procedures.

3 Methods: relevant examples

One of the challenges of MD simulations consists in choosing the most suitable granularity of the description, together with the choice of the system to simulate. We list here a few examples which might convey better the idea of suitable setup for systems of current interest.

Simulations of protein and their interactions Simulations of protein have been crucial to understand small molecular details of their structure and functioning, which influence their macroscopic behaviour.

A challenging subject in the field of enzymatic regulation is constituted by allosteric regulation, as the pathways involved are of difficult exploration. Molecular dynamics simulations can shed light on the residues responsible for transferring the information. Atomistic simulations of pre-assembled pyruvate kinase M2 (PKM2) tetramers elucidated the mecha-

nism of allosteric activation by fructose 1,6-bisphosphate (FBP) [44]: simulations of the apo and FBP bound tetramer does not show large conformational changes, however the correlated motions were extracted from both simulations and processed within a consistent information-theoretical framework, showing differences. This made possible to extract the position of some hubs, i.e. sets of amino acids, which were most influenced by the ligand presence and thus able to transport information. Such analysis necessarily requires an atomistic description as the signal comes form small modifications of the protein secondary structure which must then be accurately described. The analysis proved that residues in regions with high backbone flexibility were more prone to have an allosteric role and allowed to designed mutants with improved response to the FBP activator and less sensibility to a L-phenylalanine inhibitor.

Another particularly interesting concept explored by MD is the one of *ensemble*: proteins adopt a variety of shapes in solution, which can differ significantly from the crystal-structures available. This flexibility is proven by techniques as small angle X-ray scattering (SAXS) or nuclear magnetic resonance (NMR), which results cannot been explained by a single-conformation scenario. As such, simulations can provide "snapshots" of all these conformations and their mutual interchange. Some of these changes are crucial for the protein function, as in the case of prions, proteins which become pathogenic under conversion from an α-helix rich form to a β-sheet isoform which is prone to aggregation. Atomistic MD simulations showed the (reversible) unfolding of the protein between the two phases [45], starting from the helical crystal structure available. Moreover, they identified in the C-terminus region the trigger of the conformational change, corresponding to a region which is enriched in pathogenic mutations. The results were then used to investigate the aggregation process after the conversion, again identifying critical residues promoting it [46]. A similar procedure applied to simpler amyloid forming peptides provided recently a map of all the relevant states of the assembly process [47], suggesting that the constant improvement of computational power will make this possible also for more complex systems like the prion in a near future.

Simulations of model membranes   Membrane simulations had a key role in elucidating the mechanical and dynamical characteristics of these important organelles clarifying for

14

example how the lipids composing the membrane influence its fluidity [], or elucidating the interactions between lipids and surface proteins, transmembrane ones or membrane active peptides, such as antimicrobial ones [].

The last point in particular raised the interest in simulating bacterial membranes. As these objects are very complex, simplified models can be used to approach the problem. Very often, these models use only one or two lipid species to represent membranes, e.g. bacterial ones have contains neutral phospholipids with a percentage of negatively charged ones [48–51] as key characteristic distinguishing them from mammal membranes, which possess only zwitterionic phospholipids, with the occasional inclusion of cholesterol, deemed important in achieving the flexibility typical of mammal membranes [48–52]. Because of their simplicity, these systems are extensively used also in experiments [53–55], making a direct comparison with simulations possible. Nevertheless, attempts to model more accurately cell membranes have been pursued.

The atomistic level is suitable to reproduce single components of the membrane, e.g. the isolated inner membrane of Gram-negative bacteria [56], even if atomistic computations started in recent years to be affordable also for larger systems. For example, atomistic simulation of the outer membrane of Gram negative bacteria combined with the peptoglycan layer (which is positioned between the two membranes) elucidated how the distance is variable, thanks to the presence of Braun's lipoproteins which act as a bridge between the two, and can bring them closer by bending and tilting []. Moreover, the permeability of membranes to ions and small compounds needs to be assessed at the atomistic level to get sufficient accuracy, and because of the intensive task, often enhanced MD techniques such as umbrella sampling are employed [57].

Coarse grain descriptions are instead the most suitable to represent the full bacterial envelope, especially in the case of Gram-negative bacteria, as the inclusion of all its elements results in large systems.

Accordingly, MARTINI simulations have been able to reproduce the behaviour of several transmembrane proteins which spanned both the inner and outer membrane [58] and to model

all the different components of the Gram-negative cell envelope [**Khalid2019**]. Similarly, the ability of coarse grain simulations to investigate very large systems makes them suitable to assess elastic properties of membranes, as they can access low frequency undulations with little influence from finite size effects [59].

4   Notes: validation and challenges of MD simulations

Validation of MD simulations is performed by comparison with experiments: the same properties obtained experimentally are computed from the MD trajectory as well, and the two compared. If these are correctly reproduced, it is usually assumed that the simulation is sampling the correct ensemble of states and then one can identify in the simulation the determinants responsible for the experimental outcome of interest, which are not accessible by the experiment itself.

The comparison however is not always easy: often the experimentally measured quantity is an average in time and/or space (for example Circular Dichroism spectra or SAXS profiles of a peptide in solution) and many different combinations of computationally derived structures can produce compatible results. Indeed, the experimental information is always limited in comparison with the more abundant one handled by MD simulations. This constitutes the strength of MD, but also prevents the validation of some findings.

Thus, in the validation of MD outcomes, it is important to have a critical attitude both when the results agree and when they do not. Indeed, agreement may arise from either a simulation that reflects correctly the experimental system; but also when the property examined is insensitive to the details of the simulated trajectory and thus always agrees with experiments, or again from a compensation of errors, which happens more easily for systems with a high number of degrees of freedom. Similarly, disagreement may hint at an error in the simulation (either in the model, the implementation or simply the simulation is not converged yet) or an error in the experiment (either in the result itself o its interpretation), so that both must be carefully checked to improve the agreement.

Finally, simulations still suffer from the limited computational time available: most of

the times, the real experimental system is simply too large to be reproduced and the time scale of the process too long. Simulations are thus confined to explore a restricted space, implying that the initial conditions must be chosen carefully to optimise the search and avoid any bias which might persist for the whole length of the simulation. The use of enhanced MD techniques can increase the sampling abilities, however it introduces a bias which must be removed or properly accounted for in the interpretation of the results. Finally, the force field used are far from optimal: partly because they rely on approximate functional forms, and partly because it is difficult to find experimental observables measured with the desired resolution to set the parameters, it is not unusual to see different descriptions providing disagreeing outcomes, which still all match the experimental results. This prompts at the set up of new experiment which can properly discriminate the best description, but at the same time highlights once more how only the resolution provided by simulations can disclose the fine details behind biophysical processes.

References

(1) Hess, B.; Bekker, H.; Berendsen, H. J. C.; Fraaije, J. G. E. M. *Journal of Computational Chemistry* **1997**, *18*, 1463–1472.

(2) Miyamoto, S.; Kollman, P. A. *Journal of Computational Chemistry* **1992**, *13*, 952–962.

(3) Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. *The Journal of Chemical Physics* **1984**, *81*, 3684–3690.

(4) Nosé, S.; Klein, M. *Molecular Physics* **1983**, *50*, 1055–1076.

(5) Hoover, W. G. *Physical Review A* **1985**, *31*, 1695–1697.

(6) Andersen, H. C. *The Journal of Chemical Physics* **1980**, *72*, 2384–2393.

(7) Bussi, G.; Donadio, D.; Parrinello, M. *The Journal of Chemical Physics* **2007**, *126*, 014101.

(8) Parrinello, M.; Rahman, A. *Journal of Applied Physics* **1981**, *52*, 7182–7190.

(9) Martyna, G. J.; Tuckerman, M. E.; Tobias, D. J.; Klein, M. L. *Molecular Physics* **1996**, *87*, 1117–1157.

(10) Oostenbrink, C; Villa, A; Mark, A. E.; Van Gunsteren, W. F. *Journal of Computational Chemistry* **2004**, *25*, 1656–1676.

(11) Schmid, N.; Eichenberger, A. P.; Choutko, A.; Riniker, S.; Winger, M.; Mark, A. E.; van Gunsteren, W. F. *European Biophysics Journal* **2011**, *40*, 843–856.

(12) Berendsen, H.; van der Spoel, D.; van Drunen, R. *Computer Physics Communications* **1995**, *91*, 43–56.

(13) Abraham, M. J.; Murtola, T.; Schulz, R.; Páll, S.; Smith, J. C.; Hess, B.; Lindahl, E. *SoftwareX* **2015**, *1-2*, 19–25.

(14) Abraham, M. J.; van der Spoel, D; Lindahl, E; Hess, B; the GROMACS development team, *GROMACS User Manual version 2016*, www.gromacs.org, 2018.

(15) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G. *The Journal of Chemical Physics* **1995**, *103*, 8577–8593.

(16) Tironi, I. G.; Sperb, R.; Smith, P. E.; van Gunsteren, W. F. *The Journal of Chemical Physics* **1995**, *102*, 5451–5459.

(17) Maier, J. A.; Martinez, C.; Kasavajhala, K.; Wickstrom, L.; Hauser, K. E.; Simmerling, C. *Journal of Chemical Theory and Computation* **2015**, *11*, 3696–3713.

(18) MacKerell, A. D. et al. *The Journal of Physical Chemistry B* **1998**, *102*, 3586–3616.

(19) Klauda, J. B.; Venable, R. M.; Freites, J. A.; O'Connor, J. W.; Tobias, D. J.; Mondragon-Ramirez, C.; Vorobyov, I.; MacKerell, A. D.; Pastor, R. W. *The Journal of Physical Chemistry B* **2010**, *114*, 7830–7843.

(20) Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; De Vries, A. H. *Journal of Physical Chemistry B* **2007**, *111*, 7812–7824.

(21) Monticelli, L.; Kandasamy, S. K.; Periole, X.; Larson, R. G.; Tieleman, D. P.; Marrink, S. J. *Journal of chemical theory and computation* **2008**, *4*, 819–34.

(22) De Jong, D. H.; Singh, G.; Bennett, W. F. D.; Arnarez, C.; Wassenaar, T. A.; Schäfer, L. V.; Periole, X.; Tieleman, D. P.; Marrink, S. J. *Journal of Chemical Theory and Computation* **2013**, *9*, 687–697.

(23) Machado, M. R.; Guisasola, E. E. B.; Klein, F.; Sóñora, M.; Silva, S.; Pantano, S. *bioRxiv* **2018**, 436774.

(24) Barrera, E. E.; Machado, M. R.; Pantano, S. *bioRxiv* **2019**, 627570.

(25) Berendsen, H.; Postma, J.; Van Gunsteren, W.; Hermans, J. In *Intermolecular Forces*, Pullman, B., Ed.; D. Reidel Publishing Company: Dordrecht, 1981, pp 331–342.

(26) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The Protein Data Bank., Nucleic Acids Research, 28: 235-242, 2000.

(27) Periole, X.; Cavalli, M.; Marrink, S.-J.; Ceruso, M. A. *Journal of Chemical Theory and Computation* **2009**, *5*, 2531–2543.

(28) Yesylevskyy, S. O.; Schäfer, L. V.; Sengupta, D.; Marrink, S. J. *PLoS Computational Biology* **2010**, *6*, ed. by Levitt, M., e1000810.

(29) Lee, O.-S.; Cho, V.; Schatz, G. C. *Nano Letters* **2012**, *12*, 4907–4913.

(30) Ren, P.; Ponder, J. W. *Jurnal of Physical Chemistry B* **2003**, *24*, 5933–5947.

(31) Ponder, J. W.; Wu, C.; Ren, P.; Pande, V. S.; Chodera, J. D.; Schnieders, M. J.; Haque, I.; Mobley, D. L.; Lambrecht, D. S.; DiStasio, R. A.; Head-Gordon, M.; Clark, G. N. I.; Johnson, M. E.; Head-Gordon, T. *The Journal of Physical Chemistry B* **2010**, *114*, 2549–2564.

(32) Anisimov, V. M.; Lamoureux, G.; Vorobyov, I. V.; Huang, N.; Roux, B.; MacKerell, A. D. *Journal of Chemical Theory and Computation* **2005**, *1*, 153–168.

(33) Cieplak, P.; Caldwell, J.; Kollman, P. *Journal of Computational Chemistry* **2001**, *22*, 1048–1057.

(34) Orsi, M.; Essex, J. W. *PLoS ONE* **2011**, *6*, e28637.

(35) Ahmadi, S.; Barrios Herrera, L.; Chehelamirani, M.; Hostaš, J.; Jalife, S.; Salahub, D. R. *International Journal of Quantum Chemistry* **2018**, *118*, e25558.

(36) Fraternali, F; van Gunsteren, W. *Journal of Molecular Biology* **1996**, *256*, 939–948.

(37) Zhu, J.; Alexov, E.; Honig, B. *The Journal of Physical Chemistry B* **2005**, *109*, 3008–3022.

(38) Milano, G.; Kawakatsu, T. *The Journal of Chemical Physics* **2009**, *130*, 214106.

(39) Okamoto, Y. *Journal of Molecular Graphics and Modelling* **2004**, *22*, 425–439.

(40) Huber, T.; Torda, A. E.; van Gunsteren, W. F. *The Journal of Physical Chemistry A* **1994**, *101*, 5926–5930.

(41) Laio, A.; Parrinello, M. *Proceedings of the National Academy of Sciences of the United States of America* **2002**, *99*, 12562–6.

(42) Torrie, G.; Valleau, J. *Journal of Computational Physics* **1977**, *23*, 187–199.

(43) Kirkpatrick, S.; Gelatt, C. D.; Vecchi, M. P. *Science* **1983**, *220*, 671–680.

(44) Macpherson, J. A.; Theisen, A.; Masino, L.; Fets, L.; Driscoll, P. C.; Encheva, V.; Snijders, A. P.; Martin, S. R.; Kleinjung, J.; Barran, P. E.; Fraternali, F.; Anastasiou, D. *eLife* **2019**, *8*.

(45) Chakroun, N.; Fornili, A.; Prigent, S.; Kleinjung, J.; Dreiss, C. A.; Rezaei, H.; Fraternali, F. *Journal of Chemical Theory and Computation* **2013**, *9*, 2455–2465.

(46) Collu, F.; Spiga, E.; Chakroun, N.; Rezaei, H.; Fraternali, F. *Chemical Communications* **2018**, *54*, 8007–8010.

(47) Sengupta, U.; Carballo-Pacheco, M.; Strodel, B. *The Journal of Chemical Physics* **2019**, *150*, 115101.

(48) Lipkin, R.; Pino-Angeles, A.; Lazaridis, T. *The Journal of Physical Chemistry B* **2017**, *121*, 9126–9140.

(49) Wang, Y.; Schlamadinger, D. E.; Kim, J. E.; McCammon, J. A. *Biochimica et Biophysica Acta (BBA) - Biomembranes* **2012**, *1818*, 1402–1409.

(50) Zhao, L.; Cao, Z.; Bian, Y.; Hu, G.; Wang, J.; Zhou, Y. *International Journal of Molecular Sciences* **2018**, *19*, 1186.

(51) Chen, C.; Starr, C. G.; Troendle, E. P.; Wiedman, G.; Wimley, W. C.; Ulmschneider, J. P.; Ulmschneider, M. B. *Journal of the American Chemical Society* **2019**, jacs.8b11939.

(52) Risselada, H. J.; Marrink, S. J. *Proceedings of the National Academy of Sciences of the United States of America* **2008**, *105*, 17367–72.

(53) Castelletto, V.; de Santis, E.; Alkassem, H.; Lamarre, B.; Noble, J. E.; Ray, S.; Bella, A.; Burns, J. R.; Hoogenboom, B. W.; Ryadnov, M. G.; Vandesompele, J.; Wittwer, C. T. *Chem. Sci.* **2016**, *7*, 1707–1711.

(54) Tang, M.; Hong, M. *Molecular BioSystems* **2009**, *5*, 317.

(55) Glukhov, E.; Stark, M.; Burrows, L. L.; Deber, C. M. *The Journal of biological chemistry* **2005**, *280*, 33960–7.

(56) Piggot, T. J.; Holdbrook, D. A.; Khalid, S. *The Journal of Physical Chemistry B* **2011**, *115*, 13381–13388.

(57) Carpenter, T. S.; Parkin, J.; Khalid, S. *The Journal of Physical Chemistry Letters* **2016**, *7*, 3446–3451.

(58) Hsu, P.-C.; Samsudin, F.; Shearer, J.; Khalid, S. *The Journal of Physical Chemistry Letters* **2017**, *8*, 5513–5518.

(59) Fowler, P. W.; Hé, J.; Duncan, A.; Chavent, M.; Koldsø, H.; Sansom, M. S. P. *Soft Matter* **2016**, *12*, 7792.