Computational microscopy of the supramolecular organization of the respiratory chain complexes

Arnarez, Clement

CHAPTER II

Computational Methodology

C. Arnarez

Department of Molecular Dynamics, University of Groningen, The Netherlands

Abstract

In this chapter I will first introduce the general principles of the computational approach used in this thesis. The main theory and equations of molecular dynamics will then be discussed, followed by an overview of the definitions and parameterization of molecular interactions. In the final section, I will present the general simulation set-up as used in the studies included in this thesis

"There are currently three types of science: theoretical and experimental of course, which have to count now with computational. This latter would not exist without theoretical studies, and would not make sense without experimental results."

— Wanda Andreoni, Zernike Chair Lecture, 2011

Computational sciences

Since the invention of the computer — ENIAC, the first programmable computer with electronic switches built by the United States Army to compute ballistics, and later to conduct calculations on the hydrogen bomb [37] — and the demonstration of its potential for using mathematical methods to process vast computations, the field of computational sciences and numerical methods has risen to become an intellectual discipline by itself. The miniaturization and fine-tuning of electronic components, as well as the decrease in production cost, has lead to the rapid popularization of computers that we know today. Uncountable theories have been converted into algorithms in the past half-century; with the exponential increase of computational power and the revolution brought about by parallel computing, computational sciences have grown to be a powerful and indispensable tool. Computational methods are now applied daily in a wide range of established fields, such as: market prediction, fluid dynamics, nuclear engineering, medical imaging, behavioral anthropology and sociology, and, of course, chemistry, which is the field we will evolve in this thesis.

In computational chemistry, simulations are used to provide knowledge about the time and/or conformational evolution of molecules. Many techniques coming from a large variety of theoretical fields have been developed and different approaches, as well as levels of representation, are employed. Roughly, two main branches can be defined: the methods applying quantum mechanics to describe particles, including ab initio, density functional and (semi-)empirical methods; and the methods employing classical mechanics, including molecular dynamics and molecular mechanics.

The technique exclusively used in the studies presented in this thesis is molecular dynamics (MD). MD as we know it today was initially designed and tested in the mid-fifties. The first example of an MD simulation, describing a very simple system composed of a linear string made of 64 particles, was published in 1955 [38]. The forces resulting from the interactions between particles were described by non-linear potentials. Very quickly, the complexity of systems studied through this emergent technique increased: simulation of a two-dimensional system composed of spheres was published in 1957 [39] — often considered to be the first real MD paper — and the basis of the approach was finalized two years later [40]. This technique is now daily used as a complement to experiments, explaining at an atomic level the experimental observations, or as a quicker technique (in opposition to time- and resources-consuming series of experimentations) to narrow the range of experiments to perform by eliminating improbable and unfavorable research directions. Alternatively, MD simulation is a solution of choice to study fine interactions between molecules, something unreachable experimentally. This fact earned MD the nickname of "computational microscopy".

Molecular Dynamics

Equations of motion: MD is a widely used technique to study the time evolution of atoms, described as point particles, and thus the conformational space accessible to them. Different flavors of MD exist; the most common ones employ classical mechanics to describe the particles' interactions and displacements. In classical MD, Newton's laws of motion are applied to each particle in the system of interest:

$$F_i = m_i \cdot \ddot{r}_i(t) \tag{2.1}$$

Here F_i is the force, m_i the mass and $\ddot{r}_i$ the acceleration of the particle *i* with the coordinates r_i and at time *t*. The force is obtained from the derivative of the potential V_i felt by the particle *i*:

$$F_i(t) = -\frac{\partial V_i(t)}{\partial r_i(t)} \tag{2.2}$$

Solved numerically — most of the time through massively parallelized computations — these equations are used to predict the positions of the interacting particles every step of the period of time simulated. Thus, a trajectory of the system is generated and statistical mechanics can be used to compute macroscopic properties. To integrate the equations of motion, different methods can be applied; in this section, we will concentrate on two of the integrators implemented in the GROMACS simulation package [41,42] used to generate the data presented in this thesis: the *md* and *sd* integrators.

In the simplest implementation of the *md* integrator in GROMACS, a Leapfrog algorithm is applied to solve the differential equation constituting Newton's second law of motion (Eq. 2.1). This algorithm gets its name from the fact it updates positions and velocities of particles at equidistant time points — defined by the integration time step Δ*t* — positioned in such a way that they "leapfrog" over each other: the positions r_i are updated at each time point *t*, while velocities, $\dot{r}_i$, at each half time point *T* + ½ Δ*t*. The solutions of the equations of motion for each particle can be written as:

$$\dot{r}_i\left(t + \frac{1}{2}\Delta t\right) = \dot{r}_i\left(t - \frac{1}{2}\Delta t\right) + \ddot{r}_i(t) \cdot \Delta t$$

$$r_i(t + \Delta t) = r_i(t) + \dot{r}_i\left(t + \frac{1}{2}\Delta t\right) \cdot \Delta t \tag{2.3}$$

A slightly different algorithm called Velocity-Verlet was recently implemented in GROMACS. It is closely related to the leapfrog integration but with the difference that it calculates both positions and velocities at the same time frame. This algorithm is used only when extremely accurate integration is required (forces and velocities are known for the whole step, which is required for some applications), since it comes with a slightly higher computational cost. These integrators imply a deterministic description; meaning two different simulations started from the same initial configuration (positions and velocities) will generate the same trajectory (*i.e.* sample exactly the same conformational space).

Sampling the relevant phase space that is accessible to a system is one of the main challenges in MD simulations. Integrating these equations is computationally very demanding, especially for many interacting particles. This cost can be diminished by omitting degrees of freedom of the system irrelevant to the hypotheses tested — excluding the solvent for instance, which in most cases composes the majority of the system but is of a limited interest. In that case, the effects of these missing degrees of freedom can be mimicked by reintroducing energy through stochastic or fluctuating forces applied to the rest of the particles composing the system. With this approach, the time reversibility of the previous integrators and the analytical identity (determinism) is lost.

The leapfrog-based *sd* — for stochastic dynamics — integrator implemented in GROMACS adds friction and noise terms to

Newton's equations of motion, which become:

$$\dot{r}_i\left(t+\frac{1}{2}\Delta t\right)=\dot{r}_i\left(t-\frac{1}{2}\Delta t\right)\cdot\alpha_i+\ddot{r}_i(t)\cdot\Delta t+\sqrt{\frac{k_BT}{m_i}(1-\alpha_i^2)}\cdot n_i^G$$
$$r_i(t+\Delta t)=r_i(t)+\dot{r}_i\left(t+\frac{1}{2}\Delta t\right)\cdot\Delta t \tag{2.4}$$

with

$$\alpha_i=\left(1-\frac{\gamma_i}{m_i}\cdot\Delta t\right) \tag{2.5}$$

In these equations, m_i the mass of the particle *i*, γ_i is the friction constant, k_B the Boltzmann constant, *T* the temperature, and n_i^G the added noise extracted from a standard normal distribution.

Statistical ensemble and boundary conditions: Besides its approximate way(s) of describing particles and dynamics, MD has further intrinsic limitations. For instance, the simulation box must be large enough to avoid any boundary artifacts induced at its edges that would arise from having either vacuum or walls surrounding the simulated system. A solution to this problem lies in the replication of the box in all directions, the so-called periodic boundary conditions. System sizes should still be chosen large enough however, to avoid artifacts such as self-interaction with periodic images for instance, resulting in a non-physical ordering.

Like for an experimental system, well-defined state conditions have to be selected. This is the case for temperature *T* and pressure *p* for instance, together with the number of particles *N*. To maintain these quantities at a desired value, thermostats and barostats have been developed (see below). Simulations can then be run in various statistical ensembles; we will mention here only the canonical (*NVT*, conserved number of particles, volume and temperature) and isothermal-isobaric (*NpT*, conserved number of particles, pressure and temperature) statistical ensembles. More general ensembles can be described, but are not used in any of the work presented here and are beyond the scope of this chapter.

Different ways exist to algorithmically constrain a system to a fixed temperature. A relatively efficient method is to rescale the velocities obtained after solving the equations of motion, as is done with the popular Berendsen thermostat [43]. In that case, the temperature deviation exponentially decays to the desired temperature:

$$\frac{dT}{dt}=\frac{T_0-T}{\tau_T} \tag{2.6}$$

with T_0 the reference temperature one wants the system to evolve in, *T* the instantaneous temperature and τ_T the coupling constant. This approach, although intuitive and simple, does not reproduce a correct *NVT* ensemble since it "cuts out" the fluctuations of the kinetic energy. However, the extent of the deviation is inversely proportional to the number of particles simulated, and becomes negligible for the system sizes studied in this thesis. Other approaches generating a correct kinetic energy distribution (temperature) exist, such as the Velocity-Rescale algorithm [44] which corrects the kinetic energy distribution by adding a stochastic term to the Berendsen equation above.

The concept of exponential decay can be similarly used to correct the pressure in the case of isothermal-isobaric ensemble. The Berendsen barostat [43] leads to an exponential decay of the pressure deviation:

$$\frac{dp}{dt} = \frac{p_0 - p}{\tau_p} \tag{2.7}$$

where p denotes the instantaneous pressure of the system, p_0 the reference pressure, and τ_p the coupling constant. The barostat algorithm rescales the particle positions and box dimensions depending on the system's compressibility. The rescaling matrix can be diagonal (with equal components on the diagonal) in the case of an isotropic coupling, but can be more complex in the case of semi-isotropic or anisotropic coupling and with different compressibility values for each dimension. Most systems presented in this thesis contain interfaces (*i.e.* bilayers) and require a semi-isotropic pressure coupling.

Interaction potentials: The strength of interaction between particles is central to the MD method, and is defined through interaction potentials from which the forces are computed according to Equation 2.2. Two types of interactions are described: bonded and non-bonded.

The bonded interactions are reproducing the chemical links existing between particles in a molecule; bond (2-particle terms) and angle (3- and 4-particle terms) potentials keep the overall topology of a compound. Many different potential forms can be used to describe these interactions, the simplest form being the harmonic potential:

$$V_{\text{bond, angle}}(X) = \frac{1}{2} k_X \cdot (X - X_0)^2 \tag{2.8}$$

where X is the oscillating quantity, X_0 its equilibrium value and k_X the related force constant. This description is of course the simplest implemented; more complex forms of bonded potentials can be defined. Sometimes bonded interactions are constrained to a fixed equilibrium value; typically to remove vibrations with high frequencies that get difficult to integrate using large time steps (> 1 fs).

The non-bonded interactions describe terms extending from the bonded terms, and are thus thought to reflect the direct environment. Different types of interactions can be encountered: the London's dispersion forces describing the interactions between neutral particles, to which electrostatic interactions can be added if the particles are charged. Depending on the force field different ways of handling hydrogen bonds have been implemented (*e.g.* through explicit potentials), but are more commonly described as electrostatic interactions through the partial charges carried by the atoms. Even though more complex and detailed potentials have been derived, the Lennard-Jones (LJ) potential, computationally robust and easy to implement, is commonly used to describe the first type (interactions between particles). The Coulomb potential is used to describe the second (treatment of electrostatic if the particles are charged). These potentials have the following respective analytical forms:

$$V_{\text{Lennard-Jones}}(D_{ij}) = \frac{A_{ij}}{d_{ij}^{12}} - \frac{B_{ij}}{d_{ij}^{6}} \tag{2.9}$$

$$V_{\text{Coulomb}}(d_{ij}) = \frac{1}{4\pi\varepsilon_0} \cdot \frac{q_i q_j}{\varepsilon_r d_{ij}} \tag{2.10}$$

where d_{ij} is the distance between particles i and j, ε_0 the dielectric permittivity of vacuum. Five parameters need to be defined in these equations: the dielectric constant of the media ε_r, fixed in most cases to the experimental value of the solvent, the (partial) charges q of each particle i and j, and the interaction parameters A_{ij} and B_{ij}. These latter parameters are specific to each particle pair.

The equilibrium values and associated force constants describing the bonded interactions, as well as the parameters describing the non-bonded interactions, are gathered in large libraries called force fields. The definition of these force fields is dependent on the philosophy against which they were parameterized. General parameterization procedures are detailed in the next section.

Force field parameterization: As mentioned before, different levels of resolution can be used to describe interactions between chemical compounds: from quantum methods describing interactions on their smallest length- and time-scales, to continuum approaches focusing on macroscopic quantities. Following this scale, the most common approaches used in MD simulations are situated in the middle: the interactions are described at atomistic, or near-atomistic resolution. By ignoring quantum-chemical degrees of freedom (*e.g.*, electrons), the inter particle interactions are effective. Three ways of parameterizing these interactions are commonly used: a bottom-up approach, extracting parameters from extensive series of quantum calculations; a top-down approach, which uses quantities measured experimentally; and finally a mix between the two previous approaches. In the latter approach, one typically uses as a first step a bottom-up approaches to define an initial set of parameters for a compound, and in a second iterative step a top-down approach to tune the parameters to converge towards reproducing specific experimental data. Both bottom-up and top-down approaches have a number of shortcomings:

- Quantum calculations cannot be used to define interactions between compounds in all possible chemical environments, and are limited in size (particle-wise) due to the high computational cost.
- Experiments are in most of the cases limited to the observation of macroscopic quantities, containing many inextricable components and averaged over many molecules and long time-scales (unreachable by simulations), which do not necessarily convey the fine interactions between compounds.

The final set of parameters is therefore not unique, and not transferable between different force fields.

Not surprisingly, a large number of force fields are currently available in the literature, each of them designed and developed against a certain philosophy, giving more or less importance to theoretical calculations and/or focusing their top-down approaches on different experimental quantities and/or oriented towards certain type of compounds (lipids, proteins, DNA, *etc.*). Here again different levels of resolution can be defined: atomistic force fields use atoms as interaction centers, whereas coarse-grained (CG) force fields group atoms together in beads, which are then used as interaction centers. Commonly used atomistic force fields for biomolecules are AMBER, CHARMM, Gromos and OPLS (for a detailed definition and comparison between the philosophies behind these different force fields, see [45]). The most popular biomolecular CG force field is the Martini model developed in the laboratory of Prof. Marrink [46]. The work presented in this thesis makes extensive use of this model. It will be presented in more detail in the next section.

The Martini force field

The idea of coarse-graining: The power of MD resides in its capacity at accessing a level of detail extremely challenging to reach experimentally, namely interactions between atoms themselves. But this knowledge comes at a rather high computational cost. Analyzing interactions between protein and lipids, for instance, requires long simulations of multi-component systems, as many binding/unbinding events have to be observed before obtaining sufficient statistics. However, for such studies, the finest details of the interactions might not be needed. For instance, the fastest motion present in the system (vibrations of bonds involving hydrogen atoms for instance) is limiting the integration time step of the equations of motion; in practice and in common cases, this time step is on the order of the femtosecond. The number of steps to reach the hundreds of nanoseconds or the microseconds needed by a lipid to bind the protein surface and exchange with other lipids of the bilayer is thus tremendous, and the actual time needed to perform such simulations is not feasible in most cases.

One solution to extend the sampling simulation time scale is to describe interactions and particles in a coarser way, making use of CG force fields [47]. Instead of computing interactions between each atom composing a molecule, groups of atoms are gathered together and treated as a unique interaction center, possessing the properties of the chemical group it replaces. By performing such a conversion, the number of particles composing the system is drastically reduced, allowing the simulation of larger systems, and the highest frequency motions present in the system are now considerably reduced by the increased weight of the particles, allowing the use of a larger integration time step (several tens of femtoseconds).

Looks of the Martini model: “Martini” is one of these CG force fields, developed by the group of S.J. Marrink at the University of Groningen. Initially developed to simulate lipids and cholesterol [48,49], it has since been extended to many types of molecules such as proteins [50], carbohydrates [51], DNA and various polymers, and includes now an extensive library of solvents and small molecules. Martini inherited its appellation from the nickname of the city of Groningen, which possesses a tower with the same name; but it has been also suggested to be linked to “*the universality of the cocktail with the same name: how a few simple ingredients can be endlessly varied to create a complex palette of taste*” [46], which seems to fit given the extent of its development, and its intensive and successful use.

The Martini CG force field follows the principles previously enunciated: it gathers groups of two to six atoms in beads (*cf.* Fig. 2.1A). The hallmark of the Martini philosophy is that beads are parameterized to reproduce a thermodynamic quantity determined experimentally, namely the partition free energy. This quantity directly relates to the partition coefficient of a chemical compound, measuring the difference of solubility between aqueous and apolar phases. This parameterization philosophy has the main advantage to be derived from an experimental observable, and in consequence makes the model to perform closer to experimental behavior.

Eighteen different particle types were initially defined (slightly more in recent developments [50]), separated in four categories: apolar (C), neutral (N), polar (P) and charged (Q); each type has subdivisions to smoothly grade from extremely hydrophobic (aliphatic tails of lipids for instance) to completely hydrophilic (ions for example). The Martini approximation goes further by defining only ten (twenty counting the S particle interaction levels) levels of non-bonded interactions, modeled as LJ potentials, between these beads (Fig. 2.1B). In addition, charged beads interact through a Coulomb potential with a relative dielectric screening constant $\varepsilon_r = 15$. Some standard MD parameters were tested and tuned to obtain a maximal efficiency: for example, the non-bonded potentials are artificially shifted to reach at a cutoff $r_{cut} = 1.2$ nm, after which no interactions are calculated.

Advantages and limitations of Martini: The approximations inherent of the Martini model lead to its main advantage: the speed-up of simulations. A rough estimation leads to a gain of three to four orders of magnitude. This gain results from three main factors:

- Fewer calculations have to be performed per step since the number of particles is drastically reduced.

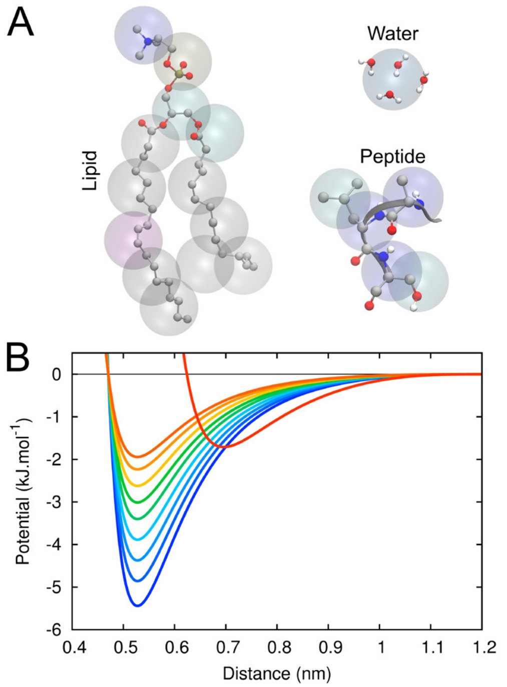

Figure 2.1 | Martini in a nutshell. **A)** Mapping of various compounds, from their atomistic representations (united atoms, ball and sticks) to their respective Martini representations (superimposed transparent balls). **B)** The ten different LJ potentials — representing the ten possible levels of interactions — used in the Martini CG force field. Note that interactions between "S" particles are not reported here for clarity.

- The decrease in the number of degrees of freedom and removal of fast vibrational motion smoothens the potential energy surface, allowing the use of a larger time step without inducing integration errors, and making the model extremely flexible to external constraints.
- The smoothened energy landscape affects the dynamics of the processes simulated, increasing in most cases their kinetics (lower energetic barriers to overcome) by removing friction.

Furthermore, by limiting the size of the CG groups to a handful of atoms, the general chemistry of the molecules is conserved, allowing Martini's extensive diversification in term of types of compounds.

But the simplicity of the Martini approach also carries several intrinsic problems:

- The smoothened free energy surface consequently affects the diffusion of compounds. A comparison of the diffusion rate of various compounds showed no systematic shift due the CG description, but a wide spread that depends on the chemical details of the system simulated [52]. Therefore, the time scale in Martini simulations has to be interpreted with care.
- The loss of structural details excludes the study of fine interactions such has specific hydrogen bonding. For instance, the conservation of the secondary structure of proteins has been shown to require elastic networks, the so-called *ElNeDyn* approach [53], linking the backbone beads by long bonds with weak force constants. Consequently, the Martini protein description does not allow for changes in secondary structure.
- Grouping atoms into beads reduces the overall entropy of the system, and the missing energy component has to be reintroduced in the enthalpy term ($G = H - TS$, where H and S are the enthalpic and entropic components of the free energy G, and T the temperature; this specific problem will be addressed in more details in Chapter VI).
- The purposely-restricted number of interaction levels induces its limitations too, leading to over/underestimated interaction strengths between certain compounds and a possible error accumulation proportional to the size of the interacting compounds.
- Finally, the standard Martini water model is not capable of explicitly screening electrostatic interactions, being essentially a neutral LJ fluid. Instead, the relative dielectric constant ε_r is used as implicit screening factor. Note that a polarizable Martini water model is also available [54], however, at the expense of larger computational cost. In this thesis, only the standard model has

been used.

Despite these limitations, the Martini model has been successfully applied to study a wide range of (bio)molecular processes. For a recent review, see Marrink & Tieleman [52].

Standard methods common to the simulations presented in this thesis

Parameters for CG simulations: Unless stated otherwise, all simulations presented in this thesis were performed using the GROMACS simulation package version 4.0.x [41] and 4.5.x [42]. The systems were described with the Martini CG force field for biomolecules (version 2.0) [46], its extension to proteins (version 2.1) [50] together with the *ElNeDyn* approach [53], which defines an elastic network between the backbone beads to control the conformation (secondary structure) of a protein. Elastic networks were built on each subunit of the protein complexes separately, *i.e.* springs are not present between the subunits. The integrity of the protein is only dependent on non-bonded interactions. Extend of the *ElNeDyn* network was 0.9 nm and the force constant of the springs was set to 500 kJ.mol^{-1} nm^{-2}. If not otherwise mentioned, conventional simulation setups associated with the use of the Martini force field were used. That includes an integration time step of 20 fs (systems containing proteins) to 40 fs (systems containing only lipids) for production runs and non-bonded interactions cutoff at a distance r_{cut} = 1.2 nm. The LJ potential is shifted to zero from r_{shift} = 0.9 nm to rcut. The electrostatic potential is shifted from r_{shift} = 0.0 nm to r_{cut}. For each system the proteins, lipid bilayer and solvent (water and salt) were coupled independently to external temperature baths using a Berendsen thermostat [43] with a relaxation time of τ_T = 0.5 ps. In the simulations performed in a *NpT* ensemble, the pressure was weakly coupled (Berendsen barostat [43]) using a relaxation time of τ_p = 1.2 ps and a semi-isotropic pressure scheme. For the studies presented in Chapter III to V, the *md* integrator was used. The approach presented in Chapter VI required the use of the *sd* integrator for reasons explained later.

Resolution transformation: To convert a CG configuration to a fine-grained (FG) configuration, we used the resolution transformation method implemented in an in-house modified version of GROMACS version 3.3.1 [55]. The systems were cooled down from an initial temperature of 1000 K to the desired target temperature in 30 ps of simulated annealing, during which the atomistic particles were coupled to their corresponding CG beads through harmonic restraints. Subsequently, the coupling was gradually removed within a time span of ~30 ps. These annealing simulations were carried out in the *NVT* ensemble. Constraints were replaced by regular bonds, and an integration time step of 1 fs was used. To control the temperature, stochastic coupling with an inverse friction constant τ_T = 0.1 ps was applied. The other parameters for the resolution transformation were set to the standard values (see Rzepiela *et al.* [55] for details).

Parameters for atomistic simulations: Unless stated otherwise, proteins in the atomistic simulations were described with the 54A7 parameter set of the Gromos force field [56]. For the lipids we used an in-house version of a new 53A6-based lipid force field [57]. The SPC water model [58] was used to model the aqueous solvent. Temperature and pressure coupling were applied in a similar manner as in the CG simulations, with time constants τ_T = 0.1 ps and τ_p = 1 ps, respectively. Non-bonded interactions within 0.9 nm were updated at every time step, and interactions between 0.9 and 1.4 nm every 10 steps. The long-range electrostatic interactions were computed with the PME algorithm [59]. In the simulations continued from reverse transformations, an integration time step of 1 fs was used for production runs. §