UNIVERSITÀ DEGLI STUDI DI TRENTO

Department of Physics

Thesis submitted for the degree of Doctor of Philosophy

COMPUTATIONAL MULTISCALE INVESTIGATIONS OF BIOLOGICAL MOLECULES

Candidate:

Giovanni Mattiotti

Supervisor:

Prof. Raffaello Potestio

Date of Discussion: 20/11/2023

Contents

Chapter 1

An Introduction to Numerical Simulations of Molecular Systems

The study and understanding of biomolecules is an interdisciplinary field that encompasses molecular biology, biochemistry, structural biology, biophysics and many other disciplines. Molecular dynamics simulations have emerged as a powerful tool in this field, providing a mechanistic view of the behavior of biological molecules. In this chapter, we will explore the principles and methods of molecular dynamics simulations and their applications in comprehending the fundamental processes that underlie life and the mechanisms underlying disease. We will also briefly introduce some of the other sectors of science that are connected to molecular dynamics simulations and explain why this tool is a valuable asset in answering questions that arise from these fields.

1.1 A brief introduction to Molecular Biology, Biochemistry and Structural Biology

Molecular biology, biochemistry, and structural biology are three closely related fields that have revolutionized our understanding of life processes: they focus on the study of biological molecules, their structures, functions, interactions and they play a crucial role in advancing our knowledge of how living organisms function at the molecular level. In the following, I will shortly introduce their contents: for an exhaustive treatment, the reader is referred to complete books, such as [2, 3, 4].

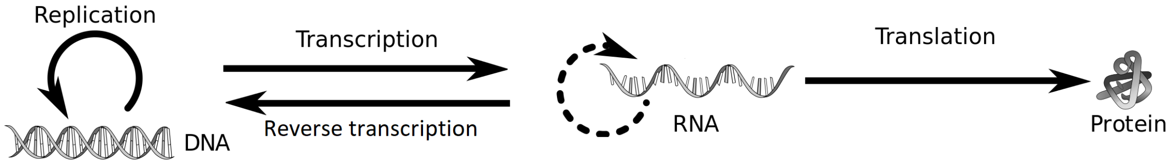


Figure 1.1: A representation of the Central Dogma in Molecular Biology [1]: the genetic information can be transferred from nucleic acids to nucleic acids, but once translated into a polymer chain of amino acids (a protein) it cannot be transferred back nor to other proteins. Credits: Wikipedia.

Molecular biology [2] is the study of the molecular basis of life. It involves the study of nucleic acids (DNA, RNA) and proteins – the building blocks of life (exemplar is the enunciation of the Central Dogma in Molecular Biology, shown in figure 1.1). The field of molecular biology has made significant contributions to our understanding of genetics, gene expression, and the regulation of cellular processes. Molecular biology techniques are used to manipulate DNA and RNA, allowing scientists to study gene expression and genetic mutations that cause disease.

Moving into the realm of chemistry, biochemistry [3] is the study of the chemical processes that occur within living organisms. It involves the study of enzymes, proteins, carbohydrates, lipids, and nucleic acids: the molecules that make up living cells (see figure 1.3). Biochemistry provides insights into the metabolic pathways that drive cellular processes, such as energy production and nutrient utilization. Biochemistry techniques are used to isolate and study individual molecules, providing a detailed understanding of their structure and function.

Structural biology [4] is the study of the three-dimensional structures of biological molecules. It involves the use of techniques such as X-ray crystallography (see *e.g.* figure 1.2), NMR spectroscopy, and electron microscopy to determine the structures of proteins, nucleic acids, and other biological species. Structural biology provides insights into how these compounds in-

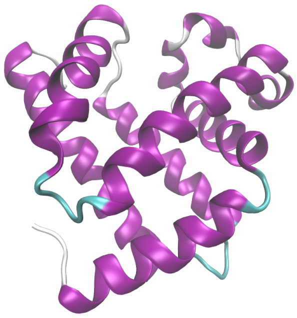


Figure 1.2: Cartoon representation of myoglobin, the structure of which is one of the first ever solved with the technique of X-ray crystallography [5] (PDB ID of the structure shown here: 1MBN).

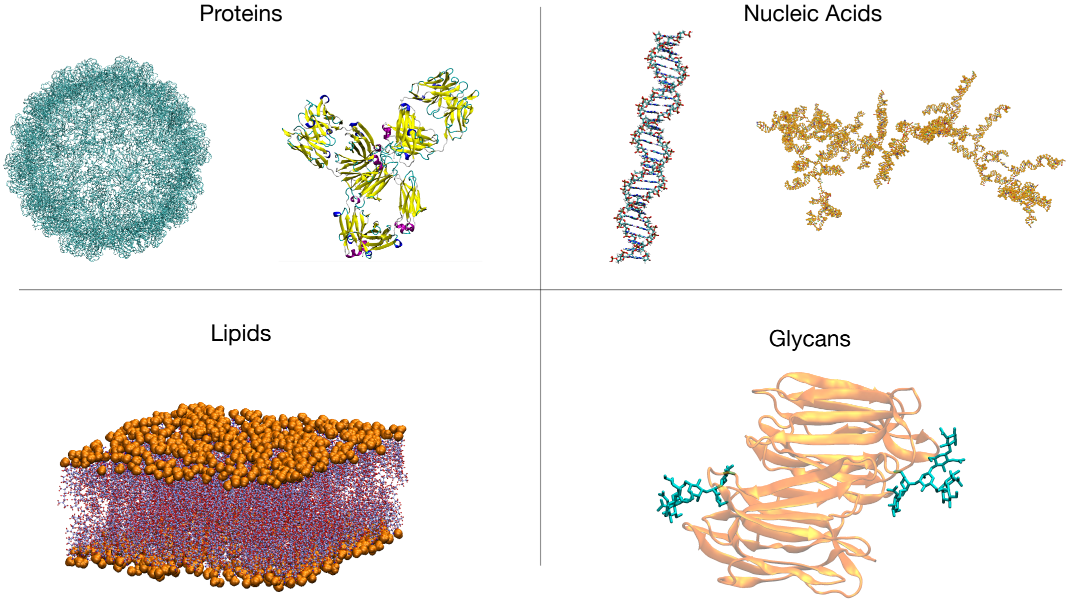


Figure 1.3: The four classes of biochemical compounds that are typically treated in biophysical simulations.

teract with each other and with other molecules in the cell: this information is crucial for understanding how biological systems function at the molecular level.

The fields of molecular biology, biochemistry, and structural biology are closely intertwined. They share many common techniques and concepts, and their findings often complement each other. For example, biochemistry techniques can be used to study the function of a protein, while structural biology techniques can be used to determine its three-dimensional structure. Molecular biology techniques can then be used to shed more lights on how the protein interacts with other molecules in the cell. Despite the plethora of experimental techniques that has been developed in the last century and the successes achieved with their use, which led to the birth of the field of *biotechnology*, some limits of these disciplines still remain today. One of the most peculiar one is related to the time and length scales associated with many relevant biological processes: in fact, from chemical reactions to large-scales motions of molecular complexes and proteins, these processes involve the motion of electrons and atoms that happen in the order of *femtoseconds* $(10^{-15}s)$ or even smaller; one can easily imagine that it is very hard to be able to experimentally observe every step of such processes with enough resolution and without disturbing the process itself. It is here that *molecular dynamics simulations* come into help: mixing experimental data, *ab initio* calculations and the laws of dynamics (both from classical and quantum mechanics) scientists are able to simulate what happens to every degree of freedom of a biological system, every femtosecond or more. Among others, we report two examples where molecular dynamics revealed to be essential to help scientific progress and human technology:

1. Prediction of tertiary structures of proteins [6], also whose native structure has not yet been resolved, in arbitrary environmental conditions (such as pH [7], ionic concentration and more): when we will be able to simulate processes on time scales of seconds/minutes, given the genetic code of a species it will be possible to predict the structures of all proteins for which the genome encodes information (modulo the intrinsic weaknesses of the model used, of course). Although many argue that the problem has already been solved with the advent of *AlphaFold* [8], many others (including myself) believe that such an extreme static and data driven approach as AlphaFold risks losing essential information about the rules and first principles underlying folding, which involves the reconstruction of the dynamical pathways and, in turn, can be clarified more deeply via the use of molecular dynamics simulations.

2. Binding free energy calculations: this has potential applications in the pharmaceutical field (computational drug design, see *e.g.* [9]) and also in more fundamental science sectors, to characterize for example chemical reactions involved in charge transport, as happens in complexes responsible for photosynthesis [10].

1.2 A Short Overview of Molecular Dynamics

After anticipating the potential of molecular dynamics applied to biology, in this section we briefly review the main theoretical reasons that leads to treat a molecular system with the engine of *classical* molecular dynamics (the theoretical framework of my whole thesis), and not a more fundamental theory. Since it is a well-established topic, I will not seek for completeness but try to be exhaustively short. A rich and elaborated treatment of the topic can be found in many textbooks: for example, I took inspiration from [11] and from [12] for the derivation of Newton's equations for the nuclei from an *ab initio* framework.

1.2.1 From quantum mechanics to classical force fields

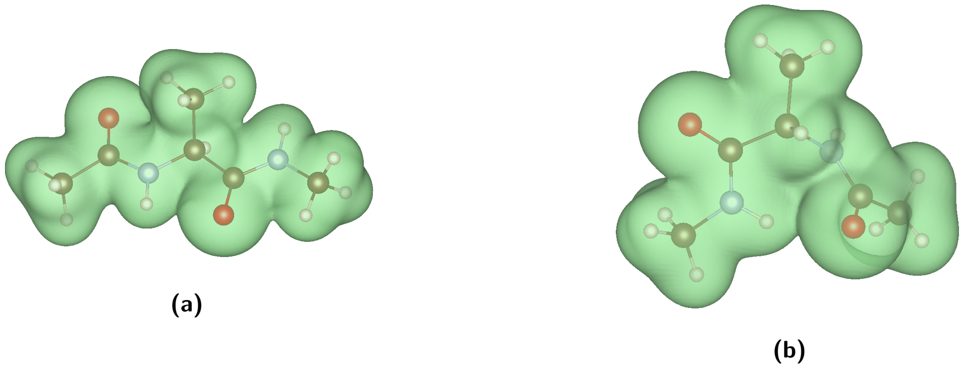


Figure 1.4: Isosurfaces of the electronic charge densities, obtained at the DFT level, for two different configurations of the alanine dipeptide, **(a)** more open and **(b)** with an hydrogen bond connecting the oxygen (in red) on the right-hand side to the hydrogen (in white) bonded to the left-hand sided nitrogen (in light blue).

A reasonable starting point that a theorist would take to model a molecular system would be non-relativistic quantum mechanics. Although it is known from decades that coupling special relativity to quantum mechanics leads to the most precise and predictive theory that humans have built up to now, *i.e.* the Standard Model of Elementary Particles [13], it would be unnecessarily laborious and probably practically unfeasible to start from such a fundamental and deep description to investigate biomolecular phenomena. We will then assume that relativistic effects can be neglected and that the only fundamental interaction that is necessary to treat molecular systems at our scales of interest is the electromagnetic interaction, and to be more precise the electrostatic part of it. We can neglect weak and strong nuclear interactions because we do not look into spatial or energetic scales where their effects become relevant. Moreover, we are interested in modelling atoms as classical point-like particles (describing their nuclei): the electrons will be in fact reduced to hidden degrees of freedom whose contribution will be indirectly taken into account. As a consequence, we will not take into accounts excited states and we will not allow for chemical reactions to occur.

After this preface, we can write down the generic quantum Hamiltonian (omitting the operator symbol $\hat{}$ for convenience) of a molecular system:

$$H(\mathbf{r},\mathbf{R},\mathbf{p},\mathbf{P}) = \sum_{I=1}^{N_a} \frac{\mathbf{P}_I^2}{2M_I} + \sum_{i=1}^{N_e} \frac{\mathbf{p}_i^2}{2m_i} + \frac{e^2}{4\pi\epsilon_0}\left(\sum_{I\neq J}^{N_a} \frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|} + \sum_{i\neq j}^{N_e} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_{i,I}^{N_a+N_e} \frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|}\right) \tag{1.1}$$

where $\mathbf{R} = \{\mathbf{R}_I\}$ and $I, J = 1, \ldots, N_a$ are indices for each atom's nucleus, $\mathbf{r} = \{\mathbf{r}_i\}$ and $i, j = 1, \ldots, N_e$ for the electrons and the potential energies V are sums of two-body terms.

Born-Oppenheimer Approximation

The time-evolution of a system would be in principle described by the Schrödinger equation involving the wave function $\Phi(\mathbf{r},\mathbf{R};t)$ that fully describe the system:

$$i\hbar\frac{\partial}{\partial t}\Phi(\mathbf{r},\mathbf{R};t) = H[\Phi(\mathbf{r},\mathbf{R};t)] \tag{1.2}$$

but the applicability of this picture is limited if the number of degrees of freedom becomes larger and larger. In this scenario, the Born-Oppenheimer approximation comes into help by recognizing that the motion of the electrons and the nuclei can be decoupled because of the substantial differences among the masses of these two objects. Considering the worst-case situation with one single nucleon:

$$\frac{m_e}{M_n} \simeq \frac{1}{1836} \lesssim 0.00055 \tag{1.3}$$

With this in mind, one can reasonably assume that the motion of the electrons happen essentially with fixed positions of the nuclei. In other words, the wave function can be expressed in a separated way as follow:

$$\Phi(\mathbf{r},\mathbf{R};t) \simeq \Phi_{bo}(\mathbf{r},\mathbf{R};t) := \psi(\mathbf{r}|\mathbf{R};t)\chi(\mathbf{R};t) \tag{1.4}$$

where $\psi(\mathbf{r}|\mathbf{R};t)$ is the electronic wave function and its modulus square is a conditional probability, *i.e.* the probability of the electrons to be measured in position $\mathbf{r}$ at time t *provided that* the nuclei wave function has the form $\chi(\mathbf{R};t)$.

We can push the approximation even further. In fact, if we assume that there is no external source of photons, we can consider the electrons to be in their ground state. In fact, even with a thermal bath, at the typical temperatures that are considered in biochemical systems (those of interest in this work) the thermal energy corresponds to few kJ/mol:

$$T \sim 300K \quad \Rightarrow \quad k_B T \sim 2.5 kJ/mol \tag{1.5}$$

while, as a reference, the energy needed for the electron in the ground state of an hydrogen atom to be excited to the first excited state is $\Delta E_{0,1} \sim 10eV \sim 965kJ/mol$. We can use this argument to strongly simplify the description of molecular motion by removing the time dependency from the electronic component of the wave function and to consider the electronic problem as a ground state problem. Equation (1.4) reduces to:

$$\Phi_{bo}(\mathbf{r},\mathbf{R};t) \simeq \psi_0(\mathbf{r}|\mathbf{R})\chi(\mathbf{R};t) \tag{1.6}$$

and the Schrödinger equation (1.2) can be decoupled into two equations:

$$\begin{cases} \left[T_e(\mathbf{r}) + V_{ee}(\mathbf{r}) + V_{ae}(\mathbf{r};\mathbf{R})\right]\psi_0(\mathbf{r}|\mathbf{R}) = E_{el}(\mathbf{R})\psi_0(\mathbf{r}|\mathbf{R}) \\ i\hbar\dfrac{\partial}{\partial t}\chi(\mathbf{R},t) = \left[-\dfrac{\hbar^2}{2M_I}\nabla^2_{\mathbf{R}} + V_{aa}(\mathbf{R}) + E_{el}(\mathbf{R})\right]\chi(\mathbf{R},t) \end{cases} \tag{1.7}$$

Here, the variable $\mathbf{R}$ in $V_{ae}(\mathbf{r};\mathbf{R})$ is essentially a fixed set of parameters, because the nuclei's positions are fixed, in the sense that every time that the nuclei move, the electrons "instantly" reaches a new ground state. This is also called *adiabatic approximation* and it holds as long as one is interested in timescales of the order of femtoseconds and more, which are the typical scales of nuclei's motion. $E_{el}(\mathbf{R})$ is nothing but an effective potential (or also called potential energy surface) because it is obtained by integrating over the electronic wave function.

Molecular Force Fields and classical Molecular Dynamics

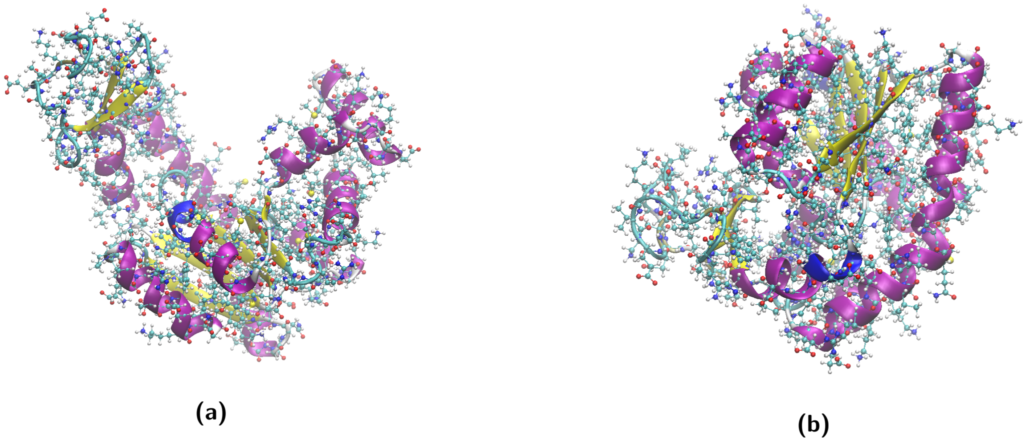


Figure 1.5: Two very different conformations ((**a**) named open, (**b**) named closed) of the Adenylate Kinase (PDB code: 1AKE), shown in two different representations: New Ribbon and CPK. In this way, one can see that even if the secondary structures are preserved, the configurations that are obtained with classical molecular dynamics carry a huge amount of information, provided by the cartesian coordinates of every atom in the molecule at every simulated time step.

The second equation in (1.7) in principle would give the dynamics of the nuclei's wave function. However, quantum delocalization effects are usually neglected for the nuclei (see [12] for a formal derivation based on the classical formal limit $\hbar \to 0$), and one is satisfied with a classical description of the nuclei's motion, via Newton's equations:

$$M_I \frac{d^2}{dt^2}\mathbf{R}_I = -\nabla_I \left[V_{aa}(\mathbf{R}) + E_{el}(\mathbf{R}) \right] \tag{1.8}$$

This protocol goes under the name of *Born-Oppenheimer Molecular Dynamics* and its use for large systems is often unpractical, because it requires the knowledge of $E_{el}(\mathbf{R})$ everywhere for it to be differentiated or, alternatively, to numerically calculate (by discretizing the differential equations) it for every step of the nuclei. Another significant example of a simulation protocol that is similar to this picture, but that does not rely on the adiabatic approximation, is the *Car-Parrinello Molecular Dynamics* protocol [14]: it couples Newton's dynamics of the nuclei to the famous self-consistent approach, named the *density functional theory* (DFT) [15], for the electronic problem by considering a time-dependent surrogate of the electronic wave function of the valence electrons alone, with its own Newton-like equations for the dynamics coupled to those of the classical nuclei (see examples of representations of electron densities obtained at the DFT level in figure 1.4b). All these methods, which are identified as *ab initio molecular dynamics* methods, requires a partial quantum description of the system and thus become computationally expensive, limiting their applications to small systems (around 10^3 at most).

For bigger systems (up to 10^9 atoms), like explicitly solvated biochemical molecules and complexes (proteins, nucleic acids, lipids, carbohydrates), the scientific community decided instead to put a lot of effort into a simplified characterization of $V_{aa}(\mathbf{R}) + E_{el}(\mathbf{R})$, firstly by making an *ansatz* for a suitable functional basis with free parameters and then finding an appropriate, mainly experimentally based, parametrization. From more than thirty years ago until now it is become a standard [16] to approximate it with two terms, called *bonded* and *non-bonded* potentials:

$$V_{aa}(\mathbf{R}) + E_{el}(\mathbf{R}) \quad \rightarrow \quad V(\mathbf{R}) = V_b(\mathbf{R}) + V_{nb}(\mathbf{R}) \tag{1.9}$$

$V_b(\mathbf{R})$ is a sum of two- three- and four-body terms that depend on distances, angles and torsional angles between nuclei, while $V_{nb}(\mathbf{R})$ is a two-body term. The set of forces that are obtained by minus the gradient of $V(\mathbf{R})$ is commonly called a *force field*. More explicitly, (one of) the typical functional form(s) of a molecular force field for a system of N nuclei (or atoms, equivalently) is:

$$\begin{aligned} V_b(\mathbf{R}) = & \sum_{i,j} \frac{B_{ij}}{2}(r_{ij} - r_{0,ij})^2 + \sum_{i,j,k} \frac{A_{ijk}}{2}(\theta_{ijk} - \theta_{0,ijk})^2 + \\ & + \sum_{i,j,k,l} \sum_{n} \frac{D_{ijkl}}{2}\left[1 + \cos(n \cdot \chi_{ijkl} + \gamma_n)\right] + \sum_{i,j,k,l} \frac{I_{ijkl}}{2}(\omega_{ijkl} - \omega_{0,ijkl})^2 \end{aligned} \tag{1.10}$$

$$V_{nb}(\mathbf{R}) = \sum_{i,j} \left[\frac{1}{4\pi\epsilon_0} \frac{q_i q_j}{r_{ij}} + 4\epsilon_{ij} \left(\frac{\sigma}{r_{ij}}\right)^{12} - 4\epsilon_{ij} \left(\frac{\sigma}{r_{ij}}\right)^{6} \right] \tag{1.11}$$

These seven terms are chosen mainly by following two criteria: simplicity and heurism. We can spend few words for each of them:

1. the first one, called the *bond* term, approximates the energy of the covalent bond between two atoms with a quadratic potential; this approximation is reasonable for distances close the minimum one, $r_{0,ij}$ (it is, after all, the first non-zero, non-constant term of a Taylor expansion around a minimum reference distance $r_{0,ij}$).

2. the second one, called the *angle* term, its another quadratic term that aims at preserving specific geometries between specific, mutually bonded triads of atomic species; the same argument on the quadratic form of the bond term also holds here.

3. the third one is called the *torsional* term and it involves indeed the torsional angle formed by four atoms; it has the same geometric purpose of the angle term, *i.e.* to maintain some geometrical properties among the atomic species involved in the torsional angle taken into account, but the functional form is now a truncated Fourier series (the sum over n is a finite sum, with usually 4/5 terms at most, per each torsional angle).

4. the fourth one is called the *improper* term and it involves other torsional angles; its purpose is to impose a given planarity (meaning: keeping the improper ω_{ijkl} close to the equilibrium value $\omega_{0,ijkl}$ of the quadratic function) to some specific fragment of a molecule, a thing that would not be guaranteed by the sole torsional term.

5. the fifth term is clearly a Coulomb potential, with q_i and q_j *partial* charges of the atoms involved in the interaction; it usually act to couples of atoms that are not involved, together, in a bonded interaction (with some exception for the torsional terms that we will neglect here). In fact, one could argue that every interaction between atoms is electrostatic in nature (as we assumed at the beginning of this section for the quantum treatment), but after all the approximations, in the classical MD picture the concept of bare charge itself loses its meaning: this term is suggested by fundamental intuitions but its interpretation is tricky and not always as simple as it looks like.

6. the sixth term is the short-range, hard core repulsion term of the Lennard-Jones potential and its aim is to avoid overlapping of atoms, as dictated by the Pauli exclusion principle.

7. the seventh and last term introduces the so-called *London dispersion forces* which are caused by dipole-dipole interactions of instantaneous dipoles; it is an attractive terms that nevertheless vanishes pretty fast as the distance between the couple of atoms involved increases.

I want to spend few words on the meaning of the parameters that appear in all these terms. By looking at (1.10) and (1.11) one could legitimately think that each of them depends on the *specific atoms* that are involved, for example atom i and j for k_{ij} and $r_{0,ij}$. In reality, one of the power of classical force fields is that the parametrization is based on the nature of the atom (H,N,O,S,C,...) and its *chemical environment*. This is a huge difference, because it makes the parameters no more system specific and ideally transferable to whatever system one is interested to simulate. Unfortunately, this is an optimal situation because in the end every set of parameter is affected by a bias that depends on the specific experiments used to make the fit.

Another weakness of classical force fields is that transferability in thermodynamical variables is not guaranteed. In some sense, in fact, the huge difference between *ab initio* and classical MD to find forces that govern the motion of the nuclei is that the parametrization of classical force fields are done in certain experimental conditions, which are typically $T \sim 300K$, $P \sim 1atm$ and $I = 150mM$ of salt concentration in solution. In some sense, for the quantum derived forces one can either assume that thermodynamics does not affect the dynamics of the electrons or one can include them explicitly in the electrons' dynamics (in the same way it can be included in the dynamics of the nuclei, as we will discuss in the next paragraph). In other words, one should always keep in mind that $V(\mathbf{R}) = V(\mathbf{R}|T, P, I, \dots)$ should be intended more properly as a *free energy*.

Nevertheless, in the last decades classical MD proved to be a promising and powerful tool to access time and spatial scales that are prohibited to experimentalists, trying to help answering questions that go from more biological and fundamental sectors to more application-oriented ones, like computer-aided drug design (see *e.g.* [17]) and material design (see *e.g.* [18]).

1.2.2 Statistical Mechanics and thermostats

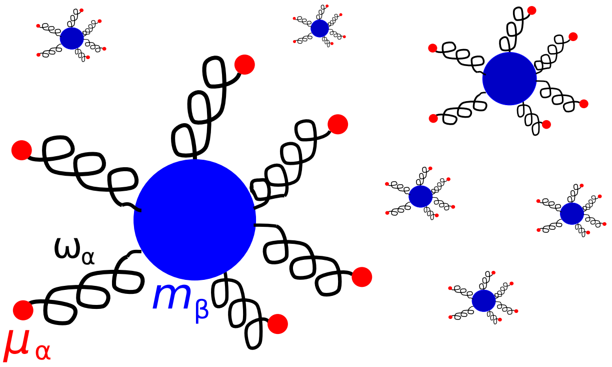


Figure 1.6: Schematic representation of a system (blue particles) in thermal contact with an external bath (red particles) via harmonic couplings, as described in the Caldeira-Leggett semi-empirical model ([19]) to treat *e.g.* a system in contact with a thermostat.

In the last paragraph we concluded that, under certain conditions, we can treat the molecular systems as made by the sole classical *atomic* degrees of freedom (one per each spatial coordinate of each atom in the molecular system under study). We also derived the potential energy term of an Hamiltonian that we have not yet explicitly written. It reads:

$$\mathcal{H}(\mathbf{R},\mathbf{P}) = \sum_{i=1}^{N} \frac{\mathbf{P}_i^2}{2m_i} + V(\mathbf{R}) \tag{1.12}$$

However, the framework provided by Newton's equations alone is not enough for a proper treatment: in fact, especially when we speak about biomolecules, the experimental conditions and the natural environments are incompatible with an Hamiltonian, energy conserving description. Moreover, a typical biomolecule is immersed into a solvent (water molecules and ions) that strongly affects its dynamics. Consequently, we are led to conclude that the proper framework is characterized by the two following properties:

1. The possibility to control experimental conditions, such as constant Temperature, constant Pressure, constant Volume ⇒ we need to be consistent with the framework of Statistical Mechanics
2. The possibility to include the effect of the environment (solvent or similar) *explicitly* ⇒ *e.g.* for solvated biomolecules, we need to treat it preserving bulk properties (as close as possible to the experimental conditions), so we need a good set of boundary conditions (discussed in the next paragraph)
3. In case we cannot include the solvent explicitly, we can make use of an *implicit solvent* model; this argument will be addressed in Chapter 3 of this thesis

We restrict the treatment to the *canonical ensemble* (N,V,T fixed), because for large systems it is considered a reasonable approximation for different experimental conditions, like NPT or μVT ensembles. In order to be consistent with the framework of Statistical Mechanics, we need to find a way to calculate the Boltzmann distribution of our system, and the related partition function:

$$P_{\mathrm{NVT}}(\mathbf{R},\mathbf{P}) = \frac{1}{\mathcal{Q}} e^{-\beta\mathcal{H}(\mathbf{R},\mathbf{P})} \qquad \mathcal{Q} := \int_{V^N} d\mathbf{R} \int_{\mathbb{R}^N} d\mathbf{P}\, e^{-\beta\mathcal{H}(\mathbf{R},\mathbf{P})} \tag{1.13}$$

so that, in turn, we can calculate the *ensemble averages* that are assumed by the theory to be the theoretical counterpart of experimental measures of the system at equilibrium. For a generic observable O:

$$\langle O(\mathbf{R},\mathbf{P})\rangle := \int_{V^N} d\mathbf{R} \int_{\mathbb{R}^N} d\mathbf{P}\, P_{\mathrm{NVT}}(\mathbf{R},\mathbf{P})\, O(\mathbf{R},\mathbf{P}) \tag{1.14}$$

Due to the complexity of the integrals in (1.13),(1.14), even for very simple Hamiltonians, one is obliged to fall back to numerical sampling: this is one of the main driving forces that lead to take advantage of Molecular Dynamics simulations. In fact, assuming that one can perform an MD simulation by keeping N,V and T under control, after an initial transient, the phase space points generated by this simulation would be distributed according to $P_{\mathrm{NVT}}(\mathbf{R},\mathbf{P})$ and in turn one would be able to estimate $\langle O(\mathbf{R},\mathbf{P})\rangle$ as follow:

$$\langle O(\mathbf{R},\mathbf{P})\rangle \simeq \frac{1}{T_{sim}}\sum_{t=1}^{T} O(\{\mathbf{R}(t)\},\{\mathbf{P}(t)\}) = \langle O(\mathbf{R},\mathbf{P})\rangle_T \tag{1.15}$$

where T_{sim} indicates the total number of time steps performed in the simulation.
To conclude, historically scientists developed a lot of ways to couple MD to algorithms that generates configurations sampled according to $P_{\mathrm{NVT}}(\mathbf{R},\mathbf{P})$: these methods go under the name of *thermostats* and in this thesis we mainly used the *Langevin thermostat* (as used in *e. g.* [20]) and the *stochastic velocity rescaling thermostat* [21]. These methods are coupled to the integration of the equations of motion (1.8) and act so to preserve the distributions of the momenta of the particles consistent with the *equipartition theorem*, which relates the average of the kinetic energy of the system to the temperature:

$$\langle K\rangle_T \equiv \left\langle \sum_{i=1}^{N} \frac{\mathbf{P}_i^2}{2m_i} \right\rangle_T = \frac{3}{2}Nk_B\tilde{T} \tag{1.16}$$

Of course, here $\tilde{T}$ is the target value, while the instantaneous value $T(t) = \dfrac{2K(t)}{3Nk_B}$ can fluctuate along the simulation.

1.2.3 Periodic Boundary Conditions and Particle-Mesh Ewald

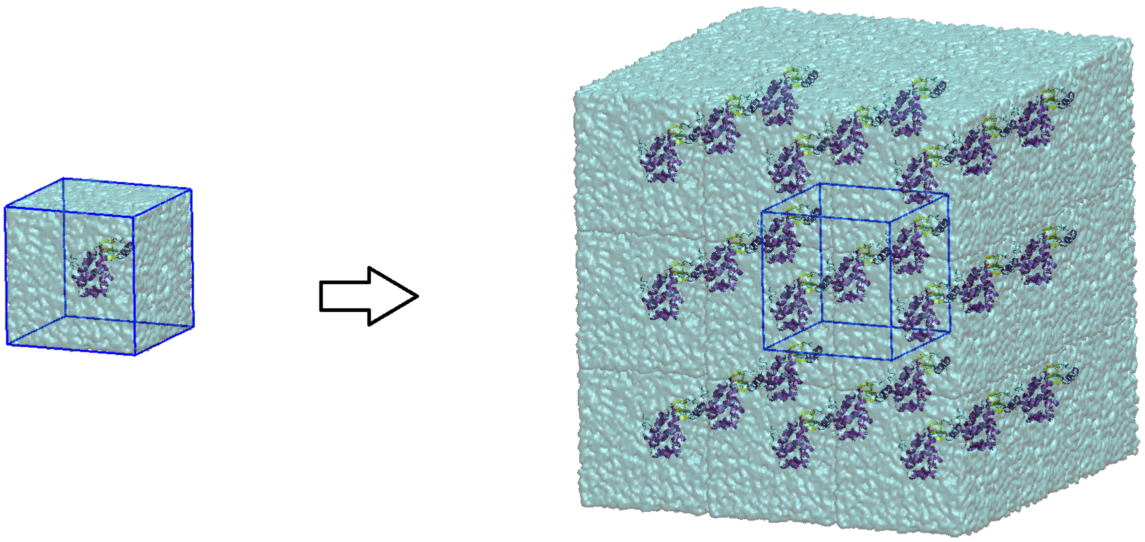


Figure 1.7: An example of partial visualization (only 26 out of the infinite copies of the cubic box) of the effect of using periodic boundary condition for simulating a solvated T4 lysozyme.

The problem of how to treat boundary conditions in a finite-size box simulation arises from the boundary effects that are typically artifacts of the simulation. In fact, in an realistic experiment of a system with N,V and T fixed, the size of N_{exp} is $\sim N_A \sim 10^{23}$, while in our simulations the groups with the highest computational power available in the world are still bound to $N_{sim} \sim 10^9$. As a consequence, putting them in a realistic box with reflecting walls translates into a proportion of solvent molecules (focusing on biomolecular systems) that is subject to non-bulk behavior much larger than the one in the experiment, whose effect on the sample can be considered negligible. To overcome this limitation, a it is a good practice to use the so-called *periodic boundary conditions* (PBC), which essentially consist in assuming the simulation box of the system of N_{sim} to be surrounded by (ideally) an infinite number of identical copies of the same simulation box, in every direction (see figure 1.7 for the representation of the first $26 = 3^3 - 1$ copies around the box).

Using PBC has the advantage to reproduce the properties of the bulk system without the need to simulate $N \sim N_{exp}$ particles [11], but one has to carefully choose the dimensions of the box in order to avoid self-interactions of the biomolecule with itself. Moreover, some problems strictly related to the use of PBC in NVE simulations [22] and to a finite-size effects to the diffusion coefficient of the particles also in NVT [23] remain unsolved, although the scientific community tends to consider them negligible.

One very powerful consequence that the use of PBC in biomolecular simulations have is the possibility to use them in combination with the Ewald Summation derived methods to calculate long-range interactions (here Coulomb interactions). In the last years, the most used protocol is the *Particle Mesh Ewald* (PME) [24]: for the Coulomb term, it essentially assumes that every charge interacts with every other charge in the infinite set of copies of the simulation box, excluding only itself. In formal terms, the total Coulomb potential is assumed to be the sum of the interactions between the particles within the "principal cell" and the interactions between each particle i in the principal cell and all the other particles in the copied cells:

$$V_{coul}(\mathbf{R}) = \frac{1}{2}\sum_{i,j\neq i}^{N} \frac{q_i q_j}{4\pi\epsilon_0|\mathbf{R}_{ij}|} + \frac{1}{2}\sum_{i,j}^{N}\sum_{\mathbf{n}\neq\mathbf{0}}{}' \frac{q_i q_j}{4\pi\epsilon_0|\mathbf{R}_{ij}+\mathbf{n}L|} \tag{1.17}$$

where the sum over $\mathbf{n}$ is over every vector of the lattice generated by the principal axes defining the box and its is a primed sum because it avoids the self-interactions of the charges; L is a generic box dimension (the box is only required to have a shape that perfectly tile into a three-dimensional crystal, whatever the lengths of the sides). This specific shape of the Coulomb potential is admitted thanks to the periodicity imposed by the PBC, and it turns out that this sum can be well approximated by the following quantity, called the *Ewald summation*, which is a sum of a short-range term in real space $V_{sr}(\mathbf{R})$, a long-range term in reciprocal (Fourier) space and a self-interaction correction term:

$$V_{coul}(\mathbf{R}) \simeq \frac{1}{2}\sum_{i,j\neq i}^{N} \frac{q_i q_j}{4\pi\epsilon_0|\mathbf{R}_{ij}|}\operatorname{erfc}\left[\sqrt{\alpha}|\mathbf{R}_{ij}|\right] + \frac{1}{2}\frac{1}{V}\sum_{\mathbf{k}\neq\mathbf{0}}\frac{4\pi}{\mathbf{k}^2}\left|\sum_{i=1}^{N} q_i e^{i\mathbf{R}_i\cdot\mathbf{k}}\right| e^{-\mathbf{k}^2/4\alpha} - \sqrt{\frac{\alpha}{\pi}}\sum_{i=1}^{N} q_i^2 \tag{1.18}$$

which can be shown [11] to have a computational cost that scales as $O(N^{3/2})$ in N. In a nutshell, the PME method is nothing but the use of Ewald summation for the calculation of long range interactions, making use of fast Fourier transform algorithms to calculate the long-range, reciprocal space part of the sum: the computational cost in this case scales as $O(N \cdot \log(N))$.

Chapter 2

The Impact on Structure and Dynamics of point-wise Missense Mutations in SBDS Protein

2.1 Introduction

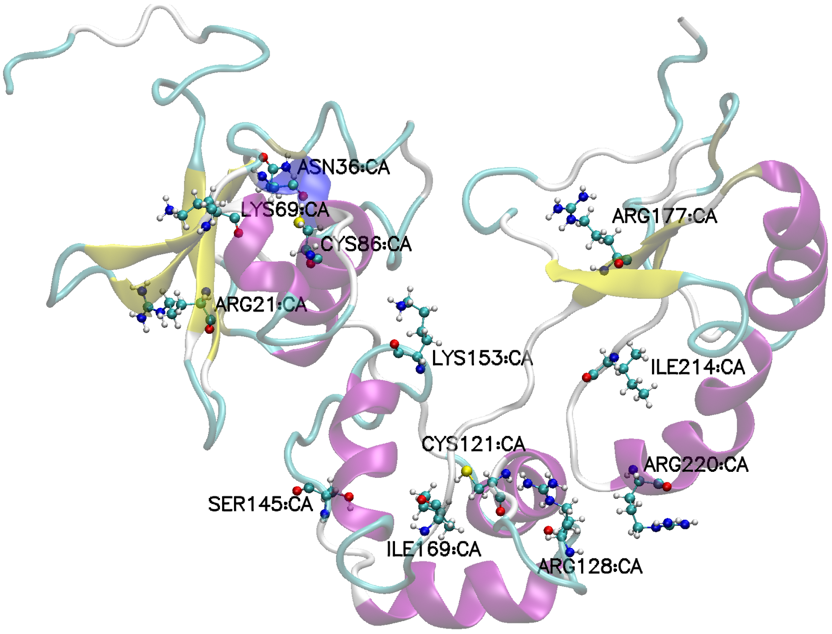


Figure 2.1: Cartoon representation of the SBDS protein (frame V taken from NMR structure [25], N-terminal on the left, C-terminal on the right). On top of it, the residues undergone mutation in this work, whose α carbons are labelled in black.

In this chapter we give an overview of a work that couples atomistic molecular dynamics simulations and small-angle x-rays scattering (SAXS) experiments of a specific protein, called *Shwachman Bodian Diamond syndrome* protein (from now on called only SBDS). SBDS is involved in a lot of biological processes (summarized in 2.2, image taken from [26]). In particular, multiple works [27, 28, 26] correlate the presence of mutations in the human gene of SBDS (and, in turn, also in the protein itself) with a rare disease, called Shwachman-Diamond Syndrome (SDS).

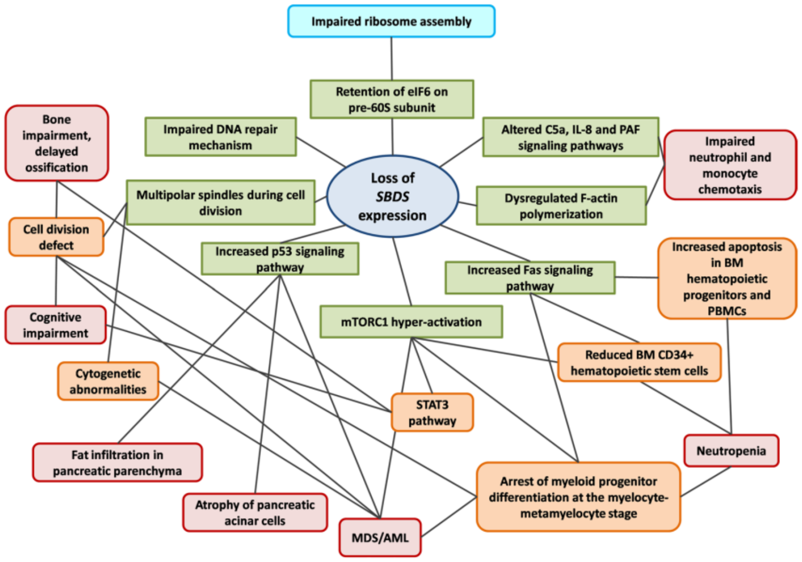

Impaired ribosome assembly
Retention of eIF6 on pre-60S subunit
Impaired DNA repair mechanism
Altered C5a, IL-8 and PAF signaling pathways
Bone impairment, delayed ossification
Impaired neutrophil and monocyte chemotaxis
Multipolar spindles during cell division
Loss of *SBDS* expression
Dysregulated F-actin polymerization
Cell division defect
Increased p53 signaling pathway
Increased Fas signaling pathway
Increased apoptosis in BM hematopoietic progenitors and PBMCs
mTORC1 hyper-activation
Cognitive impairment
Reduced BM CD34+ hematopoietic stem cells
Cytogenetic abnormalities
STAT3 pathway
Neutropenia
Fat infiltration in pancreatic parenchyma
Arrest of myeloid progenitor differentiation at the myelocyte-metamyelocyte stage
Atrophy of pancreatic acinar cells
MDS/AML

Figure 2.2: Cellular functions involving SBDS. Image taken from [26].

We decided to focus on 12 point-wise mutations of the SBDS gene found in patients with SDS, namely: R19Q, N34I, K67E, C84R, C119Y, R126T, S143L, K151N, I167T, R175W, I212T, R218Q (highlighted in 2.1). These are missense mutations, meaning that they cause the substitution of one single amino acid in the polypeptide chain without interfering with the synthesis process. Mutations in the SBDS protein associated with diseases are typically truncating mutations rather than missense mutations. Missense mutations are often coupled with truncating mutations, whereas missense mutations in conjunction with the WT form are never identified in patients with SDS. This observation suggests that the wild-type form and missense mutations result in a healthy phenotype. Despite the earlier observation, it is noteworthy that the presence of both a truncating mutation and a missense mutation still leads to the manifestation of the disease. This finding suggests that missense mutations are not functionally equivalent to the wild-type (WT) form, indicating their non-functionality in disease pathology. For these reasons, a study of the missense mutations involved in the disease could shed lights on the molecular origins that lead to the disease. More details on the involvement of SBDS in the genesis of a ribosome, and the relation of this process with the SDS disease are reported in the next paragraphs.

2.2 SDS and the ribosome maturation protein SBDS

2.2.1 SDS

SDS is a rare genetic disorder that affects multiple organ systems, including the bone marrow, pancreas, and skeletal system. SDS was first described in 1964 by two physicians, Harry Shwachman and Louis Diamond, who observed a cluster of patients with similar clinical features, including pancreatic insufficiency, skeletal abnormalities, and bone marrow disfunction.

The clinical features of SDS can vary widely, but some of the most common symptoms include failure to thrive, recurrent infections, anemia, neutropenia, and thrombocytopenia. Patients with SDS may also develop skeletal abnormalities, such as short stature, scoliosis, and rib cage abnormalities. Another hallmark of SDS is pancreatic insufficiency, which can lead to malabsorption of nutrients and chronic diarrhea.

Diagnosis of SDS typically involves a combination of clinical evaluation, laboratory tests, and genetic testing. Treatment is largely supportive and aimed at managing the various complications of the disorder.

Patients with pancreatic insufficiency may require enzyme replacement therapy to aid in digestion, and those with bone marrow dysfunction may require transfusions or bone marrow transplantation. Some patients may benefit from growth hormone therapy to improve growth and development.

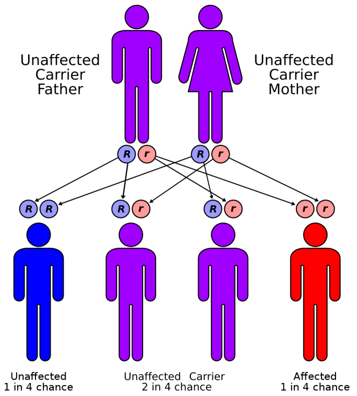

Unaffected Carrier Father
Unaffected Carrier Mother
R r
R r
R R
R r
R r
r r
Unaffected 1 in 4 chance
Unaffected Carrier 2 in 4 chance
Affected 1 in 4 chance

Figure 2.3: Mechanism of inheriting of autosomal recessive diseases, like SDS. R indicates the dominant allele, while r the recessive one. Credits: Wikipedia.

2.2.2 SBDS and EFL1 cooperativity for eIF6 release from the 60S subunit

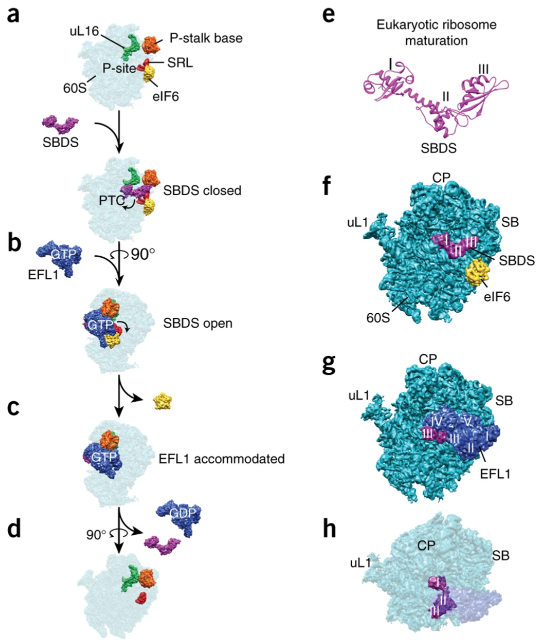


Figure 2.4: Image adapted from [29].

The ribosome is a complex molecular machine involved in protein synthesis, and its formation requires a series of complex steps. One of the last steps in this process involves the detachment of eIF6 from the upper subunit of the ribosome itself, called 60S subunit, allowing it to attach to the 40S subunit and form the mature ribosome. This detachment process is also mediated by the proteins SBDS and EFL1.

Mutations in the SBDS gene have been shown to cause dysfunction in this process, leading to the development of symptoms associated with ribosomopathies. One example of such a ribosomopathy is SDS. However, the precise molecular mechanisms by which these mutations affect the function of SBDS are not yet fully understood. The most supported version of this action mechanism can be summarized in the following steps [29, 30]:

1. initially the eIF6 factor is bound to the 60S subunit, around a zone that is named *P-site* (part **a** of 2.4) [29]

2. at some point, the free SBDS binds to the 60S subunit, thanks to its RNA binding affinity provided by domain I (part **a** of 2.4) [29]

3. successively, EFL1 binds the 60S subunit, close to SBDS (part **b** of 2.4) [29]

4. after the binding, SBDS cooperate with EFL1 to perform its GTP-ase function by converting a GTP molecule (which was already bonded to the free EFL1) to a GDP and a Pi group [30]

5. the Pi group is released from the complex, and this release induces a conformational change in SBDS which consists in a 180^o of domain III and a smaller rotation of domain II around the hinges connecting I-II and II-III (part **b** of 2.4) [30]

6. this conformational change leaves to EFL1 the space to compete with eIF6 in binding the P-site, eventually leading to the removal of eIF6 (part **b** and **c** of 2.4) [29, 30]

7. in the final step, both SBDS and EFL1 unbind from the 60S subunit which is free to merge with the 40S subunit to form the mature ribosome (part **d** of 2.4) [29, 30]

To gain insight into this process, we have simulated the behavior of SBDS in solution both in its wild-type (WT) form and in the presence of 12 of these known mutations, by means of all-atom explicit solvent molecular dynamics. By doing so, we aimed to determine how the protein's conformational changes are influenced by the mutations and whether these changes affect the protein's ability to attach to the 60S subunit and perform its function. By comparing the ensemble of conformations of the WT and mutated SBDS in solution, we were able to gain high-resolution insights into the mechanisms of action of SBDS mutations in SDS. This approach may eventually help identify new targets for therapeutic intervention in patients with SDS and other ribosomopathies.

2.3 In Silico Simulations

In this section we firstly describe the observables we decided to calculate and what is the information we aim to extract from them. Then, we show the results obtained and we discuss them, pinpointing strengths and weaknesses of the claims that can be done by looking at the values of the computed quantities. We omit to describe the details on the simulations (force fields, protocols and so on): an exhaustive treatment of this matter is reported in the appendix.

2.3.1 Workflow of the investigation

As for every molecular dynamics study of complex biological systems, and more generally, in order to solve the differential equations that describes the time-evolution of a dynamical system, one needs an initial configuration. In our case, we picked 2 structures out of the 20 from the PDB file with ID 2KDO (Figure 2.5). These are NMR structures of the human SBDS protein solved in the following conditions: $pH = 7.2$, $T = 293K$, ionic salt concentration $I = 0.071M$ (1.0 mM DTT-2, 50 mM sodium phosphate-3, 20 mM sodium chloride-4). It is known from Small Angle X-rays spectroscopy studies [28] on SBDS from *A. fulgidus* as well as from human SBDS that the protein, free in solution, explores three main conformational basins, named *stretched, closed, and relaxed* conformations (Figure 2.6).

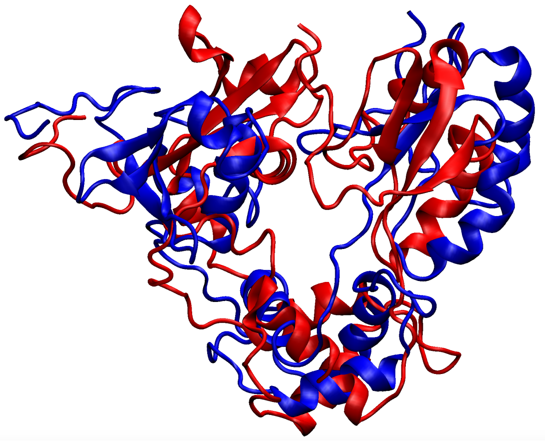


Figure 2.5: NewCartoon visualization of the *open* (blue) and *closed* (red) conformation (indices II and V in the PDB ID: 2KDO). These are the starting configurations used for the sets of the so-called open and closed simulations presented in this work.

The three basins are characterized by different angles between Domains I,II and III and an overall different size of the protein. These basins can be related to the conformational variability observed in SBDS when bonded to the 60S subunit, which is necessary for its interaction with ELF1 during eIF6 release, although the absence of the whole environment that is instead present during SBDS functioning makes a direct, quantitative comparison arguable.

The variability observed in the SAXS structures is at the origin of the choice for at least two different starting configurations, among the 20 available from NMR: it is a compromise between the computational resources we had access to and the aim to make a configuration space exploration as vast and complete as possible (for each starting configuration we wanted to produce a 500ns-long production run), with the highest number of replicas (*i.e.* higher chances to explore different conformations). As already anticipated and as explained in more

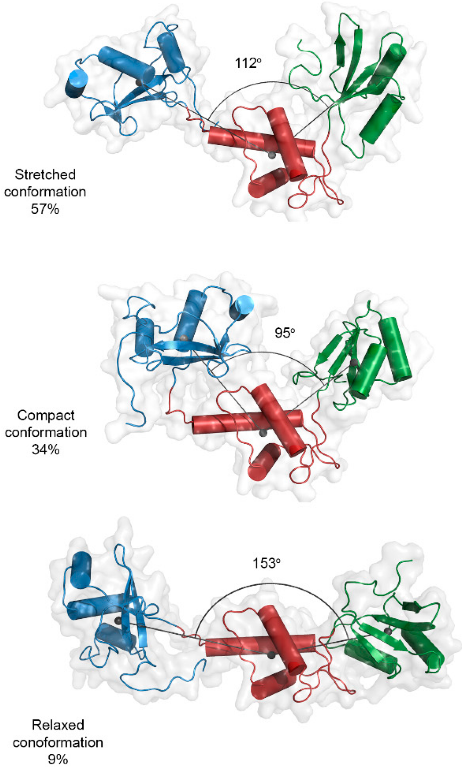


Figure 2.6: Adapted from [28].

detail in the appendix of this chapter, we then proceeded to mutate each of the two starting configurations according to the 12 mutations listed before, for a total of 26 starting configurations (considering also the wild type).

In this paragraph I will skip the details on the equilibration and simulation protocols followed. These are discussed in the appendix. I will instead review the *rationale* that stands at the basis of the analyses we performed, in order to probe some physical properties emerging from the simulations. The aim is twofold: to compare these properties to experimental facts and to see if there are matches/mismatches; to make insightful observations on the dynamical behaviour of the protein variants with respect to the WT, interpreting high-resolution information coming from atomistic molecular dynamics that are inaccessible to experiments, and use them to shed lights on the mechanisms of action of the mutations on the physiological function of SBDS.

After equilibrating the solvent (pressure $1atm$, temperature $310K$), during which the protein was constrained to remain in the starting configuration, we calculated the RMSD with respect to the first frame of the production run (500ns). The purpose of this analysis is to qualitatively assess the reach of a stable, equilibrated conformational basin in each simulation, to use as a reference for the subsequent analyses. In fact, we expect the protein to be far away from a (local) minimum of its free energy. This is due to two main facts: the presence of a new amino acid (not in the WT simulations of course) as a consequence of the *in silico* mutation, a fact that surely requires a settlement process; the potential discrepancies between the structure resolved with NMR and the parametrization of the force field used for the run. We are aware that there exist more sophisticated and quantitative methods that are prone to be precise in this kind of assessments, but we decided to keep it simpler, for the sake of interpretability. More details on this point are discussed in the Results and Discussion section.

After discarding the equilibration part of the production run, we proceeded to monitor the RMSF per residue. In general, we focused on the mobility of the hinges and specifically on the hinge connecting domain I and domain II. In fact, as pinpointed by the cryoEM structures reported in [29], after SBDS binds to the 60S ribosomal subunit, it undergoes a conformational change that requires a certain flexibility in the above-mentioned hinge and this motion is necessary to let EFL1 carry out its function. According to this observation, we wanted to probe this flexibility in our runs (protein alone, in solution) to check if it is already particularly affected (lower RMSF on the residues of the hinge with respect to the WT). We speculate that a reduced mobility of the hinge in the protein free in solution can be correlated to a similar reduced mobility of the protein bound to the 60S subunit.

We then monitored a free energy profile that emerges using the radius of gyration and the angle between the center of mass of the 3 domains as collective variables (CVs). The goal of this analysis was to have a low-dimensional representation of the conformations explored in the equilibrated part of every production run. Thanks to that, we were able to visually inspect the basin(s) explored and make qualitative and quantitative comparisons between the simulations, exploiting similarities and discrepancies. Moreover, we expanded the free energy by including a third CV, the dihedral angle formed by the center of mass of, respectively: domain I, hinge connecting domain I and II, hinge connecting domain II and III, domain III. We used this analysis to perform a clustering and to detect the most populated clusters within each simulation. We then used the configurations belonging to these clusters to calculate another quantity, which we are about to present.

We defined a quantity *alpha*, named *SASA-based binding affinity estimator* (see appendix for the formal definition) that is based on the van der Waals area of the atoms of each charged residue that is exposed and accessible to the solvent. This quantity's definition is in fact based on the well-known solvent accessible surface area (SASA), an observable typically used in molecular dynamics simulations also to model the free energy of solvation (the reader will find a more exhaustive discussion on this topic in the next chapter). The scope of calculating α, averaged on the frames that belongs to the most populated cluster (as anticipated before), was to have a semi-quantitative measure of the affinity of non-specific Coulombic binding of domain I to the ribosomal RNA that is present in the 60S subunit. It is in fact hypothesized by de Oliveira *et al.* [25] that domain I (and not domain III as was supposed before by homology considerations) is responsible for the binding of human SBDS to the 60S subunit, thanks to the diffused positive charges on this domain and the negatively charged phosphate groups on the backbone of rRNA (see Figure 2.7).

A last analysis focuses on the dynamics of the mutated and WT proteins and is based on the principal component analysis (PCA) performed on the equilibrated part of the trajectory. We built a symmetric matrix of values, each one corresponding to a pair of mutations. These values the output of the calculations of a quantity, called Ω, is a function of the eigenvalues and eigenvectors of the covariance matrix (a detailed definition is reported in the appendix). It was introduced by [32] as a proxy to estimate the similarity of two PC spaces, in order to get a quantitative measure of the similarity of the dynamical (fast and slow) modes of a protein, starting from samples of its conformational space. The goal was to further deepen the comparisons between the simulations to see whether the similarities already pointed out were

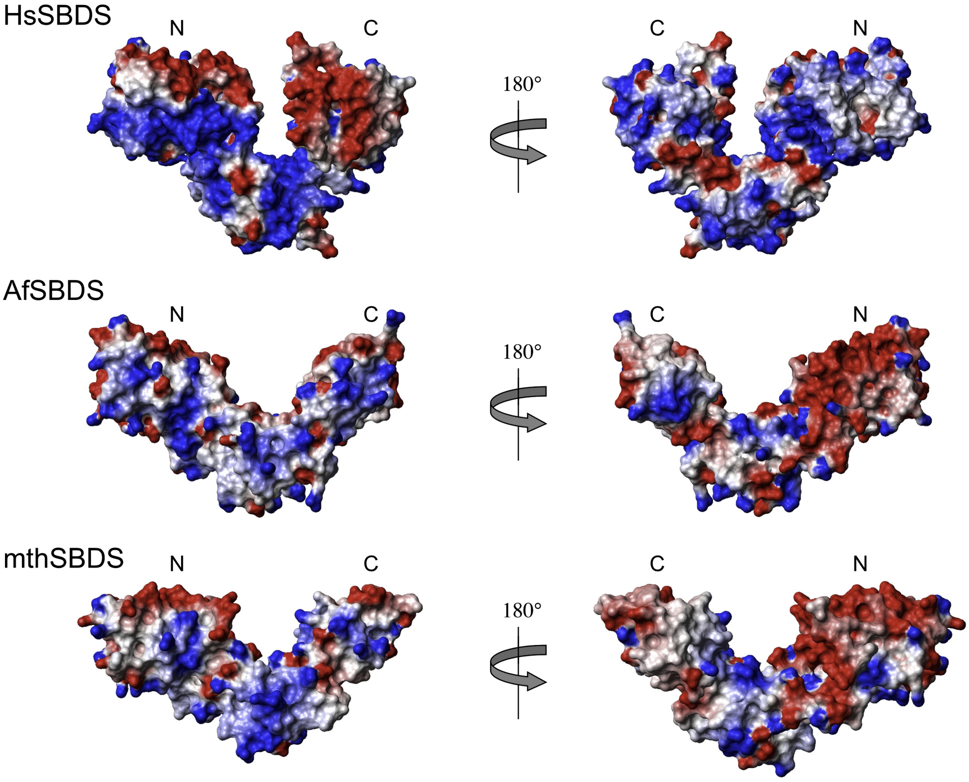


Figure 2.7: Image taken from [25]. These charge maps are generated with the software MOLMOL [31] and show positive (in blue) and negative (in red) charge distributions on SBDS surface (two views, front and back) among different species (human (Hs), A. *fulgidus* (Af), *M. thermautotrophicus* (mth)).

consistent or not with those found here, in order to make the first steps into a mechanistic categorization of the mutations.

2.3.2 Results and discussion

In this paragraph we present the results of the analyses mentioned before. For the sake of clarity, we kept each analysis and the relative discussion separated from the others and introduced by a subtitle.

RMSD analysis

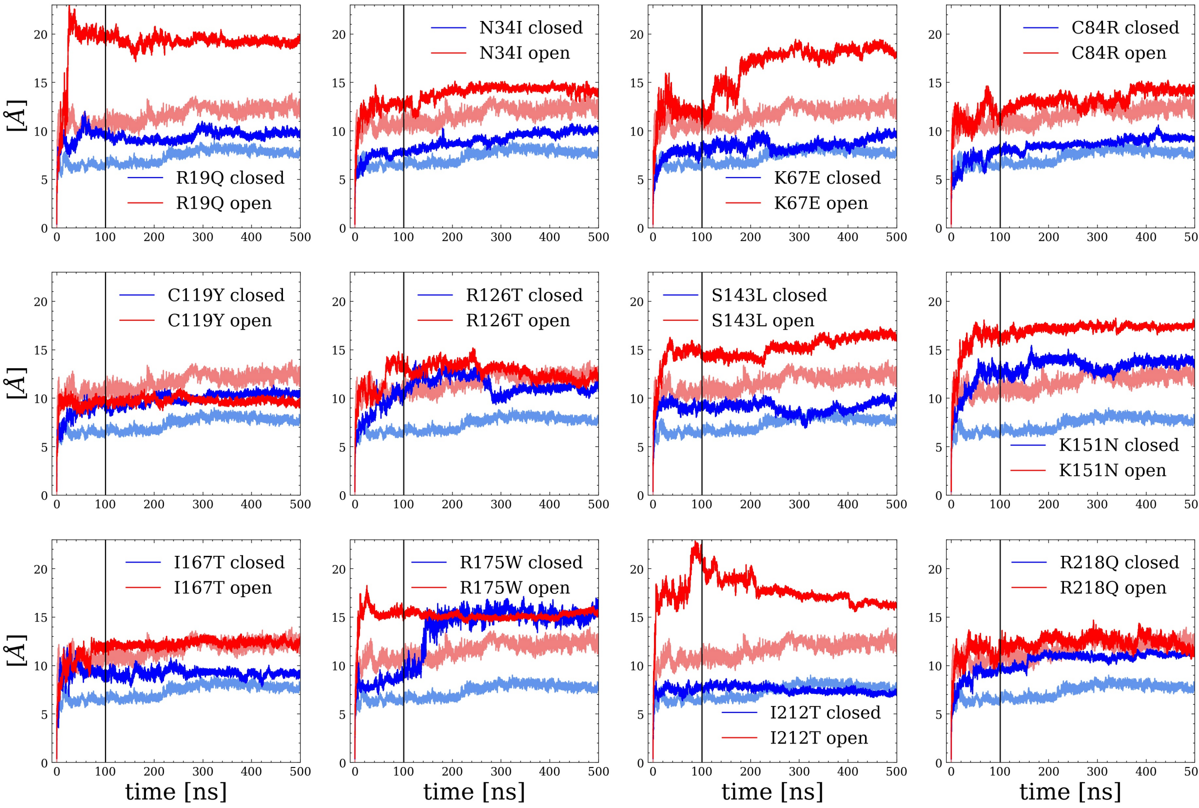


Figure 2.8: RMSD values calculated using the first frame of each trajectory as a reference. As indicated by the labels, blue curves correspond to closed trajectories, while red curves to the open ones. The light red and blue coloured lines refer to the values obtained in the WT runs, for direct comparison. The black vertical line separates the equilibration phase from the the sampling phase (at 100ns).

The calculation of the RMSD is performed by aligning all frames to the first one. Essentially, the first frame is the same for all simulations (between open and closed, of course), since during the solvent equilibration phases the protein coordinates are constrained. To determine which part of the trajectories to assume as sufficiently equilibrated for subsequent analyses, we employed an ad hoc criterion based on visual inspection. We chose to truncate all trajectories at the same point, considering the trajectory lengths were the same. This point corresponds to the first 100ns, as stated beforehand and shown in the figure 2.8 (black vertical line), which subsequently corresponds to relatively stable RMSD values, with slight variations observed in some trajectories and the most evident one in the R175W closed trajectory. This last transition highlighted by the RMSD with respect to the initial frame indicates that the protein in this simulation may reach a more stable local minimum after the first 100ns, around 200ns. Nonetheless, to remain consistent with the criterion, we decided to keep the frames before the transition, taking that into account for subsequent analyses. To strengthen our choice, we also checked the values of the *cosine content* (introduced by Hess [32] in order to identify non-relaxed trajectories by means of PCA). The values, reported in the Appendix, are below the dangerous range $[0.7, 1]$ and so we can deduce that at east this test do not identify pathologies in the choice of considering the $[100ns, 500ns]$ interval as equilibrated. These analyses will reveal a peculiar behavior that can be attributed to the transition. To support even further the assumption that our trajectories relaxed to a metastable state, in the Appendix (figures 2.16,2.17) we also reported the RMSD calculated on the α carbons of domain I (residues 9-95), domain II (residues 107-167) and domain III (residues 173-236) using the convention reported by de Oliveira *et al.* [25], after local alignments.

However, we would like to emphasize that we do not claim to establish that thermodynamic equilibrium (intended as an absolute minimum of the free energy) has been reached. It is nowadays generally accepted that, at equilibrium, a protein's landscape is rugged, and thus the protein has the possibility to jump from one conformational basin to another, and hence, the sampling of each of them provides useful information of the protein's equilibrium properties. What definitely remains a limit of molecular dynamics simulations is the inability to establish whether a trajectory can be effectively considered ergodic, so as to leverage the ergodic principle to establish that the averages over the trajectory frames (if sufficiently uncorrelated) are equivalent to the averages over the ensemble. In spite of that, it remains in principle predictive in characterizing the observables in the explorable metastable basins. Nevertheless, we consider one of the strengths of molecular dynamics to be the dynamic character of the trajectories generated by it. Therefore, we deem it useful to attempt to extract insights from in-depth analyses of trajectories that cannot be regarded as ergodic but are reasonably long enough to provide sampling that accesses the dynamic behavior of the system, albeit partially. Based on this consideration, we believe that the scientific contribution that molecular dynamics (in general, and in particular as the tool employed in this work) provides is to be an high-resolution, cost-effective "probe" that can guide important choices to be made in future experiments and suggests action mechanisms that can be further investigated and clarified.

RMSF analysis

In this paragraph, we attempt to rationalize the dense information content presented in the figures representing the RMSF per residue. Initially, we characterize the curves corresponding to wild type runs, which we use as a reference. Subsequently, we constructed a table 2.1 where, for each region of the protein (domain I, hinge I-II, domain II, hinge II-III, domain III), we indicate the corresponding behaviors that are considered "anomalous", i.e., those that differ substantially from the behavior of the corresponding wild type. In the table, we compare open and closed trajectories, with each mutation listed in rows, thus enabling us to highlight similarities and differences in the behavior of open and closed runs. Finally, we proceed to advance hypotheses that are drawn from this analysis through direct comparisons with observations from experiments and simulations.

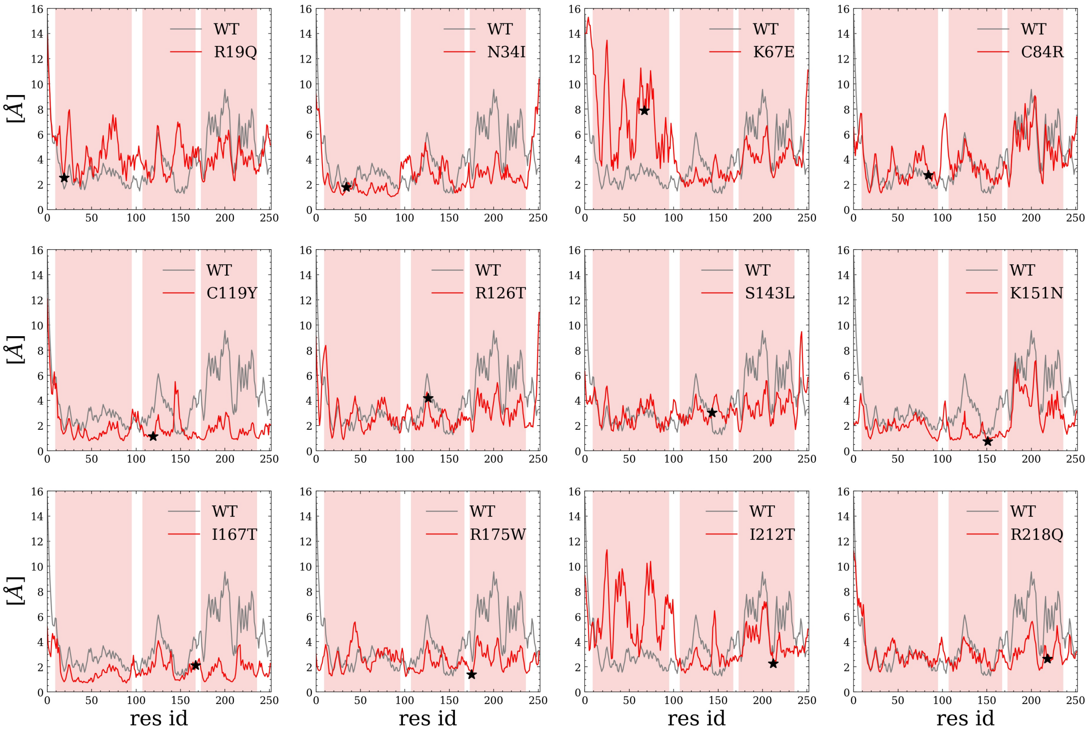


Figure 2.9: RMSF per residue, relative to the open trajectories for each mutation. In the background, we reported in grey the WT RMSF for direct comparison. We also highlighted the residue involved in the mutation with a black star and the relative domains/unstructured regions with red/white background.

concerning the wild type, in the open simulation the main feature that is shown is a domain III substantially fluctuating more than the other two. This pattern is not observed in the closed simulation, where the most fluctuating domain is the II (neglecting the N- and C-terminal that are expected to move a lot with respect to their average position, due to their unstructured nature). Other than these facts, however, no noteworthy behaviour in the amino acids of the hinge I-II arises from this analysis. We are therefore induced to think that the flexibility required for the above-mentioned hinge to perform the conformational change that is functional to SBDS attached to the 60S subunit (see step 2. in the discussion presented in paragraph 1.2.2) is modest. On the other hand, by looking at the third and fourth columns of 2.1, one can notice a common, statistically relevant effect of the mutations on the flexibility of the hinge: for every mutation, in fact, in at least one of the two runs, the fluctuations of hinge I-II is enhanced with respect to the wild type. We are led to speculate that *one pathological effect of the mutations*

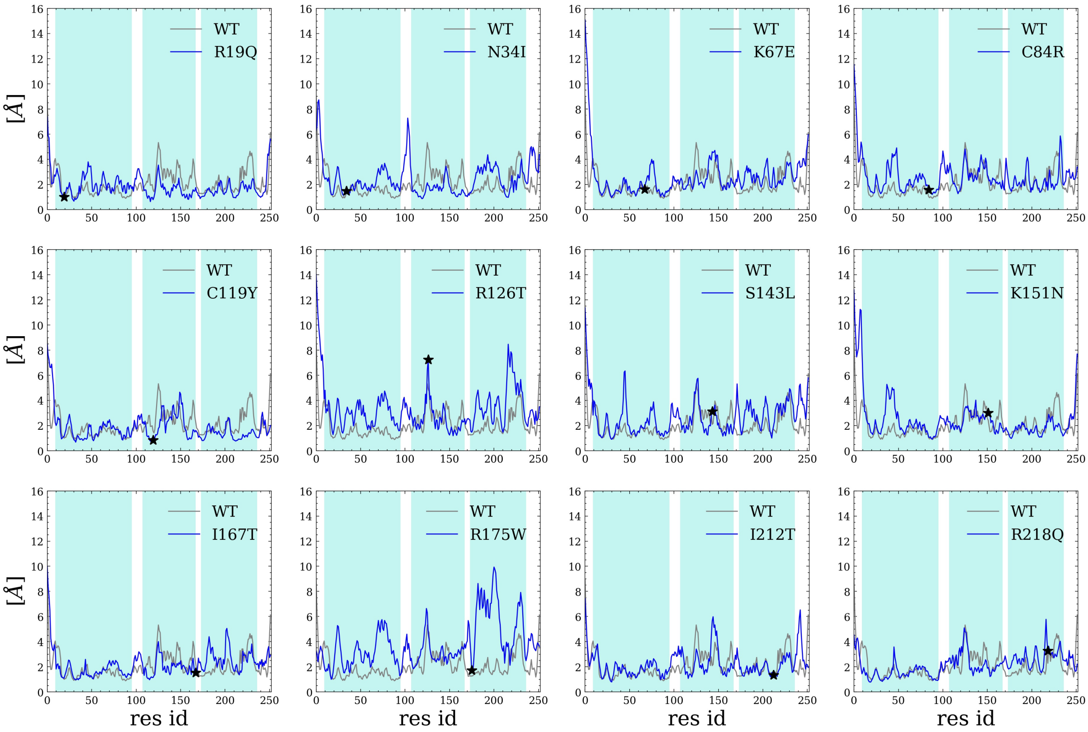


Figure 2.10: RMSF per residue, relative to the closed trajectories for each mutation. In the background, we reported in grey the WT RMSF for direct comparison. We also highlighted the residue involved in the mutation with a black star and the relative domains/unstructured regions with lightblue/white background.

studied here could be to impose an excessive mobility to the hinge I-II that, in turn, leads to a partial loss of the ability to perform the conformational change mentioned before. In this view, some sort of long-range communication pathway is expected to exist among these residues and domain III.

Another feature that can be easily extracted from table 2.1 is that for the open trajectories hinge II-III and domain III are over-stabilized, with respect to the wild type. This fact was also observed in the simulations of Spinetti *et al.* [33]: in fact, they noticed an over-stabilization of domain III in all the mutations investigated (R19Q, R126T and I212T).

Given this compatibility in spite of the differences in their setup (NPT ensemble and the force field used) we are led to interpret this over-stabilization as a general dynamical feature of the mutations, rather than a statistical outlier.

	domain I		hinge I-II		domain II		hinge II-III		domain III	
	O	C	O	C	O	C	O	C	O	C
R19Q	↑	↑	↑	↑	↑	↓	∼	↓	↓	↓
N34I	↓	↑	↑	↑	∼	↓	↓	∼	↓	×
K67E	↑	↑	↑	∼	↓	×	↑	↓	↓	↑
C84R	∼	↑	↑	↑	∼	∼	∼	↑	∼	∼
C119Y	↓	∼	∼	∼	↓	↓	↓	∼	↓	↓
R126T	∼	↑	↑	↑	∼	↑	↓	↑	↓	↑
S143L	∼	↑	↑	↑	∼	×	↓	↑	↓	↑
K151N	∼	↑	↑	∼	↓	∼	↓	∼	↓	×
I167T	↓	∼	↓	↑	↓	↓	↓	↑	↓	×
R175W	∼	↓	∼	↑	∼	↓	↓	↑	↓	↓
I212T	↑	∼	∼	↑	×	×	∼	↑	↓	∼
R218Q	∼	∼	↑	∼	∼	×	↓	↑	↑	×

Table 2.1: Summary of the discrepancies highlighted by RMSF between the WT and the mutated trajectories, region-wise along the sequence of the protein. ∼ indicates that the RMSF in the given region, for the respective run, shows fluctuations comparable to those of the WT, while ×, ↑ or ↓ indicate respectively different, generally higher or generally lower fluctuations in that region, with respect to the WT. A color code has been used to relate mutated residues to the belonging domains.

Free Energy surfaces and clustering

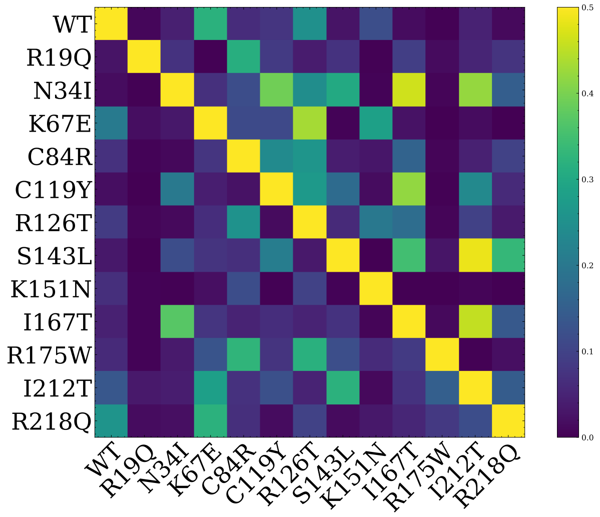


Figure 2.11: Jensen-Shannon divergence based similarity matrix of the free energy landscape sampled in the open (lower-left values) and closed (upper-right values) runs. The color map was compressed in the range $[0, 0.5]$ to highlight the relative differences, although in principle the divergence can take values in $[0, 1]$.

In this section we report the free energy landscape for each simulation, which are built by considering as collective variables (CVs) the radius of gyration and the angle formed by the centers of mass of domain I,II and III. The plots reported in figures 2.12 and 2.13 give a direct picture of the low-dimensional conformational variability explored in the simulations and in fact, based also on these values (see the definition of the V-space in Appendix) we decided to perform a clustering that is functional to the next analysis of the binding propensity.

In order to make a quantitative comparison of the frequentist probabilities sampled by the various mutations and the WT, we made use of the Jensen-Shannon (JS) divergence [34] (see Appendix for the definition). This quantity is interpreted as a distance (that takes only values in $[0, 1]$) based on which we built two dendograms (see 2.25 and 2.26) that are preparatory for a hierarchical clustering based on the average linkage algorithm [35].

The mutual values of the JS distances are reported in a compact way (thanks to the symmetric nature of the distance matrix) in 2.11, where the lower-left part of the matrix shows the values from the open trajectories and upper-right from the closed ones. This analysis reveals that the R19Q simulation, among the open ones, and the R175W simulation, among the closed ones, are characterized by the most peculiar and different frequentist probability on the space of (R_g, θ). Curiously, K151N is highlighted as the second most different mutation in both the groups, a fact that will be discussed in more detail later on. The WT simulations are found to be grouped with K67E both the times, suggesting that K67E could affect negatively the function of SBDS not directly from a structural point of view, but in other way: this observation will be explored more deeply in the next analysis.

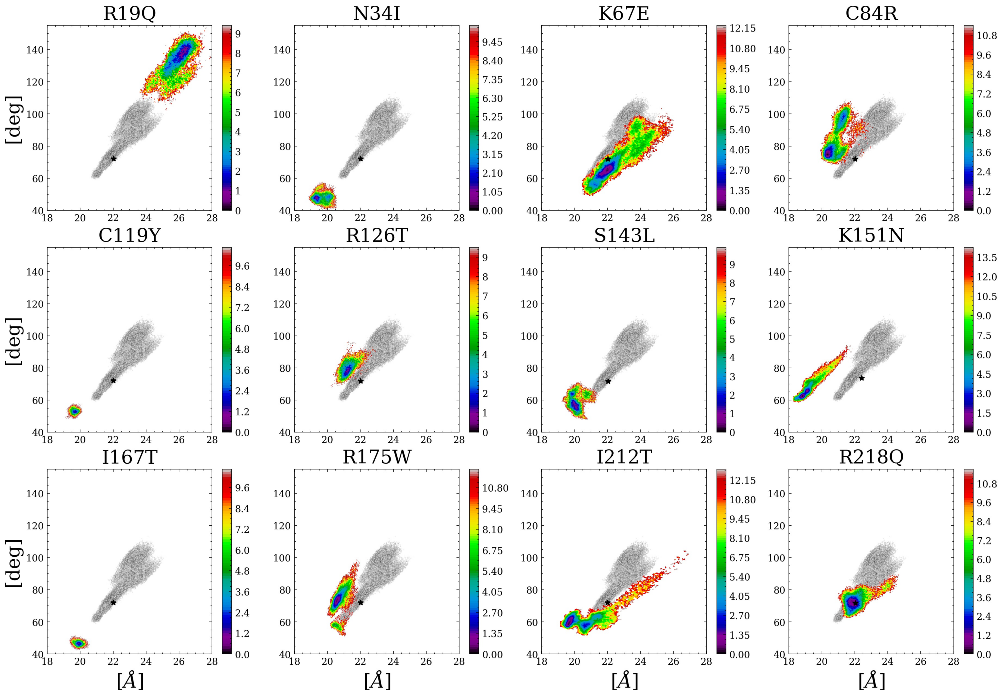


Figure 2.12: Free energy landscapes for each mutations, built from the histograms $h[R_g, \theta]$ sampled in the open runs. R_g and the angle between the domains θ are plotted along the x and y axis, respectively. The color maps report values in $kcal/mol$, normalized to be consistent with a temperature $T = 310K$ (as explained in the documentation of the pyEMMA package).

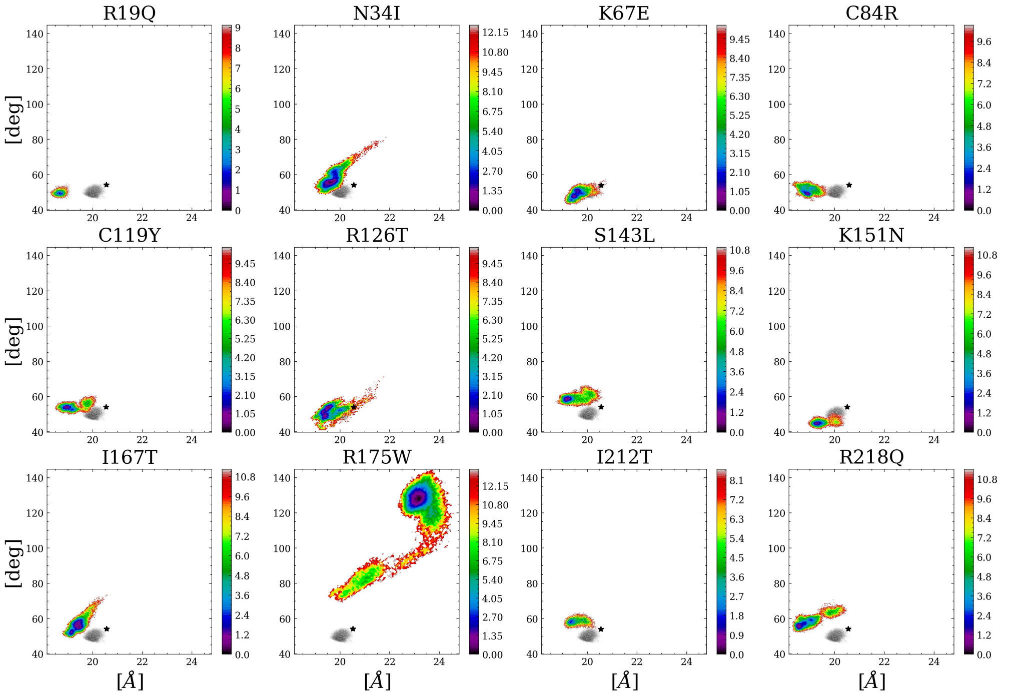
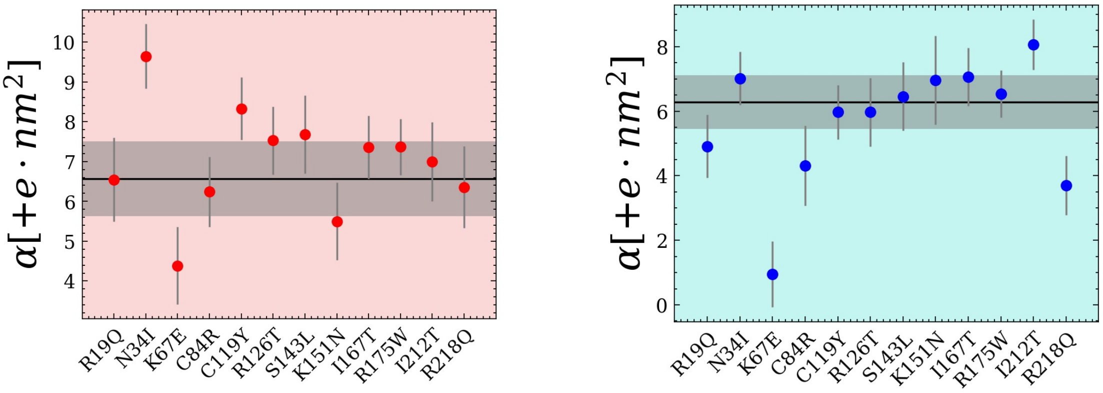


Figure 2.13: Free energy landscapes for each mutations, built from the histograms $h[R_g, \theta]$ sampled in the closed runs. R_g and the angle between the domains θ are plotted along the x and y axis, respectively. The color maps report values in $kcal/mol$, normalized to be consistent with a temperature $T = 310K$ (as explained in the documentation of the pyEMMA package).

Solvent Accessible Surface Area and binding propensity



(a)

(b)

Figure 2.14: Values of α related to domain I of the open **(a)** and closed **(b)** simulations. The black horizontal lines correspond to the value of α for the WT runs, while the grey area covers the standard deviation of them.

In de Oliveira *et al.* [25], where the structure of human SBDS is resolved via NMR experiment, they noted that, like other organisms that carry the same gene, the protein is characterized by three main domains. However, by making a static analysis of the charge distribution and by performing *ad-hoc* experiments (chemical shifts analyses) to probe the binding affinities, a substantial difference is identified: contrary to expectations that the C-terminal domain (domain III) is the site of RNA binding and therefore putatively responsible for attaching to rRNA to carry out its function, they observe that the RNA binding affinity of residues on this domain is very low and discard this hypothesis. Not only that, but they instead measure good binding affinity between RNA and some residues of the N-terminal domain I, and postulate that it is responsible for the above-mentioned binding. Starting from this hypothesis, as previously mentioned, we conducted a study of the SASA exposed by positively and negatively charged residues in domain I for each of the 26 simulations. The aim was to investigate whether certain mutations resulted in a greater or lesser exposure to solvent and the environment by charged residues, and to determine whether any exhibited higher coverage of positive residues and greater exposure of negative residues. This could indicate a lower binding affinity between this domain and rRNA, as the starting hypothesis assumed a Coulombic and non-specific binding primarily mediated by charge distribution.

In the first step of this analysis we performed a DBSCAN clustering [36] of the structures sampled in the production run, based on the euclidean distance in the space of 3 CVs: the two already used in the previous paragraph and the dihedral angle formed by the centers of mass of domain I, hinge I-II, hinge II-III and domain III (see Appendix). The results of the clustering are reported in 2.19 and 2.20 and clearly show the presence of multiple conformational basins in some of the trajectories. Based on this evidence, we decided to perform the SASA-based analysis only on those frames belonging to the most populated cluster for each trajectory. The reason behind this choice is to try to avoid mixing (or, in statistical terms, averaging on) structures that are too dissimilar. We then calculated the average SASA (using the command *gmx sasa* of GROMACS v2018) of each charged residue in domain I. The are reported in figures 2.21, 2.23, 2.22, 2.24 with their standard deviation. After that, we introduced a quantity named *SASA-based binding affinity estimator*, indicated with α (see Appendix for the formal definition), to track the total value of the SASA of these residues but weighted by their charge: in this way, higher values of the SASA for a positively/negatively charged residue contribute to the binding to the negatively charged phosphate groups on the rRNA backbones by favouring/disfavouring it. The values of α related to each mutation are reported in 2.14a and 2.14b: the standard deviation have been propagated by the square root of the single deviations squared.

As mentioned before, we notice that in the open simulations this analysis highlights the K67E one as a disfavouring mutation and N34I as a favouring one. For the closed one, we notice again K67E together with R19Q, C84R and R218Q as disfavouring mutations. An interesting fact is that precisely R19Q and K67E were hypothesized by de Oliveira *et al.* to be potential mutations affecting RNA binding. Another interesting fact is that in Gijsberg *et al.* [28], a study on the binding affinity between SBDS and EFL1 in solution, they found that R19Q and K67E are crucial in what they call the second binding event between the two proteins. Moreover, Wies *et al.* [29] identify K67 as a binding site of SBDS to the 60S subunit's P loop. Based on our simulations, we can corroborate these observations by saying that this loss of affinity can be caused by a lower exposure of positive charged patches on domain I.

Comparisons of the Principal Components' space

The last analysis we performed is somewhat orthogonal to the one based on the free energy space built from $p(R_g, \theta)$ and the Jensen-Shannon divergence. It relies on the *principal component analysis* (PCA), which is a simple yet very informative method of dimensionality reductions that finds many applications in molecular simulations, due to its cheap but effective operating principles. The idea is to make a change of variables from the $3N \cdot T$-dimensional space of the trajectory (N atoms, T frames used for the sampling, equilibrated part) to a new $3N \cdot T$-dimensional space, called the PC's space. The power of this change of variable is that it allows to build a hierarchy of directions in this space that are weighted by the eigenvalues of the transformation matrix used for the change of variable. It can be shown that this matrix is nothing but the *covariance matrix* and the new variables (called PC, *principal components*) keep track of the directions that, if the trajectory would be projected onto, show higher (or lower) variability. The directions with the highest eigenvalues (usually called λ_k) are those that retain the highest variability of the trajectory, while those with the lowest are the other way around. In this way, the PC space is able to extract the signal (directions with highest λ_k) from the noise (directions with lowest λ_k). We decided to used the information extracted by the PCA to compare the PC spaces (intended as dynamical space because they are directly built from the trajectories) using a quantity called *covariance overlap*, introduced by Hess [32]. The quantity, indicated with the symbol $\Omega(T_A, T_B)$ (and defined also in the Appendix), gives a degree of similarity and is expressed as a real value in the interval $[0, 1]$, with 0 corresponding to totally orthogonal spaces and 1 to identical spaces. Based on it, we calculated the pseudo-distances $d_\Omega(T_A, T_B) := 1 - \Omega(T_A, T_B)$ between each couple of the trajectories, within the same set of open or closed simulations (as done for the Jensen-Shannon divergence based analysis). The results are reported in 2.15.

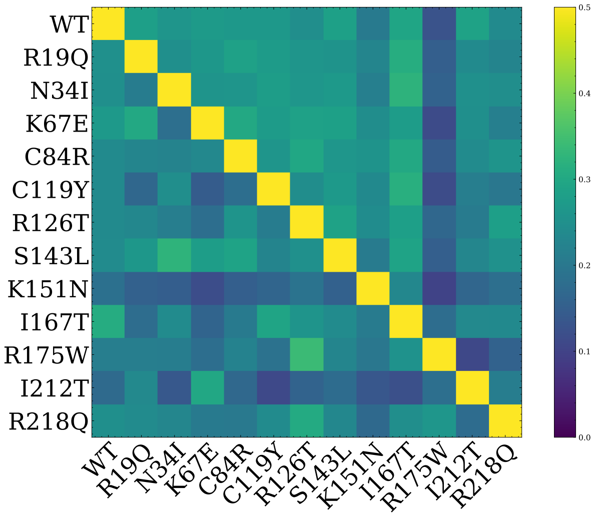


Figure 2.15: Ω based pseudo-distance matrix of the free energy landscape sampled in the open (lower-left values) and closed (upper-right values) runs. The colormap was compressed in the range $[0, 0.5]$ to highlight the differences, although in principle the distance can take values in $[0, 1]$.

As for the Jensen-Shannon divergence, we also built two dendograms (see 2.27 and 2.28) that are preparatory for a hierarchical clustering and based on the average linkage algorithm. The dendograms highlight what was already perceivable from 2.15: in both the closed and the open trajectories K151N is highlighted as an outlier, although for the closed trajectories the most different one is R175W, in full agreement with the corresponding dendogram based on the JS divergence. The DBSCAN clustering 2.20 and visual inspection of the FEP 2.13 give a clear explanation for the diversity of R175W: in the equilibrated part of the production run, the mutated protein clearly explore two different conformational basins and the PCA embraces the directions of both the local minima explorations as well as the transition between them. Regarding K151N, it was observed by Weis *et al.* [29] that residues K151 and R218 have a key role in stabilizing the conformation that SBDS assumes after interacting with EFL1 (see figure 2.2 part **b**) and so we can make the guess, based on the observations from our simulations where K151N has a dynamical behaviour that differs substantially from the WT, which the mutation in this residue brutally disrupts the ability of SBDS to stabilize after the above-mentioned conformational change.

2.4 Conclusions and perspectives

This work is a massive comparative study of the behavior of 12 pathogenic mutations of SBDS and WT, through atomistic molecular dynamics simulations in explicit solvent. Simulations of 500ns of production run were performed for two replicas of each mutation, for a total of $13\mu s$ of sampling. The aim of the simulations was to characterize the conformational space of the mutations, highlight similarities or differences and try to correlate their dynamic behavior in simulation to experimental facts previously reported in the literature. Below we summarize the main observations that we were able to do by looking at the analyses performed on the trajectories produced.

1. The analysis of RMSF highlights a common trend among all mutations to increase residue fluctuations in the hinge I-II. This fact can negatively interfere with the conformational change that involves domains II-III reorienting to make room for EFL1 to stretch out and induce eIF6 to detach from the 60S subunit.

2. We observe that K67E behaves in a structurally similar manner to the WT, as also shown by free energy analysis, despite the fact that in the analysis of binding affinity, it is highlighted in both cases as a mutation with a lower exposed positive charge. Weis *et al.* suggested that K67 was essential in RNA binding, and the message derived from my simulations is that the pathological mechanism associated to the mutation of this residue may be more closely linked to a decrease in the value of α (SASA-based estimator of binding affinity) rather than to structural deformation.

3. Other mutations (R19Q and C84R), only in the closed trajectories, are identified as having lower binding affinity: the same mutations were highlighted by de Oliveira *et al.* as potentially reducing the bond with RNA. Therefore, our simulations corroborate a hypothesis advanced by these experimental observations.

4. Weis *et al.* noted that K151 and R218 are essential in stabilizing the conformation assumed by SBDS after EFL1 binds to the 60S subunit: it is interesting to note that not only in the clustering of distances in PC space, but also in that of distances in free energy, K151N is actually highlighted as being different from the others, both in open and closed simulations; we are led to speculate that the impact of these mutations has repercussions on the dynamics and that compromised dynamics leads to an inability of the protein to stabilize a functional conformation for its cooperation with EFL1.

Among the prospects for expanding studies based on the results of this work, one could certainly perform docking between conformations sampled from my simulations and the cryoEM maps [29] of the 60S subunit, to see if the conformations extracted from the dynamics are more or less similar to what is identified as the binding site in cryoEM. Additionally, one could further explore the binding affinity with EFL1 by examining the value of $\alpha^{(II)}$ and $\alpha^{(III)}$ on domains II and III respectively, as they are hypothesized to be the mediators of the first binding event with EFL1 and interesting data on this affinity for many of the mutations simulated for this work are reported in Gijsberg *et al.* [28]. A last, but not least, observation is related to the lack of replicas per each mutation, among our simulations. It is in fact well documented in literature (see *e.g.* [37, 38]) that the statistical relevance of the deductions that arise the phase space sampling of a system by a single, long run is low, due to its lack of reproducibility. On the other hands, multiple short runs revealed to produce more reliable results. A way to valorise the work done here is to select few among the most interesting mutations, based on the results obtained, and perform short replicas to extract more reliable statistics from them, in order to test the observations made here.

The numerical experiments presented here can serve as a guide to explore in detail specific mutations and their mechanism of action in order to shed light on the still obscure points of the pathogenesis of SDS and, in the future, to support the search for a cure for this rare disease. Despite ongoing research, the underlying molecular mechanisms of SDS are not yet fully understood, and there is currently no cure for the disorder. However, continued research into the biology of SBDS and ribosome biogenesis may ultimately lead to the development of new therapies for SDS and other disorders that affect ribosome function.

2.5 Appendix

In this section I report the simulations setup; the plots of RMSDs of the three domains per mutation and conformation; the plots of the configurations in the (R_g, θ) space, highlighting the clusters identified by the DBSCAN algorithm; the SASA values (per charged residue, adding up the contributions of each atom); the dendograms used to perform the hierarchical clustering; the observables calculated for the analysis.

Simulations Setup

The human SBDS protein structure with PDB ID: 2KDO was selected for MD simulations. We extracted 2 structures from the 20 conformers from NMR data (II and V frame contained in the PDB, respectively): we referred to the *open* and *closed* conformations and simulations in the text. We generated 12 missense point mutants [26, 28] of both the open and closed conformations: R19Q, N34I, K67E, C84R, C119Y, R126T, S143L, K151N, I167T, R175W, I212T, R218Q. These mutations were performed in VMD (v1.2.3) [39] using the Mutator Plugin (v1.5). At the end of this process we obtained 26 different conformations (12 mutants and the wild type, each in both the open and closed conformations) that we used as input to build the topology for the MD simulation in GROMACS (v2018) [40]. The force field adopted was Amber ff99SB-ILDN [41] and the water model was TIP3P [42]. The protonation states of the charged residues was automatically selected by GROMACS consistent with a pH of 7. Each structure was solvated in a water box having at least 15Å from the closest box side. We ionized the solvent with 0.15 M of KCl and neutralized the whole box. We used a cutoff distance of 12Å for van der Waals (vdW) interactions, while long-range electrostatic forces were computed using the particle-mesh Ewald method [24]. A time step of 2 fs was used for every simulation. Each solvated system underwent an energy minimization, using the steepest descent algorithm with maximum tolerance force: 500.0 $kJ/mol/nm$. The last two step before the production run consisted in the equilibration of the solvent, performed restraining the protein to the minimized configuration. The first was a 500ps-long simulation in the NVT ensemble with the modified velocity rescale thermostat [21], with 310K reference temperature, while the second was another 500ps-long simulation in the NPT ensemble with the Parrinello-Rahman barostat [43] (and the same thermostat), with 1 atm reference pressure. We performed 500ns-long production runs for each of the 26 starting structures, for a total of 13μs, in the NVT ensemble (modified velocity rescale thermostat, 310K reference temperature).

RMSD on the domains

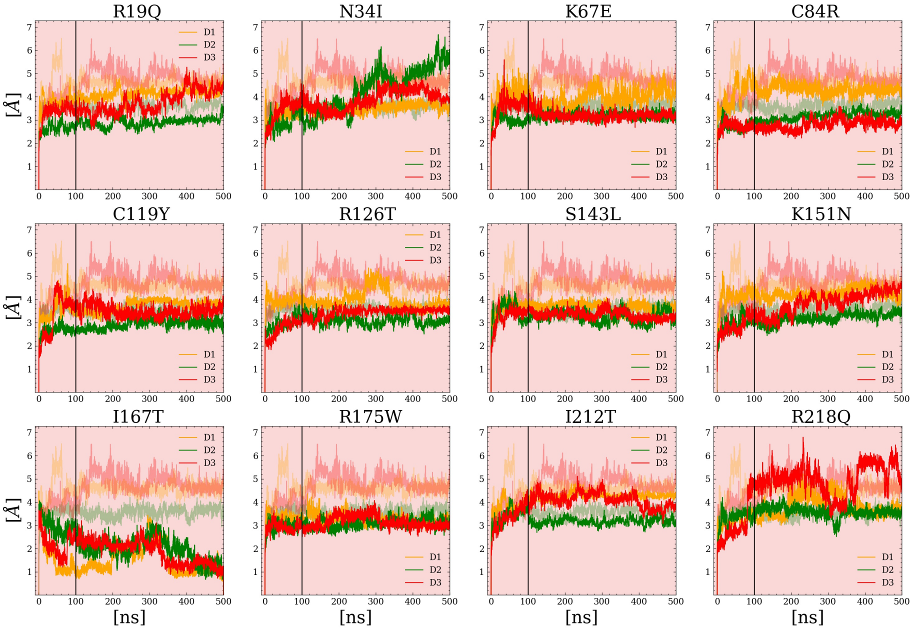


Figure 2.16: Open.

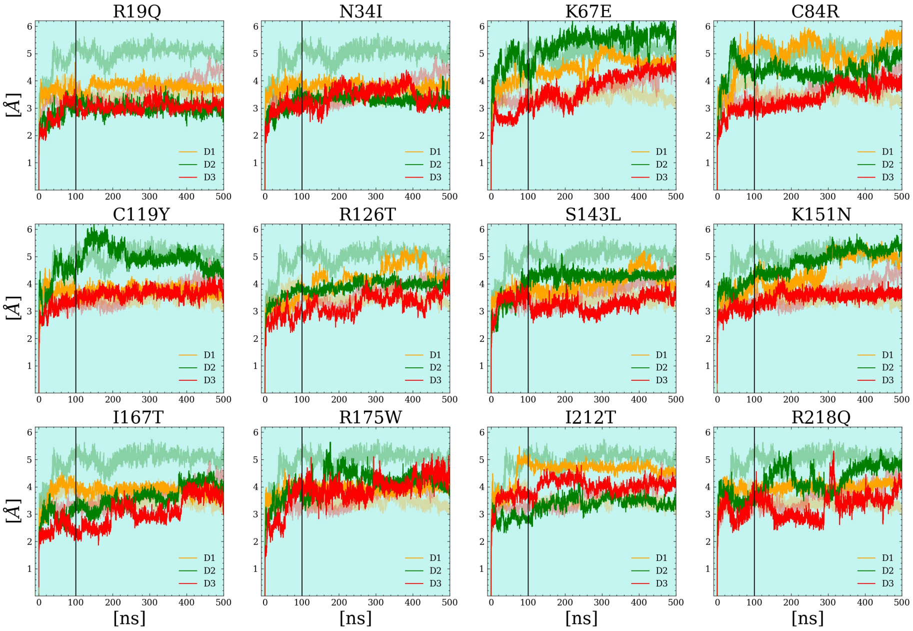


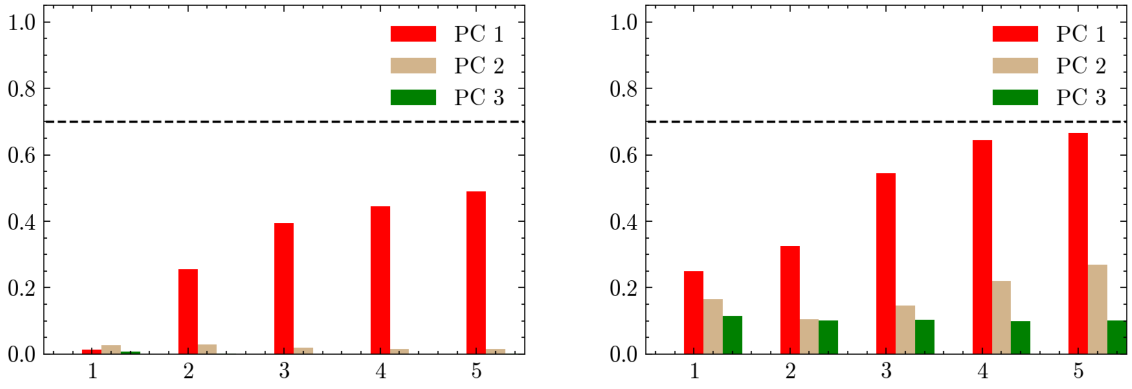
Figure 2.17: Closed.

Cosine content analysis

As exhaustively discussed in [32], the cosine content quantifies the similarity of the variables, associated to the dynamics of a system, to those of a random diffusion process. If we consider the principal components (PCs) $p_1(t)$, $p_2(t)$, ... as these dynamical variables, it is as follow:

$$cc[p_i;t_f] := \frac{2}{t_f}\left\{\int_0^{t_f} \cos\left[\frac{(i+1)\pi t}{t_f}\right] p_i(t)\,dt\right\}^2 \left(\int_0^{t_f} p_i^2(t)\,dt\right)^{-1} \tag{2.1}$$

where t_f is the ending frame of the trajectory: by changing this parameter ($t_f = 100ns$, $200ns$, $300ns$, $400ns$, $500ns$), we quantified the cosine content for different, progressively longer blocks of the trajectory, in order to mainly compare the values with $t_f = 100ns$ and $t_f = 200ns$ and to check if there is a trend in the quantity. The results for the R175W trajectories are reported in figure 2.18.



(a) **(b)**

Figure 2.18: Cosine content values of the first 3 PCs, for the **(a)** open and **(b)** closed trajectory of R175W-mutated SBDS. The dashed black line highlights the value 0.7 indicated by Hess [32] as the threshold below which the cosine content has to be considered dangerously high, and so the dynamics dangerously similar to a random diffusion process.

The first observation that we can make is that all the values reported in the figures are below 0.7, from which we can state that the two dynamics are not highlighted as too much similar to random diffusion processes, *regardless of the length of the trajectory's blocks.* Another relevant observation to be made is that the values of the cosine content grow with the length of the trajectory's blocks, a trend that is hard to be interpreted but that could indicate that a trajectory reaching a timescale of microseconds could have values higher than 0.7. A future perspective of this work is to check these values on a longer simulation.

DBSCAN clustering

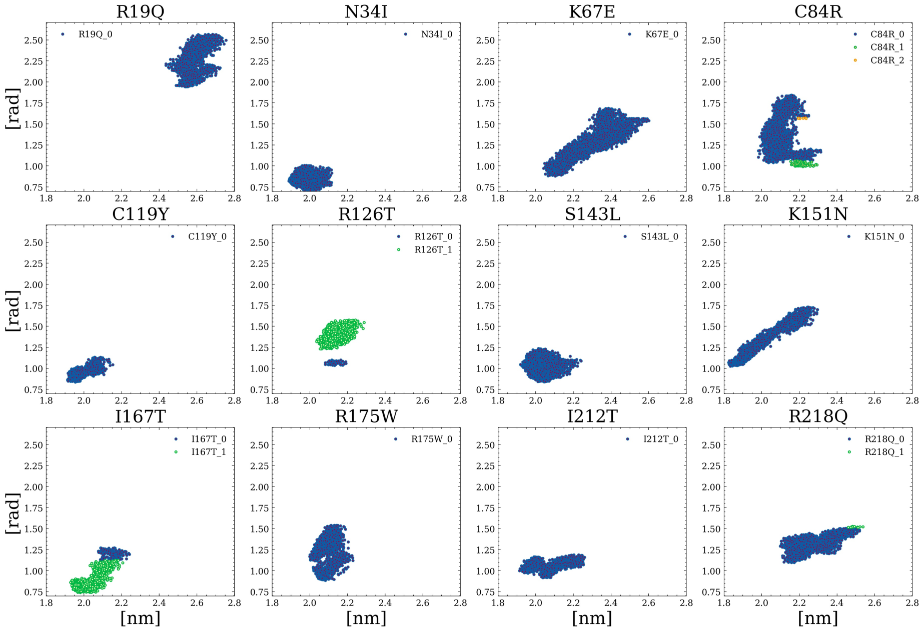


Figure 2.19: Open. $\epsilon = 0.5$. $N_{\text{min}} = 100$.

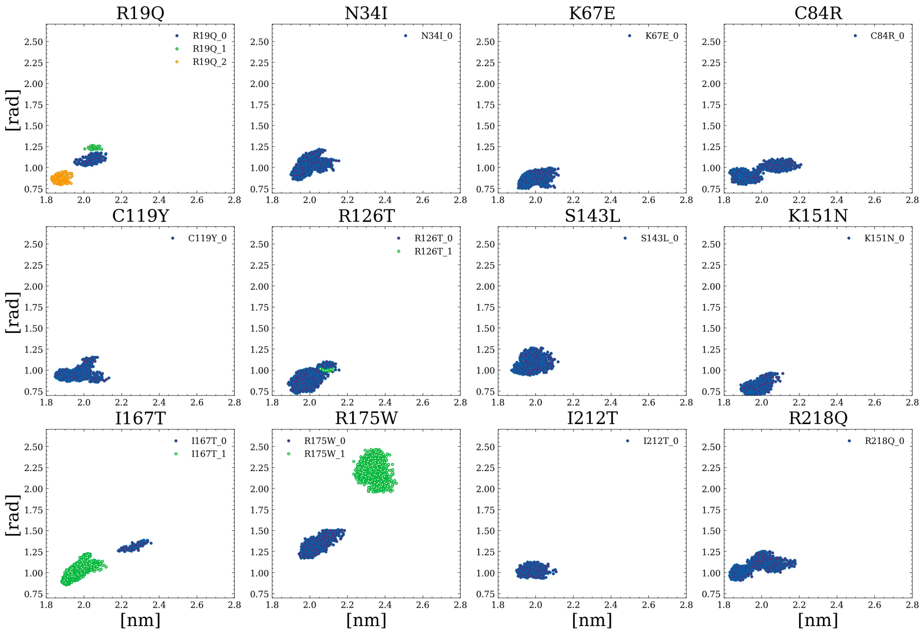


Figure 2.20: Closed. $\epsilon = 0.5$. $N_{\min} = 100$.

SASA Analysis

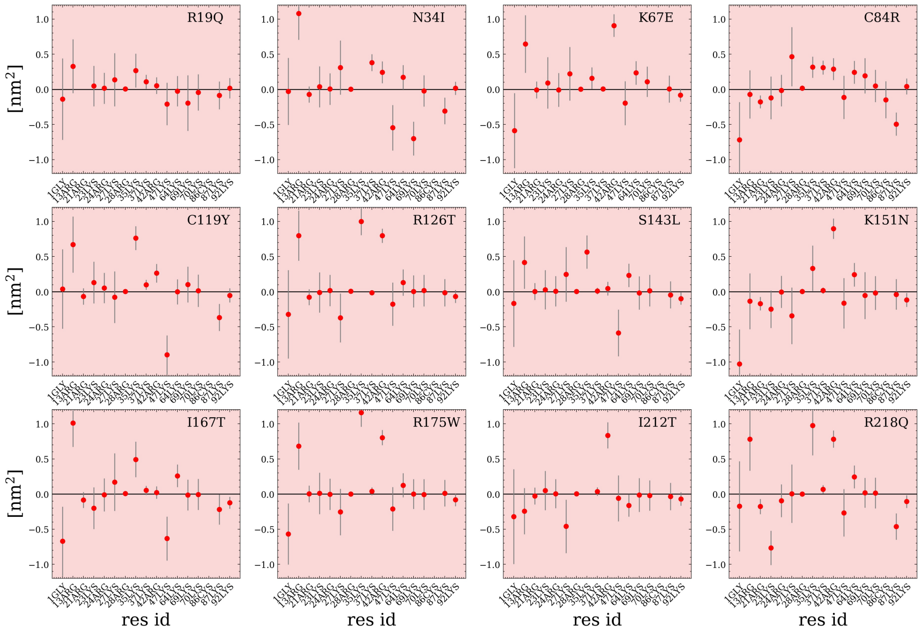


Figure 2.21: Open, positively charged.

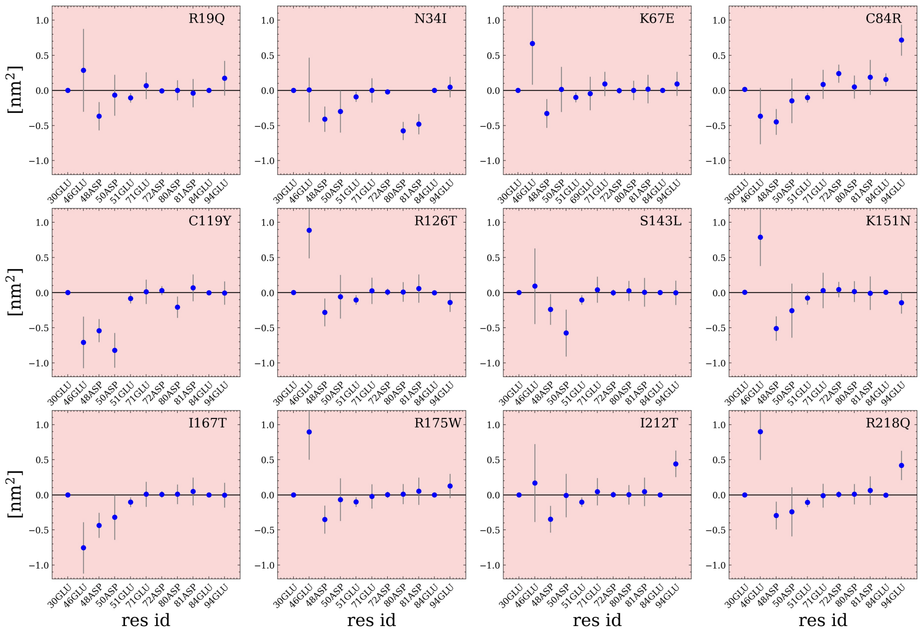


Figure 2.22: Open, negatively charged.

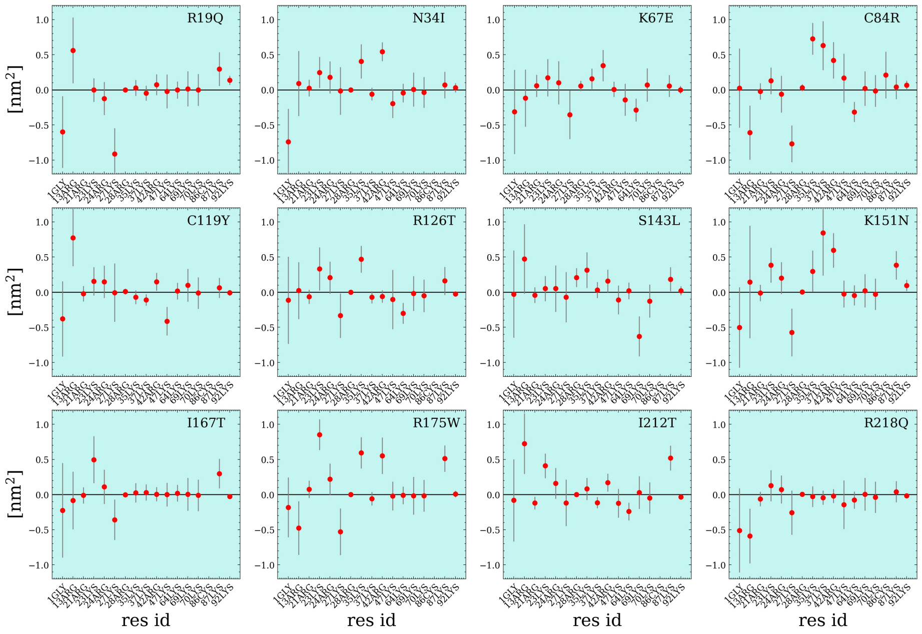


Figure 2.23: Closed, positively charged.

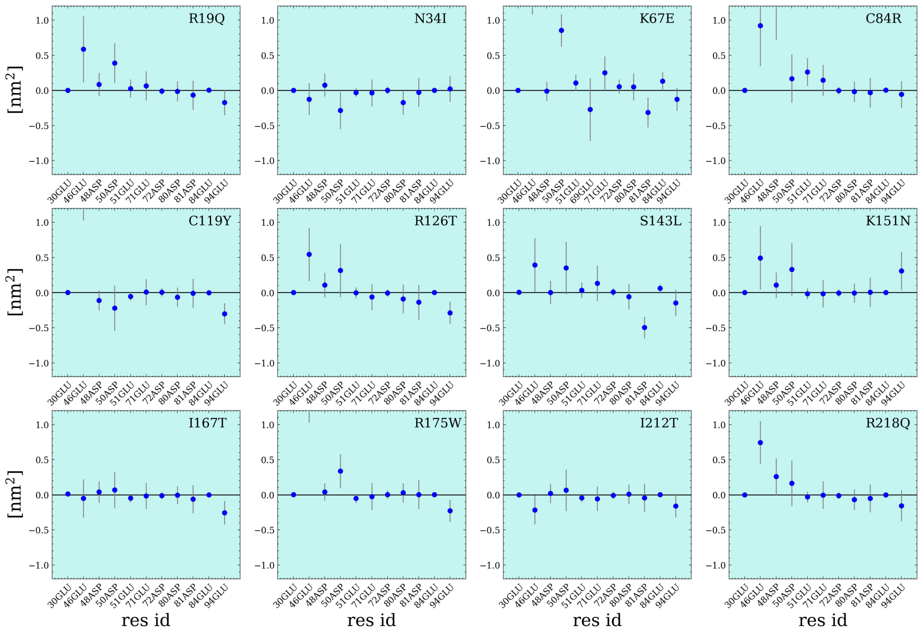


Figure 2.24: Closed, negatively charged.

Dendograms based on JS and d_Ω

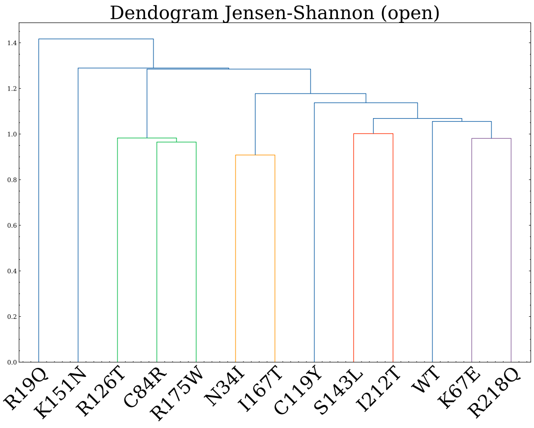


Figure 2.25

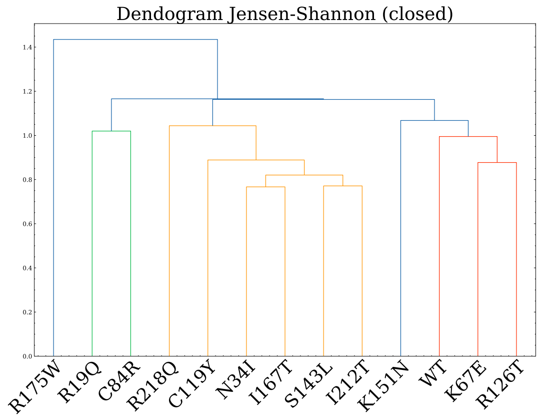


Figure 2.26

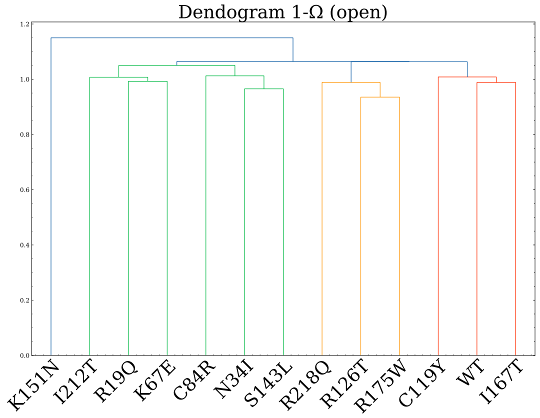


Figure 2.27

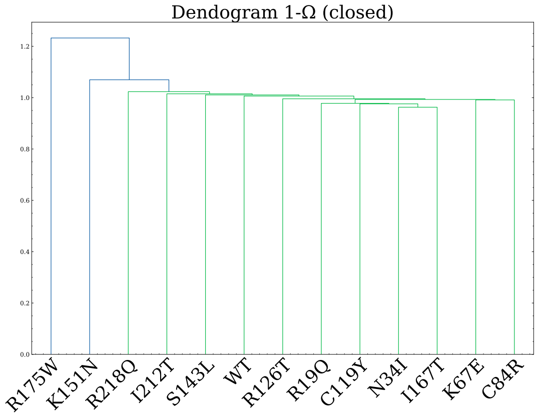


Figure 2.28

Observables used in the Analysis

- **Root Mean Squared Deviation:**

  given two atomistic configurations (*e.g.* two different frames of the same trajectory at times t, t') of a molecular system M, the RMSD is defined as a normalized euclidean distance between the sets $\{\mathbf{r}\}_M$ of every atom of the structures:

$$\mathrm{RMSD}\left(\{\mathbf{r}(t)\}_M, \{\mathbf{r}(t')\}_M\right) := \min\left\{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{r}_i(t)-\mathbf{r}_i(t')\|^2}\right\} \tag{2.2}$$

  where min indicates a minimization of the quantity itself with respect to the rigid rotations and translations onto one of the two structures.

- **Root Mean Squared Fluctuations:**

  given a residue r (usually traced by the position of its C_α atom), the RMSF is defined as the standard deviation of its position with respect to te average position, along a trajectory with T frames:

$$\mathrm{RMSF}[r] = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{r}_r(t)-\mathbf{r}_{r,\mathrm{av}}\|^2} \tag{2.3}$$

$$\mathbf{r}_{i,\mathrm{av}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{r}_r(t) \tag{2.4}$$

- **Free Energy:**

  given couples of values of observables, *e.g.* the two dihedral angles (Φ, Ψ) of an amino acid, one can calculate a free energy surface that essentially quantifies the relative stability (in energetic terms) of the states explored by the system, as follow:

$$F[\Phi, \Psi] = -k_b T \log\left[\rho(\Phi, \Psi)\right] + F_0 \tag{2.5}$$

where k_b is the Boltzmann constant, T is the temperature of the thermostat used to sample the values of the collective variables (CV) (Φ, Ψ), ρ is the histogram created from the values of the CV and F_0 is an arbitrary constant used to set the zero of the free energy. In this work, we used the Python package PyEMMA [44] to perform the calculations of F.

- **Jensen-Shannon Divergence:** the JS divergence is an estimator of the similarity between two probability distributions, and it is well defined also for discrete cases (histograms). Given two histograms $p = [p_1, \ldots, p_n]$ and $q = [q_1, \ldots, q_n]$ it is defined as follow:

$$JS[p, q] := \frac{1}{2}\left[\sum_{i=1}^{n} p_i \log_2\left(\frac{p_i}{m_i}\right) + \sum_{i=1}^{n} q_i \log_2\left(\frac{q_i}{m_i}\right)\right] \tag{2.6}$$

$$m_i := \frac{1}{2}(p_i + q_i) \tag{2.7}$$

- **Similarity in Free Energy (based on Jensen-Shannon divergence):**

$$JS[p, q] := \frac{1}{2}\left[\sum_{i} p_i \log\left(\frac{p_i}{m_i}\right) + \sum_{i} q_i \log\left(\frac{q_i}{m_i}\right)\right] \tag{2.8}$$

$$m_i := \frac{1}{2}(p_i + q_i) \tag{2.9}$$

$$\eta[p, q] := 1 - JS[p, q] \tag{2.10}$$

- **Euclidean distance in the V-space**, with $V := \left(\tilde{R}_g, \tilde{\theta}, \tilde{\chi}\right)$:

$$d\left(V_i, V_j\right) := \sqrt{(\tilde{R}_g^{(i)} - \tilde{R}_g^{(j)})^2 + (\tilde{\theta}^{(i)} - \tilde{\theta}^{(j)})^2 + (\tilde{\chi}^{(i)} - \tilde{\chi}^{(j)})^2} \tag{2.11}$$

where $\tilde{R}_g$ is the radius of gyration expressed in nanometers, $\tilde{\theta}$ and $\tilde{\chi}$ are the angles formed by the centers of mass of the 3 domains and domain I-III with the 2 hinges, expressed in radiants. The choice of these units is dictated by the fact that we wanted to have comparable absolute values among the components of this distance, and in this particular system this requirement is satisfied by this choice.

- **Solvent Accessible Surface Area (SASA):**
The SASA is defined as the surface area of a molecule that is accessible to a solvent probe, typically represented by a spherical probe with a defined radius (usually 1.4 Å, corresponding to the radius of a water molecule). The probe is rolled over the molecular surface, and the accessible area it covers is measured. Several computational algorithms are available to calculate the SASA of a molecule. One widely used method is the Shrake-Rupley algorithm (used also here), which approximates the molecular surface by dividing it into smaller triangles and then calculating the exposed surface area of each triangle. The SASA calculation involves the following steps, starting from the given 3D molecular structure:

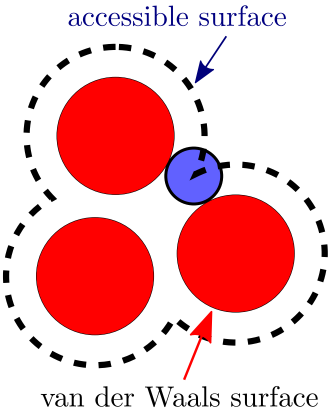


Figure 2.29: Taken from Wikipedia.

1. Generating a set of points on the molecular surface to represent the atoms.
2. Placing the solvent probe, usually a spherical probe, at each point on the molecular surface and determining whether it overlaps with any other atoms. This process accounts for the effective solvent accessibility of the surface.
3. Summing up the areas covered by the probe at each point to obtain the total SASA value.

- **SASA-based binding affinity estimator** based on domain I charged residues, relative to the simulation of the mutation μ with initial configuration γ ($\alpha^{(I)}_{\mu,\gamma}$):

$$\alpha^{(I)}_{\mu,\gamma} := \sum_{r=1}^{N_I} q_r \left\langle SASA^{(\mu,\gamma)} \right\rangle \equiv \sum_{r=1}^{N_r} q_r \left(\frac{1}{T} \sum_{t=1}^{T} SASA^{(\mu,\gamma)}(t) \right) \tag{2.12}$$

- **Principal Component Analysis:** consider the 3N degrees of freedom of the solute molecule, made by the cartesian coordinates of each atom. Calling $\{\mathbf{r}(t)\}_M$ the row vector made by those coordinates at time t, with $t = 1, \ldots, T$, one can build the $3N \times 3N$ *covariance matrix* $\mathbf{C}[\{\mathbf{r}(t)\}_t]$, each of whose elements $C_{\mu\nu}$ is defined as:

$$C_{\mu\nu} := \frac{1}{T} \sum_{t=1}^{T} (r_\mu(t) - r_{\mu,\text{av}})(r_\nu(t) - r_{\nu,\text{av}}) \tag{2.13}$$

with $\mu, \nu = 1, \ldots, 3N$. By diagonalizing $\mathbf{C}$ one gets the $3N \times 3N$ diagonal matrix $\mathbf{\Lambda} = \mathbf{P}^T\mathbf{C}\mathbf{P}$ containing the eigenvalues and the $3N \times 3N$ $\mathbf{P}$ matrix with the row eigenvectors.

The power of this analysis lies in the fact that the eigenvectors relative to the highest eigenvalues can be used to project the space of 3N cartesian coordinates of the trajectory onto a low-dimensional space. This space can be used to represent the principal collective directions taken by the system, based on the sampling provided by the trajectory itself. Here we used the *PCA* class implemented in the Python package MDAnalysis [45].

- **Similarity in Dynamics** (Ω from PCA):
  given two trajectories T_A and T_B (of two systems with the same number of degrees of freedom $3N$), and given their PCs $\mathbf{u}_k^A$,$\mathbf{u}_k^B$ and eigenvalues λ_k^A,λ_k^B, the similarity quantity Ω is defined as follow:

$$\Omega(T_A, T_B) := 1 - \left[\frac{\sum_k (\lambda_k^A + \lambda_k^B) - 2 \sum_k \sqrt{\lambda_k^A \lambda_k^B} (\mathbf{u}_k^A \cdot \mathbf{u}_k^B)^2}{\sum_k (\lambda_k^A + \lambda_k^B)} \right]^{1/2} \tag{2.14}$$

Chapter 3

From the All-Atom to the Coarse-Grained Resolution

All-atom molecular dynamics (AA-MD) simulations have become invaluable tools for studying the behavior and properties of biomolecules at an atomic level. However, these simulations are not without limitations: in fact, understanding the inherent limits of AA-MD is crucial for interpreting results and designing appropriate numerical experiments. Firstly, the computational cost of all-atom MD is a significant challenge since simulating large biomolecular systems over long timescales requires substantial computational resources. The number of atoms and the complexity of the potential energy landscape result in extensive calculations, limiting the length and timescales that can be explored. While advances in hardware and algorithms have improved the efficiency [24, 46, 47], all-atom MD remains computationally demanding for very large systems. Another aspect comes from the fact that the limited timescales accessible by all-atom MD pose a constraint on the study of biologically relevant processes. Many important biomolecular events, such as protein folding, ligand binding, or conformational transitions, occur on timescales ranging from microseconds to even minutes and hours [48, 49, 50]. Capturing these processes within the temporal constraints of all-atom MD simulations is challenging and sometimes even impossible nowadays, potentially leading to incomplete representations or missing critical events. Additionally, all-atom MD simulations may suffer from sampling issues: in fact, the exploration of complex conformational spaces can be hindered by energy barriers, resulting in limited sampling of rare events or transitions. This sampling problem can lead to biased results or incomplete descriptions of the system's behavior. Despite these limitations, all-atom MD remains a valuable tool for studying many molecular phenomena.

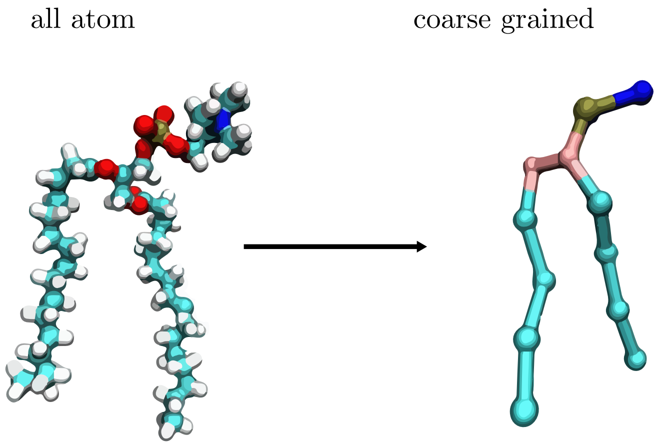


Figure 3.1: A generic pictorial representation of the process of coarse-graining in biomolecular modelling: the all-atom description is mapped into a simplified, coarse-grained one by reducing the number of explicit degrees of freedom included in the model and by introducing the proper interactions among them, in order to be able to predict the dynamics of the simplified form as good as possible.

However, to address these challenges, researchers have developed alternative methods such as enhanced sampling techniques, coarse-grained models, and multiscale modeling approaches. While the first strategy aims at accelerating the process of exploring the conformational space of a system, the second and the third strategies aim to overcome the limitations of all-atom MD by reducing the system's intricacy and/or incorporating information from higher-resolution techniques.

In this chapter, we discuss some of the scientific achievements done in the field of *coarse-graining* (CG) in molecular dynamics simulations, focusing on the applications to biomolecules since this is the direction taken by this thesis work.

3.1 A formal introduction to coarse-graining

The concept of CG has its foundations in the theory of *Renormalization Group* (RG) [51, 52], which consists in a series of methodologies that allow systematic investigation of the different scales that are intrinsically present in physical system. By also requiring that the energetic spectrum of a given system is characterized by well-identifiable, separable *scales*, this framework can be used in *e.g.* quantum field theories to obtain a description of the system by means of only the "relevant" degrees of freedom, where "relevant" has to be interpreted from an experimental point of view: if we want to use the theory to predict observables that will be compared to the outcome of an experiment whose apparatus is able to capture only a portion of the energetic spectrum of the system, then the predictions can be done by using the so-called renormalized theory. In the derivation of the classical equations of atomistic molecular dynamics in chapter 1 we used similar arguments (although classical MD is not a renormalized theory), by saying that we can assume a trivial quantum dynamics of the electrons and even neglect the explicit presence of the electrons themselves by using an effective description of the nuclei's dynamics, by means of the force field.

The simplification can be pushed even further. We can be interested *e.g.* in the study of the large-scale dynamics of macroscopic portions of a system, without requiring the knowledge of the microscopic dynamics; nevertheless, to do so we want to incorporate somehow the details that arise from the description of the theory at the microscopic level. In [53], for example, the authors show how it is possible to obtain a Langevin-like dynamics for the positions of the centers of mass (CoM) of group of atomistic particles: starting from the Hamiltonian dynamics of these particles, they use the *projection operator method* from Mori and Zwanzig [54], assuming that the time scales (or, equivalently, the energy scales) of the CoM are well separated from those of the microscopic constituents of the system. By calling $(\mathbf{R}_\alpha, \mathbf{P}_\alpha)$ the CoM coordinates of groups of atoms, they get (equation (74) of [53]):

$$\frac{d}{dt}\mathbf{P}_\alpha = \frac{1}{\beta}\frac{\partial}{\partial \mathbf{R}_\alpha}\log[\omega(\mathbf{R})] - \frac{\gamma}{M_\alpha}\mathbf{P}_\alpha + \delta\mathbf{F}^{\mathcal{Q}}_\alpha \tag{3.1}$$

where $\delta\mathbf{F}^{\mathcal{Q}}_\alpha$ is approximated with a random fluctuating force that satisfies the usual properties of white noise in Langevin dynamics. It is interesting to notice that the first term in the right-hand side of the equation is defined in terms of the normalized portion of the microscopic configuration space that is compatible to a given configuration $\mathbf{R}$ of the CoM coordinates, in the canonical ensemble picture:

$$\omega(\mathbf{R}) := \frac{\displaystyle\int d\hat{\mathbf{r}} \left(\prod_\alpha \delta[\hat{\mathbf{R}}_\alpha - \mathbf{R}_\alpha]\right) e^{-\beta U}}{\displaystyle\int d\hat{\mathbf{r}}\, e^{-\beta U}} \tag{3.2}$$

In other terms, this quantity is nothing but the probability to observe a given CG configuration $\mathbf{R} := \{\mathbf{R}_\alpha\}$, by summing up over all the Boltzmann weighted configurations of the microstates

$\hat{\mathbf{r}}$ compatible with $\mathbf{R}$. Moreover, U is the potential energy in the Hamiltonian $H = K + U$ of the microscopic description. In the jargon of the field of CG, the term:

$$W(\mathbf{R}) := -\frac{1}{\beta}\log[\omega(\mathbf{R})] \tag{3.3}$$

is called the *many body potential of mean force* (MB-PMF or simply PMF) and can be shown [55] to lead to the mean (in thermodynamical sense) force acting on the CoM coordinates, due to the microscopic degrees of freedom. This quantity will be important for the arguments treated in the next section.

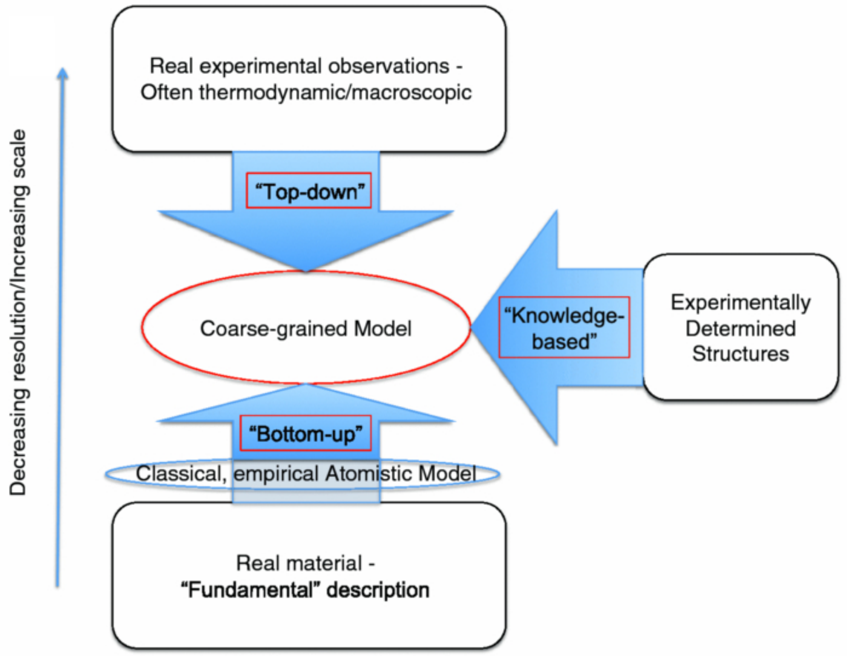

Real experimental observations -
Often thermodynamic/macroscopic
"Top-down"
Coarse-grained Model
"Knowledge-based"
Experimentally Determined Structures
"Bottom-up"
Classical, empirical Atomistic Model
Real material -
"Fundamental" description
Decreasing resolution/Increasing scale

Figure 3.2: Diagram showing 3 different approaches that are followed in the field of coarse-graining. Taken from [56].

This introduction is aimed at qualitatively justifying the process of coarse graining in the field of classical molecular dynamics simulations: given a Hamiltonian of the atomistic system and assuming the separability of energetic scales between atomistic and coarse-grained degrees of freedom, one is formally allowed to (try to) construct a model based on these CG degrees of freedom. Then, depending on the interest of studying either the equilibrium properties or the kinetic one, one can examine this lower-resolution model through *e.g.* Monte-Carlo techniques, the Langevin equation (and its generalized version [57, 58, 59]) or other equivalent pictures that are compatible with the thermodynamical properties to be reproduced (*e.g.* constant temperature T, fixed volume V, *et cetera*). The overview given here is very generic and in fact one may still ask some questions: is it always reasonable and possible to build a CG model based on the CoM coordinates? How to do the best choice of CG degrees of freedom (called *CG sites*)? How to construct the PMF or, if not possible, how to best approximate it? Although some of these questions remain open, in the last decades researchers put a lot of effort to answer them [60, 61, 62]. It is worth to notice that recently Giulini *et al.* [63] proposed an unsupervised way to find the best choices of CG sites, based on a finite sampling of the *fine-grained* representation of the system. One important prescription of this selection method, which is based on a quantity called *mapping entropy*, requires the *mapping* from fine-grained to coarse-grained representation to be a *decimation* one. A *mapping* is the mathematical relation that links the fine-grained and the coarse-grained degrees of freedom. Explicitly, its action can be represented as a linear operator $\mathbf{M}$ (a matrix) such that $\mathbf{M}[\mathbf{r}] = \mathbf{R}$, where $\mathbf{r} = [\mathbf{r}_1, \ldots, \mathbf{r}_n] \in \mathbb{R}^{3n}$ are the atomistic coordinates and $\mathbf{R} = [\mathbf{R}_1, \ldots, \mathbf{R}_N] \in \mathbb{R}^{3N}$ are the CG coordinates, with $N < n$. A decimation mapping is a linear function (*i.e.* one with $\mathbf{M}$ that is a matrix in $\mathbb{R}^{3N} \times \mathbb{R}^{3n}$) that consists in selecting some of the fine-grained degrees of freedom to "survive" as they are in the CG representation, and discarding the others. Another example of mapping is the already mentioned CoM mapping, which is another example of linear mappings.

Historically, it was custom to categorize coarse-graining approaches into two main groups: bottom-up and top-down methods (see 3.2), although nowadays it is also customary to mix the two approaches. Bottom-up methods [55, 64, 65] involve constructing a simplified representation based on a higher-resolution "reference" model using systematic, analytical rules based on the theory of coarse-graining. On the other hand, top-down methods [66, 67, 68] propose empirical models guided by macroscopic observables, without necessarily requiring a microscopic foundation. However, these models can be refined by incorporating higher-level knowledge, such as known structures or thermodynamic properties obtained from experiments. For the sake of simplicity, we chose to include the *knowledge-based* type of CG models into the top-down models (the interested reader is referred to *e.g.* [69] for a review of the CG'ing methods). In the next sections we discuss these two classes in more details and we will also treat particular cases of CG models: the so-called *implicit solvent models*, where the degrees of freedom of the solvent surrounding the solute molecule are removed (or "integrated out") and substituted by special terms in the PMF.

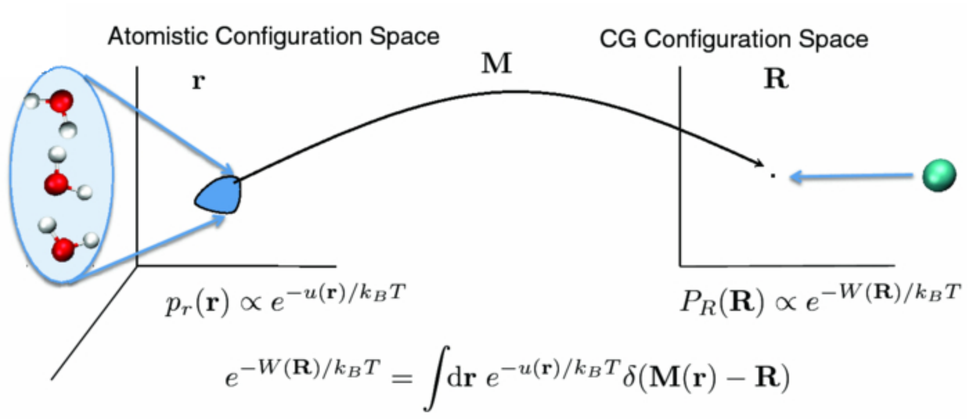

Atomistic Configuration Space
CG Configuration Space
M
r
R
$p_r(\mathbf{r}) \propto e^{-u(\mathbf{r})/k_BT}$
$P_R(\mathbf{R}) \propto e^{-W(\mathbf{R})/k_BT}$
$e^{-W(\mathbf{R})/k_BT} = \int d\mathbf{r}\, e^{-u(\mathbf{r})/k_BT}\delta(\mathbf{M}(\mathbf{r}) - \mathbf{R})$

Figure 3.3: Schematic representation of the mapping process in bottom-up coarse-graining procedures. Taken from [56].

3.2 Bottom-up coarse-graining

The bottom-up coarse-graining [55, 64] procedures are all characterized by the underlying assumption that a fine-grained model exists, with its own degrees of freedom (dof)$(\mathbf{r}, \mathbf{p})$ and with its own Hamiltonian $h(\mathbf{r}, \mathbf{p}) = k(\mathbf{p}) + u(\mathbf{r})$. Given a mapping $\mathbf{M}[\mathbf{r}] = \mathbf{R}$, the bottom-up procedures are based on the *consistency criterion*, which links the equilibrium thermodynamics of the fine-grained dof to the CG ones. Calling $(\mathbf{R}, \mathbf{P})$ the CG coordinates and assuming that they can be described by an Hamiltonian of the form $H(\mathbf{R}, \mathbf{P}) = K(\mathbf{P}) + W(\mathbf{R})$, the criterion requires the sampling of the CG coordinates through the canonical probability density $P_R(\mathbf{R})$ to be equal to the sampling of the fine-grained coordinates through the canonical probability density $p_r(\mathbf{r})$ *projected* into the low-dimensional space of the CG coordinates, $p_R(\mathbf{r}) = p_R(\mathbf{R})$:

$$p_R(\mathbf{R}) \equiv P_R(\mathbf{R}) \quad \text{with} \quad p_R(\mathbf{R}) := \int d\mathbf{r}\, e^{-u(\mathbf{r})/k_BT}\, \delta(\mathbf{M}[\mathbf{r}] - \mathbf{R}) \tag{3.4}$$

From this criterion and assuming a canonical-like form for $P_R(\mathbf{R})$ (see also figure 3.3), one can derive the *ideal* choice of the effective potential $W(\mathbf{R})$, which is the already mentioned *multi-body potential of mean force* (MB-PMF):

$$W(\mathbf{R}) = -\frac{1}{\beta}\log\left[p_R(\mathbf{R})\right] \equiv -\frac{1}{\beta}\log\left[\int d\mathbf{r}\, e^{-u(\mathbf{r})/k_BT}\, \delta(\mathbf{M}[\mathbf{r}] - \mathbf{R})\right] \tag{3.5}$$

From its definition, it is clear that the MB-PMF is not a potential energy but a free energy, due to the dependence on T and, in principle, also on V and n. As explained in [56], however, the exact knowledge of $W(\mathbf{R})$ is impossible for almost every realistic system since the integration of the degrees of freedom lead to the appearance of up to N-body terms in the PMF, which make its calculation intractable. As a consequence, one usually introduces an approximating potential term $U(\mathbf{R})$ that can be generally written as follow:

$$U(\mathbf{R}) := \sum_{\xi} \sum_{\lambda} U_{\xi}[\Psi_{\lambda}(\mathbf{R})] \tag{3.6}$$

where the U_{ξ} are general functions of the collective variables Ψ_{λ}, which of course are functions of the Cartesian coordinates for the CG dof. The U_{ξ} usually depend on some set of parameters that are optimized so to get as close as possible to $W(\mathbf{R})$.

In the next section we will briefly recap the most commonly used protocols in the context of bottom-up CG.

3.2.1 Force Matching

The *multi-scale coarse-graining* (MS-CG) method has been introduced by Izvekov and Voth [70], inspired by a method used to obtain atomistic force fields from *ab initio* calculations [71]. The basic idea is to construct the best approximation to the PMF by a variational protocol, aimed at optimizing (*i.e.* minimizing) a functional that is a sort of distance between the fine-grained forces and the (parameter-dependent) coarse-grained forces, acting on the CG sites. By including the explicit dependence on the parameters to be optimized, $\{\gamma_i\}$, one can introduce the CG force obtained by differentiating $U(\mathbf{R}|\gamma_i)$ with respect to $\mathbf{R}$, which we call $\mathbf{F}(\mathbf{R}|\gamma_i)$, and express the functional χ^2 as follow:

$$\chi^2[\mathbf{F}(\mathbf{R}|\gamma_i)] = \frac{1}{3N} \left\langle \sum_{I=1}^{N} |\mathbf{f}_I(\mathbf{r}) - \mathbf{F}_I(\mathbf{M}(\mathbf{r})|\gamma_i))|^2 \right\rangle_{\mathbf{r}} \tag{3.7}$$

where $\mathbf{f}_I(\mathbf{r})$ is a weighted sum of the forces acting on the atoms mapped in the same CG site I and the average $\langle . \rangle_{\mathbf{r}}$ is in principle an ensemble average, but is in practice calculated as an average over an atomistic trajectory that sampled fine-grained configurations distributed according to the microscopic probability. It can be proven that the global minimum of χ^2 is the force obtained from the MB-PMF, called $\mathbf{F}^0$, and so minimizing χ^2 with respect to γ_i is equivalent to look for the best choice of the parameters *at fixed mapping and choice of the basis function for* $U(\mathbf{R}|\gamma_i)$. This approach has found countless applications in the field of soft matter: see *e.g.* [56] for a summary of the most relevant.

3.2.2 Relative Entropy

Another bottom-up CG approach that rests on a variational principle is the one based on the *relative entropy* (RE) S_{rel}, introduced by Shell [64]. The RE is a *Kullback-Leibler divergence* [72], which quantifies the "distance" of two probability distribution functions (despite not being symmetric, as required to a proper distance in mathematical terms). In the context of the RE, the two probability distributions are the already mentioned $p_R(\mathbf{R})$ and $P_R(\mathbf{R}|U)$, which is the probability density function of the CG configuration space, sampled by the putative CG potential $U(\mathbf{R}|\gamma_i)$:

$$S_{rel}[\{\gamma_i\}] := k_B \int d\mathbf{R}\, p_R(\mathbf{R}) \log\left(\frac{p_R(\mathbf{R})}{P_R(\mathbf{R}|U)}\right) \tag{3.8}$$

One can easily show that $S_{rel} \geq 0$ and $S_{rel}[W] = 0$, so the best choice is the MB-PMF as expected. In the case, as before, of canonical ensemble distributions for both the fine-grained and the CG systems in terms of their potentials, minimizing S_{rel} with respect to the parameters γ_i can be shown to be equivalent to the following condition:

$$\left\langle \frac{\partial U}{\partial \gamma_i} \right\rangle_{AA} = \left\langle \frac{\partial U}{\partial \gamma_i} \right\rangle_{CG} \quad \forall i \tag{3.9}$$

which represents the equivalence between the averages of the derivatives of the potential U with respect to the parameters γ_i, as sampled by the microscopic and the CG theory.

In [73], Rudzinski and Noid showed that the MS-CG and the RE frameworks are similar but not identical, in general. In fact, defining $\Phi(\mathbf{R}|U) := \log\left(\frac{p_R(\mathbf{R})}{P_R(\mathbf{R}|U)}\right)$, they show that, while the RE approach aims at minimizing the average of Φ, the MS-CG one aims at minimizing the average of $|\nabla\Phi|^2$: in the end, as said in [56], in the scenario of a complete basis set for U, the many-body potential of mean force (PMF) can be determined with both the MS-CG and relative entropy variational principles. These principles yield identical approximate potentials (but for an additive constant), assuming the CG potential follows a quadratic form in Cartesian coordinates [56].

Correlation functions methods

The last class of bottom-up CG approaches were introduced by a seminal work by Tscho and coworkers [74] and is based on a simple yet effective idea: given the atomistic 2-body, 3-body, 4-body and so on correlation functions, for the CG variables (*e.g.* R, Θ, X that are euclidean distances, angles between 3 CG sites and dihedral angles made by 4 consecutive sites), one can perform a so-called *direct Boltzmann inversion* (DBI) [56, 74] to obtain an approximated form of the 2-body, 3-body, 4-body *et cetera* terms of the MB-PMF:

$$U_2(R) \equiv -\frac{1}{\beta}\log\left(\frac{g_2(R)}{J_2(R)}\right) \quad U_3(\Theta) \equiv -\frac{1}{\beta}\log\left(\frac{g_3(\Theta)}{J_3(\Theta)}\right) \quad U_4(X) \equiv -\frac{1}{\beta}\log\left(\frac{g_4(X)}{J_4(X)}\right) \quad \dots \tag{3.10}$$

with J_2, J_3, J_4 proper Jacobians. Subsequent works [75] introduced an iterative approach, called *iterative Boltzmann inversion* (IBI), to refine the potentials obtained with the DBI. Another successful iterative approach is the so-called *Inverse Montecarlo* (IMC) method [76], which is based on calculations inspired by the theory of RG.

3.3 Top-down coarse-graining

In top-down models, interactions are commonly parameterized in absence of a more detailed, fine-grained model. These interactions are not intended to precisely approximate the many-body potential of mean force for a specific system. Instead, they are often quantitatively determined starting from physicochemical intuition, generic physical principles, or the need to reproduce observed emergent structural or thermodynamic properties at larger scales. Although not standard, we include here also the so-called *Structure-based* (SB-CG) or *Native state-based* CG models, of whom we will describe the *Gō-Models* and the *elastic network models*, which are two historically successful classes of SB-CG models in the field of protein folding and protein dynamics.

3.3.1 Gō models

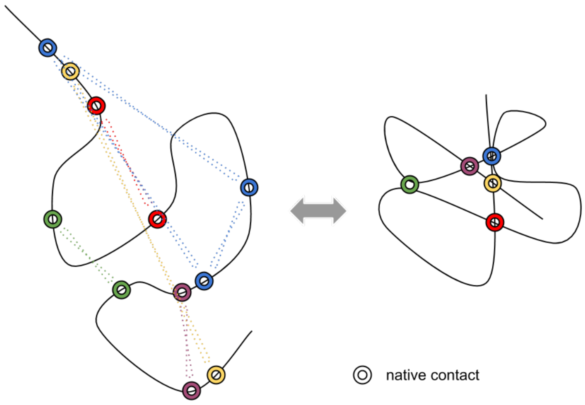


Figure 3.4: Pictorial representation of the base concept of the Gō models: given the native contacts among the residues (as indicated, for example, by a structure from the PDB), specific attractive forces are imposed on the dynamics for example via a Lennard-Jones like potential (3.12). In this way, one can characterize the transition paths that lead the system from a starting conformation (*e.g.* a random coil) to the native conformation, which by construction is a minimum of the Hamiltonian. Image taken from https://www.blopig.com/blog/tag/go-model/.

Gō models [77] where introduced to simulate the process of protein folding, with simplified (*i.e.* coarse-grained) representations of the proteins. By transposing the idea in the continuum space (the first models where lattice-based), one can represent the potential energy of a very simple Gō-like protein as follow (see *e.g.* [78]):

$$U_{\text{Gō}}(\mathbf{R}; C_{IJ}) = \sum_{I<J} \frac{1}{2} K_b \left(R_{IJ} - d\right)^2 \cdot \delta_{J,I+1} + \tag{3.11}$$

$$+ \sum_{I<J} \sigma[C_{IJ}] \cdot \epsilon \cdot \left[\left(\frac{R_0}{R_{IJ}} \right)^{12} - \frac{(2R_0)^6}{(R_{IJ} - R_0)^6 + (2R_0)^6} \right] + \sum_{I<J} (1 - \sigma[C_{IJ}]) \cdot \epsilon \cdot \left(\frac{R_0}{R_{IJ}} \right)^{12} \tag{3.12}$$

The first term is an harmonic spring connecting consecutive beads along the chain (like in a very simple polymer model); the second term is a LJ-like term acting only on those residues that are in native contact (in fact, $\sigma[C_{IJ}] = 1$ if I, J are in contact and $\sigma[C_{IJ}] = 0$ otherwise) and it depends on the choice of a reference distance R_0; the third term is an excluded volume term acting on non-native residues. Here the mapping is a *one-bead-per-residue* mapping, dictated by chemical intuition, as often happens in the process of building a CG model. Due to their exceptional simplicity and outstanding computational efficiency, these models have gained immense popularity and success in studying the folding, fluctuations, and interactions of proteins, protein complexes, and, to a lesser degree, nucleic acids with known equilibrium structures [79, 80, 81]. On the other side, since the model does not include non-native interactions, the potential energy is minimally frustrated, and this fact makes simulations with these models easily subjected to kinetic traps.

3.3.2 Elastic network models

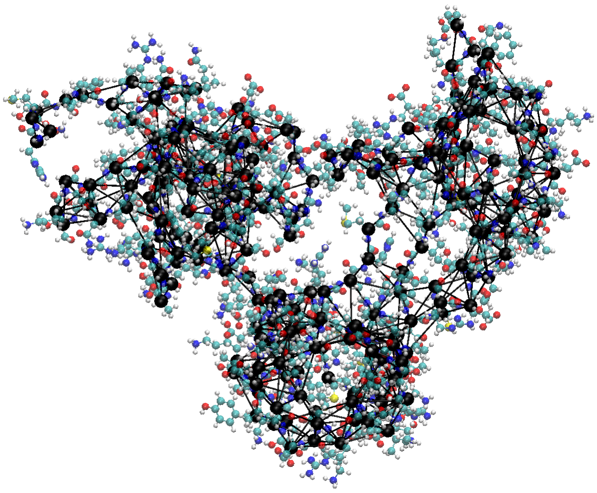


Figure 3.5: An example of elastic network model (black spheres and lines) built for the open conformation of the SBDS protein (all-atom structure in CPK style, coloured balls; see chapter 2), by employing a cut-off for the springs at $R_c = 6\text{Å}$.

The whole idea of the elastic network models (ENM) for biomolecules has its foundations in the theory of normal mode analysis (NMA). Starting from the Hamiltonian of a classical molecular system, for example the one shown in chapter 1, one can assume the existence of one or more minima of it. By diagonalizing the Hessian matrix of the potential, one can in principle construct a simplified model of the harmonic dynamics of the protein around the selected minimum of the potential by finding the normal modes of its vibrational dynamics. In a seminal paper [82], Tirion showed that, by assuming a very simple description of the potential energy of the system, it is possible to predict accurately enough the temperature factors and cumulative density of modes of simple proteins like the G-actin:ADP:Ca^{++} studied in the paper. It was then become common to use a free energy, acting on the C_α atoms alone or the CoM of the residues, with the following form (after a second-order expansion), $\mathbf{R}$ referring to the positions of the CG sites:

$$U_{ENM}(\mathbf{R};\mathbf{R}^0_{IJ}) = \sum_{(IJ)} \frac{1}{2} C \left[\left(\mathbf{R}_{IJ} - \mathbf{R}^0_{IJ} \right) \cdot \hat{\mathbf{R}}^0_{IJ} \right]^2 \tag{3.13}$$

It depends parametrically on the native distances $\mathbf{R}^0_{IJ}$, on the general spring constant C and on a cut-off R_c, by selecting only those pairs (IJ) with $R^0_{IJ} < R_c$. A straightforward generalization of this potential is the one with pair specific spring constants C_{IJ}. Another version of network models for biomolecules, closely related to the one based on U_{ENM}, is the so-called *Gaussian network model* (GNM) [83], with potential energy:

$$U_{GNM}(\mathbf{R};\mathbf{R}^0_{IJ}) = \sum_{(IJ)} \frac{1}{2} C_{IJ} \left(\mathbf{R}_{IJ} - \mathbf{R}^0_{IJ} \right)^2 \equiv \sum_{(IJ)} \frac{1}{2} C_{IJ} \left(\delta\mathbf{R}_J - \delta\mathbf{R}_I \right)^2 \tag{3.14}$$

where $\delta\mathbf{R}_I := \mathbf{R}_I - \mathbf{R}^0_I$. Assuming one single a-specific spring constant C, it can be shown that U_{GNM} can be rewritten as $U_{GNM}(\mathbf{R};\mathbf{R}^0_{IJ}) = \frac{1}{2} C \sum_{(IJ)} \delta\mathbf{R}_J \Gamma_{IJ} \delta\mathbf{R}_I$, where Γ_{IJ} is called *connectivity matrix*. An interesting property of the U_{GNM}-based models is the analytical integrability of its canonical configurational partition function:

$$Z^{(NVT)}_{GNM} = \int d\mathbf{R}\, e^{-\beta U_{GNM}(\mathbf{R};\mathbf{R}^0_{IJ})} = (2\pi)^{3N/2} \left| \frac{k_B T}{C} \Gamma^{-1} \right|^{3/2} \tag{3.15}$$

where Γ is the trace of Γ_{IJ}. Another useful property of GNMs is the direct relation between the diagonal elements $\mathbf{\Gamma}_{II}$ and the temperature factors of the residues I:

$$B_{I,GNM} \equiv \frac{8\pi^2 k_B T}{C} (\Gamma^{-1})_{II} \tag{3.16}$$

It is worth to mention that in the work presented in chapter 4 we made use of a peculiar version of a GNM, called $\beta-$GNM. Moreover, the AA/CG (All-Atom/Coarse-Grained) multiple resolution CANVAS model for proteins, which will be introduced in this chapter and used in chapter 6, makes use of an ENM to model the CG part.

3.3.3 MARTINI force field

The MARTINI coarse-grained model [66, 84] is a widely used approach for simulating biomolecules, from lipid membranes to proteins. It provides a compromise between computational efficiency and accuracy by grouping several atoms into a single "bead" representation. Its level of coarse-graining reduces the number of particles in the simulation, enabling longer timescales and larger systems to be studied compared to traditional atomistic models. In the MARTINI model, biomolecules are represented using four to five interaction sites per bead. Each bead accounts for several atoms, and its properties, including mass and charge, are determined by averaging the characteristics of the corresponding atoms. This approach simplifies the system while preserving key chemical and physical properties. The model employs a coarse-grained force field that describes the interactions between beads. The force field parameters are derived from *ab-initio*/atomistic simulations and experimental data, ensuring a good balance between accuracy and computational efficiency: this fact poses the model in between the bottom-up and top-down CG approaches.

MARTINI simulations have proven successful in studying a wide range of biomolecular processes, including membrane anchoring [85] and self-assembly of lipids [86]. Due to the reduced particle number, MARTINI simulations can cover longer timescales, allowing researchers to investigate slower biological phenomena. The model has also been applied to study complex biological systems, such as cell membranes and viruses.

3.3.4 oxDNA and oxRNA for nucleic acids

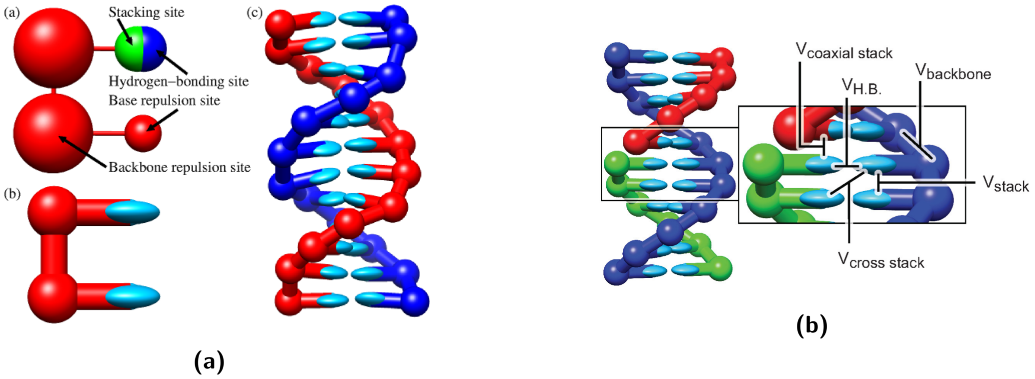


Figure 3.6: Mapping **(a)** and interactions **(b)** of the oxDNA and oxRNA CG models for nucleic acids. Taken from the oxDNA website.

The oxDNA [87] model and its twin oxRNA [88] model are two examples of top-down CG models for nucleic acids. They are built to match structural, mechanical, and thermodynamical properties of short and long single-stranded (ss) and double-stranded (ds) nucleic acids (NA).

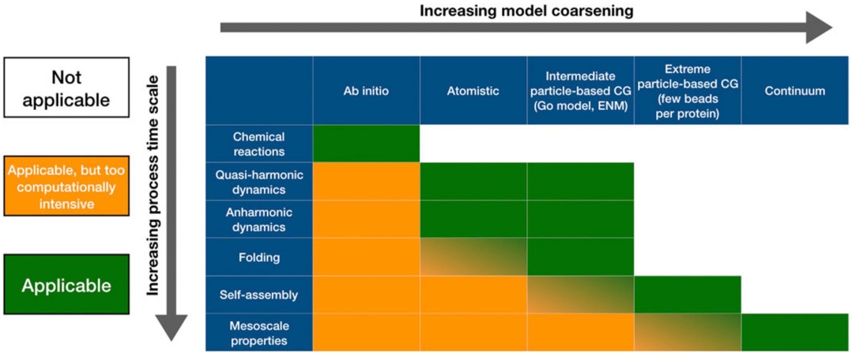
In particular, the oxDNA model aims at reproducing the dynamics of *e.g.* DNA origami, DNA-based nanostructures that have gained increasing interest in the last years for their potential technological applications [89]. On the other hand, the oxRNA model is built to match, among other properties, extension curves as a function of twist and stretching force, also including the behaviour of plectoneme structures [88].

The mapping is based on chemical intuition and consists in a two-beads per nucleotide mapping. The backbone atoms are all mapped into the same spherical bead, while the nucleobase atoms are mapped into another bead that is splitted into two, distinguishable interaction sites (both linearly dependent on the current position of the nucleobase bead); see figure 3.6 for a pictorial representation. The interaction terms are divided into two classes: the nearest neighbors interactions, which act only on pairs of those nucleotides that are consecutive in the same chain/molecule; and the non-bonded interactions, which involve all the other pairs. In summary, all the interactions in the oxDNA and oxRNA models are pairwise. The terms can be distinguished even further, as follow:

$$V^{oxDNA/oxRNA} = \sum_{ij \in \langle ij \rangle} (V_{back} + V_{stack} + V_{ex}) + \tag{3.17}$$

$$+ \sum_{ij \notin \langle ij \rangle} (V_{HB} + V_{DH} + V_{cr.-stack} + V_{co.-stack} + V_{ex}) \tag{3.18}$$

where the first sum involves the nearest-neighbors interactions, and the second all other pairs. V_{back} describes the bonds among the backbone CG sites; V_{stack} is a directional term that accounts for the stacking of consecutive nucleobases; V_{ex} is an excluded volume term; V_{HB} is also directional and aims at stabilizing hydrogen bonds between AU,CG (canonical) and UG (wobble) base pairs; V_{DH} is a Debye-Huckel potential (see last section of this chapter) that takes into account the electrostatics screened by an effective salt concentration (since the models are implicit solvent-based, see again last section of this chapter); $V_{cr.-stack}$ accounts for the cross-stacking effects that stabilize the structure of helices in dsNA; $V_{co.-stack}$ is the analogous of V_{stack} but for nucleotides that are not nearest neighbors. In the last chapter, we show how to make use of the very fast and efficient GPU-based implementation of the model into the native software (itself called oxDNA) to simulate the process of folding of a 2774 bases-long ssRNA viral fragment, in order to get an equilibrated structure to be use to construct a model for a full virion particle.



☐ Not applicable	🟧 Applicable, but too computationally intensive	🟩 Applicable

	Ab initio	Atomistic	Intermediate particle-based CG (Go model, ENM)	Extreme particle-based CG (few beads per protein)	Continuum
Chemical reactions	🟩				
Quasi-harmonic dynamics	🟧	🟩	🟩		
Anharmonic dynamics	🟧	🟩	🟩		
Folding	🟧	🟧🟩	🟩		
Self-assembly	🟧	🟧	🟧🟩	🟩	
Mesoscale properties	🟧	🟧	🟧	🟧🟩	🟩

Figure 3.7: A schematic illustration of the relation between a model's accuracy and its capacity of reproducing long time-scale phenomena. In principle, an extremely accurate model might reproduce all phenomena that take place at a characteristic length and time scale that lies above that of its fundamental constituents; however, practical limitations make its usage impossible beyond a certain limit. The coarser the model, the longer the time scale that can be achieved, at the expenses of a shorter and shorter list of processes that it can manage to produce. Image taken from [90].

3.4 Multi-resolution models

There is a promising class of methods that consists in mixing multiple levels of resolution in the same setup, called *multi-resolution models*. The resolutions can be mixed at various levels: Quantum with classical AA (QM/MM) [91, 92], all-atom and coarse-grained (MM/CG) [93] and also all-atom or coarse-grained with continuum models of *e.g.* the solvent, which is treated as a field variable [94]. These methods incorporate the intrinsic multi-scale nature of real systems typically found in the realm of Biology (which involves all the scales represented in 3.7) and for these reasons they can be the best theoretical framework to be adopted in biomolecular modelling and simulations. However, due to its young age, the field's maturity is still far from being comparable to that of the all-atom modelling: from my point of view, it is hard to predict whether it will spread over the scientific community to become a valid competitor of the AA models or if the ongoing explosion of increasing computational power accessible all over the world will make it obsolete, leaving the stage to the AA modelling as unique, out of reach protocol. Nevertheless, in the following we report some examples of these approaches and later on in chapter 6 we will show and discuss the application of a specific multi-resolution model for proteins, namely the the *CANVAS* model [93], which is briefly introduced in this section.

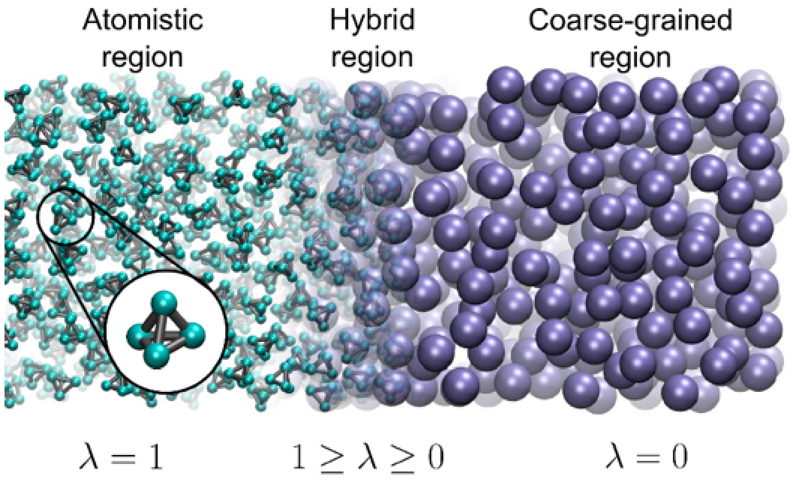


Figure 3.8: Schematic representation of the multi-resolution Hamiltonian-AdResS model: the parameter λ, which is a function of the points in the space of the simulation box, determines the degree of resolution of the particles located there. Image taken from the H-AdResS website.

3.4.1 Solvent models

The AdResS (adaptive resolution simulation) [95, 96] and H-AdResS (Hamiltonian-adaptive resolution simulation) [97] models are powerful multi-resolution (or more precisely *adaptive* resolution) approaches for simulating liquids. These models are based on a seamless transition between atomistic and coarse-grained representations, allowing for efficient simulations of liquids at different levels of detail. The AdResS model divides the simulation setup into three regions: atomistic, hybrid, and coarse-grained. The atomistic region contains molecules that require high-resolution representation, such as solute molecules or interfaces. The coarse-grained region represents bulk solvent or less critical regions. It has been shown [98] that the CG region can be treated with the very simplistic and computationally light description of an ideal, non-interacting gas, which makes that region essentially a *reservoir* of particles for the AA one. The hybrid region acts as an interface between the atomistic and coarse-grained regions, allowing for the exchange of particles and information. In mathematical language, the force acting between the CG site I and J is expressed as a sum of the AA (CoM force, assuming a mapping onto the CoM) and CG forces [96]:

$$\mathbf{F}_{IJ}^{(Ad)} = [\lambda(\mathbf{R}_I)\lambda(\mathbf{R}_J)] \cdot \mathbf{F}_{IJ}^{(AA)} + [1 - \lambda(\mathbf{R}_I)\lambda(\mathbf{R}_J)] \cdot \mathbf{F}_{IJ}^{(CG)} \tag{3.19}$$

where $\lambda(\mathbf{R}_I)$ is a smooth function that takes values in the range $[0, 1]$: a value of 0 is associated to the CG region, while 1 to the AA one. By dynamically adapting the resolution of each region based on the position of the particles, AdResS achieves a balance between accuracy and computational efficiency, also relying on the addition of compensation terms to keep the density/pressure constant in the AA region. The H-AdResS model extends the capabilities of AdResS by introducing Hamiltonian-based coupling between the atomistic and coarse-grained regions. The Hamiltoninan reads [97]:

$$H^{(H-Ad)} = \sum_{i,I} \frac{\mathbf{p}_{iI}^2}{2m_{iI}} + \sum_{I} \left\{ \lambda(\mathbf{R}_I) V_I^{(AA)} + \left[1 - \lambda(\mathbf{R}_I)\right] V_I^{(CG)} \right\} + V_{int} \qquad (3.20)$$

where $V_I^{(AA)}$ is the mean potential energy of the CoM-site I and $V_I^{(CG)}$ is the CG potential. This approach allows for the accurate representation of thermodynamic properties, such as energy and temperature, across the different regions. The H-AdResS model provides a more rigorous treatment of the interfaces and maintains the Hamiltonian formalism throughout the simulation, allowing for the use of the model in Monte Carlo simulations [99]. Both the AdResS and H-AdResS models have been successfully applied to study a wide range of liquids and molecules in solution, including aqueous solutions and organic solvents [100]. These models have been used to investigate various phenomena, such as solvation dynamics, phase transitions, and chemical reactions. The AdResS and H-AdResS models offer several advantages, including the ability to simulate large systems over long timescales while retaining accuracy in critical regions. By incorporating multi-resolution representations, these models overcome the limitations of purely atomistic or coarse-grained approaches. In principle, they provide a versatile framework for studying fluids with varying levels of detail, enabling researchers to gain insights into complex molecular processes that occur in realistic environments. However, unfortunately, there is still no unique and consolidated implementation of these models that makes it suitable to couple them with biomolecular simulations efficiently, although the methods have been already successfully applied to the study of *e.g.* proteins [101].

3.4.2 The CANVAS model for proteins

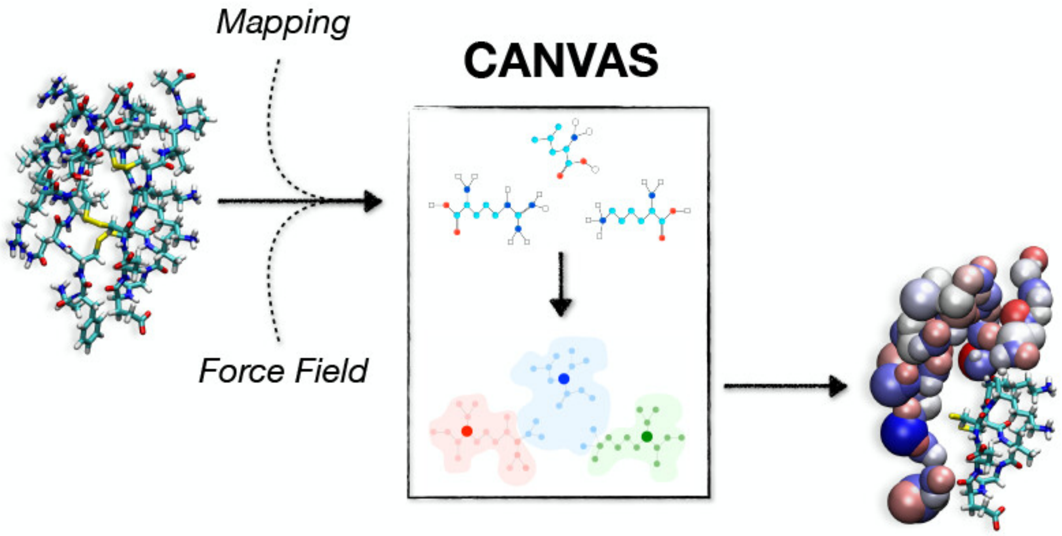


Figure 3.9: Graphical abstract of [93] that illustrates the logic behind the construction of a CANVAS model of a protein.

The CANVAS model is a multiple-resolution model for proteins developed in my current research group, and is extensively presented and described in [93]. The model allows one to describe the chemical components of a protein with 3 different levels of resolution: atomistic, medium grained and coarse-grained. The first one keeps every atom as explicit degrees of freedom; the second level keeps only the heavy atoms of the backbone of the amino acids as explicit beads; the third level, the lowest, treats an amino acid as a single degree of freedom, centered onto the α Carbon. While the interaction terms among the atomistic level are those found in typical atomistic force fields, the low levels make use of harmonic springs with variable stiffness in order to impose fluctuations about specific relative distances, taking them from the starting structure provided by the user. Moreover, the medium and lowest resolution beads are modelled so as to preserve chemical properties of their neighborhoods (partial charges and van der Waals parameters). It is important to notice that the choice of the resolution distribution among the structure can be decided *a priori* by the user, and we are currently working to a new method that couples the mapping entropy minimization approach [63] with the resolution-relevance theoretical framework to optimally select the resolution distribution in a totally *unsupervised* and *data driven* manner.

3.5 Implicit solvent models

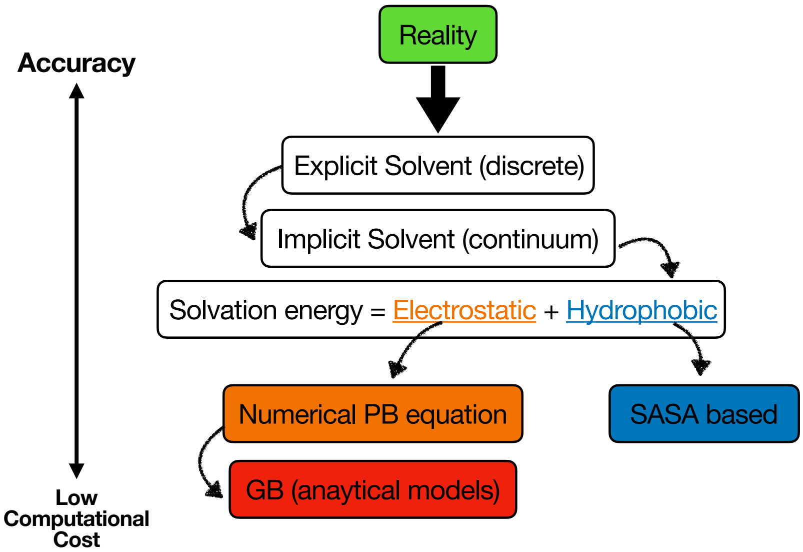

Accuracy
Reality
Explicit Solvent (discrete)
Implicit Solvent (continuum)
Solvation energy = Electrostatic + Hydrophobic
Numerical PB equation
SASA based
GB (anaytical models)
Low Computational Cost

Figure 3.10: Hierarchy of solvation models for biomolecules, inspired by [102]. The terms named Electrostatics and Hydrophobic are discussed in the text.

In molecular dynamics simulations of biomolecules it is already well-established that the solvent environment (typically water molecules and ions) plays a crucial role in the accurate prediction of the dynamics of the solute under study. However, modelling the solvent explicitly (*i.e.* as explicit degrees of freedom in the simulation) is computationally very demanding: on average, the number of atoms in a simulation box with PBC can be more than one order of magnitude higher than the number of atoms of the solute itself, in order to avoid self-interactions between the multiple copies of the solute. Alternatives to explicit solvent model are offered by multi-resolution models (already discussed) or the more common *implicit solvent* (IS) models. In these models, the free energy of solvation of the solute is estimated via approximations, the main goal being to reduce drastically the number of degrees of freedom integrated along the simulation. These approaches decrease the computational cost of energies and forces calculations for the solute and enhance the calculation of averages of observables by removing the need for averaging over the solvent degrees of freedom.

The free energy of solvation can be seen in a bottom-up CG fashion as a PMF: the process of integrating out the solvent degrees of freedom is nothing but a decimation mapping of the solvated system onto the unsolvated system, with the solute alone. In fact, the goal of every implicit solvent model is to construct an interaction potential that necessitates of information from solely the solute atoms, which is able to retain average (or *mean field*) information of the effect of the solvent “as if” it were explicitly taken into account. We can formalize the idea in mathematical terms. We can call $U_{tot}[\{\mathbf{r}\}_S, \{\mathbf{r}\}_M] = U_{SS}[\{\mathbf{r}\}_S] + U_{MS}[\{\mathbf{r}\}_S, \{\mathbf{r}\}_M] + U_{MM}[\{\mathbf{r}\}_M]$ the total potential energy of the system composed by the solute M and the solvent S. In this framework, the PMF of solvation $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]$ can be defined from the consistency criterion:

$$e^{-\beta\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]} \equiv \int d\{\mathbf{r}\}_S\, e^{-\beta(U_{SS}[\{\mathbf{r}\}_S] + U_{MS}[\{\mathbf{r}\}_S, \{\mathbf{r}\}_M])} \tag{3.21}$$

where $d\{\mathbf{r}\}_S := \prod_{j=1}^{\mathcal{N}} d\mathbf{r}_j$, if we call $\mathcal{N}$ is the number of atoms in the solvent. $U_{SS}[\{\mathbf{r}\}_S]$ and $U_{MS}[\{\mathbf{r}\}_S, \{\mathbf{r}\}_M]$ are the “real” potential energies of interaction of the solute with itself and the solvent with the solute, respectively (and depends on the choice of the force field and the model of the solvent used). Clearly, the knowledge of this PMF is essential if we want to mimic the thermodynamical properties of the protein system in the implicit solvent *exactly*. However, as one can imagine, a general solution to extract $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]$ from this criterion is essentially unfeasible, neither analytically nor numerically.

In the past decades there have been a lot of attempts to construct a quantity that is able to well approximate that PMF while being sufficiently general and computationally light [102, 103], in a more top-down fashion. The main group of this methods starts generally by one first approximations, i.e. to represent the free energy of solvation $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M] \approx \Delta G_{solv}[\{\mathbf{r}\}_M]$ as a sum of two terms (see also figure 3.10):

$$\Delta G_{solv}[\{\mathbf{r}\}_M] = \Delta G_{el}[\{\mathbf{r}\}_M] + \Delta G_{hydro}[\{\mathbf{r}\}_M] \tag{3.22}$$

Here, $\Delta G_{hydro}[\{\mathbf{r}\}_M]$ is the free energy of solvating a solute from which all charges have been removed (i.e. partial charges of every atom are set to zero, essentially the Lennard-Jones interactions), while $\Delta G_{el}[\{\mathbf{r}\}_M]$ is the free energy of first removing all partial charges in the vacuum, and then adding them back in the presence of a mean field solvent environment. The common approximation widely in use today [104, 103] for estimating the second addend of this sum assumes it proportional to the total solvent accessible surface area $\Delta G_{hydro}[\{\mathbf{r}\}_M] \simeq \sigma \cdot A[\{\mathbf{r}\}_M]$ (defined in chapter 2), with a proportionality constant σ fitted against experimental data on small peptides or molecular fragments. The second addend, on the other hand, is by nature long-ranged and requires more care. Among others, two approaches are recurrent in literature: the so called Poisson-Boltzmann (PB) method and the generalized Born (GB) method, which is a simplified version of the PB one. The first one is very general, physically speaking: it is in fact a direct application of electrostatic treatment of the solvent as a continuum dielectric in the linear response regime. The idea is to solve the PB equation:

$$\nabla\left(\epsilon(\mathbf{r})\nabla(\phi(\mathbf{r}))\right) = C\cdot\left[\rho_M(\mathbf{r}) + |e|\sum_j n_j z_j \exp(-\phi(\mathbf{r})|e|z_j/k_BT)\right] \tag{3.23}$$

with a given solute partial charge distribution (assumed to be static, step by step), at every time step of the simulation. Here the sum over j takes into account multiple ionic species in the solvent, n_j is the density, z_j is the ionic number and $|e|$ is the elementary charge. While being fully general, its computational cost makes it hard to use in most of the practical cases.

A simple ilnearization of the PB equation leads to the so-called *Debye-Huckel* (DH) potential for the screened electrostatic potential. By introducing the *ionic strength* $I := \frac{1}{2}\sum_j n_j z_j$, approximating $\epsilon(\mathbf{r}) \simeq \epsilon_r\epsilon_0$ (with typical values of $\epsilon_r \sim 80$ for water bulk), and defining $\kappa^2 := \frac{2I|e|}{\epsilon_r\epsilon_0 k_BT}$ one gets the following Helmholtz equation (which holds under the hypothesis of neutrality conditions):

$$\nabla^2\phi(\mathbf{r}) = C\cdot\kappa^2\phi(\mathbf{r}) \quad\Rightarrow\quad \phi(\mathbf{r}) = C'\frac{\exp(-\kappa r)}{r} \tag{3.24}$$

whose general solution is the Debye-Huckel potential written above. So, in turn, the DH potential between charge q_i and charge q_j of the solute takes the form $\phi(|\mathbf{r_i}-\mathbf{r_j}|) = C''\frac{q_iq_j\exp(-\kappa|\mathbf{r_i}-\mathbf{r_j}|)}{|\mathbf{r_i}-\mathbf{r_j}|}$. Other than being the implicit solvent model implemented in oxDNA and oxRNA, the DH potential will be also adopted by us (see chapter 6) as IS together with the CANVAS model, so as to explore the feasibility and the accuracy of this combination.

Alternatively, one can push the approximations even more, obtaining the GB potential. In the GB model, $\Delta G_{el}[\{\mathbf{r}\}_M]$ is calculated as follow:

$$\Delta G_{el}[\{\mathbf{r}\}_M] \simeq -\frac{1}{2}\left(1-\frac{1}{\epsilon_w}\right)\sum_{ij}\frac{q_iq_j}{\sqrt{r_{ij}^2 + R_iR_je^{\frac{-r_{ij}^2}{4R_iR_j}}}} \tag{3.25}$$

where q_i,q_j are the partial charges of the solute's atoms, ϵ_w is the relative dielectric constant of the bulk water; R_i and R_j are the so-called *Born radii*; an effective Born radius corresponds to a spherical ion having the same ΔG_{el} as the same solute with partial charges set to zero would have for all atoms except the atom of interest. A plethora of sub-methods starts from (3.25), each one of them consisting in different ways of calculating the Born radii. The plain speed up in computation for the GB method is clear: apart from calculating R_i, the free energy of solvation is analytically differentiable and leads to a quantity that can be efficiently implemented for the calculation of the forces in an MD engine. However, a number of limitations that we will not discuss here (see *e.g.* [103]) have been already pointed out on the accuracy of these methods. In chapter 5 we introduce a new method to approximate $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]$ by making use of a simple artificial neural network, discussing applications, pros and cons extensively.

Chapter 4

In Search of a Dynamical Vocabulary: A Pipeline to Construct a Basis of Shared Traits in Large-Scale Motions of Proteins

Note: The content of this chapter is entirely taken from the work: "In Search of a Dynamical Vocabulary: A Pipeline to Construct a Basis of Shared Traits in Large-Scale Motions of Proteins", *published in* Applied Sciences in 2022 [105]. *Accordingly, I here acknowledge Dr. Thomas Tarenzi, Dr. Marta Rigoli and Dr. Raffaello Potestio for their crucial contribution to this chapter.*

4.1 Introduction

Internal motions of proteins are intimately linked to protein function [106]. Such conformational movements span a wide range of spatial and temporal scales, going from local sidechain rotations and loop motions (ps to ns), to conformational transitions involving unfolding/refolding processes (ms to hours) [107]. In between these two extremes, internal large-scale protein fluctuations happening on timescales of the order of ns-μs [108] typically involve the collective movements of secondary structure elements; such fluctuations lead to a variety of potential conformational states, which might promote the exposure of specific binding sites [109, 110] or facilitate the induced fit of the protein upon interaction with partner molecules [111, 112]. It has been shown not only that this large-scale dynamics is essential for a protein to carry out its biological role [113], but also that a remarkable correlation exists between a protein's function and its specific dynamical signature [114], thus strengthening the view of dynamics as a link between a protein's structure and its specific function. This is particularly evident for the case of allosteric proteins, where the binding of a ligand conveys a signal that is propagated within the protein structure through a modulation of its internal dynamics, resulting in alternative conformational states and an altered protein function [115, 116, 117].

Several computational methods exist for the study of collective dynamics in proteins [118, 119, 120]; however, in order to develop a more general view of how dynamics bridges structure and function, it is necessary to build a dataset-wise approach for the comparison of such large-scale dynamics among proteins sharing different degrees of sequence and structural similarity. Attempts in this direction have been performed in several works [121, 122, 123, 124, 125, 126]. Maguid et al. [127] based their analysis on a dataset of pairs of homologous proteins; comparison of vibrational backbone dynamics within each pair led to the remarkable observation of correlation between dynamics and evolutionary conservation. Velázquez-Muriel et al. [128] performed a comparison between the protein flexibility shown by the structurally aligned members of a CATH superfamily [129] and the protein flexibility sampled by molecular dynamics simulation of a reference protein belonging to the same superfamily. Single value decomposition was used to capture the essential components of the two spaces, which show different size and complexity, and are therefore suggested to be combined for a thorough exploration of protein deformations. Analyses of the distance in dynamics have also been performed in the case of structurally and functionally diverse sets of proteins; in this regard, Hensen et al. [114] introduced the notion of "dynasome", namely an ensemble of observables computed from molecular dynamics (MD) simulations of a structurally heterogeneous protein dataset. The method highlights a striking correlation between the dynasome descriptors (which include 34 observables for each protein, ranging from the first five eigenvalues of the covariance matrix of C_α fluctuations to the average ruggedness of the energy landscape) and the proteins functional classification. However, this approach relies on time-consuming MD simulations, which limits its applicability to large protein datasets. In addition, the large number and sophistication of the descriptors employed does not enable a straightforward recognition and visualization of the similarities in dynamics between proteins in term of conformational movements.

To overcome these limitations, in this work, we set-up and validate a novel pipeline for the identification of a basis set of conformational motions in an enzymatic family, representing a common vocabulary of their large-scale dynamics. To this aim, we investigated internal, collective protein dynamics in terms of fluctuations at the level of single residues. Our approach does not require the acquisition of expensive MD simulations, since it is based on the topology of native contacts derived from a protein's experimental structure; specifically, we made use of the normal mode analysis (NMA) [130], which represents, together with the principal component analysis (PCA) [131], one of the main protocols employed to identify the most relevant patterns in the large-scale dynamics of proteins. While PCA requires a large set of configurations (for example from MD trajectories) to build the covariance matrix, NMA can be performed with the sole knowledge of an equilibrium configuration of the system. For this reason, NMA is often used in combination with simplified quadratic models, such as the linearized versions of elastic network models (ENMs) [82]. Another degree of simplification can also be introduced by building coarse-grained (CG) models of the protein, where the atomistic degrees of freedom are replaced by a smaller number of physically relevant representative beads. In spite of this simplicity, the collective, large-scale dynamical features obtained by NMA of ENMs of proteins showed to be successful to predict experimental B-factors [132] and also conformational changes [133, 134].

Given the nature of the ENM, the proposed pipeline is particularly suited for the study of collective dynamics in globular proteins; ENM might indeed show limitations for biomolecules whose dynamics is strongly anharmonic, as in the case of intrinsically disordered proteins. For this reason, the validation of the method is here performed on a set of globular enzymes, namely chymotrypsin-related proteases, for which in-depth analyses of evolutionary relationships and structural similarities are available in the literature [135, 136, 137, 138]; in addition, ENM-based NMA has been successfully applied on chymotrypsin-like proteases in previous works, both in Cartesian space [139, 140] and in torsion space [141]. In our approach, normal modes are computed from the β-Gaussian elastic network model of the dataset members [142]. In the β-Gaussian model, each residue is described in a simplified representation as two beads: one corresponds to the C_α atom and represents the mainchain, while the second, describing the sidechain, is positioned according to the degrees of freedom of the first bead. An effective quadratic potential energy is used to model the bead fluctuations from the native conformation. We made use of this information to perform a dynamics-based alignment between all pairs of proteins from the dataset; the results from the alignment were used to construct a distance matrix in the space of protein dynamics and to cluster together proteins with similar large-scale motions, thus adding an additional layer of information to clustering procedures based on sequence identity [143, 144] or structural similarity [145, 146, 147].

Moreover, we developed a way to represent each protein's large-scale normal mode as a vector field on the 3D space. Thanks to this representation, we were able to build a high-dimensional basis set of large-scale protein modes. The basis set is validated by comparison with results from MD simulations, with the perspective of applying this methodology to a dataset comprehensive of a large number of protein classes, differing in structure and function. In this way, common fluctuations between distant proteins can be correlated to the presence of local structural elements, with implications in protein engineering for the design of scaffolds that are able to perform controlled conformational changes in functional enzymes [148, 149]. In addition, the large-scale dynamics might serve as a guide in the identification of those patterns where the preservation of a high resolution is of paramount importance in the construction of simplified, multiscale models [150, 151, 152, 153, 90] that retain the original dynamics. In particular, by describing at an atomistic level the structural elements identified as important for the desired conformational movements and simultaneously coarse-graining the remainder of the protein, it might be possible to obtain a simplified and computationally inexpensive protein model that shows the conformational dynamics of the high-resolution one.

4.2 Overview of the workflow

In our approach, the identification of a common set of conformational motions among different proteins is based on the analysis of their dynamics in a CG representation; from here, a representative set of normal modes is identified through a dynamics-based clustering of the proteins comprising the initial dataset. The selected, representative modes are then orthonormalized and ordered, so as to obtain the final basis set. An overview of the workflow is given in Figure 4.1 and explained in detail in the following paragraphs.

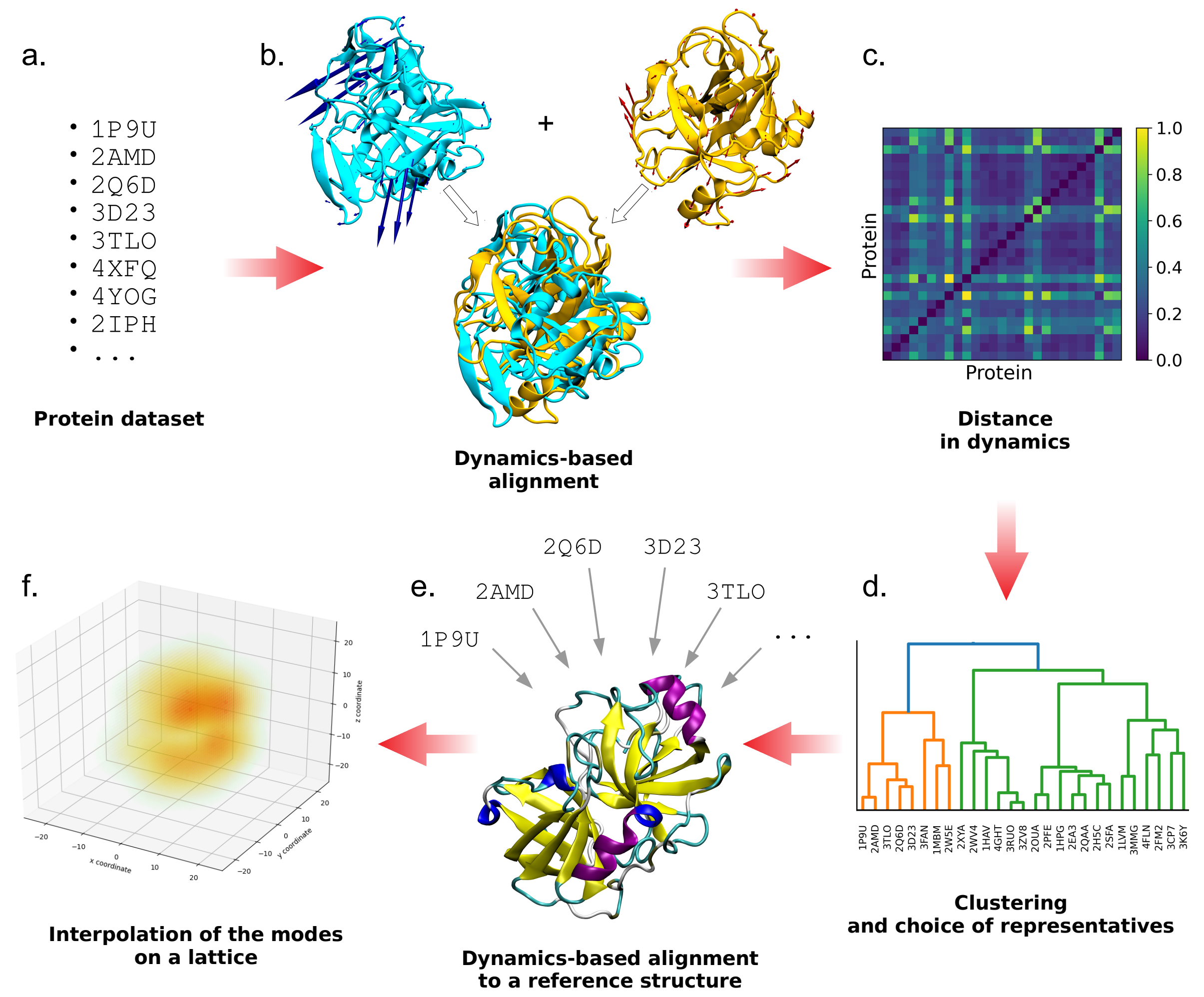


Figure 4.1: Schematic representation of the workflow proposed, from the choice of the protein dataset to the creation of the vector fields on a grid. Once orthonormalized and ordered, the latter are used to construct the final basis set.

The starting point is the identification of a set of proteins (Figure 4.1.a). The choice of this dataset is arbitrary and independent on the pipeline; however, the number of proteins that the dataset contains is supposed to be large enough so as to be representative of the families or superfamilies that are included, meaning that the more distant are the members in terms of homology, the larger should be the dataset. This is necessary to ensure sufficient generality of the resulting basis set of conformational motions.

The selected set of structures is used to run pairwise dynamics-based protein alignments with the ALADYN software developed by some of us [154] (Figure 4.1.b). ALADYN takes two input structures and performs the maximization of a score function that takes into account the spatial superposition of protein regions that have similar motion. The dynamical information is encoded in the low-energy (large-amplitude) eigenvectors obtained from the diagonalization of the interaction matrix M_{ij} of the Hamiltonian function of the β-Gaussian Network Model:

$$H = \frac{1}{2} \sum_{ij} \delta\vec{x}_i M_{ij} \delta\vec{x}_j \tag{4.1}$$

where $\delta\vec{x}_i$ is the displacement vector of the i-th bead with respect to the equilibrium configuration. Once the eigenvectors have been obtained, the extent and consistency of the alignment are quantified through the root-mean-square inner product (RMSIP) between the spaces given by the first 10 modes of each aligned protein. If we call N_i and N_j the total number of residues in the chains of the two aligned proteins, the RMSIP calculation is limited to a subset $q < N_i, N_j$ of marked C_α. These subset of amino acids are chosen by firstly grouping the amino acids in groups of 10 subsequent ones; then maximizing a single scoring parameter via the standard Metropolis criterion over the space of possible pairs of groups among the two proteins' sequences, as exhaustively explained in [154]. Specifically, the RMSIP is defined as:

$$\mathrm{RMSIP}(\{\vec{v}_l^k\}_i, \{\vec{w}_m^k\}_j) = \mathrm{RMSIP}_{ij} := \sqrt{\frac{1}{10} \sum_{l,m=1}^{10} \left| \sum_{k=1}^{q} \vec{v}_l^k \cdot \vec{w}_m^k \right|^2} \tag{4.2}$$

The RMSIP $\in [0, 1]$ takes on the value 1 in case of perfect correspondence of the spaces, and 0 in case of their complete orthogonality. The quantity (1.0 - RMSIP), which still takes values in the interval $[0, 1]$, is therefore suitable to define a distance in dynamics between two proteins after alignment. Statistical significance of the alignment, quantified by means of a z-score, is taken into account by weighting the RMSIP by the hyperbolic tangent of the module of the z-score, so as to give more importance to the most reliable results. The distance in dynamics between two aligned proteins i and j is therefore defined as:

$$d_{ij} = 1.0 - (\mathrm{RMSIP}_{ij} \cdot \tanh|z_{ij}|) \tag{4.3}$$

After all the pairwise alignments between the elements of the dataset are performed, a distance matrix that expresses differences in the large-scale dynamics is obtained (Figure 4.1.c); then the dataset undergoes hierarchical clustering [155] based on this distance matrix, in order to identify groups of dynamics-related proteins (Figure 4.1.d). The optimal number of clusters is identified from the interplay between *resolution* and *relevance* [156, 157, 158, 159, 160]. These two quantities are entropies that are related to each other and depend on the clusterization procedure adopted. We exploited them to select the number of clusters to retain, by considering the smallest number of clusters (hence the lowest resolution) that gives the highest relevance (Figure 4.2). Specifically, given a labeling $\hat{s} := (s_1, \ldots, s_\eta)$ (e.g. a clustering) to a sparse dataset made by $N \geq \eta$ data points (in our case the single proteins in the dataset), the resolution is defined as an entropy $\hat{H}_{res}$ representing the relative amount of information loss in the process:

$$\hat{H}_{res}[\hat{s}] := -\sum_{s} p_s \cdot \log_2(p_s) \quad p_s := \frac{k_s}{N} \tag{4.4}$$

where k_s is the number of data points that fall into the same cluster s. It is proven [157] that $\hat{H}_{res}$ increases monotonically with the number of clusters, in accordance with the idea that the coarser is our clustering, the more information we loose. On the other hand, the relevance $\hat{H}_{rel}$ is defined as:

$$\hat{H}_{rel}[\hat{k}] := -\sum_{k} \frac{k \cdot m_k}{N} \cdot \log_2\left(\frac{k \cdot m_k}{N}\right) \tag{4.5}$$

where m_k is the number of clusters containing the same amount $k = 0, \ldots, N$ of data points, for a given clustering process. By choosing the lowest resolution value corresponding to the largest relevance (Figure 4.2), we can rely on the most compact clusterization (thus increasing the statistics within each cluster) that preserves the highest empirical information content.

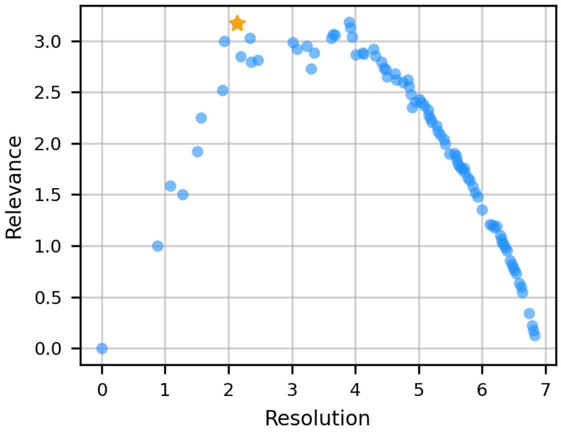


Figure 4.2: Resolution-relevance curve used to determine the optimal number of clusters in the dynamics-based clusterization of the protein dataset. Each point corresponds to a different number of clusters. The optimal subdivision, indicated with an orange star, corresponds to 9 clusters.

Once the optimal number of clusters is derived, protein representatives of each cluster are identified as the cluster centroids, namely the proteins with the shortest distance to every other protein of the cluster itself. In addition, a representative for the whole dataset is selected as the protein with the most characteristic dynamics, expressed in terms of the lowest distance with respect to all the other dataset members. The other protein structures are then dynamically aligned to this one with ALADYN, so as to have a consistent orientation in space (Figure 4.1.e).

From an ENM representation of each of these newly oriented structures, normal modes are computed. In order to facilitate the comparison between modes belonging to proteins with a different sequence length, the first 5 reoriented normal modes of the cluster representatives are placed on a cubic lattice, and interpolated on the grid points so as to obtain a smooth vector field (Figure 4.1.f). In this way, we move from comparing the $3N$-dimensional modes of different proteins (where N is the number of residues, different for each protein), to comparing vector fields defined on identical 3D lattices having the same dimension. More details on the lattice construction and interpolation are given in Section 4.3. Proteins belonging to the dataset employed in this work, despite displaying a range of sequence length and radius of gyration, do not grandly differ in size; therefore, the modes interpolated on the lattice can be directly compared. However, it might be the case that the dataset includes proteins with very different size; this would require a rescaling of the protein coordinates before the interpolation on the lattice, so as to compare motions occupying similar volumes in space.

The interpolated modes are orthonormalized using the Gram-Schmidt algorithm [161]. The components of the basis are finally ordered according to decreasing entropy, considered as a measure of their degree of collectivity. The entropy S of a mode k is defined as:

$$S_k = -\frac{\sum_i \phi_i^k \ln \phi_i^k}{\ln N}, \tag{4.6}$$

where N is the number of lattice sites and ϕ_i^k is the square modulus of the k-th mode on the lattice site i. S_k takes a maximum value of 1 if the mode is delocalized on all the lattice sites, and a minimum value of 0 if the mode is localized on a single site.

The final set of orthonormalized and ordered vector spaces represents the basis of protein dynamics. In the next section, technical details of the methods employed are presented.

4.3 Materials and methods

4.3.1 Preprocessing of the dataset

A dataset of 116 chymotrypsin-related proteases, for which structural experimental information is available, was selected. This dataset is based on the one used in ref. [138], from which proteins with sequence identity $>$ 70% were removed. The dataset comprises serine proteases from bacteria, eukaryotes, archaea, and viruses, in addition to chymotrypsin-related cysteine proteases from positive-strand RNA viruses. The full list of proteins' PDB IDs is given in Table S1. The structures were downloaded from the Protein Data Bank, and the coordinate files were cleaned-up from heteroatoms, from copies of the protein in the crystallographic cell, and from residue-configurations with low occupancy. The position of missing atoms was rebuilt and the protein conformations were optimized using the software FoldX [162]. Non-terminal missing residues were modelled with MODELLER [163, 164]. An analysis of the first 3 normal modes for each protein was run using an elastic network model with a cutoff of 10 Å, in order to identify the problematic cases in which the flexible protein termini impaired the analysis of the motion of the protein core. Such analysis was conducted by visual inspection of the modes on the protein structures. In those cases, flexible tails were not considered in the following analyses, which thus focused on globular structures. Moreover, in the case of multi-domain structures, only the domain known to have protease activity was retained.

4.3.2 Dynamics-based alignment and clustering

The dynamics-based alignment of all the pairs of protein structures was performed with the ALADYN software [154], using as input the cleaned coordinates files. From the resulting alignment scores, clustering of the structures was performed with the Python library SciPy, using the ward linkage method. The calculation of relevance and resolution, used to identify the optimal number of clusters, was performed with an in-house script.

4.3.3 Lattice interpolation and basis construction

Normal modes of each protein of the dataset have been computed with an in-house code. The first 5 reoriented normal modes of the cluster representatives were placed on a cubic lattice, with a lattice constant of 1 Å(for a total of 45 modes, namely vector fields). The vector on each protein C_α was translated on the nearest lattice grid point. The mode vectors were interpolated on the lattice in order to create a smooth vector field (Figure 4.1), using Gaussian functions with σ=0.8 Å and truncated at a distance of 2 Å. This distance is slightly smaller than the lowest spatial distance between two C_α atoms to make sure that the vector coming from the original protein mode is not spuriously modified during interpolation. The chosen value of σ ensures that in correspondence of the cutoff the mode field is close to zero. The resulting vector at each grid point ijk is the sum of the mode fields centered on the nearby C_α grid points, calculated at ijk, within the cutoff. Eventually, orthonormalization and ordering of the modes was performed with Python scripts.

4.3.4 Molecular dynamics simulations

Molecular dynamics simulations have been performed on the representatives of each cluster, using the software Gromacs 2019 [40]. The proteins were described with the Amber99sb-ildn force field [41], and the TIP3P model [42] was used for water molecules. Sodium and chloride ions were added at a concentration of 0.15 M, and balanced so as to neutralize the charge in the simulation box. All systems were energy minimized for 100 steps by steepest descent. The solvent was then equilibrated for 500 ps with positional restraints on the protein heavy atoms, using a force constant of 1000 $\text{kJ}\cdot\text{mol}^{-1}\cdot\text{nm}^{-2}$. MD simulations were carried out in the NPT ensemble for 250 ns for each system. Protein and solvent were coupled separately to a 300 K heat bath with a coupling constant of 0.1 ps, using the velocity-rescaling thermostat [165]. The systems were isotropically pressure-coupled at 1 bar with a coupling constant of 2.0 ps, using the Parrinello-Rahman barostat [43]. Application of the LINCS [166] algorithm on hydrogen-containing bonds allowed for an integration time step of 2 fs. Short-range electrostatic and Lennard–Jones interactions were calculated within a cut-off of 1.0 nm, and the neighbor list was updated every 10 steps. The particle mesh Ewald (PME) method was used for the long-range electrostatic interactions [167], with a grid spacing of 0.12 nm.

The calculation of the root-mean-square fluctuations from the trajectory coordinates was performed on the protein C_α atoms using the Gromacs tool *gmx rmsf*. The dynamic cross-correlation was computed with a Python script, using the library MDTraj [168]. Plots were produced with Python libraries, and protein images were rendered with VMD [169].

4.4 Results and discussion

4.4.1 Overview of the protein dataset

Proteases are enzymes catalyzing the reaction of hydrolysis of peptide bonds. The independent evolutionary origin of these enzymes [170] is reflected in their large variety of sizes, shape and specificity [171]. In this work we focus on a specific superfamily, namely the chymotrypsin-related proteases. The latter share a common structure with two β-barrel-like domains accommodating the binding site (Figure 4.3); however, the size and structural completeness of the β-barrels and the length of the turns and loops connecting the sheets greatly vary. The result of this structural variability is a range of sequence lengths and protein sizes among the 116 proteins included in our dataset (Figure S8). The proteolytic reaction is performed by a catalytic triad of residues, located between the β-barrels. The type of amino acid playing the role of nucleophile in the mechanism of catalysis determines the class of proteases: in the serine proteases, the catalytic triad contains His, Asp/Glu, and Ser residues [172]; in the cysteine proteases, the triad is composed of His, Asp/Glu, and Cys or of a dyad of His and Cys residues [173].

The classification used in the remainder of the paper is based on MEROPS, a hierarchical classification scheme for proteases [174, 175]. In the MEROPS database, chymotrypsin-related proteases constitute the PA clan, which contains 9 families of cysteine proteases (representing proteases of positive-strand RNA viruses) and 14 families of serine proteases (representing proteolytic enzymes from eukaryotes, bacteria, some DNA viruses and eukaryotic positive-strand RNA viruses). Families are defined on the basis of sequence similarity and/or resemblance of the folds among their protein members. However, experimental structural information is available for a limited number of these families; therefore, not all of them are represented in the dataset employed in this work.

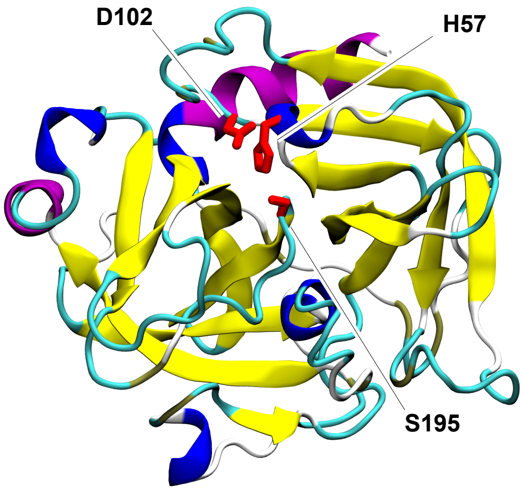


Figure 4.3: Cartoon representation of chymotrypsin from *Bos taurus* (PDB ID: 2CGA). Colors are used to differentiate the structural elements; in particular, the two β-barrels are distinguishable in yellow. The catalytic triad is represented in licorice and colored in red.

4.4.2 Results of the dynamics-based alignment

We performed an alignment based on the dynamical information entailed into the first 10 lowest frequency modes obtained by the NMA on the β-Gaussian Network Model of each pair of proteins in the dataset. The alignment consists in the optimization of a score function that maximizes the RMSIP of the two sets of normal modes. For each pair of dynamically-aligned proteins, matching regions in the two structures are identified as the subset of residues giving the best overlap. The number of residues belonging to these cores shows great variability (Figure S9), and their RMSD values range from 0.6 to 4.0 Å; these results are indicative of heterogeneity in dynamics within the dataset.

The distance matrix obtained from the pairwise dynamics-based alignments of all proteins of this dataset is used as a measure of similarity in dynamics. This can be compared to the MEROPS classification by computing the average distance between protein pairs that fall into the same family. Following such a procedure, it is apparent that the average distance in dynamics is lower within each family, with respect to the total average (Figure 4.4). In other words, proteins belonging to the same family are significantly closer in dynamics than they are to members of other families.

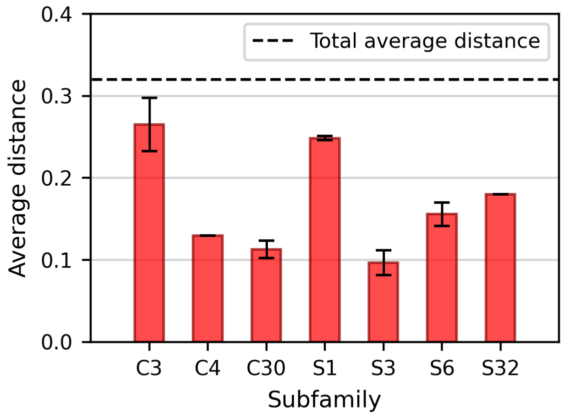


Figure 4.4: Average distances (in terms of dynamics) between proteins of the dataset belonging to the same family. Only those subfamilies including more than one representative member are displayed here. The histograms show that proteins are significantly closer in dynamics within the same family then they are to members of other families.

The distance matrix is used as input for the division of the dataset into dynamically homogeneous protein clusters. The outcome of the hierarchical clustering is graphically expressed by the dendrogram in Figure S10. On the basis of the resolution-relevance plot, 9 clusters were identified (Figure 4.2); this corresponds to a threshold of ≈ 0.58 in the clustering dendrogram. The resulting clusters appear to be quite homogeneous in terms of protease classification (Figure S11). Importantly, the dynamics-based clustering automatically tends to group proteins belonging to the same subfamily. Figure 4.5.a shows that in most of the cases (17 of the 19 subfamilies represented in the dataset) all the members of each subfamily fall into the same cluster, thus suggesting that these proteins share a similar conformational dynamics and strengthening the idea of homogeneity in dynamics between homologous proteins [176, 177]. On the other hand, each cluster groups several subfamilies, and only 4 clusters out of 9 include proteins belonging to only one subfamily (Figure 4.5.b). Therefore, the clustering procedure proves able to effectively group different protein subfamilies that, despite the different evolutionary origin, share similar dynamics.

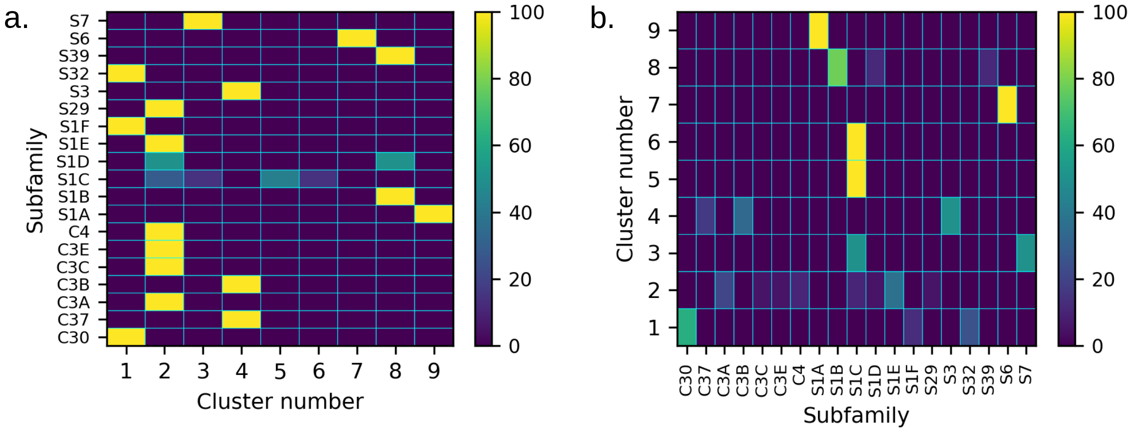


Figure 4.5: **a.** Distribution of the members of each subfamily among the different clusters, expressed as percentage with respect to the total number of members of the subfamily. In **b.** each row represents the content of each cluster classified on the basis of the function (in percentage with respect to the total population of the cluster). The results show that the dynamics-based clustering automatically tends to group proteins belonging to the same subfamily.

4.4.3 Comparison between the dynamics-based and the structure-based clustering

We compared the results from the dynamics-based clustering on the proteases of the PA clan with the structure-based distance tree calculated in the work of Mönttinen et al. [138]. There, the authors identified a common structural core of 72 residues for the set of PA clan proteases taken into account; according to the structural similarities of this common core, they built a distance tree between the members of the dataset. 5 different clusters were identified, contrary to the 9 cluster found in this work.

Despite the two different approaches, the results present several similarities, showing a close relation between structure and dynamics. The S1A subfamily, which includes both bacterial and eukaryotic proteases, forms a clearly distinct and compact cluster both in terms of structure and dynamics. On the other hand, the S1D subfamily, which includes bacterial proteases, is split in two different groups in terms of structure as well as dynamics: in both cases, the S1D *Achromobacter* protease I (1ARB) is close to the bacterial S1B proteases, while the S1D protease AL20 of *Nesterenkonia abyssinica* (3CP7) is close to the members of the bacterial S1E subfamily. This difference between members of the S1D subfamily has been explained on the basis of the different evolutionary history of the bacteria in which they are expressed [138].

Another common feature emerging from the two clustering approaches is the similarity between the S39 subfamily of positive-strand RNA viruses and the bacterial S1B proteases; interestingly, such degree of similarity is higher than between S39 and the other viral proteases, as already reported on the basis of structural comparisons [178]. Moreover, the bacterial S6 family forms an independent group in both clustering approaches. This peculiarity has been attributed to the presence of a long β-stalk structure at the C-terminus (Figure S12), which is absent in all the other proteases of the PA clan [179, 138]; the protease domain alone, instead, shares high structural similarity with that of the S1A subfamily. However, the β-stalk domain was cut before the dynamics-based alignment, meaning that our analysis of dynamics of the S6 protease domain alone is able to distinguish this subfamily from the other members of the PA clan.

Importantly, the two types of clustering present also some differences. In the case of the structure-based analysis, the cysteine proteases tend to be grouped together; however, in the dynamics-based alignment, the similarity is only at the level of one of the two large groups in which the dataset is divided, as evident from the dendrogram in Figure S11. Within this group, C families are mixed with S families, and appear to be more distributed among different clusters than in the distance tree built on the basis of the structural features. This is indicative of a clear differentiation of the C proteases in terms of dynamics, despite their structural similarity in the protein core. This can be explained not only by the fact that different classes of C-proteases are involved in the processing of different viral polyproteins (therefore requiring adaptation to the substrate), but also because some of them have additional functions, playing the role of inhibitors of host cell protein synthesis [180]. Another difference regards the heat-shock proteases S1C, which include proteins from bacteria, chloroplasts, and mitochondria; even though structurally similar in the proteolitic core, members of this subfamily appear very scattered in the dynamics-based clustering. Specifically, the observed similarities in dynamics accentuates the structural relatedness already observed between some eukaryotic S1C proteases and different viral protease subfamilies, inasmuch that these similarities are stronger than the similarity within the S1C subfamily itself. This relatedness has been previously explained on the basis of exchanges of protease genes between eukaryotic viruses and their hosts [138].

In the structure-based distance tree, proteases from flavivirus (families S29 and S7) and from togavirus (family S3) are grouped together, even though the two viruses belong to different families; on the opposite, S29/S7 and S3 are placed in different clusters when their dynamics is included in the analysis. This distinction might arise from the difference in function: the S3 protein togavirin, in fact, does not only function as a viral protease, but plays also the structural role of capsid protein of the virus [181]. S29 and S7 proteases, on the other hand, possess only proteolitic function and do not work as structural components.

Overall, the inclusion of dynamics in the comparison of the proteases from the PA clan adds therefore an additional level of classification, which seems appropriate to bridge structural and functional similarities.

4.4.4 Creation and validation of the basis set of the high-dimensional space of protein dynamics

The representative proteins of the 9 clusters are identified by the PDB codes: 3D23, 1HPG, 2YOL, 1VCP, 3QO6, 1L1J, 1WXR, 4JCN, 4I8H. Their structures are represented in Figure S13. Protein 1GDQ is chosen as the reference structure of the whole dataset, against which the other representatives are dynamically aligned prior to lattice interpolation of their normal modes (see Section 4.3). In the latter, the oriented protein modes are placed and interpolated on a cubic lattice, orthonormalized, and finally ordered. The interpolation on the grid allows us to easily compare the dynamics of any pair of proteins, irrespectively on the number of residues. For instance, modes from proteins with a different number of C_α cannot be directly compared in terms of scalar products, while different vector fields on the grid have the same dimensionality.

We investigated the quality of the orthonormalized modes as a basis set for the dynamics of the whole dataset, by computing the overlap between the spaces given by the protein modes and by the basis. To this aim, the RMSIP was computed between the space spanned by the first 5 modes of each protein in the dataset (after their interpolation on the lattice) and the first n components of the basis. For each protein, the components of the basis are ordered so as to maximise the RMSIP with the protein modes. The resulting RMSIP for each protein is plotted in Figure 4.6.a as a function of the number n of basis vectors considered for the calculation of the RMSIP. From the distribution of the values attained when using the full basis set (45 vector fields), the RMSIP is greater than 0.5 for $\approx$ 94% of the proteins, showing in those cases a good agreement between the dynamics of the protein and the one expressed by the basis [182]. The agreement is excellent (RMSIP$>$ 0.7) for $\approx$ 61% of the proteins; therefore, we can conclude that the identified basis is indeed able to describe with a good generality the large-scale conformational dynamics of the dataset. For each protein, we also computed the normalized RMSIP, by dividing each value of RMSIP with the value obtained with the use of the full basis set. The normalized RMSIP curves show that, for each dataset member, as few as

15 basis components are sufficient to reproduce the 80% of the dynamics that would be attained with the use of the full basis set (Figure 4.6.b); however, such components differ from protein to protein, meaning that there are no vector fields in the basis that can be considered more essential than others. This suggests that a further reduction in the dimension of the basis set would lead to a loss of generality in the description of the dynamics of this class of proteins.

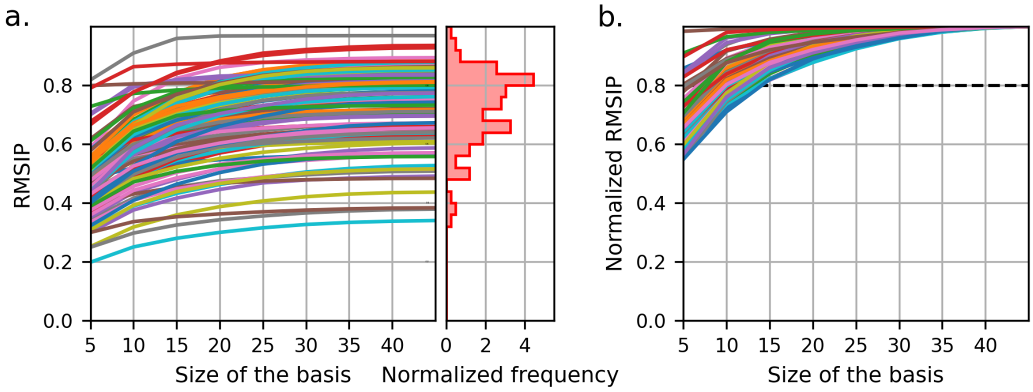


Figure 4.6: **a.** Root-mean-square inner product (RMSIP) between the subspaces spanned by the first 5 modes of each protein and the first n basis vectors, as a function of the basis size n. The histogram on the right represents the distribution of the RMSIP values attained when the full basis is used. The RMSIP shows a good overlap of the subspaces (RMSIP>0.5) for $\approx$ 94% of the proteins. **b.** RMSIP normalized with respect to the value attained from the use of the full basis. For each dataset member, as few as 15 basis components are sufficient to reproduce the 80% of the dynamics that would be attained with the use of the full basis set.

4.4.5 Comparison with MD simulations

In order to better assess the ability of the basis to reproduce the general dynamics of chymotrypsin-like proteases, we performed MD simulations of four proteins belonging to the same family, and compared the per-residue fluctuations emerging from the simulations with those obtained by filtering the trajectory along the vectors of the basis; a good agreement would be indicative of the ability of the basis to describe the large-scale dynamics of the protein. Two of the proteins used as test-case belong to the dataset; these are 1EKB [183] and 1NPM [184], eukaryotic proteases belonging to the S1A subfamily. The other two proteins, 4YOG [185] and 3W94 [186], are external to the dataset, and as such have not been used to define the basis. 4YOG is a C30 protease from the bat coronavirus HKU4, while 3W94 is an S1A enteropeptidase. These two proteins have been included here in order to test the generality of the identified basis for the description of the dynamics of the PA clan, independently on the specific members of the initial dataset.

For each of the four proteins we compared the root-mean-square fluctuations (RMSF) as computed from the simulation, and as computed from the same trajectory filtered along the "modes" given by the backmapping of the protein structure on the basis vectors. The comparison shows a good qualitative agreement (Figure 4.7 and Figure S14), in particular in correspondence of all the secondary structure elements. In the unstructured regions, the comparison is slightly less accurate; this is particularly true for long loops, which are more sensitive to the limitations of the ENM and of the NMA employed to define the modes of the basis, since both assume small-amplitude fluctuations from a well-defined reference structure. From the two sets of trajectories, namely the original MD simulations and the filtered ones, we also computed the dynamic cross-correlation matrices (Figure S15 and Figure S16), which give a measure of the degree of correlation between each pair of C_α atoms in terms of fluctuations from their average position. When comparing the original and filtered trajectories, the intensity of the resulting correlations are different, with higher correlations/anti-correlations emerging form the trajectory filtered on the basis; however, the patterns of correlation are strikingly similar between the two trajectories for all of the four proteins. In addition, we computed the RMSIP between the first n modes obtained from the PCA of the MD simulation and of the filtered trajectory, where n is the number of components that capture the 80% of the variance in the original simulation (Table S2); in all cases, the results show a good overlap of the two subspaces, with RMSIP>0.5. Therefore, the basis set appears to be able to describe the relevant large-scale dynamics of the considered protein systems.

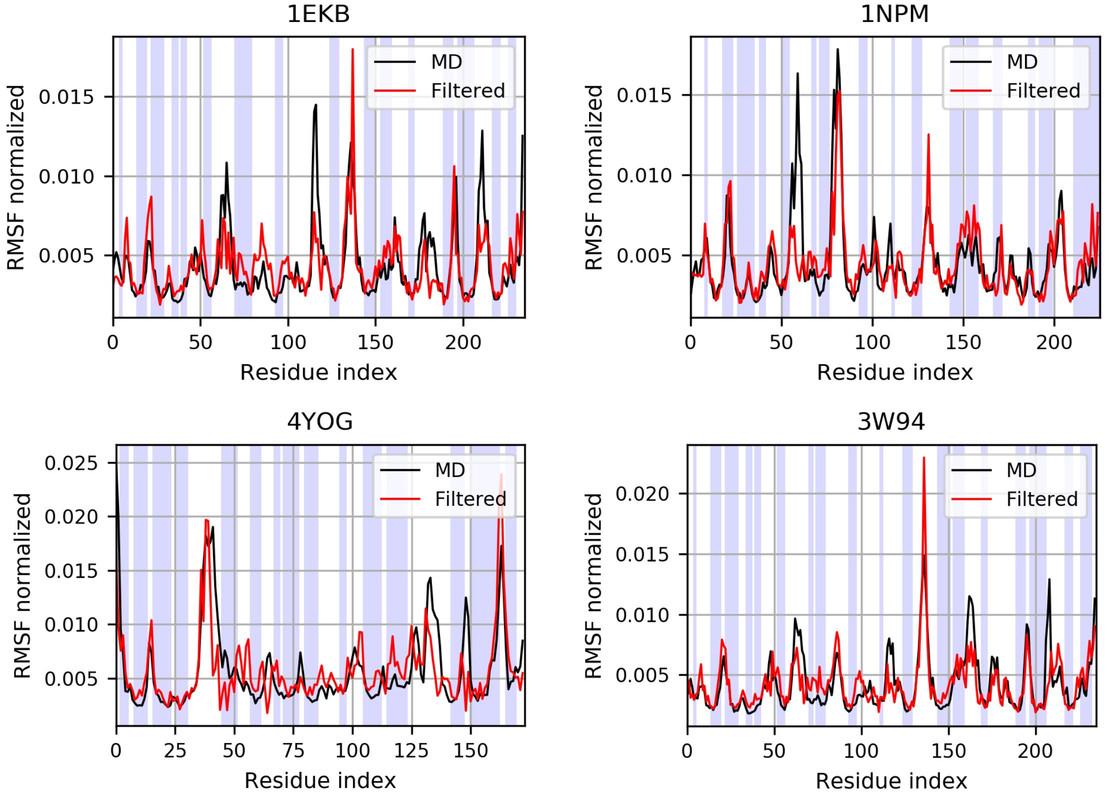


Figure 4.7: Root-mean-square fluctuations (RMSF) of the C_α atoms, normalized with respect to their sum, computed on proteins belonging to the initial dataset (1EKB, 1NPM) and external to it (4YOG, 3W94). The shaded areas correspond to structured regions, identified with the DSSP algorithm [187, 188]. The comparison shows a good qualitative agreement, particularly in correspondence of secondary structure elements.

4.5 Conclusions

In this work, we proposed a workflow for the identification of common large-scale conformational motions in a set of proteins. Specifically, we performed a dynamics-based clusterization of 116 chymotrypsin-related proteases, belonging to the PA clan, and compared the resulting clusters to the MEROPS classification and to a more recent structure-based classification of the same dataset of proteases. The clustering based on the dynamics adds interesting information to that known on the basis of structural and evolutionary relationships between the members of the protein family, thus facilitating the interpretation of dynamics as a bridge between protein structure and function. In addition, we used NMA and the β-GNM to build a basis set of vectors of the high-dimensional space of the PA clan large-scale dynamics, and tested the basis set to demonstrate that it is sufficiently complete to describe the main large-scale dynamical features of the members of the dataset. The basis set of conformational motions was also successfully validated by comparison with results from MD simulations of proteins internal and external to the initial dataset.

In this regard, the method proved to deal particularly well with the conformational dynamics of structured regions; loops and disordered regions are by definition challenging to describe with an ENM, which is able to reproduce only small-amplitude fluctuations with respect to a well-defined reference structure; the dynamics of such regions, however, is qualitatively different from the functional one of the structured part, which is the one responsible to carry out the biological function in the proteins under examination. Additionally, we note that the dataset employed contained only a number of proteins belonging to the family of chymotrypsin-related proteases: a larger dataset is expected to lead to more general results; however, the number of proteins included was limited by the availability of experimental structures and by the choice to remove proteins with too high sequence identity. The natural development of the methodology presented and discussed in this work is its application to a larger dataset of proteins, comprehensive of multiple enzyme superfamilies, with the aim of building a basis set of conformational motions that represents a general vocabulary of proteins' common dynamics. Once mapped on a protein structure, the basis components can help to identify the most common –but diverse among each other– movements that better describe the common large-scale dynamics of the proteins belonging to the dataset. The dynamics of any protein not belonging to the initial set can be projected on the basis, so as to describe it in terms of a few general movements and thus facilitating the comparison between the dynamical features of different proteins. In addition, the method can be employed to identify those common structural signatures that characterise the dynamics encoded in the basis components, and relate them to specific biological functions.

4.6 Appendix

4.6.1 Additional Figures

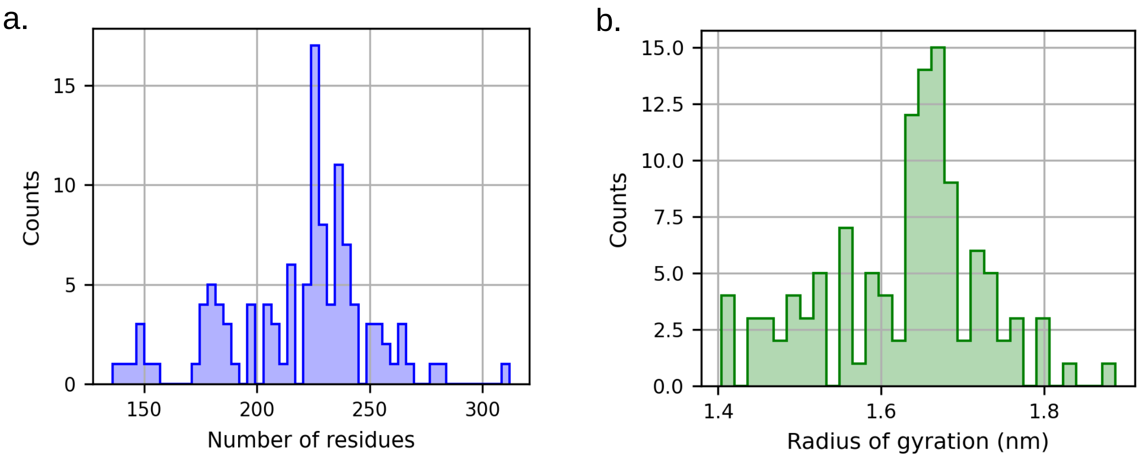


Figure S8: Histograms of the sequence length (**a**) and radius of gyration (**b**) of the proteins in the dataset.

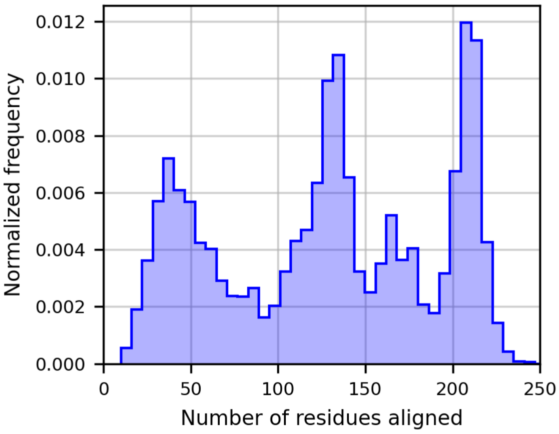


Figure S9: Histograms of the number of residues belonging to the superimposed protein cores, defined from the dynamics-based alignment of each pair of proteins from the dataset.

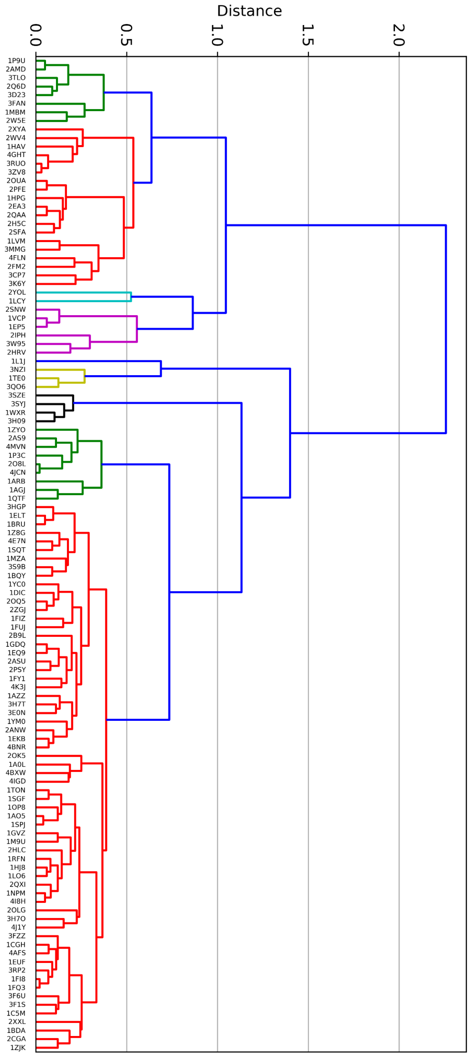


Figure S10: Dendrogram resulting from the hierarchical clustering, performed on the basis of the distance in dynamics between the dataset elements. The labels represents the PDB IDs, and colors are used to differentiate the clusters.

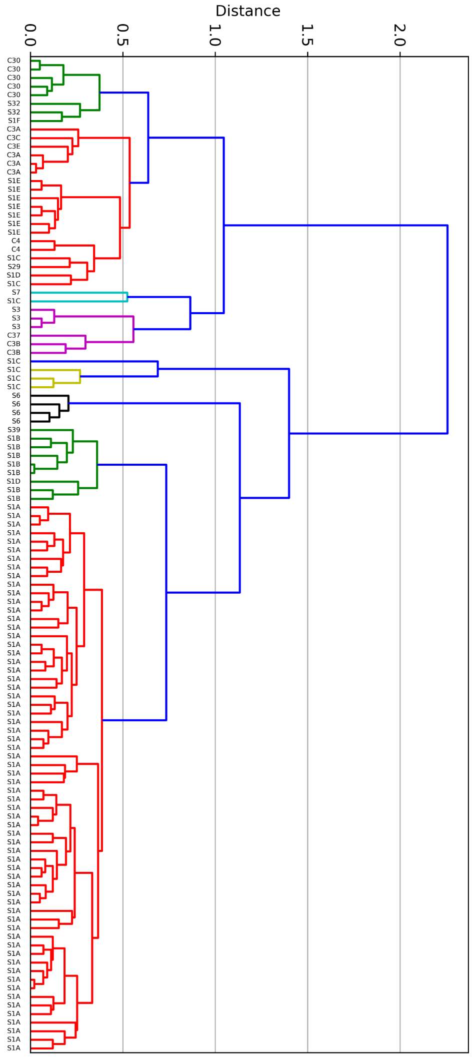


Figure S11: Dendrogram resulting from the hierarchical clustering, performed on the basis of the distance in dynamics between the dataset elements. The labels represents the protease subfamily of each protein, and colors are used to differentiate the clusters.

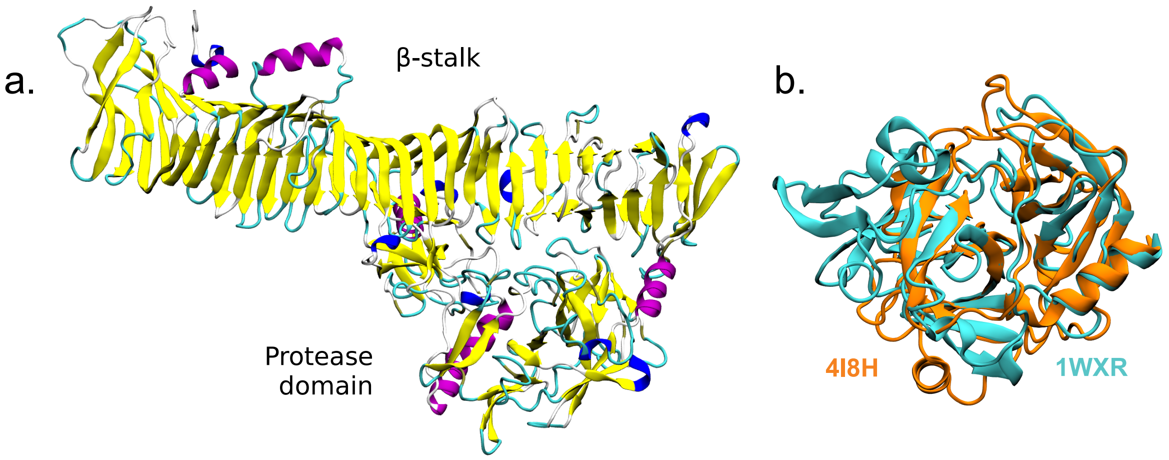


Figure S12: **a.** Full structure of the 1WXR protease from subfamily S6, displaying the long β-stalk domain at the C-terminus. **b.** Structural alignment of 1WXR (in cyan) and 4I8H from subfamily S1A (in orange), showing the similarity of their protein core.

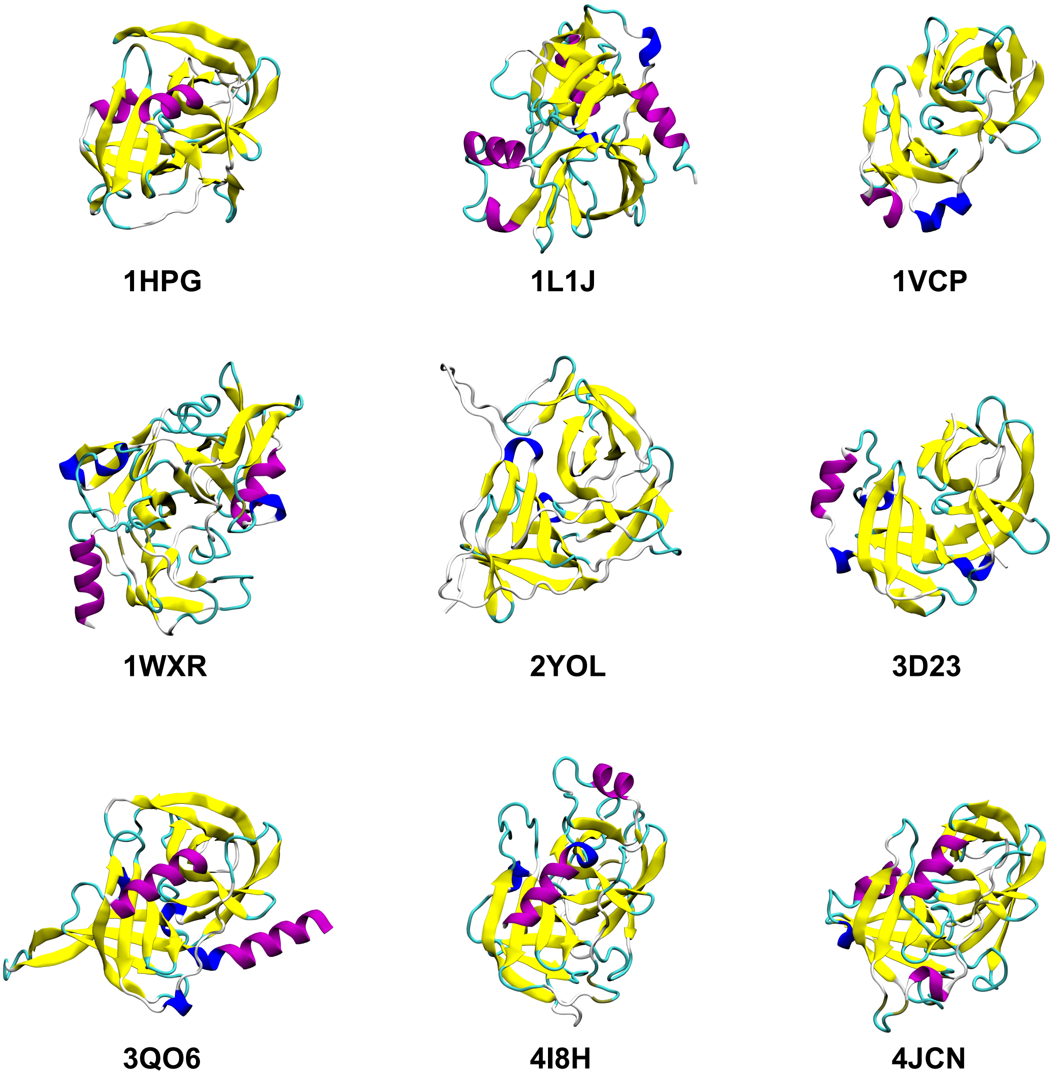


Figure S13: Structure of the representatives of each protein cluster, resulting from the dynamics-based alignment. The color corresponds to the type of secondary structure element: β-sheets in yellow, α-helices in magenta, 3-10 helices in blue and loops in cyan.

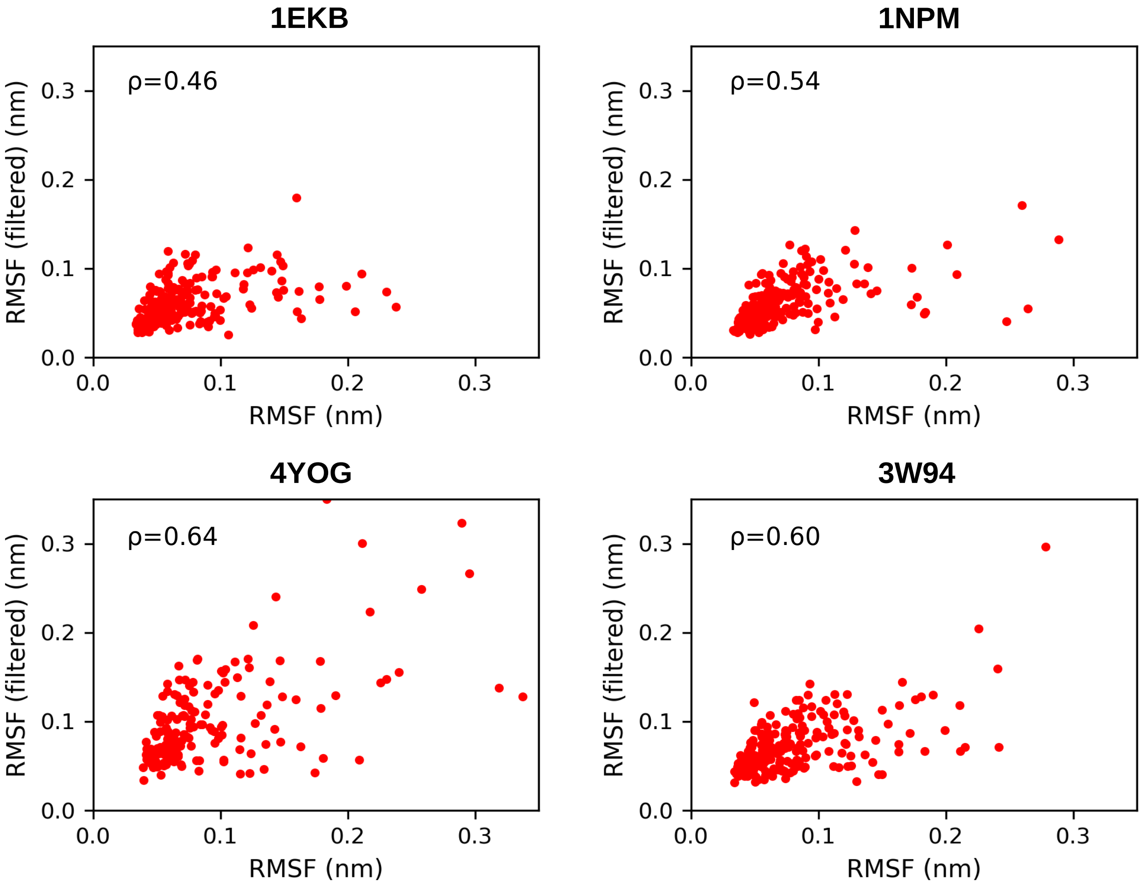


Figure S14: Scatter plots of the root-mean-square fluctuation (RMSF) values, computed on the C_α atoms, from the MD simulations of the protein and from the same trajectories filtered on the basis set. ρ indicates the value of Pearson Coefficient computed between the two sets of fluctuations. All cases show satisfactory results.

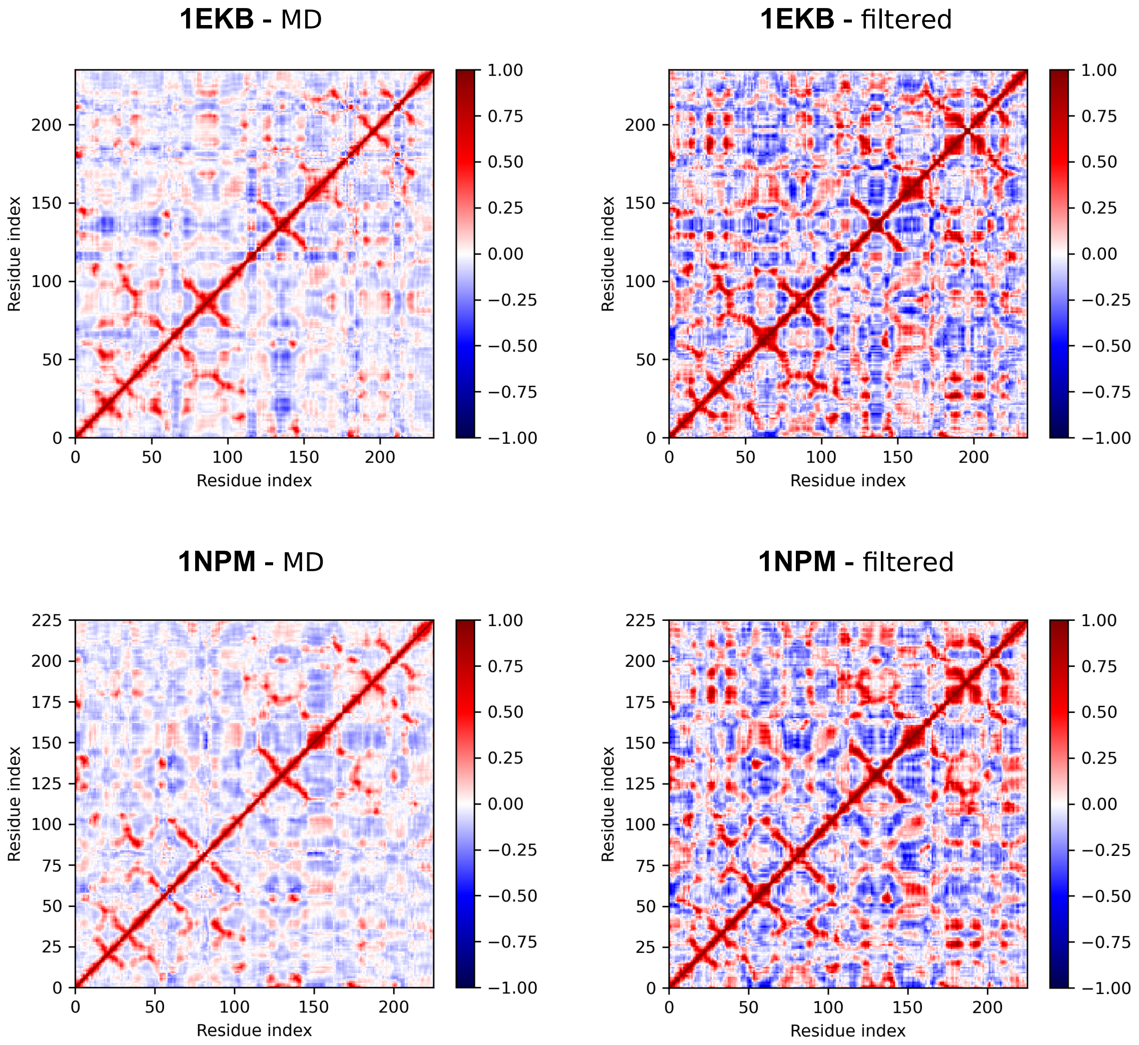


Figure S15: Cross-correlation computed from the simulations of the proteins 1EKB and 1NPM, both on the original and filtered trajectories. Both proteins belong to the dataset.

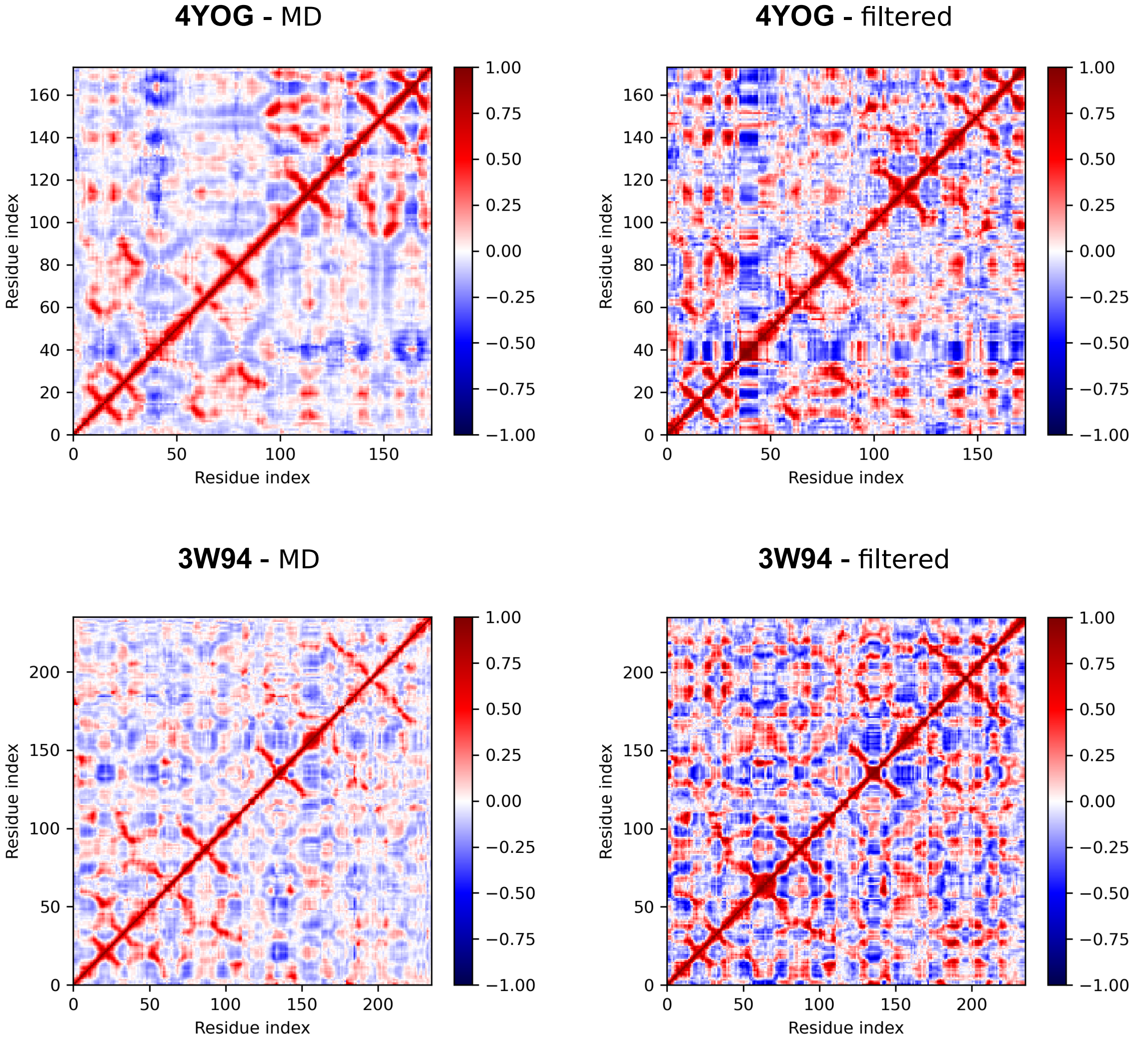


Figure S16: Cross-correlation computed from the simulations of the two proteins 4YOG and 3W94, both on the original and filtered trajectories. The two proteins are not part of the dataset from which the basis set is derived.

4.6.2 Additional Tables

Table S1: List of the PDB IDs of the proteins comprising the dataset.

1A0L 1CGH 1FY1 1MBM 1TE0 2AS9 2O8L 2SNW 3F1S 3QO6
4BXW 1DIC 1GDQ 1MZA 1TON 2ASU 2OK5 2W5E 3F6U 3RP2
4E7N 1EKB 1GVZ 1NPM 1VCP 2B9L 2OLG 2WV4 3FAN 3RUO
4FLN 1AGJ 1ELT 1HAV 1OP8 1WXR 2CGA 2OQ5 2XXL 3FZZ
3S9B 4GHT 1AO5 1EP5 1HJ8 1P3C 1YC0 2EA3 2OUA 2XYA
3H09 3SYJ 4I8H 1ARB 1EQ9 1HPG 1P9U 1YM0 2FM2 2PFE
2YOL 3H7O 3SZE 4IGD 1AZZ 1EUF 1L1J 1QTF 1Z8G 2H5C
2PSY 2ZCH 3H7T 3TLO 4J1Y 1BDA 1FI8 1LCY 1RFN 1ZJK
2HLC 2Q6D 2ZGJ 3HGP 3W95 4JCN 1BQY 1FIZ 1LO6 1SGF
1ZYO 2HRV 2QAA 3CP7 3K6Y 3ZV8 4K3J 1BRU 1FQ3 1LVM
1SPJ 2AMD 2I6Q 2QXI 3D23 3MMG 4AFS 4MVN 1C5M 1FUJ
1M9U 1SQT 2ANW 2IPH 2SFA 3E0N 3NZI 4BNR

Table S2: RMSIP computed between the first n modes obtained from the PCA of the MD simulation and of the filtered trajectory, where n is the number of components that capture the 80% of the variance in the original trajectory. The results show a good overlap of the two subspaces in all the simulated systems.

PDB ID	N. of components	RMSIP
1EKB	48	0.61
1NPM	43	0.62
4YOG	24	0.60
3W94	39	0.59

Chapter 5

An Efficient Deep Neural Network Approach to Implicit Solvation

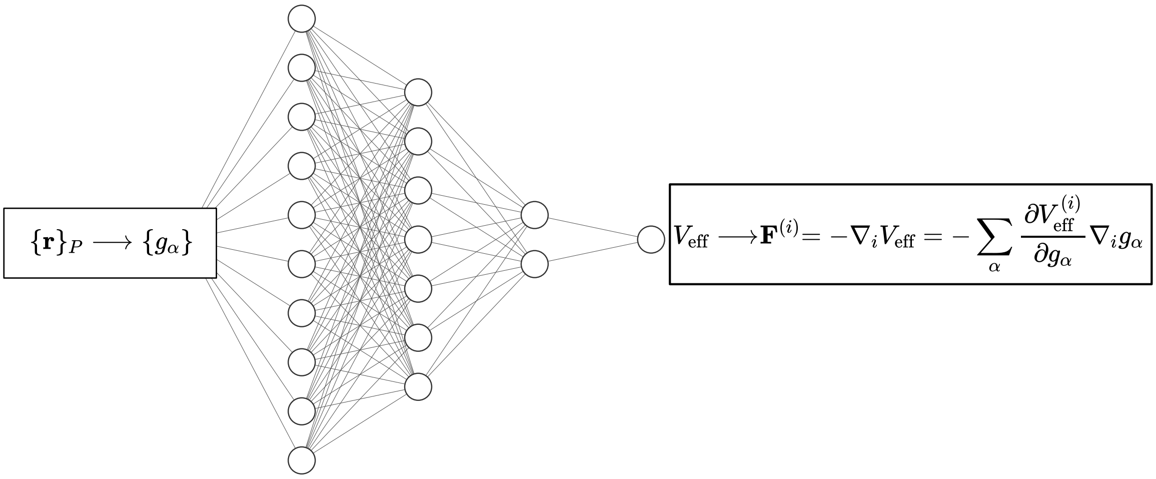

$\{\mathbf{r}\}_P \longrightarrow \{g_\alpha\}$
$V_{\text{eff}} \longrightarrow \mathbf{F}^{(i)} = -\nabla_i V_{\text{eff}} = -\sum_\alpha \frac{\partial V_{\text{eff}}^{(i)}}{\partial g_\alpha} \nabla_i g_\alpha$

Figure 5.1: Scheme of the simplest neural network architecture adopted to learn $V_{\text{eff}}[\{\mathbf{r}\}_P] \simeq \mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_P]$ (here P stand for *Protein*, while more generally we can use M indicating the solute *Molecule*). As input layer, the first 10 neurons receive the values of the 10 SF calculated with different values of η. The activation function used is the tanh and each value of the hidden neurons is used in order to calculate the gradient of the output with respect to the input, as explained in the Appendix.

In this chapter, we describe our attempt to answer to the question formulated in chapter 3 about the implicit solvent models, following a totally different perspective. We show a way to estimate the PMF of solvation by relying on an *artificial neural network*. The core ideas of the project originate from a pioneering work by Behler and Parrinello [189]. It's important to stress that the choice of building a model for the PMF (and not, for example, directly the mean *force* itself) is not casual: being $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]$ a scalar, it is translationally and rotationally invariant and so independent on the choice of the frame of reference.

Nota Bene: since its promising results in almost every field of scientific research, machine learning methods such as ANN-based regression acquired a lot of visibility. For these reasons, it is expectable that the same idea of applications can be developed from different research group around the world independently. This is what happened for the core idea of this project: while we were working on the MALIS model, Noé and coworkers came up with a similar implementation of essentially the same idea, which is explained in [190]. At the end of the chapter we dedicate a paragraph to recap similarities and differences between the two works.

5.1 Methods

5.1.1 Deep Neural Network PMF

A generic deep neural network can be seen as a function $f : \mathbb{R}^n \to \mathbb{R}^m$ that depends parametrically on a tensor $\mathbf{W}$ of weights. The dimensions of $\mathbf{W}$ depend on the number of hidden layers and the number of neurons chosen for the specific application. The dataset used to train the neural network is a set of input and output values, which are used to parametrize the network itself (by tuning $\mathbf{W}$). The goal of the training is to make the network reproduce the same input-output relation, together with being able to predict new outputs from inputs that are in a region of the domain not visited by the training dataset. In our case, we have $m = 1$ because the target function $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]$ is a scalar function and, in general, we should have $n = 3N$ (solute degrees of freedom). By the way, a first typical approximation that is done in these contexts is to assume that $\mathscr{V}_{\text{eff}}$ is separable into the contribution coming from each solute atom i independently, *i. e.* $\mathscr{V}_{\text{eff}} \simeq V_{\text{eff}}[\{\mathbf{r}\}_M] = \sum_{i=1}^{N} V_{\text{eff}}^{(i)}[\{\mathbf{r}\}_i]$, where $\{\mathbf{r}\}_i$ indicates the neighborhood of atom i, within a certain cutoff. This simplification opens the door to the possibility of building a NN that is able to predict each term $V_{\text{eff}}^{(i)}[\{\mathbf{r}\}_i]$ without any specific bias, using input functions of the atomic coordinates that are able to describe the environment of each atom of the solute, taking into account for example the spatial neighborhood and their chemical properties. The functions we chose to use are called *symmetry functions* (SF) [189] and they work as descriptors of each atom's local environment. The choice of the functional form is dictated also by the property of translational and rotational invariance of the SF. To be more explicit, the final form of the ANN PMF of solvation of each atom i will be $V_{\text{eff}}^{(i)}[\mathbf{g}(\{\mathbf{r}\}_i)]$, where $\mathbf{g} := [g_1(\{\mathbf{r}\}_i), \dots, g_S(\{\mathbf{r}\}_i)]$ represents a specific choice of S of these SFs (introduced below). We now want to introduce another approximation. As already anticipated, the input values of our NN are the SF, calculated for each atomic environment. For the output values used in the training, we chose to train the NN to reproduce the instantaneous energies of interaction between the atoms of the solute molecule and the solvent molecule. In the ideal case, concerning classical force fields for molecular systems, this quantity consists in the so-called non-bonded interactions for each atom at each frame of the trajectory:

$$U_{MS}^{(i)}[\{\mathbf{r}\}_S, \{\mathbf{r}\}_M] = \sum_{j=1}^{\mathcal{N}} U_{\text{sol}}[r_{ij}] = \sum_{j=1}^{\mathcal{N}} \left[k\frac{q_i q_j}{r_{ij}} + \epsilon_{ij}\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \epsilon_{ij}\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6} \right] \tag{5.1}$$

while for practical purposes in molecular dynamics softwares like those used by us for this work (LAMMPS and GROMACS) this terms are either calculated up to a cutoff distance or approximated with various techniques, such as Particle Mesh Eward [24]. We consider this the *ideal* case in the sense that we use the all-atom description as the reference one and the goal is to build a CG model (coarse-grained in the sense discussed in chapter 3) that is able to properly sample the configurational space and energetics of the all-atom solute. We picked the mean squared error (MSE, indicated by $\mathcal{L}$) as an error function to be minimized during the training process of the NN. The final form of the error function is then:

$$\mathcal{L}(\{U_{MS}\}, \{V_{\text{eff}}[\mathbf{g}]\}; \mathbf{W}) = \left\langle \sum_{i=1}^{N} \left(U_{MS}^{(i)}(\{\mathbf{r}\}_M) - V_{\text{eff}}^{(i)}[\mathbf{g}(\{\mathbf{r}\}_M); \mathbf{W}] \right)^2 \right\rangle \tag{5.2}$$

where $< \cdot >= \dfrac{1}{T}\sum_{t=1}^{T} \cdot$ is the average on all the frames collected in the training dataset. We notice that this error function recalls the one proposed in [191], with some differences. To match our formalism with theirs we can do the following:

$$E\left(\mathbf{M}(\{\mathbf{r}\})\right) = U_{MM}(\{\mathbf{r}\}_M) + V_{\text{eff}}(\{\mathbf{r}\}_M) \qquad u(\{\mathbf{r}\}) \equiv U_{\text{tot}}(\{\mathbf{r}\}_M, \{\mathbf{r}\}_S) \tag{5.3}$$

because in an implicit solvent the mapping is essentially a decimation one that removes every degree of freedom of the solvent, $\mathbf{M}(\{\mathbf{r}\}) = \{\mathbf{r}\}_M$. By manipulating a bit the quantity $\chi^2[E]$

they use as cost function, one can obtain:

$$\chi^2[E] = \left\langle \left| \sum_{i=1}^{N} \left(U_{MS}^{(i)}(\{\mathbf{r}\}_M) - V_{\text{eff}}^{(i)}[\mathbf{g}(\{\mathbf{r}\}_M)] \right) + U_{SS}(\{\mathbf{r}\}_S) \right|^2 \right\rangle = \tag{5.4}$$

$$= \mathcal{L}(\{U_{MS}\}, \{V_{\text{eff}}[\mathbf{g}]\}) + \left\langle |U_{SS}(\{\mathbf{r}\}_S)|^2 \right\rangle + \tag{5.5}$$

$$+ 2 \left\langle \sum_{i \neq j} \left| \left(U_{MS}^{(i)}(\{\mathbf{r}\}_M) - V_{\text{eff}}^{(i)}[\mathbf{g}(\{\mathbf{r}\}_M)] \right) \left(U_{MS}^{(i)}(\{\mathbf{r}\}_M) - V_{\text{eff}}^{(i)}[\mathbf{g}(\{\mathbf{r}\}_M)] \right) \right| \right\rangle \tag{5.6}$$

Since the second term is independent of the weights, minimizing $\mathcal{L}(\{U_{MS}\}, \{V_{\text{eff}}[\mathbf{g}]\})$ differs from minimizing $\chi^2[E]$ only by the third term.

The next section is dedicated to introduce these SF, to explain how they are treated in the ANN and the way how to calculate the force term from the whole model of $\mathscr{V}_{\text{eff}}[\{\mathbf{r}\}_M]$, which we already called $V_{\text{eff}}[\{\mathbf{r}\}_M]$.

5.1.2 Symmetry Functions

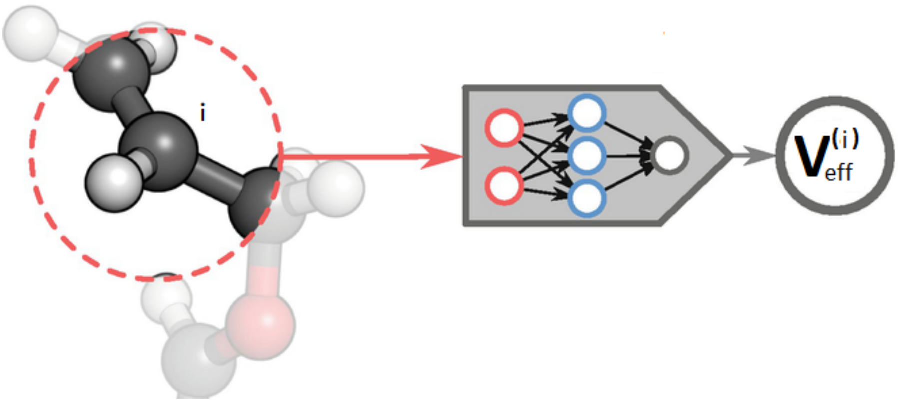


Figure 5.2: Generic representation of the workflow of MALIS: after defining the local environment of atom i (within a cutoff), the atomic positions and partial charges of the atom's neighborhood are treated with the SF and passed to the NN, which in turn is able to calculate $V_{\text{eff}}^{(i)}$ and $\mathbf{F}_{\text{eff}}^{(i)}$.

In our applications we will mainly focus on two different kinds of SF, all dependent on some parameters that can be changed in order to give diversity to the structural information we want to extract and pass to the ANN. Following the literature [189], these are named $G_1^{(i)}$ and $G_2^{(i)}$, for each atom i. We define them below:

$$G_1^{(i)} := q_i \sum_{\{j\}_i} q_j \, f_c(R_{ij}) \tag{5.7}$$

where: $R_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$; q_i indicates the partial charge of the specific atom; $\{j\}_i$ restricts the sum over all other atoms with a distance less than a given cut-off R_c from the i-th atom (the neighbors); and

$$f_c(R_{ij}) := \begin{cases} \frac{1}{2}\left[\cos\left(\pi \frac{R_{ij}}{R_c}\right) + 1\right], & \text{if} \quad R_{ij} \leq R_c \\ 0, \quad \text{otherwise} \end{cases} \tag{5.8}$$

$G_2^{(i)}$ is a more complex version of $G_1^{(i)}$ that includes a Gaussian filter that highlights the presence of atoms with distance close to a new parameter R_s:

$$G_2^{(i)} := q_i \sum_{\{j\}_i} q_j \, e^{-\eta(R_{ij}-R_s)^2} \, f_c(R_{ij}) = q_i \sum_{\{j\}_i} q_j \, \mathcal{F}_c(R_{ij}) \tag{5.9}$$

where we introduce $\mathcal{F}_c(R_{ij}) := e^{-\eta(R_{ij}-R_s)^2} \, f_c(R_{ij})$ for simplicity (see Appendix). We pinpoint that this version of SF is slightly modified from that found in literature due to the presence of the partial charges q_i,q_j of the atomic species involved in the Gaussian filter: the goal for it is to include chemical information in the inputs of the network. Here of course the other new parameter η is set to control the width of the Gaussian. In our applications, we used the following couples of parameters to calculate the 10 SF used as input for the NN: $(R_s, \eta) = (0, 0)$, $(10, 22)$, $(10, 12)$, $(10, 8)$, $(10, 4)$, $(10, 3)$, $(10, 2)$, $(10, 1)$, $(10, 0.05)$, $(10, 0.01)$, [Å, Å^{-2}].The first is in fact equivalent to a G_1-type.

5.2 Implementation

5.2.1 Training process

In this paragraph we summarize the paradigm followed to prepare the training dataset and to train the ANN's parameters. We can divide the process into 4 main steps:

1. Explicit solvent simulation: the first step is to obtain a trajectory of the system (solute and solvent) that can be considered long enough for the solute to explore its conformational space (or at least the basin of the first local minimum reached by the molecule from the initial configuration). In our applications this step was made relying on GROMACS; the specific setup of the simulations is reported in the Appendix.

2. Symmetry function calculation: the second step was performed by a Python v3 [192] script, which calculates a set of 10 different $G_2^{(i)}$, as anticipated. This step produces the set of $\{X_{train}\}$ input data used during the training process of the ANN, together with the $\{y_{train}\}$ labels.

3. Solute-solvent interaction energy calculation: this second step was made again relying on GROMACS, and specifically by using the rerun functionality, which allows to define new atom groups and to calculate single atom-solvent interaction energies for each frame of a given trajectory, which was of course the one obtained in step 1. This step is required to obtain the set $\{y_{train}\}$ of target labels that the ANN is asked to associate to the relative set of $\{X_{train}\}$, during the training process.

4. Training of the ANN: this step was performed by a Python v3 script using the Keras framework for Tensorflow [193]. Before the construction of the model, which has 3 hidden layers with 10, 7 and 2 neurons (see 5.1), the dataset was shuffled, divided in two groups (80% training and 20% testing) and finally normalized to values in the range $[0, 1]$. More details on the choices made in the training step and on the results of the training are reported in the Application section.

5.2.2 Force Calculation

We can calculate the instantaneous force acting on each atom i by performing the gradient of $V_{\text{eff}}[\mathbf{g}(\{\mathbf{r}\}_M)]$. First of all, as anticipated, by construction $V_{\text{eff}}[\mathbf{g}(\{\mathbf{r}\}_M)]$ can be decomposed in the sum of the PMF relative to each single solute atom i:

$$V_{\text{eff}}[\mathbf{g}(\{\mathbf{r}\}_M)] = \sum_{i=1}^{N} V_{\text{eff}}^{(i)}[\mathbf{g}(\{\mathbf{r}\}_M)] \tag{5.10}$$

Now we can focus on the calculation of the solvent mean force. As an example, we can consider the force acting on atom i along the $\hat{x}$ direction:

$$F_x^{(i)} = -\frac{\partial V_{\text{eff}}^{(i)}}{\partial x_i} \tag{5.11}$$

We know that $V_{\text{eff}}^{(i)} = V_{\text{eff}}^{(i)}[\{g_\alpha(\ldots, x_i, \ldots)\}]$ and so this derivative can be performed with the chain rule:

$$\frac{\partial V_{\text{eff}}^{(i)}}{\partial x_i} = \sum_{\alpha} \frac{\partial g_\alpha}{\partial x_i} \frac{\partial V_{\text{eff}}^{(i)}}{\partial g_\alpha} \tag{5.12}$$

In the appendix we show how to calculate both $\frac{\partial V_{\text{eff}}^{(i)}}{\partial g_\alpha}$ and $\frac{\partial g_\alpha}{\partial x_i}$ analytically. In particular, we show how the first term can be calculated as the derivative of the ANN output with respect to its inputs that are indeed the SF. We stress the fact that having an analytic form for both these two quantity enhances the algorithmic implementation of the method, since there is no requirement of calculating derivatives numerically.

5.3 Applications

Molecule	$\mathbf{T_{train}}$ **[ns]**	δ**t[ps]**	$\mathbf{R_{cut}}$**[nm]**	$\mathbf{T_{test}}$ **[μs]**	**# runs**
Alanine Dipeptide	10	1	10	2	1
Icosalanine	50	10	20	1	20
ssRNA fragment	50	10	20	1	50

Table 5.1: Table summarizing the parameters involved in building the NN training datasets and in generating the trajectory used for the comparison with the IS simulations (testing). $T_{\text{train}}/T_{\text{test}}$ indicates the length of the ES training/testing trajectory; δt indicates how often we selected each frame in the training trajectory in order to extract $\{X_{train}\}$ and $\{y_{train}\}$; R_{cut} is the cut-off used to calculate the SF; # runs is the number of IS parallel runs launched for the comparison.

In this section we show some standard analysis performed on the trajectories obtained by the GROMACS simulations used for training the ANN and the trajectories obtained by the LAMMPS simulations with the implicit solvent model. Three systems have been chosen for the comparison: a typical benchmark molecule, alanine dipeptide; a simple polypeptide chain made up of 20 alanines (Ala)$_{20}$; and a single-stranded RNA fragment, with sequence *UUUAUCCGUACUCAGCCAUUGUACACUACCG* (31 nucleotides). Every calculated quantity is explained in more details in the Appendix.

5.3.1 Alanine Dipeptide

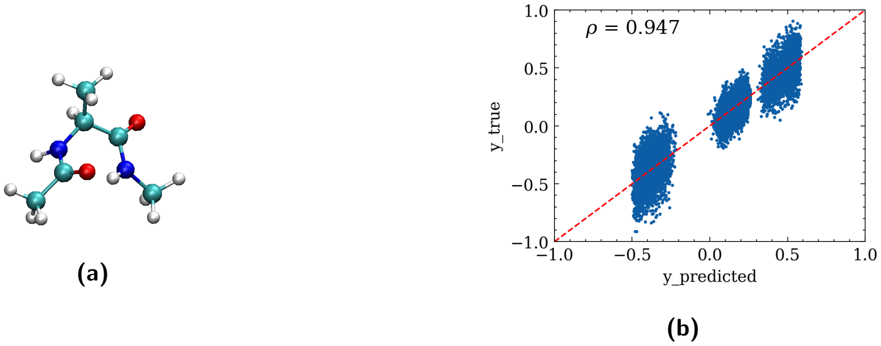


Figure 5.3: **(a)** VMD [39] CPK visualization of alanine dipeptide in a configuration of those belonging to the basin with $\Phi \simeq 50^o$ (MALIS simulation). **(b)** scatter plot of the normalized output of the NN $\{y_{predicted}\}$ after the last step of training, compared to the true output $\{y_{train}\}$ (equivalent to $\{y_{true}\}$) with the relative ρ.

One of the most typical benchmark molecule used to test new methodological frameworks in theoretical and computational biophysics is alanine dipeptide [194, 195, 196, 197]. The two main reasons why it adapts so well are: 1) it's very small (22 atoms) and therefore it requires minimal computational effort to be simulated and 2) it is well known that it describes a peculiar free energy profile (FEP) by using the (only) two dihedral angles of its backbone as collective variables (CV) [194]. This FEP is of course different if one simulates the peptide surrounded by solvent molecules or not, and for this reason it adapts to our purpose of testing a novel IS model. As shown in figures 5.4 **(a)** and **(b)**, one can see that in the ES case the FEP has 5 distinct minima in the Ramachandran angle space (out of 6 explorable with longer simulations, see [198]). The peculiar FEP of alanine dipeptide originates from an interplay of non bonded interactions: the formation/disruption of the hydrogen bond involving the right-hand side nitrogen (in blue) as donor, with its hydrogen (in white), and the left-hand side oxygen (in red) as acceptor; and the steric repulsions due to the van der Waals interactions. As a consequence, an implicit solvent that is able to reproduce the proper FEP is in principle reliable also from a chemical perspective.

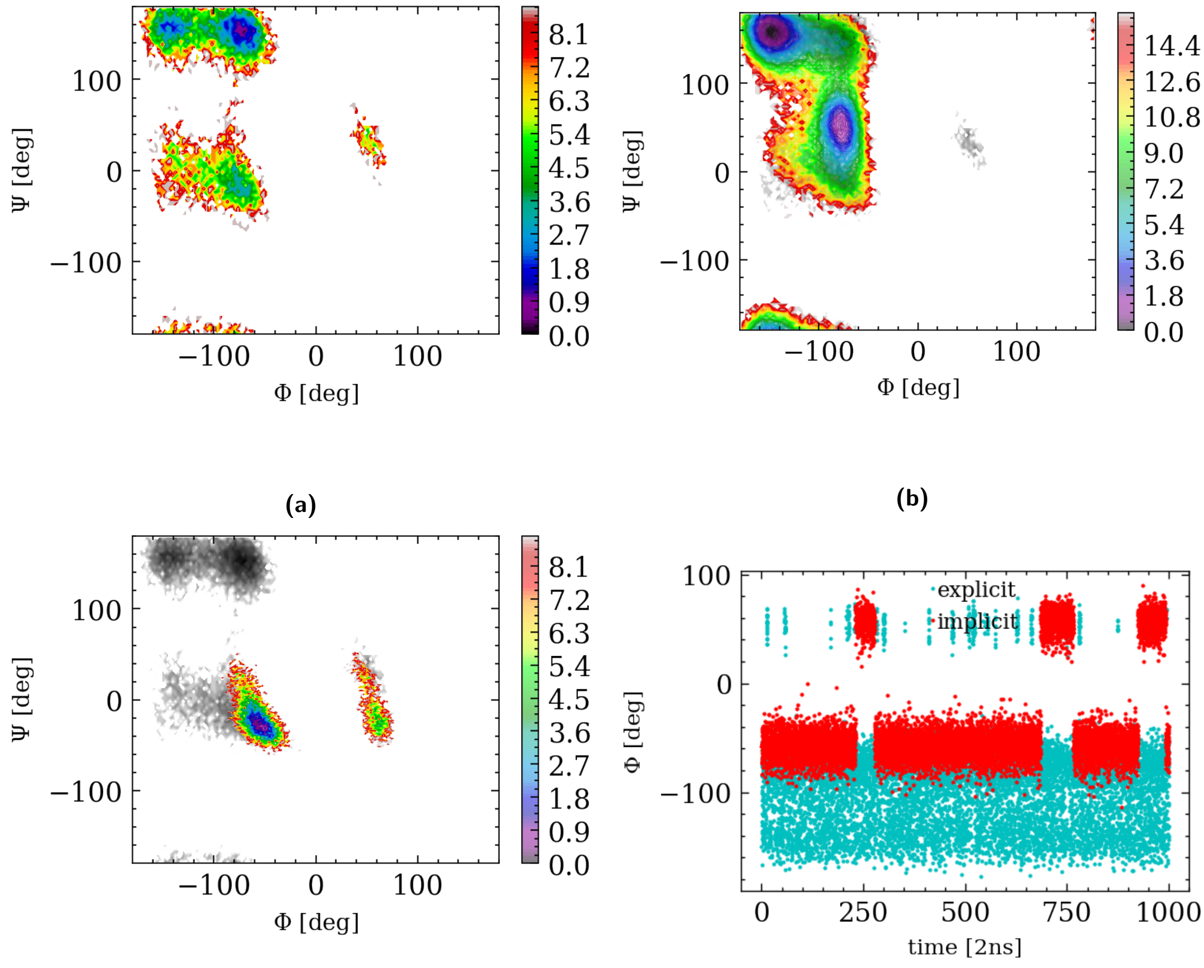
For these reasons, we decided to apply the MALIS framework to this peptide and we used the Ramachandran FEP as analysis to assess the goodness of the implicit solvent model. In figure 5.4 **(c)** we report the FEP obtained through a MALIS simulations with a NN trained on a dataset with summary properties described in table 5.1. Clearly, the two minima explored with a $2\mu s$-long simulation are on one hand compatible with the corresponding one explored by the ES simulations, while on the other hand they are too enduring, in a kinetic sense (see the time windows spanned by the values of Φ in figure 5.4 **(d)**). In spite of that, however, we want to point out that it is hard to do kinetic considerations and comparisons between an implicit solvent simulation and an explicit solvent one: the integration of the solvent degrees of freedom could certainly lead to an acceleration of dynamical processes, like jumping from one minimum to another. Curiously, by looking at 5.4 **(d)**), one is induced to think that the characteristic time of jumping is increased in the IS simulation, fact that contradicts the above-mentioned consideration. At this stage, we are not able to establish what are the real causes of this observation and so the predictive power of the method, from a kinetic perspective, remains unclear and we will keep it in mind for future developments.

In figure 5.4 we reported the dihedral space sampled in a simulation performed with the simplest implicit solvent model available in the LAMMPS package version we used (v2018). It is based on an heuristic mean-field description of the shielding action of the solvent on the solute, described as a continuum dielectric: the relative dielectric permittivity is assumed to be a linear function of the relative distance of the 2 partial charges i, j involved in each Coulomb interaction, $\epsilon_r := \epsilon_r(r_{ij}) = \alpha r_{ij}$ ($\alpha = 1$Å is introduced to make ϵ_r adimensional), so that the Coulomb pairwise potential and force become the following:

$$V_{\text{rdie}}(r_{ij}) \equiv \frac{1}{4\pi\epsilon_0}\frac{q_i q_j}{\alpha r_{ij}^2} \quad \Rightarrow \quad \mathbf{F}(r_{ij})_{\text{rdie}} = 2\frac{1}{4\pi\epsilon_0}\frac{q_i q_j}{\alpha r_{ij}^3}\hat{\mathbf{r}}_{ij} \tag{5.13}$$

where the pedices *rdie* refer to the nomenclature used in [199] and stand for *linear r dielectrics*. The 2 basins sampled are clearly wrong and resemble a lot those sampled by a simulation performed in vacuum (*i.e.* $\epsilon_r = 1$). This fact allows us to at least claim that our implementation of MALIS in LAMMPS is a step forward in modeling implicit solvation for biomolecules MD simulations for this package (at least up to the v2018 used here). Keep in mind that this IS model is considered unreliable because it leads to dynamical motion suppression at high ($\sim 300K$) temperatures [199].

Nevertheless, the central issue in the MALIS simulation, which in turn causes a wrong sampling of the dihedral space, is the observation of a kinetic trap, which is supported by visual inspection of the trajectory. We identify the formation of the aforementioned hydrogen bond as the underlying cause of the kinetic trap, for it leads to a structural constraint that is shown to be irreversible, at least at the temporal scale investigated by us. This phenomenon of excessive hydrogen bond persistence in implicit solvent models has been well documented in previous literature [200]. For a comprehensive examination of this issue, the reader is directed to the



(a)

(b)

(c)

(d)

Figure 5.4: **(a)-(b)-(c):** FEPs calculated using the two dihedral angles Ψ and Φ as collective variables. Plot **(a)** and the grey plots in **(b)** and **(c)** refers to the explicit solvent simulation and shows the well-known [194] pattern of alanine dipeptide dihedral angles. The colored plots in **(c)**, from the implicit solvent simulation, points out that the system is somehow trapped into two of the five minima that are instead explored by the explicit solvent simulation; in the linear dielectric approximation simulation **(b)**, however, the two minima explored are clearly different; **(d)** time series of the sole dihedral angle Φ. It is interesting to notice that for the explicit case at least three ranges can be identified, which are related to the minima seen in **(c)**; on the other hand, the implicit solvent values cover only two of the three ranges of values.

discussion section. As we will show in the next sections, further investigations were conducted using two other systems which are expected to manifest hydrogen bonds along their chains, but with an increased flexibility (due to their polymeric nature), to assess the behavior of the model with longer chains.

5.3.2 Icosalanine

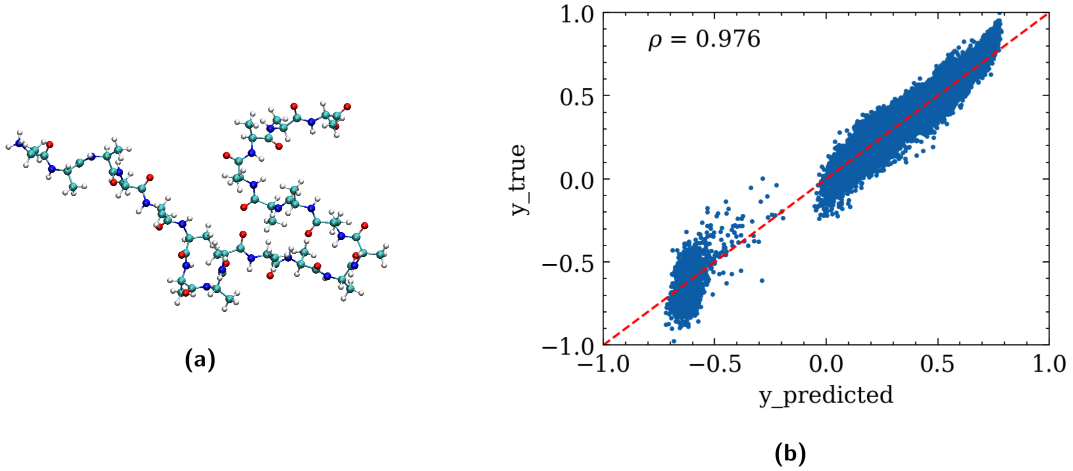


Figure 5.5: **(a)** VMD [39] CPK visualization of Icosalanine in the last conformation after the 50ns of simulation used to train the MALIS neural network. **(b)** scatter plot of the normalized output of the NN $\{y_{predicted}\}$ after the last step of training, compared to the true output $\{y_{train}\}$ (equivalent to $\{y_{true}\}$) with the relative ρ.

The second application of the MALIS that we want to discuss involves a simple, biologically irrelevant molecule, Icosalanine, which is an amino acid chain made of 20 alanines. This system is chosen for some reasons. For being non-exotic from a chemical point of view, given the simple nature of the monomers. It is a natural subsequent step after alanine dipeptide, since it is nothing but a polymer made by copies of it. It has a predisposition to form secondary structural motifs, mainly α-helices, and this feature can manifest in a molecular dynamics simulation; by the way, we can say *a posteriori* that the limits of the timescales explorable by the current implementation of the MALIS method don't allow for the formation of any secondary structure in the peptide.

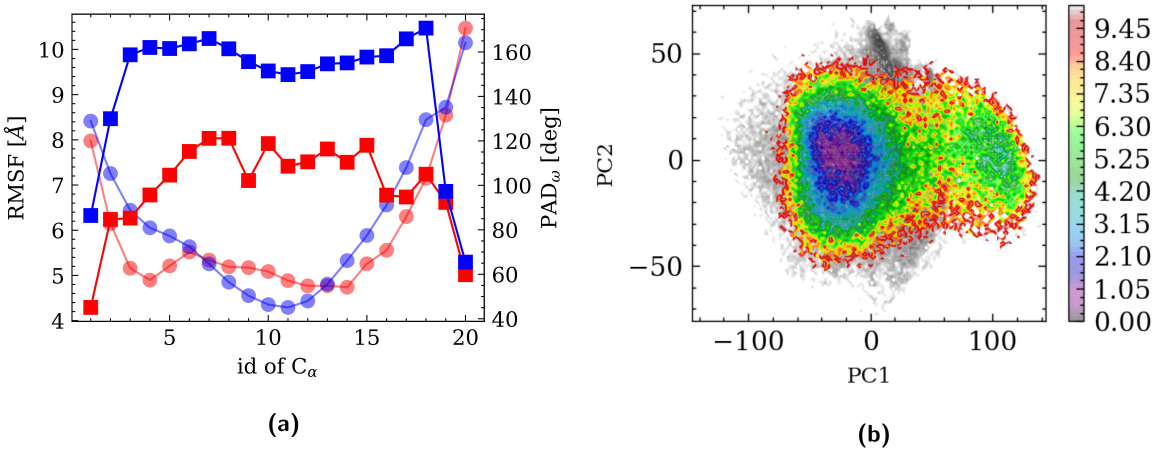
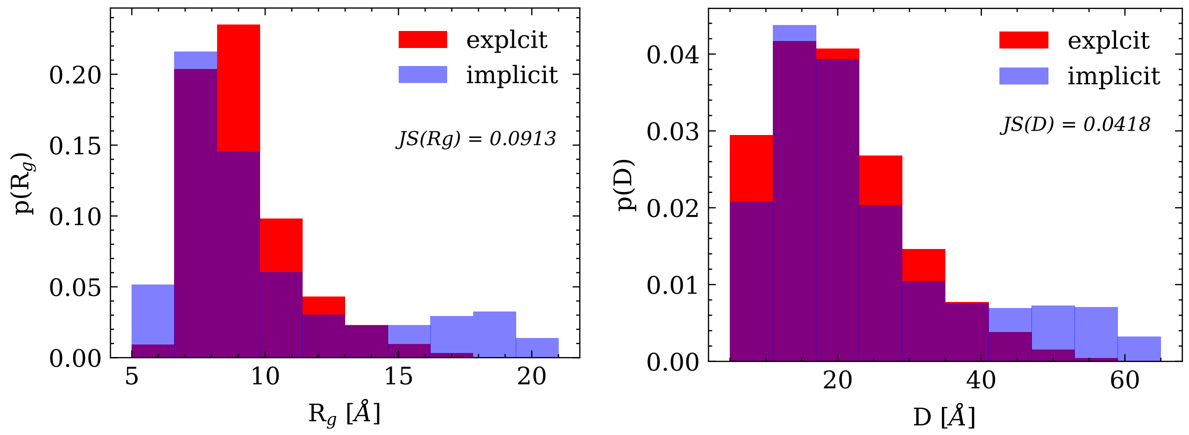


Figure 5.6: **(a):** RMSF (red circles from ES simulation, blue circles from IS one) calculated after the best alignment, performed with MDAnalysis; PAD$_\omega$ (red squares from ES simulation, blue squares from IS one), performed with a VMD script provided by the authors of [201]. **(b):** Plots of the FEP calculated on the first two PCs. As for the alanine dipeptide, the grey plot refers to the ES trajectory, while the coloured one to the IS. The ranges of values are comparable, but for the IS case there is a clear formation of two clusters of values, a feature that is not present in the ES case.

By looking ate the root mean square fluctuation (RMSF) values 5.6a, one can notice that even if there is a similarity between the explicit and implicit cases, there is a quantitative difference in the curves' behavior. In particular, the explicit case exhibits a double-well shape, whereas the implicit case displays a single-well shape. Intriguingly, the PAD$_\omega$ curves (see appendix for the definition) show a different trend with respect to RMSF, with the IS curve displaying very high values, nearly 180^o, indicating high variability in the dihedrals of the central carbon alpha, while the external ones show lower values in both cases. A potential explanation for this phenomenon is that the polymer's extremities behave like rigid rods that are highly agitated at the junction with the central core of the polymeric chain. Although this may seem unrealistic, it is worth noting that this quantity is better suited for structured proteins, rather than random coil polymers: the interpretability of these values is not guaranteed to be the same as for systems sampling well defined local minima of the free energy. Additionally, the values of the first two principal components (PCs) 5.6b are comparable, but, in this case, a double-well is also observed in the IS case, whereas the values are concentrated in a single cluster in the explicit solvent (ES) case. Even in this case, by the way, the use of PCA is not optimal: the principal components strongly depend on a successful alignment of the frames in the trajectory, but this process is not accurate when dealing with fast-fluctuating polymers.



(a)

(b)

Figure 5.7: **(a):** Histograms of the values of R_g assumed along the simulations. While the range of values covered by the two solvation model is very similar, it is notable the bimodal behaviour of the implicit solvent case, which is not present in the explicit solvent one. **(b):** Histograms of the values of the end-to-end distance assumed along the simulations. It is interesting to notice that the behaviour of the distribution for the implicit case resembles a lot the one of p(R_g).

Our analysis of the radius of gyration 5.7a and end-to-end distance 5.7b values revealed a consistent trend between them. Although the similarity between the two solvent models was confirmed by the Jensen-Shannon divergence, a slight bimodality was observed in the IS case that was not visible in the ES one. Visual inspection of the simulations confirms this trend, revealing more extended and stretched conformations in the IS case, a fact that we believe to be consistent with the bimodality of the distributions and the presence of two distinct clusters in the first two Principal Components (PC) plot: these are able to capture the presence of two main conformational basins representing a *stretched* and a *compact* class of structures. The fact that this aspect is not present in the ES histograms suggests that the forces arising from the MALIS model favor more extended conformations than the explicit solvent simulation.

5.3.3 ssRNA fragment

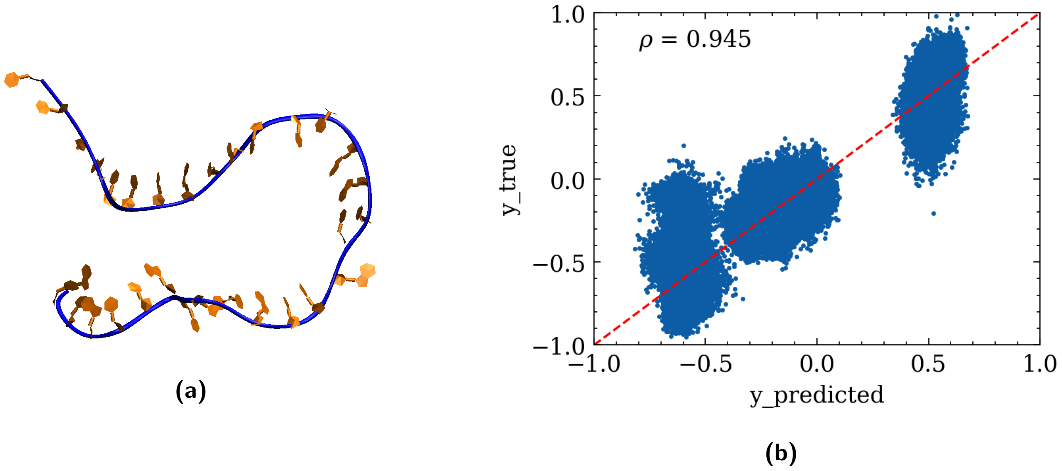


Figure 5.8: **(a):** VMD [39] NewCartoon visualization of Icosalanine in the last conformation after the 20ns of simulation used to train the MALIS neural network. **(b):** scatter plot of the normalized output of the NN $\{y_{predicted}\}$ after the last step of training, compared to the true output $\{y_{train}\}$ (equivalent to $\{y_{true}\}$) with the relative ρ.

The third system chosen as a benchmark for MALIS is another polymer, with a biochemically distinct nature from icosalanine. In this case, we aimed to test the method on a single-stranded RNA molecule consisting of 31 nucleotides and a total of 975 atoms. The reasons for this choice are primarily threefold: size, as the number of atoms is approximately 5 times that of icosalanine, allowing the method to be tested on systems that approach the size biological relevant molecules; practicality, as the sequence was selected and extracted from a much longer filament that was already available to us (viral RNA2 of CCMV, see next chapter) and that has a very simple secondary structure (predicted by the RNAfold webserver 5.9); biochemical nature, as the propensity of single-stranded RNA fragments to be both flexible yet with a conformational space with many local minima makes this type of system a good challenge to test the ability of the IS model to simultaneously predict the geometric and mechanical properties (shape, flexibility) as well as the chemical properties (formation of hydrogen bonds that determine the secondary and tertiary structure of the system).

By looking at the scatter plot that shows the results of the training process, 5.8b, one can immediately notice that the matching is worse than in the other cases already analyzed. Apart from the central region (values in the range [-0.4,0.1]), the other clusters of points are

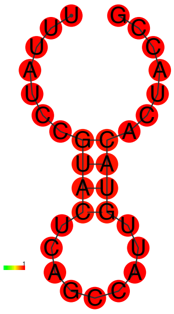
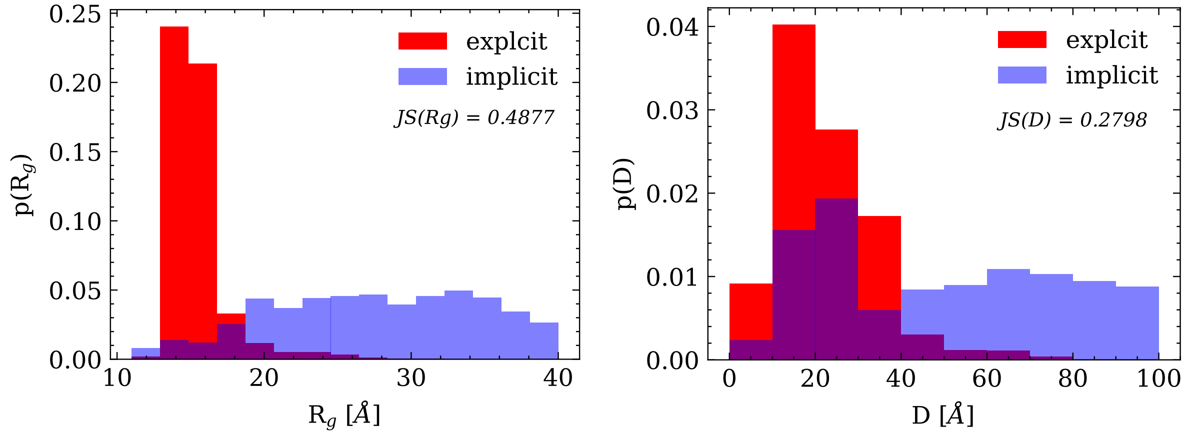


Figure 5.9: Secondary structure predicted by RNAfold [202] of the ssRNA simulated with GROMACS and MALIS: given the sequence, the system is expected to form few Watson-Crick bonds that cooperate to form a single duplex of 4 base pairs and an hairpin in correspondence to the central nucleotides of the sequence.

far from being ellipsoidal with the main axis oriented along the red dashed line (which is the ideal behaviour). The conclusion we can get from it is that the training process was not as successful as in the other cases: the causes could be several, from the size of the training dataset to the architecture of the neural network. About this last point, we remark that for simplicity of the implementation we decided to stick with the choice of using the same architecture for every system tested with the method: this is for sure a limitation of the predictive power of the algorithm, considering the variability of the amount of information carried by the symmetry functions that is necessary to learn in order for the proper forces to be calculated, from system to system.



(a) **(b)**

Figure 5.10: **(a):** Histograms of the values of R_g assumed along the simulations of the ssRNA fragment; **(b):** Histograms of the values of the end-to-end distance assumed along the simulations of the ssRNA fragment. Even if the superposition of the D histograms is slightly better than the R_g values, the discrepancy is evident: the ES simulation consists in one single run reaches a compact conformation of (metastable) equilibrium, while the IS runs are too short to reach something analogous.

To compare the IS simulations to the ES one, we firstly observed the histograms of the radius of gyration 5.10a and end-to-end distance 5.10b values, as for the icosalanine. The distributions differ a lot, particularly in the case of R_g values, as pointed out by the JS divergence value. The reason for this gap is evident also by looking at the values of R_g in time 5.11: for a system of this size, the simulation time used for the single parallel runs of MALIS (20ns each) is not enough for it to relax to a state of even metastable equilibrium. On the other hand, it is clear that in the $1\mu s$-long ES run the system had the chance to relax to some equilibrium configuration. To further explore this issue, we decided to complement the histogram analyses with additional analyses. In particular, we selected configurations sampled by MALIS that had a radius of gyration value compatible with the range of values most explored by the simulation with explicit solvent 5.10a: by convention and visual inspection, we chose the range $R_g \in [10, 20]$Å. These configurations (about a hundred) were compared with a subset of frames from the simulation in ES (uniformly sampled along the trajectory), from the first 200ns onwards.

For comparison, we chose to calculate the RMSD between each pair of frames, after optimal alignment taking into account all the atoms. This procedure does not claim to be informative in quantitative terms, but mainly aims to identify, where they exist, those frames that are sufficiently similar to allow the alignment process to return low RMSD values.

As can be seen from Figure 5.12a, the values obtained are all greater than 8Å, indicating a substantial average diversity between the positions of all the atoms, even for the most similar structures (or, in other words, those that were best superimposed in the alignment process). As an example, we report the licorice-style representation of the structures with the minimum RMSD among those calculated in the map 5.12b: it is evident that, given the diversity in shape and 3D structure, the alignment process is not capable of identifying similar structures with low RMSD values.

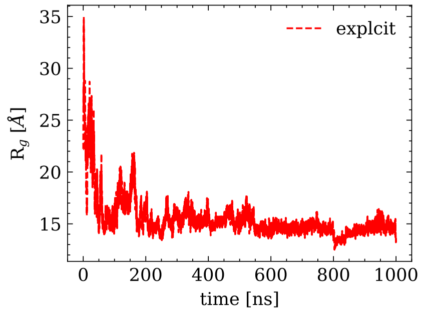
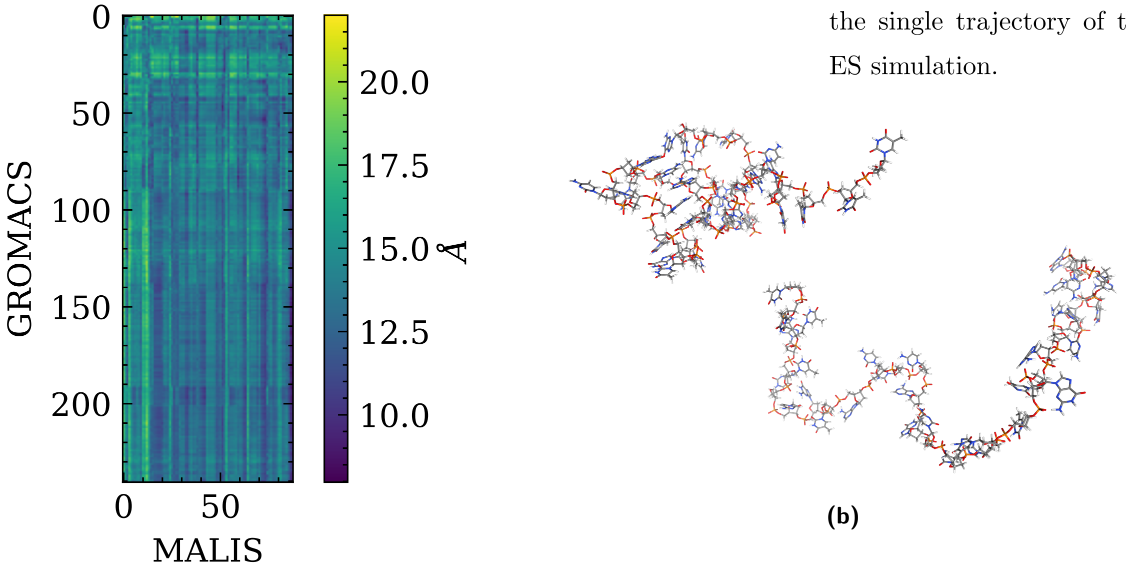


Figure 5.11: Radius of gyration as a function of the time frames along the single trajectory of the ES simulation.



(a)

(b)

Figure 5.12: **(a):** RMSD values calculated between those ES and IS configurations with $R_g \in [10, 20]$Å. **(b):** Licorice representation of the two most similar configurations taken from the RSMD map shown in **(a)**: the value of the RMSD is still $\sim$ 10 Å and in fact the two conformations are visibly different.

To complement the analysis of the simulations, we

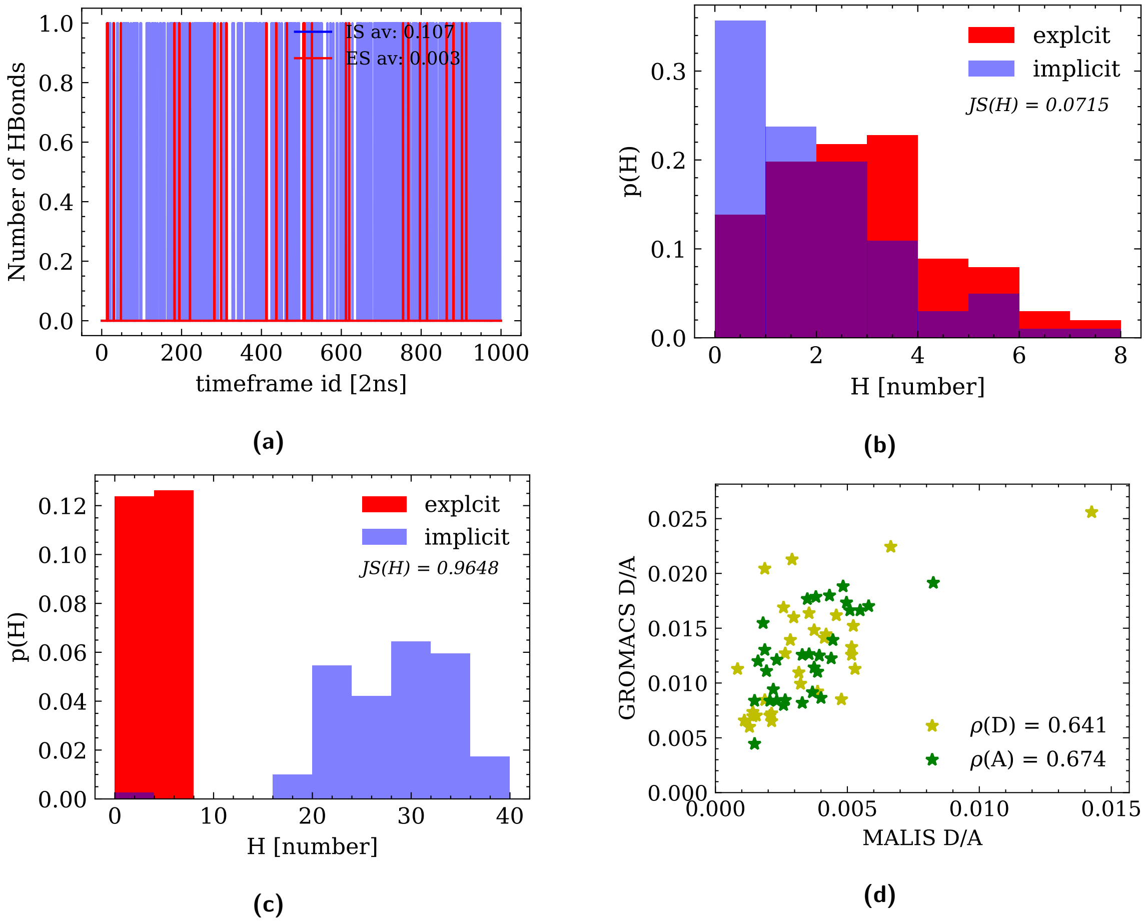
tracked the base pairings using BARNABA [203], a Python library specifically designed to perform analyses of atomistic structures and trajectories of RNA systems.
Interestingly, none of the base pairs pinpointed by it are Watson-Crick in nature, both for the ES and the IS case. As a consequence, the program is not able to construct any contact map based on canonical base pairings. Moreover, the pairs of nucleotides involved in the base pairs that are pinpointed don't ever match those predicted by the minimum free energy secondary structure of RNAfold (5.9). Visual inspection of the trajectories confirms that the configurations explored in both cases do not manifest any stable secondary structure.

5.4 Discussion

In this section we proceed in discussing three main points regarding the pros and cons of the MALIS model. In particular, we expand the considerations on the hydrogen bond network and persistence by performing other analyses. We also report the results obtained in applying the method to small protein [6] that are known to fold into stable structures computationally: in fact, the whole idea of making a machine learning based implicit solvent model was conceived for applications to protein folding and/or sampling of different free energy basins of even larger macromolecules, taking advantage of the theoretical gain in computational speed due to the integration of the many solvent degrees of freedom.

5.4.1 Hydrogen Bonds

As anticipated, hydrogen bonds formation is a key feature in testing an implicit solvent model. The reason is quite simple: in an explicit solvent simulation, water molecules constantly act as donors and acceptors of hydrogens, involving both their oxygen and their two hydrogen atoms. Moreover, once an hydrogen bond is established between one atom (being it an acceptor or a donor) of the solute and a water molecule, the chance of making another one with other solute atoms is essentially forbidden. This holds even in classical molecular dynamics simulations, where electrons are not taken into account explicitly: the way how we interpret a geometric disposition of atoms to form an hydrogen bond (see 5.16) and the real interactions involved (especially the excluded volume term in the Lennard-Jones non bonded potential) make a double involvement of a single acceptor into two hydrogen bonds very unlikely. The main consequence of this fact is that by integrating the degrees of freedom of solvent molecules, one has to bee sure that hydrogen bonds within atoms of the solute molecule itself are not overestimated. In our IS model, the assumption that we decided to make and to test was that the neural network would have been able to recognize this over-esteem, by looking ate the environmental information provided by the symmetry functions, and to adjust the energies (and eventually the forces) in order to decrease the probability of “self” hydrogen bonding formation of the solute.



(a)

(b)

(c)

(d)

Figure 5.13: **(a):** comparison between IS and ES runs about the formation of the only hydrogen bond predicted to potentially form in alanine dipeptide (more details in the text); **(b):** histogram of the frames with a given number of hydrogen bonds in the Icosalaine runs; **(c):** histogram of the frames with a given number of hydrogen bonds in the ssRNA runs; **(d):** scatter plot of the frequency of involvement of each residue whose atoms appear either as donors (D) or acceptors (A) in the ssRNA, directly comparing the ES and IS runs. **NB:** for histograms built on arrays of integers, each bin counts the frequency of appearance of the lower extreme (*e.g.* the bin plotted in the range $[0, 1]$ contains the p(0)).

The quantitative analysis of hydrogen bond formation in our simulations has been carried out with the MDAnalysis Python package. In particular, we tracked the number of hydrogen bonds per each frame in each trajectory, letting the algorithm to select *a priori* those atoms suitable for being donors/acceptors. In the case of alanine dipeptide, the pair of D and A is one: the nitrogen in the N-terminal and the Oxygen in the C-terminal of the small peptide. This pair's hydrogen bonding is in fact responsible for the kinetic trap that leads to the bad sampling of the dihedral space. By looking at 5.13a one can notice that the fraction of frames with the hydrogen bond is much higher in the IS simulation (indicated with *IS av* or *ES av* in the figure). For the icosalanine and the ssRNA simulations we decided to look at the histograms that track the fraction of frames with a given number of HBs. In the case of icosalanine 5.13b, the IS shows a higher number of frames with 1 or 2 HBs, while the ES histogram is more populated between 2 and 4. Here the HBs are expected to form involving atoms of the backbone, which are in turn responsible for the formation of the secondary structure motifs in proteins in general, and in particular in a potential α-helix in this system [204]. The major discrepancy is actually highlighted in the analysis of the ssRNA system 5.13c. There we have two very highly populated columns in the lowest-value range for the ES, while the distribution is clearly shifted to higher values in the IS case. This shows that the number of HBs in the MALIS simulation has been largely overestimated. However, we wanted to refine this analysis, to see if at least there is correlation between the involvement of a specific atom as donor/acceptor in the ES and the IS simulation. To do so, we tracked the fraction of frames where each donor/acceptor atom was involved in an HB, we attributed those to the relative nucleotide (lowering the resolution of HB participation to the single nucleotides: each star in 5.13d is related to a single nucleotide) and we calculated the Pearson's coefficient of this fraction related to the IS and the ES simulation. The result shows a good correlation: we can deduce that even if the HBs are overestimated, at least the atomistic interactions in the MALIS model preserve the proper chemical specificity of donor/acceptor along the chains, which participate to the HBs.

In conclusion, as we already anticipated and as these analyses support, the MALIS model apparently is not able to compensate for the absence of water molecules, where their presence is essential to keep the right balance in hydrogen bonding formation. One future attempt to improve this aspect will certainly be to try to train the MALIS with also G_5-type SF: one of the missing bricks for the HB prediction in fact could be the geometrical information on the angular disposition of atoms (in other words including an effective 3-body term).

5.4.2 Structured vs Unstructured molecules

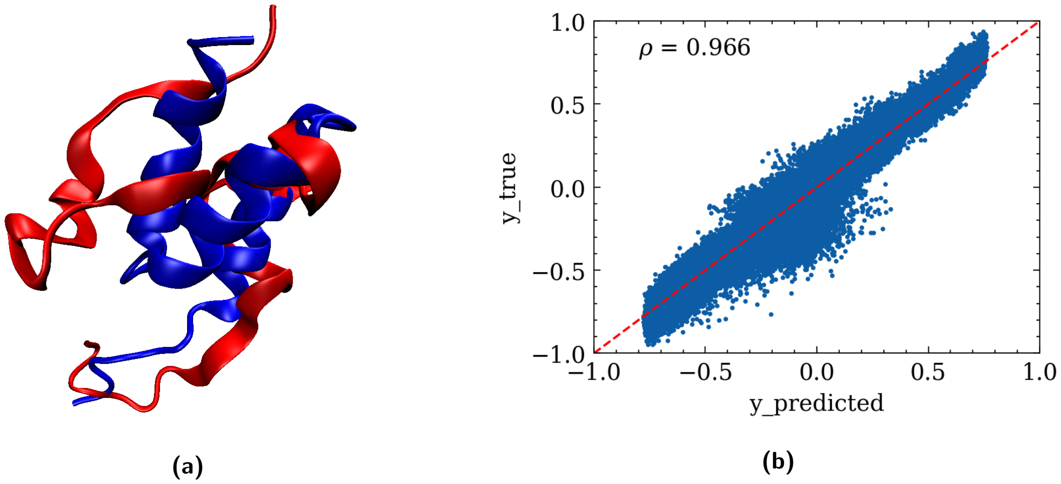


Figure 5.14: **(a):** starting configuration (in blue) and configuration after $\sim 1ps$ of MALIS simulation (in red); **(b):** scatter plot of the normalized output of the NN $\{y_{predicted}\}$ after the last step of training, compared to the true output $\{y_{train}\}$ (equivalent to $\{y_{true}\}$) with the relative ρ.

In addition to testing the hybrid deep learning and molecular dynamics approach on random coil polymers, we also tested the method on structured proteins by training on trajectories of both folded and unfolded states. We chose to use two small, fast-folding proteins, 1PRB and 2JOF. Interestingly, our training/testing approach yielded excellent results even when mixing datasets, coming from different basins of the free energy profile. For example, the network was able to predict the solvation energy values of unfolded conformations using a network trained on a folded trajectory, and vice versa. However, a major issue arose when applying the MALIS model in the molecular dynamics simulations of the structured proteins. Within just a few tens of femtoseconds, the folded protein became de-structured, rendering the results of the simulations unreliable (an example of it is shown in figure 5.14a). This behaviour appeared also using a network trained on a dataset from a simulation of the already folded protein (native state). One possible explanation to this counter-intuitive behaviour could be that we are over-fitting the potential in the small region explored by the system in the short simulation used for training. However, this could not explain the success in training the network with a dataset obtained by a simulation of the *e.g.* unfolded state, and successfully predict the energies of the folded dataset. Further work is needed to address this challenge, and to explore the limits and capabilities of the hybrid deep learning and molecular dynamics approach for studying structured proteins.

One interesting observation from our results is the discrepancy between the excellent results obtained during training and the weaknesses observed during testing. This is particularly striking when it comes to fast-folding proteins, where the model's predictive power is limited. Another intriguing finding from our training process is the presence of clusters of energy values, ranging from -1 to +1, which gradually merge to form a single ellipse as the system size increases (see 5.3b, 5.5b, 5.8b, 5.14b). This phenomenon is especially prominent in the case of fast-folding proteins, where the distribution becomes essentially continuous. One potential improvement we consider is to train the model to learn forces instead of energies, which could preserve more directional information and lead to more consistency between the predictive power of the instantaneous values of the observables and the impact of them on the dynamics of the systems. In order to to so, one idea would be to define relative coordinate systems around each atom and train the forces projected onto those systems to maintain rotational and translational invariance. These future perspectives could enhance the model's performance and applicability.

5.4.3 Computational Gain and Speed up

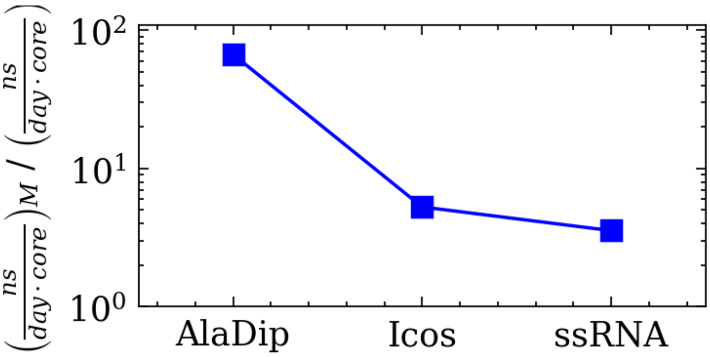


Figure 5.15: On the y axis (logarithmic scale): ratio between the *ns* per *day* and per *core* (the number of cores used is > 1 only in the GROMACS runs) of MALIS (numerator) and ES (denominator) simulations. The higher the ratio, the higher is the computational gain obtained by using the MALIS on that specific system (labelled on the x axis).

In this paragraph, we discuss a direct comparison of the MALIS method with GROMACS simulations for systems with different numbers of atoms. As we report in the appendix, a rough estimated ratio between the simulation times of an explicit and an implicit solvent model is approximately 10^2. This estimate, however, does not take into account many facts, such as the speed of the algorithms used to integrate the equations of motions and those used to calculate non-bonded interactions between molecules, which is the most expensive process in molecular dynamics simulations. Specifically, in our case we compare simulations made in LAMMPS with simulations made in GROMACS, which is highly optimized for water calculations, allowing it to scale better as water molecules are added. Additionally, in GROMACS the PME algorithm is used, which scales better ($\mathcal{N} \cdot \log(\mathcal{N})$) then the brutal cut-off LJ and Coulomb terms. So, given this facts, the comparison between the efficiencies of simulations is to be considered qualitative: unfortunately, a more unbiased comparison (using LAMMPS with cut-off non-bonded terms in ES simulations) would have been too expensive to perform sufficiently long runs for benchmarking purposes. Nevertheless, we compared the simulation times of the ES and IS simulations of the 3 different systems reported in the applications section. The total number of atoms in the simulation boxes are: 22 (IS) and 2344 (ES) for alanine dipeptide; 203 (IS) and 46682 (ES) for the icosalanine; 975 (IS) and 105783 (ES) for the ssRNA fragment. The ratios of the simulation times are plotted in 5.15 and shows a monotonically decreasing trend. The two main reasons we can give for this trend are: the increase of the cutoff radius used to calculate the SFs in the icosalaine and ssRNA (2nm, while it is 1nm in the alanine dipeptide case) and increase in the scaling quality of GROMACS with the tested sizes of the systems, using 48 cores as we did. Anyway, in general we can state that the MALIS performances remain good with respect to the GROMACS ones, considering that the implementation we did in LAMMPS was not thought to be efficient but rather it aimed for test and validation of the algorithm.

5.5 Perspectives and Conclusions

The current project presents several limitations that need to be addressed in order to improve its performance. Firstly, the workflow is convoluted and requires the use of different software with few automated steps. This makes the process more time-consuming and prone to errors. Additionally, the implementation is not optimized, and the tests conducted were done in serial simulations on a single core, making it challenging to parallelize, especially for larger systems. The quality of the results obtained, when compared to ES models, is also poor, and the reasons for this could be due to intrinsic limitations in the model and/or the size of the dataset used for training. Furthermore, there are doubts about the method itself, such as whether PME is suitable for generating the dataset and whether SF of type 1-2 is sufficient. The PME method, in fact, relies on the knowledge of both short-range and long-range (or low-frequency) information to be addressed, while the SF bring only short-range information. This way, it is hard to expect that the network's predicted energy correlates well with the PME energy, given this lack of input information.

On the other hand, the project also has some significant strengths. The code runs much faster when compared to GROMACS. The code in LAMMPS is relatively easy to use and, once formatted correctly, the use of the pair-style is as simple as the others. Finally, although the project has not shown much efficiency in predicting dynamics, it could be beneficial for estimating solvation free energy from unknown structures, an aspect that is very useful in many fields of applications, such as drug design. In this case, a network trained on several different proteins (i.e., a generalized network) would be needed, but from our results this seems feasible given the transferability shown, for example, from unfolded to folded proteins in the 1PRB. In the future, we plan to extend the method to increase both quality and performance, with the aim of obtaining an IS model that is a good compromise between chemical accuracy and computational speedup, with the goal of reducing the gap between the timescales attainable with simulations and experimental techniques.

5.5.1 Comparison with the work of Noé and coworkers

We report here a comparison summarizing pros and cons of the MALIS model and the ISSNet [190]. We divide them in 4 categories.

1. quality of the results (predictive power compared to ES): concerning this aspect, the work of Noé and coworkers is clearly effective, in both system under investigation (alanine dipeptide, chignolin); not only they are able to reconstruct the proper FEP in the dihedral space of alanine dipeptide, but they also manage to predict the proper melting temperature of chignolin.

2. computational cost: as reported in Figure S5 of [190] Supplementary Materials, their model is faster than ES simulations only for alanine dipeptide and with certain choices of batch size, while it is always slower in chignolin simulations (which has a number of atoms comparable with icosalanine); on the other hand, as we showed in figure 5.15, we are able to over-perform GROMACS in every case tested.

3. interpretability of the model: although the complexity of their graph neural network is higher than ours, we cannot claim that our network can be easily interpreted in physical terms; on the contrary, by studying the use of inputs such as atom types or the atomic partial charges, they at least exhibit the relevance of retaining these information. From

this point of view, we can only state that combining radial distribution quantities alone (like the SF G_2) is not enough: we can deduce that a mean field description of the solvent effect is by nature many-body, which is reasonable.

4. accessibility of the model: in our case, we prepared a *pair style* that can be compiled in the LAMMPS package (v2018); in order for a new user to play with the MALIS, however, all the stages before the simulations in LAMMPS are still intricate: from the atomistic simulations in ES to the creation of the parameters files of the network, the pipeline is still newborn and based on a multiplicity of different scripts. We were not able to find any reference to github repositories or similar of ISSNet in [190], even if in the previous works on SchNet they refer to a github and a well-written documentation can be easily found on the web.

5.6 Appendix

NN derivative of the Output (with respect to the Input)

In this section we give some nomenclature to the variables involved into the NN calculations. After that, we dig into performing the derivative of the (single) output of the network with respect to one of the inputs. This calculation are useful in order to extract the mean field forces from the potential of mean force that is calculated by the NN.

We focus on a fully-connected neural network with a generic number of inputs. In our example, the NN has 3 hidden layers with, respectively, N_1, N_2 and N_3 neurons. We call the output of a generic neuron $y_i^{(n)}$, where (n) represents the specific n-th hidden layer and j indicates the j-th neuron of that specific layer. Each neuron is characterized by an activation function $f(z)$, where z is a combination of the input values that each generic neuron receives. More explicitly:

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \qquad y_i^{(n)} = f(z_i^{(n)}) = f\left(\sum_{k=1}^{N_{n-1}} w_{i,k}^{n,n-1} y_k^{(n-1)} + b_i^{(n)}\right) \tag{5.14}$$

Imagine now that we want to extract the derivative of the single output neuron, F, with respect to one of the input neurons, e.g. that labelled with the symbol x (and whose associated variable is x). In other words, we want to know $\frac{\partial F}{\partial x}$, where F is a chain composition of functions that eventually would directly depend on x. The first step to do is to express F in terms of the outputs of the neurons belonging to the third hidden layer:

$$F(y_1^3, \ldots, y_{N_3}^3) = f\left(\sum_{k=1}^{N_3} w_{F,k}^{4,3}\, y_k^{(3)} + b_F^{(4)}\right) \tag{5.15}$$

where we introduced the weight $w_{F,k}^{4,3}$, related to a connection from the k-th neuron of the 3rd hidden layer to the "F-th" neuron of the 4th layer (which in this case is not hidden, because is the single-neuron output) and the bias $b_F^{(4)}$, of the 4th (output) layer of the "F-th" neuron (the only output neuron). We can perform the derivative of this quantity to obtain the following:

$$\frac{\partial F}{\partial x} = \frac{df}{dz_F}\left[\sum_{k=1}^{N_3} w_{F,k}^{4,3}\, \frac{\partial y_k^{(3)}}{\partial x}\right] \tag{5.16}$$

where we also introduced the notation $\frac{df}{dz_F} := \frac{df}{dz}$. In the specific case of $\tanh(z)$ this derivative can be express as function of the function itself:

$$\frac{d}{dz}\tanh(z) = \frac{d}{dz}\frac{e^z - e^{-z}}{e^z + e^{-z}} = 1 - [\tanh(z)]^2 \tag{5.17}$$

so that for the output neuron, we have $\dfrac{df}{dz_F} \equiv 1 - F^2$. This is nice because we begin to see that these derivatives can be expressed in terms of quantities that we have under control when we use the NN.

The next step is to calculate the derivative of each $y_k^{(3)}$ with respect to x. As one can imagine, this will generate a cascade of sums of weights and derivatives. Each hidden layer will have its own sum of products of weights and other derivatives. Moreover, new derivatives of the activation function will appear and these will be evaluated at the proper values of z. So for example:

$$\frac{\partial y_k^{(3)}}{\partial x} = \frac{df}{dz_k^{(3)}} \left[\sum_{i=1}^{N_2} w_{k,i}^{3,2} \frac{\partial y_i^{(2)}}{\partial x} \right] \tag{5.18}$$

where now $\dfrac{df}{dz_k^{(3)}} = \left.\dfrac{df}{dz}\right|_{z=z_k^{(3)}} \equiv 1 - (y_k^{(3)})^2$.

This can be done also for the 1st hidden layer and finally for the single input we took into account. For it, there will be no sum (because the “0-th” layer has $N_0 = 1$ which is the neuron that shoots the value g_α) and we can wrap up the final result:

$$\begin{aligned}\frac{\partial F}{\partial g_\alpha} = [1 - F^2] \Bigg\{ \sum_{k=1}^{N_3} w_{F,k}^{4,3} \left(1 - (y_k^{(3)})^2\right) \Bigg[\sum_{j=1}^{N_2} w_{k,j}^{3,2} \cdot \\ \cdot \left(1 - (y_j^{(2)})^2\right) \left(\sum_{i=1}^{N_1} w_{j,i}^{2,1} \left(1 - (y_i^{(1)})^2\right) w_{i,\alpha}^{1,0} \right) \Bigg] \Bigg\}\end{aligned} \tag{5.19}$$

Output gradient in terms of matrix products

It can be useful to reformulate the product in (5.19) as a product of matrices; a good reason would be, for example, the need to implement this calculation in a Python script, which is optimized for matrix algebra and performs poorly with *for* loops.

If we introduce the diagonal matrix *diag* $\left[\left(1 - (y_1^{(\alpha)})^2\right), \ldots, \left(1 - (y_{N_\alpha}^{(\alpha)})^2\right) \right]$, made by the derivative of *tanh* for each neuron output, at each layer α, we can write:

$$\nabla_{\mathbf{g}} F = [1 - F^2] \cdot \begin{bmatrix} w^{4,3}_{F,1} \\ \vdots \\ w^{4,3}_{F,N_3} \end{bmatrix} \cdot \begin{bmatrix} \left(1 - (y^{(3)}_1)^2\right) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \left(1 - (y^{(3)}_{N_3})^2\right) \end{bmatrix} \cdot$$
$$\cdot \begin{bmatrix} w^{3,2}_{1,1} & \dots & w^{3,2}_{1,N_2} \\ \vdots & \ddots & \vdots \\ w^{3,2}_{N_3,1} & \dots & w^{3,2}_{N_3,N_2} \end{bmatrix} \cdot \begin{bmatrix} \left(1 - (y^{(2)}_1)^2\right) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \left(1 - (y^{(2)}_{N_2})^2\right) \end{bmatrix} \cdot \dots \tag{5.20}$$

and this is the general formula of the gradient of the output with respect to the vectorial input $\mathbf{g}$, which in our case will be a vector of scalar symmetry functions $\mathbf{g} = [g_1, \dots, g_{N_0}]$.

Derivatives of the Symmetry Functions

In this paragraph we report the analytical calculations of the derivatives of the SF with respect to the Cartesian coordinates of the atoms. We also introduce a third kind of SF, which we did not use in the applications reported here, but that are reccuring in other works based on the Beheler-Parrinello-like NN potentials. This kind is named $G^{(i)}_5$ and requires the introduction of a last new parameter ξ and its defined as follow:

$$G^{(i)}_5 := 2^{1-\xi}\, q_i \sum_{\{j\}_i} \sum_{\{k\}_i} q_j\, q_k\, \mathcal{F}_c(R_{ij})\, \mathcal{F}_c(R_{ik}) \cdot [\cos\theta_{ijk} + 1]^{\xi} \tag{5.21}$$

In the calculation of the forces, we anticipated that we need to calculate the terms $\dfrac{\partial g_\alpha}{\partial x_i}$. Now we know that g_α can have the form of one of the Gs defined above, so that this derivative will be of one of the three different kinds of SF. All we have to do so is to perform the derivative of these Gs with respect to a generic Cartesian coordinate of a generic i-th atom:

$$\frac{\partial R_{ij}}{\partial x_i} = \frac{(x_i - x_j)}{R_{ij}}, \qquad \frac{\partial f_c(R_{ij})}{\partial x_i} = -\frac{\pi}{2R_c} \sin\left(\pi \frac{R_{ij}}{R_c}\right) \cdot \frac{\partial R_{ij}}{\partial x_i} \tag{5.22}$$

$$\frac{\partial G_1^{(i)}}{\partial x_i} = q_i \sum_{\{j\}_i} q_j \, \frac{\partial f_c(R_{ij})}{\partial x_i} \tag{5.23}$$

$$\frac{\partial G_2^{(i)}}{\partial x_i} = q_i \sum_{\{j\}_i} q_j \, \frac{\partial \mathcal{F}_c(R_{ij})}{\partial x_i} = q_i \sum_{\{j\}_i} q_j \, \left\{ e^{-\eta(R_{ij}-R_s)^2} \left[\frac{\partial f_c}{\partial x_i} - 2\eta(R_{ij} - R_s) f_c(R_{ij}) \frac{\partial R_{ij}}{\partial x_i} \right] \right\} \tag{5.24}$$

$$\frac{\partial G_5^{(i)}}{\partial x_i} = 2^{1-\xi} \, q_i \sum_{\{j\}_i} \sum_{\{k\}_i} q_j \, q_k \, \frac{\partial}{\partial x_i} \left[\mathcal{F}_c(R_{ij}) \, \mathcal{F}_c(R_{ik}) \cdot (\cos\theta_{ijk} + 1)^{\xi} \right] = 2^{1-\xi} \, q_i \sum_{\{j\}_i} \sum_{\{k\}_i} q_j \, q_k \cdot \tag{5.25}$$

$$\cdot \left\{ (\cos\theta_{ijk} + 1)^{\xi} \left[\frac{\partial \mathcal{F}_c(R_{ij})}{\partial x_i} + \frac{\partial \mathcal{F}_c(R_{ik})}{\partial x_i} \right] + \mathcal{F}_c(R_{ij}) \, \mathcal{F}_c(R_{ik}) \frac{\partial}{\partial x_i} \, (\cos\theta_{ijk} + 1)^{\xi} \right\} \tag{5.26}$$

where the last missing part is given by:

$$\frac{\partial}{\partial x_i} \left(\frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{R_{ij} R_{ik}} + 1 \right)^{\xi} = \xi \left(\frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{R_{ij} R_{ik}} + 1 \right)^{\xi - 1} \frac{\partial}{\partial x_i} \left(\frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{R_{ij} R_{ik}} \right) \tag{5.27}$$

$$\frac{\partial}{\partial x_i} \left(\frac{\mathbf{R}_{ij} \cdot \mathbf{R}_{ik}}{R_{ij} R_{ik}} \right) = \frac{1}{(R_{ij} R_{ik})^2} \Big[(2x_i - x_j - x_k) \, (R_{ij} R_{ik}) - \mathbf{R}_{ij} \cdot \mathbf{R}_{ik} \left(\frac{\partial R_{ij}}{\partial x_i} R_{ik} + R_{ij} \frac{\partial R_{ik}}{\partial x_i} \right) \Big] = \tag{5.28}$$

$$= \left[\frac{(2x_i - x_j - x_k)}{R_{ij} R_{ik}} - \cos\theta_{ijk} \left(\frac{1}{R_{ij}} \frac{\partial R_{ij}}{\partial x_i} + \frac{1}{R_{ik}} \frac{\partial R_{ik}}{\partial x_i} \right) \right] \tag{5.29}$$

Of course, this can be done for every Cartesian coordinate of interest in order to find the component acting on the atom of interest.

We conclude by stressing the fact that these calculations require knowledge of the Cartesian coordinates for every neighbor of the atom on which the force is acting.

Molecular dynamics simulation details

Molecular dynamics simulations have been performed using the softwares GROMACS 2018 [40] and our modified version of LAMMPS (12Aug18) [205]. The molecules were modeled with the Amber99sb-ildn force field [41]. For the explicit solvent simulations the TIP3P model [42] was used for water molecules; sodium and chloride ions were added at a concentration of 0.15 M, and balanced so as to neutralize the charge in the simulation box. In the GROMACS simulations, all systems were energy-minimized for 1000 steps by steepest descent. The solvent was then equilibrated for 1 ns (500 ps of NVT and 500 ps of NPT ensemble simulations) with positional restraints on the protein heavy atoms, using a force constant of 1000 kJ·mol^{-1}·nm^{-2}. During the NPT, the systems were isotropically pressure-coupled at 1 bar with a coupling constant of 2.0 ps, using the Parrinello-Rahman barostat [43]. In the production run, solute and solvent were coupled separately to a 300 K heat bath with a coupling constant of 0.1 ps, using the velocity-rescaling thermostat [165] (same as equilibration). Application of the LINCS [166] algorithm on hydrogen-containing bonds allowed for an integration time step of 2 fs. Short-range electrostatic and Lennard–Jones interactions were calculated within a cut-off of 1.0 nm, and the neighbor list was updated every 10 steps. The particle mesh Ewald (PME) method was used for the long-range electrostatic interactions [167], with a grid spacing of 0.12 nm. In the LAMMPS simulations, however, we were bound to use a cut-off description for the full electrostatic contribution. As already anticipated in the Application section, we kept the cut-off high enough (depending on the system) to minimize artifacts arising from cut-off long range Coulomb potentials. As initial configurations for the LAMMPS parallel runs, we extracted (uniformly) frames from the short runs performed to build the training dataset.

LAMMPS implementation

To create the *pair style* for the MALIS, in LAMMPS, we modified the *lj/cut/coul/cut* file, present in original the code. Roughly speaking, it is structured in a series of functions, one of which calculates the energies and forces of the non-bonded terms in a cut-off scheme. To do so, for the *i*-th atom in the system the function calculates the pairwise terms using three for cycles: one for those atoms directly bonded to it, one for those atoms that share a 3-body angular bonded interaction with it and the last one that involves those atoms that are involved in a 4-body bonded interaction and every other non-bonded atom to it, within the cut-off. We inserted the calculation of the $G_{2,\alpha}^{(i)}$ and the $\frac{\partial G_{2,\alpha}^{(i)}}{\partial x_i}$ at the level of each one of these for loops to be sure that every atom of the surrounding region (within the SF cutoff, which coincides with the pairwise non-bonded interactions) is considered. After collecting the values of the SFs and their derivatives, we defined a new function that reads a *.csv* file containing the weights of the already trained neural network, and uses it to compute the force components $\left[f_x^{(i)}, f_y^{(i)}, f_z^{(i)}\right]$ as scalar products with $\frac{\partial V^{(i)}}{\partial g_\alpha}$. Then the forces are added to the variables that are sent to the integrator of Newton's equations.

How to deal with normalized datasets

What we saw in equations (5.11),(5.12) is the recipe to directly calculate the forces (in Cartesian components) acting on each atom, using both numbers that comes from the NN and the values of the derivatives of the g's. The unit of measurement of the force component will be coherent with those of $V_{\text{eff}}^{(i)}$ and x_i and, indirectly, those of the g's. This is an important fact, because the dataset used in the training is composed by normalized values of inputs/outputs, and this is reflected in the calculation of $\dfrac{\partial V_{\text{eff}}^{(i)}}{\partial g_\alpha}$. The "true" values and the normalized ones are related by two affine trasformations:

$$g_{\alpha,\text{nor}} = \frac{2}{g_{\max} - g_{\min}} g_\alpha - \frac{g_{\max} + g_{\min}}{g_{\max} - g_{\min}} = A \cdot g_\alpha + B \quad \Rightarrow \quad \partial g_{\alpha,\text{nor}} = A \cdot \partial g_\alpha \tag{5.30}$$

$$V_{\text{eff}}^{(i)} = \frac{V_{\max} - V_{\min}}{2} V_{\text{eff,nor}}^{(i)} + \frac{V_{\max} + V_{\min}}{2} = C \cdot V_{\text{eff,nor}}^{(i)} + D \quad \Rightarrow \quad \partial V_{\text{eff}}^{(i)} = C \cdot \partial V_{\text{eff,nor}}^{(i)} \tag{5.31}$$

The NN operates with the normalized versions of these two quantities, so if we want to calculate $\dfrac{\partial V_{\text{eff}}^{(i)}}{\partial g_\alpha}$ we have to keep in mind the following:

$$\frac{\partial V_{\text{eff}}^{(i)}}{\partial g_\alpha} = A \cdot C \cdot \frac{\partial V_{\text{eff,nor}}^{(i)}}{\partial g_{\alpha,\text{nor}}} = \frac{V_{\max} - V_{\min}}{g_{\max} - g_{\min}} \cdot \frac{\partial V_{\text{eff,nor}}^{(i)}}{\partial g_{\alpha,\text{nor}}} \tag{5.32}$$

and be aware that the NN calculates $\dfrac{\partial V_{\text{eff,nor}}^{(i)}}{\partial g_{\alpha,\text{nor}}}$. Wrapping up everything, the x-component of the force acting on the i-th atom can be directly calculated by:

$$F_x^{(i)} = -\frac{V_{\max} - V_{\min}}{g_{\max} - g_{\min}} \sum_\alpha \frac{\partial V_{\text{eff,nor}}^{(i)}}{\partial g_{\alpha,\text{nor}}} \frac{\partial g_\alpha}{\partial x_i} \tag{5.33}$$

These calculations are important since the normalization factor is dataset-dependent, forcing us to insert it as one of the values to be passed in the input script of the LAMMPS run.

Estimated Computational Gain

In MD simulations, each time step requires the update of the value of each degree of freedom explicitly taken into account in the system (i.e. the atomic positions and velocities). It is then clear that, given $\mathcal{N} \gg N$ (where $\mathcal{N}$ is the number of solvent atoms and N is the number of solute atoms), an implicit solvent model would result into a huge speed up for the simulation.

Therefore, in this paragraph we want to qualitatively compare this gain to the cost of applying our specific implicit solvent framework, just to have an idea.

For a system with $N+\mathcal{N}$ DOFs, in each time frame the distance matrix (required to calculate the forces for the integration of the equations of motions) will be made by $(N+\mathcal{N})\times(N+\mathcal{N}-1)/2$ independent terms, but we know that if we work e. g. with cut-off Coulomb and LJ potentials, this number is lower. Let's call $\overline{N+\mathcal{N}-1} < (N+\mathcal{N}-1)$ the approximate average number of neighbors for the atoms in the system. The real number of terms required for the update of the positions of the atoms becomes then $C_{\text{sol}} := (N+\mathcal{N}) \times (\overline{N+\mathcal{N}-1})/2$. By assuming that the number of solvent atoms is 10^a times the number of protein atom, $\mathcal{N} \simeq 10^a \cdot N$, and by assuming that $\overline{N+\mathcal{N}-1} \sim 10^b$, this number is around $C_{\text{sol}} \sim 10^{a+b} \cdot N$. In our framework we have instead $C_{\text{malis}} \sim (10^c \cdot 10^d) \cdot N$ terms, where 10^c is the number of SF we choose to use and $10^d \simeq \overline{N-1}$, because we are needed to construct a constrained distance matrix of the protein to calculate the SF. The speed-up is then:

$$S := \frac{C_{\text{sol}}}{C_{\text{malis}}} = \frac{10^{a+b}}{10^{c+d}} = 10^{(a+b)-(c+d)} \tag{5.34}$$

In a realistic case where $a = 2$, $b = 2$, $c = 1$ and $d = 1$, for example, the speed-up would result in $S \simeq 10^2$.

Observables used in the Analysis

- **Radius of gyration:** the radius of gyration R_g is a scalar geometrical feature of the system. It is calculated for each configuration $\{\mathbf{r}(t_j)\}_M$ of the trajectory. Given the center of mass:

$$\mathbf{r}_c(t_j) := \frac{\sum\limits_{i=1}^{N} m_i \mathbf{r}_i(t_j)}{\sum\limits_{i} m_i} \tag{5.35}$$

  of the configuration at time t_j, it is defined as the square root of the mean square displacement of each atomic position with respect to $\mathbf{r}_c(t_j)$:

$$R_g[\{\mathbf{r}(t_j)\}_M] := \sqrt{\frac{\sum\limits_{i=1}^{N} m_i \|\mathbf{r}_i(t_j) - \mathbf{r}_c(t_j)\|^2}{\sum\limits_{i=1}^{N} m_i}} \tag{5.36}$$

- **PAD$_\omega$:** similarly to RMSF, Caliandro *et al.* [201] introduced another quantity, called PAD$_{\omega_r}$, which quantifies the variability of the values of the dihedral angles of each residue r along a trajectory. PAD$_{\omega_r} \in [0, 180]$ and is 0 when the dihedral angles do not vary at all along the trajectory and is equal to 180 if the values of Φ_r and Ψ_r are randomly sampled. In formulae, we have:

$$\omega_r(t) := \Phi_r(t) + \Psi_r(t), \text{ for } t = 1, \dots, T \tag{5.37}$$

$$R_{k,\omega_r} := \frac{1}{T}\left|\sum_{t=1}^{T} e^{ik\omega_r(t)}\right| \quad \Rightarrow \quad \mathrm{CS}_{\omega_r} := \frac{1 - R_{2,\omega_r}}{2R^2_{1,\omega_r}} \tag{5.38}$$

$$\mathrm{PAD}_{\omega_r} := \frac{180}{\pi} \arccos\left(\frac{1 - \mathrm{CS}_{\omega_r}}{1 + \mathrm{CS}_{\omega_r}}\right) \tag{5.39}$$

- **End-to-end distance:** this quantity is simply defined as the distance between the positions of the first $\mathbf{r}_1(t)$ and the last $\mathbf{r}_N(t)$ atom of a polymeric chain at each frame:

$$D(t) = \|\mathbf{r}_1(t) - \mathbf{r}_N(t)\| \tag{5.40}$$

- **Hydrogen Bond Analysis:** although quantum by nature, the hydrogen bond formation can be modelled and described also in MD simulations, by looking at some geometrical dispositions of donor (D) atoms, hydrogen (H) atoms and acceptor (A) atoms. In this work we relied on the *HydrogenBondAnalysis* class implemented in the Python package MDAnalysis [45], using the default input parameters to deduce valid candidates for the determination of the presence/absence of one or multiple hydrogen bonds along the trajectory. Referring to figure 5.16, we used the following cutoffs to identify an hydrogen bond: $r_H \leq 1.2$ Å, $r_D \leq 3.0$ Å $\theta \geq 150^o$ and $q_A, q_D \leq -0.5$e (effective charges).

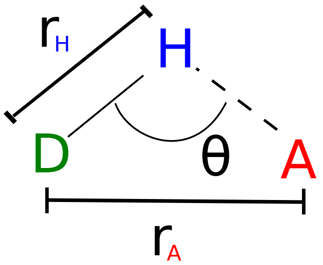


Figure 5.16: Simple scheme of an hydrogen bond: the donor and acceptor atoms are typically electronegative species (Oxygen and Nitrogen are the most common ones in biomolecules). The dashed line indicates the hydrogen bond.

It can take values in the interval $[0, 1]$, 0 being perfect matching and 1 being zero overlap (depending on the binning chosen for creating the histogram).

- **Pearson coefficient:** in this application, we used this quantity (indicated with ρ) as a measure of the correlation between the true and predicted values of normalized solvation energies during the training process of MALIS networks and to quantify the correlation between the involvement of nucleotides in hydrogen bonds in the ES and IS simulations. It is defined as follow:

$$\rho[\{y_{true}\}_i, \{y_{pred}\}_j] := \frac{\sum_i \left(y_{true}^{(i)} - \bar{y}_{true}\right)\left(y_{pred}^{(i)} - \bar{y}_{pred}\right)}{\sqrt{\sum_i \left(y_{true}^{(i)} - \bar{y}_{true}\right)^2} \sqrt{\sum_j \left(y_{pred}^{(j)} - \bar{y}_{pred}\right)^2}} \tag{5.41}$$

where $\bar{y}_{true}$ and $\bar{y}_{pred}$ indicate the averages of the raw data.

Chapter 6

A comprehensive Multi-resolution Study of a CCMV virion particle and of its constituents

6.1 Introduction

Viruses are enigmatic entities that have captured the attention of scientists and researchers for decades. These microscopic agents are a-cellular, consisting of genetic material (either DNA or RNA) encased within a protein coat called *viral capsid*. Other viruses posses another layer of coating, made of lipid bilayers: it is called *envelope* and it gives the virus more resistance; moreover, having a lipid coat allows the viruses to infect their target cells by causing the viral envelope and cell membrane to fuse. They lack the ability to carry out metabolic processes on their own and rely on host cells to replicate and propagate. Despite their small size, viruses can cause a wide range of diseases in plants, animals, and humans, making them crucial subjects of study to understand and combat infectious diseases, as the recent times demonstrated worldwide [206].

Among the vast variety of groups, plant viruses have generally a simpler structure, lacking of the lipid envelope. The mechanisms of infection and proliferation of plant viruses are intricate processes that involve many molecular interactions [207]. When a plant is infected, the virus attaches to specific receptors on the plant's cell surface, facilitating entry into the cell. Once inside, the viral genetic material takes control of the host's cellular machinery, directing it to replicate the virus's genetic material and synthesize viral proteins. These newly synthesized components then assemble into complete viral particles, which are released from the host cell to infect neighboring cells, further propagating the infection.

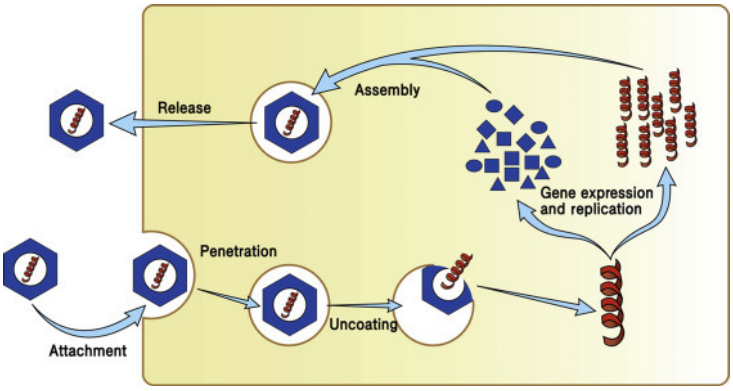


Figure 6.1: Description of a generic virus life cycle. Image taken from [207].

Although their threatening nature, the study and understanding of viruses led also to brilliant ideas on how to use them for beneficial applications. In fact viral capsids have garnered significant interest due to their unique properties and potential technological applications. Here are a few notable ones:

1. Nanoparticle-based Drug Delivery [208]: Viral capsids can be engineered to serve as nano-sized carriers for targeted drug delivery. By modifying the capsid's surface, scientists can attach specific ligands that bind to receptors on target cells. This allows precise drug delivery, reducing off-target effects and enhancing therapeutic efficacy.

2. Vaccine Development [209]: Viral capsids can be utilized as vaccine platforms. Non-infectious viral capsids can be engineered to display antigens from other pathogens, stimulating a targeted immune response. These virus-like particles (VLPs) offer a safe and effective means of vaccination without the risk of causing disease.

3. Nanoscale Imaging and Sensing [210]: The unique structural properties of viral capsids, including their defined sizes and shapes, make them ideal templates for nanoscale imaging and sensing applications. Capsids can be modified with fluorescent tags or other imaging agents, allowing researchers to track cellular processes or detect specific biomolecules.

4. Nanoelectronics and Nanomaterials [211]: Viral capsids' symmetrical and self-assembling nature has inspired research in nanoelectronics and nanomaterials. By engineering cap-

sids to encapsulate different materials, they can act as scaffolds for the creation of novel nanomaterials with unique properties.

5. Biocatalysis and Enzyme Encapsulation [212]: The interior of viral capsids can provide a confined and protected environment for enzyme encapsulation. This controlled environment enhances the stability and activity of enzymes, making them more efficient as biocatalysts for various industrial and medical applications.

6. Bioimaging and Diagnostics [213]: Viral capsids can serve as contrast agents in bioimaging techniques such as magnetic resonance imaging (MRI) and positron emission tomography (PET). Their unique properties enable improved visualization of specific tissues and organs in medical diagnostics.

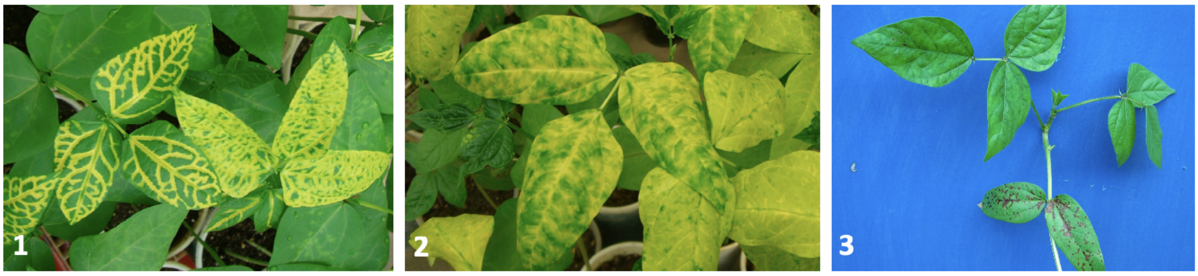


Figure 6.2: Symptoms elicited by CCMV infection in cowpea (1), yardlong bean (2) and mung bean (3). Image taken from the Supplementary Materials of [214].

One noteworthy plant virus that has piqued the interest of researchers is the Cowpea Chlorotic Mottle Virus (CCMV). Discovered in the mid-20th century [215], CCMV belongs to the Bromoviridae family and is known for infecting a variety of leguminous plants, including cowpea (Vigna unguiculata). It possesses a single-stranded RNA genome encapsulated within a robust icosahedral protein shell, forming its characteristic viral particle. One intriguing aspect of the CCMV genome is its multipartite nature. This means that instead of possessing a single continuous piece of RNA, the viral genome is divided into multiple segments, each encoding different genetic information essential for its replication and infection process. These segments are separately encapsulated in distinct virus particles. In the case of CCMV, it consists of four genome fragments referred to as RNA1, RNA2, RNA3 and RNA4. Moreover, CCMV's unique structural features and well-studied self-assembly process make it an ideal model for studying viral dynamics at the molecular level [216, 217, 218].

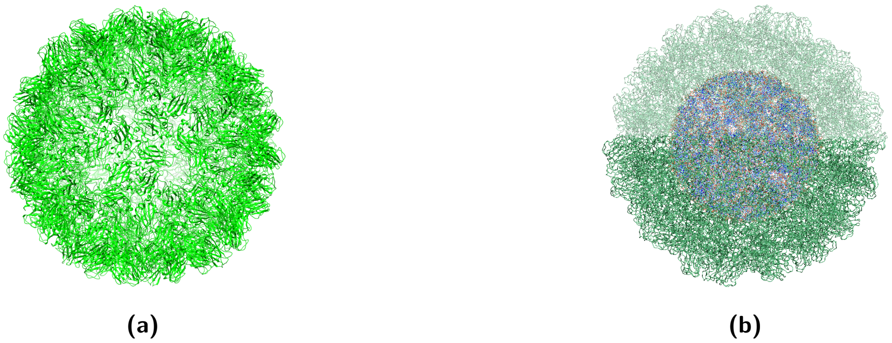


Figure 6.3: 3-dimensional structures of the Cowpea Chlorotic Mottle Virus used in this work. **(a)** reports the structure of the capsid alone, in ribbon representation, as resolved via X-ray christallography by Speir *et al.* in 1995 [219]; **(b)** shows the atomistic model of the virion particle built and used in this work, with the capsid (in dark green) containing the RNA2 fragment inside (CPK representation, in blue). All the images are obtained using VMD [39].

Driven by these facts, we decided to select CCMV as a case study for the characterization of the dynamical behaviour of a virus particle. The decision to focus on studying the CCMV was primarily driven by the desire to undertake a challenging molecular dynamics simulation that would stretch the capabilities of our in-house developed CANVAS model. The research project discussed in this chapter involves conducting molecular dynamics simulations at various levels of resolution to unravel different aspects of CCMV's structure and behavior. At the coarse-grained level, we employed the oxRNA model [88] to simulate the RNA2 fragment of the virus. The oxRNA model simplifies the representation of the RNA, allowing us to study larger timescales and explore the folding and dynamics of this specific genomic segment. Moving to the atomistic level, we conducted simulations of one capsid (the protein shell) and one complete virion particle, which includes the RNA2 fragment within the capsid. This level of detail enables us to study the interactions between the viral genome and the capsid, shedding light on the assembly and stability of the viral particle. Additionally, we studied one trimer, comprising three individual capsid proteins (CP) that collectively form a subunit of the capsid. We used this system to benchmark the CANVAS model coupled to the Debye-Hückel model for implicit solvation, with the future goal to apply it to the whole capsid both empty and containing the RNA2 fragment. By combining these different levels of simulation, our research aims to provide a comprehensive understanding of the CCMV dynamics, ultimately contributing to advancements in the field of virology.

6.2 Coarse-Grained Molecular Dynamics of the RNA2 viral fragment

The process of self-assembly (SA) of biomolecular structures, such as viral particles, is a fascinating topic whose general understanding impacts several scientific fields, from evolutionary biology to drug design and delivery. However, its intrinsic complexity sets a challenge for the high-resolution reconstruction of the assembly pathway, both to experimental and computational scientists.

While models and simulations have provided extremely insightful spatial and temporal details, they are limited by the current computational overhead. On the contrary, *in vitro* experiments cannot access the fine resolution detail that is essential to deeply understand the phenomena under study. In particular, in order to simulate the process of viral SA, it is required to reproduce many sub-steps: the folding of the CP (or the lipid membranes for some viruses), the folding of the RNA and finally the aggregation of CPs and RNA to form the assembled virion. Although the folding process of CPs could not be strictly necessary, for example if the 3D structure of them is already known experimentally, the folding of RNA can take advantage of computational modelling, since it is almost impossible to reconstruct all the conformers compatible with the given thermodynamical state.

By exploiting the numerical efficiency provided by a recently developed CG model of RNA (oxRNA [220], described in chapter 3), we have assessed the folding pathways of a ssRNA viral fragment (one of the 4 found in CCMV capsids) by means of MD simulations. By construction, the oxRNA model has been developed to consistently predict the emergence of secondary and tertiary structural motifs (such as helices and stem loops), although allowing for the sole Watson-Crick coupling of base pairs and the GU wobble base pair. In addition, the level of resolution of the model is suitable to perform an extensive exploration of the configurational space of biologically-relevant systems, expanding on the typical timescales of classic, atomistic MD assessments by more than 3 orders of magnitude. Therefore, simulations performed with the oxRNA force field might provide insightful details on the earliest stages of SA processes, which have been shown *in vitro* to occur on timescales about the order of seconds [221].

For this study, we had the following goals:

1. to give an extensive description of the conformational space of the RNA2 fragment found in the CCMV viral capsid both in the presence and in the absence of an external field that mimics the effect of the capsid itself;

2. to generate structures with realistic topologies and conformations to be back-mapped into AA representations of the system, which would be subsequently employed in high-resolution simulations of the whole virion.

An insightful feature of these simulations is the spontaneous emergence of secondary and tertiary structural motifs in the viral RNA, tracked via the evolution of the hydrogen-bond network between nucleotides. In the following we will describe the protocol followed to setup the simulations and the analysis performed. we will then carry out a comparison between the two scenarios, followed by a final discussion of the results obtained. After explaining the protocol followed for the setup of the simulations, we discuss the convergence of the potential energy as a tool to assess the achievement of equilibrium. Then we analyse the hydrogen-bond network between the nucleic bases and we focus on the formation of duplexes and their persistence, also looking for the formation of pseudoknots (in a qualitative manner). The last analyses are based on the construction of a so-called *dual graph* [222] starting from the the secondary structures' contact map. The graph is used to classify these structures in topological terms, and some features are extracted and compared. This analysis revealed to be a necessity, since the huge complexity of the system does not allow for a frame-by-frame visual inspection of the simulation to identify interesting features, as can be done for smaller system, thereby requiring me to resort to the use of automatized algorithms to filter out some information.

6.2.1 Folding of the RNA2 fragment

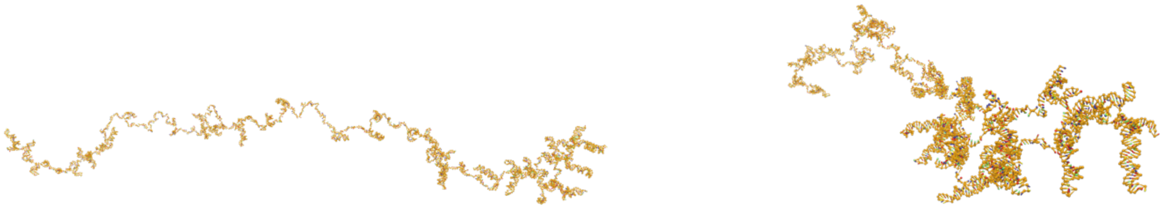


Figure 6.4: Two examples of conformations obtained by simulating the freely folding RNA2 fragment. The conformation on the left-hand side is an example of those at the beginning of the production runs, after the relaxation runs (see text); the one of the right-hand side is instead an example of the folded structures, from the final part of the production runs.

For a 2774-nucleotide viral ssRNA fragment, what kind of structures are predicted by the oxRNA model? Are there pseudoknots in the final structures? What is the variability of the secondary structures in the conformations sampled at equilibrium? We try to give exhaustive answers to these questions by analysing two runs of the same RNA2 fragment, with two different Debye screening terms in the Coulombic part of the potential energy (corresponding to two salt concentrations: 0.15 M and 0.50 M).

Simulation setup and creation of the starting structure

NB: all the runs were made using the oxDNA source code [223, 224, 225], the oxRNA2 version of the force field [220] and the sequence-dependent parametrization of the stacking terms.
The choices we made for the simulation setup are mainly based on the suggestions presented in [226]. In the following, we will report the passages as a list. The first steps are common for both scnarios, *i.e.* at $[Na^+] = 0.15\,M$, and at at $[Na^+] = 0.15\,M$. They involve: (*NB:* for units conversion, we refer to table 6.2, taken from the oxRNA's webpage on the Oxford University wiki)

1. extraction of the sequence of the ssRNA RNA2 fragment of CCMV from the National Library of Medicine's website
2. creation of a rod-like 3D structure of the fragment, using a python script provided by the oxRNA developers
3. relaxation runs: two runs (10^9 steps each, with a timestep of $\delta t = 3 \cdot 10^{-3} \tau_{ox}$) for a total of $t_r = 6 \mu s$ in converted units (see table 6.2); the temperature was T = 333K in order to favour the decorrelation of the polymer; the salt concentration was $[Na^+] = 0.15\,M$

The product of these steps was a relaxed, yet still elongated filament (see e.g. left-hand side of figure 6.4). Next, we used these structures to proceed in two directions for the two different runs.
Concerning the $[Na^+] = 0.15\,M$ run:

4. production run: one single run at temperature T=310 K, $[Na^+] = 0.15\,M$ and a total sampling time equivalent to $t_s \simeq 0.5\,ms$ (trajectory frames were sampled for the analyses every $3 \cdot 10^4\,\tau_{ox}$)

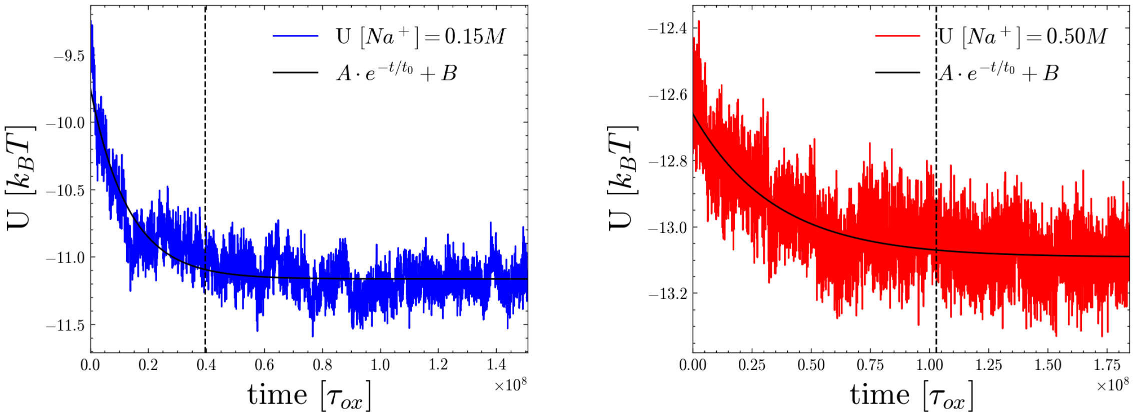
Concerning the $[Na^+] = 0.50\,M$ run:

4. increase of the salt concentration: linear increase of $[Na^{+}]$, from 0.15 M to 0.50 M, in time (at T $=$ 310 K, for a total simulation time of $t_i = 10\,\mu s$)

5. production run: one single run at temperature T=310K, $[Na^{+}] = 0.50\,M$ and a total sampling time equivalent to $t_s \simeq 0.5\,ms$ (frames sampling for the analyses: $3 \cdot 10^4\,\tau_{ox}$)

All the analyses presented below refer either to the production runs or to a subpart of it (the equilibrated part, see next paragraph).

Convergence of the potential energy



(a) **(b)**

Figure 6.5: Values of the potential energy U^{oxRNA} sampled in the production runs with salt concentration 0.15 M **(a)** and 0.50 M **(b)**. The scales of the y axes are not the same.

The first observable that we tracked along the simulations is the total potential energy, whose functional terms are reported in chapter 3 (3.18) and, more explicitly, in the related literature [220]. The values of U^{oxRNA} over simulation time are reported in 6.5, for both setups. We decided to fit the curve to an exponential function $U^{\text{theo}}(t) := A \cdot e^{-t/t_0} + B$, in order to extract the characteristic times t_0, as well as the converged energy B. To perform the fit, we used the *curve_fit()* function from the *scipy* python package [227]. It uses non-linear least squares to perform the regression. The values obtained are:

$$A[0.15\,M] = 1.41\,k_BT \quad t_0[0.15\,M] = 1.32 \cdot 10^7\,\tau_{ox} \quad B[0.15\,M] = -11.2\,k_BT \tag{6.1}$$

$$A[0.50\,M] = 0.43\,k_BT \quad t_0[0.50\,M] = 3.43 \cdot 10^7\,\tau_{ox} \quad B[0.50\,M] = -13.1\,k_BT \tag{6.2}$$

From the values of R^2 ($R^2[0.15\,M] = 0.11$ and $R^2[0.50\,M] = 0.61$), which are associated to the quality of the fit, we can deduce that the fit works better for the $0.50\,M$ case, although a visual inspection suggests that the model curve is sufficient to describe the trend in both cases.
One thing to notice is the difference in the converged energy $U^{\text{theo}}(\infty) \equiv B$ among the simulations, this being higher in the $[Na^+] = 0.15\,M$ scenario. The simplest explanation for this fact accounts for the most straightforward difference in the two setup, which is the different screening factor in the Yukawa-like potential (Debye-Hückel) that mimics the equivalent ionic strength of the solution: a higher screening leads to a minor repulsive energy between (non-consecutive) backbone sites of the chain. This argument might be valid as a rough approximation of the electrostatic contribution, although the energy value is determined by the interplay between all the terms involved. More about this point will be discussed below.
Anyway, the main goal of the fit was to establish quantitatively an equilibrated part of the trajectory, which by our convention we picked as 3 times the value of the characteristic time of the exponential, $3 \cdot t_0$. As we will discuss later on, some analyses have been performed considering only those frames sampled at $t > 3 \cdot t_0$.

Time-resolved base pairing and contact maps analysis

The core of the analyses performed in this section is based on the study of the emerging secondary structures of the fragments during the simulations. Hereafter, we will briefly report the criteria to establish whether two nucleotides are bound (via hydrogen bonding) or not.
Given two nucleotides (each made by 3 interaction sites, one for the backbone, one for the stacking and one for the hydrogen bonds as shown in figure 3.6) hydrogen bonds are formed according to the following rules:

1. distance criterion between hydrogen bond sites: an HB is formed if the distance δr_{HB} is less than $\delta r_{HB}^{cut} = 0.75\,nm$ (which is the real cut-off used in the calculation of V_{HB}, which is a Morse potential [228])

2. anti-parallelism between the versor $\mathbf{a}_1$ of both nucleotides. This versor is built by joining the backbone site with the HB site, and points towards the HB; as a consequence, the ideal HB between nucleotide i and j is formed when $\mathbf{a}_1^{(i)} \cdot \mathbf{a}_1^{(j)} = -1$. The criterion we adopted to discriminate between hydrogen-bonded and unbound nucleotides is $\mathbf{a}_1^{(i)} \cdot \mathbf{a}_1^{(j)} < -0.85$, which correspond to a discrepancy of $\sim 30^o$ from the anti-parallelism (the choice is purely arbitrary)

3. non-complementarity: we discarded every pair of nucleotides that satisfies the above-mentioned criteria but are almost consecutive in sequence ($|i-j|$ needs to be ≥ 3) as well as those pairs that are not considered as complementary (*i.e.* AU, GC or GU)

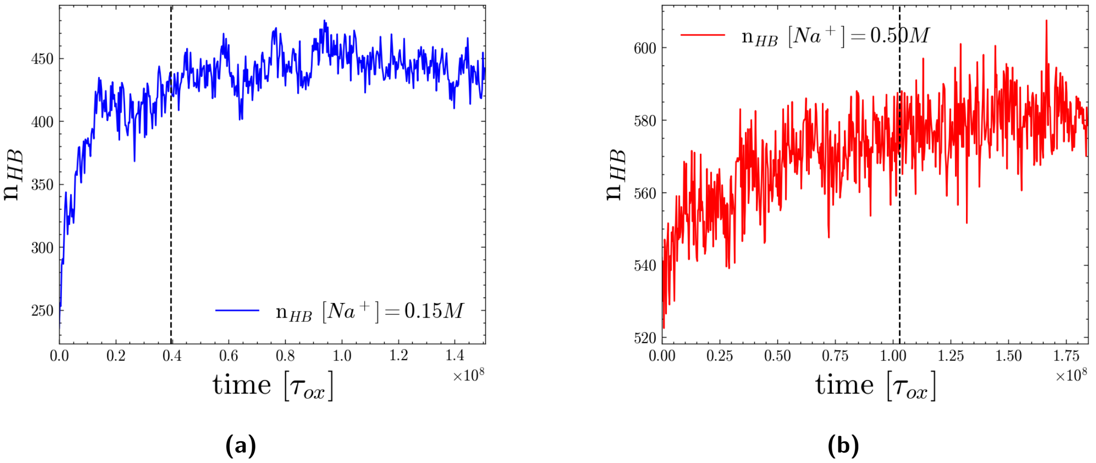


Figure 6.6: The number of HB found in the configurations sampled along the production runs: **(a)** refers to $[Na^+] = 0.15\,M$, while **(b)** refers to $[Na^+] = 0.50\,M$.

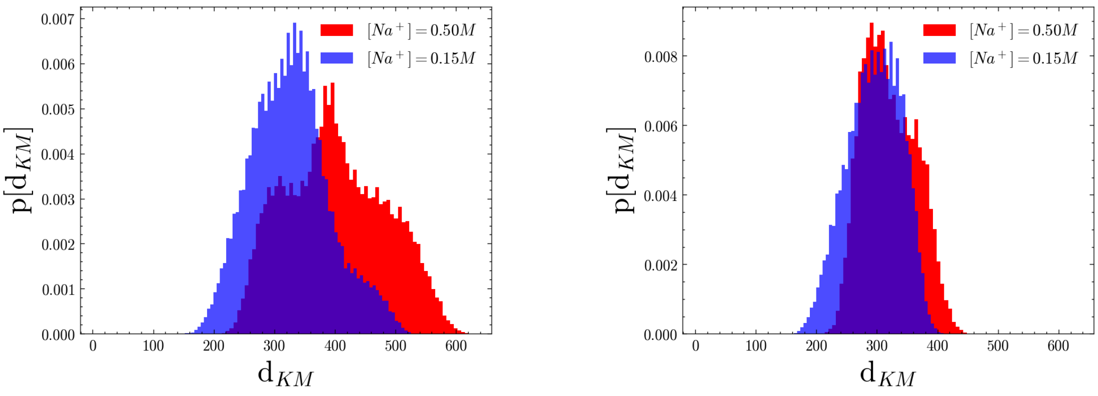
Based on these rules, we constructed a boolean contact map of values k_{ij} (*i.e.* a 2774×2774 symmetric, binary matrix of True and False values that is True if nucleotides i and j satisfy the criteria, and False otherwise). Due to the high computational cost of these calculations, we limited to 1/10 of the sampled frames for both trajectories, for a total of roughly a thousand of contact maps generated.

The most straightforward observable that can be extracted from the contact maps is the total number of HB per frame, n_{HB}: the values are reported in figure 6.6. The curves show a fast increase (which is sharpest in the $0.15\,M$ plot) in the first part, somehow reaching a plateau after the equilibration time, indicated by a black dashed vertical line. The converged values reached in the two simulations are different: $n_{HB} \sim 450$ for the $0.15\,M$ and $n_{HB} \sim 580$ for the $0.50\,M$, roughly corresponding to 1/3 and 2/5 of the total number of nucleotides of the chain involved in duplexes. This difference can be correlated to the difference in the value of $U^{\text{theo}}(\infty)$ expressed before. In fact, HB formation leads to a lower value of V_{HB} and so the highest is n_{HB} the more negative is the contribution of V_{HB} to the total potential energy.

In order to use the information contained in the contact maps to compare pairs of frames, we defined a distance d_{KM} that accounts for the fact that the two contact maps are sparse (we want to ignore the non-contacts and focus only on the non-zero part of both matrices, *i.e.* the contacts) and that they can have different absolute numbers of contacts (we need a quantity that is symmetric with respect to the exchange of the contact maps). By calling n[criterion] the total number of matrix elements that satisfy the criterion expressed as input, the distance is defined as follow:

$$d_{KM}(k^{(1)}, k^{(2)}) = \frac{n[(k^{(1)}_{i>j} == \text{True}) \wedge (k^{(2)}_{i>j} == \text{False})] + n[(k^{(1)}_{i>j} == \text{False}) \wedge (k^{(2)}_{i>j} == \text{True})]}{2} \tag{6.3}$$

d_{KM} is maximum if the two contact maps do not have any contact in common, and $d_{KM} = 0$ if the two contact maps have exactly the same contacts. The factor 2 at the denominator has been introduced to obtain values of distance that can be qualitatively interpreted as an averaged number of different pairs: with this normalization factor, in fact, d_{KM} cannot be higher than the highest n_{HB} among $k^{(1)}$ and $k^{(2)}$.



(a) **(b)**

Figure 6.7: Normalized histogram of the values of d_{KM} for both the simulations compared, including all the frames of the simulations **(a)** or only the frames relative to the equilibrated part of the simulations **(b)**.

In figure 6.7 we reported the normalized histogram of the values of d_{KM} comparing all frames and also those from the equilibrated part of the production run. Qualitatively, the full distributions are moderately different while those related to the equilibrated parts are very similar in shape. From this fact we can deduce that, at equilibrium, the variability in secondary structures is close to be independent on the salt concentration (at least for the values implemented here). Moreover, the numerical values of d_{KM} at equilibrium, which vary approximately from 200 to 400, indicate an intense variability in the contact maps, considering the total number of contacts shown before.

We notice that this distance can be used to perform clustering of the secondary structures, so to reveal the conformational basins and extract a small number of representative contact maps (*e.g.* the centroids of the clusters) that are easier to handle in order to perform further analyses. In the appendix 6.30 we report the plot of the values d_{KM} per each couple of sampled frame, which can be used as a starting point for the clustering. However, given the non-self-similarity of the system, we considered the analysis to be somewhat redundant or of secondary importance for the scope of this work.

Duplex persistence

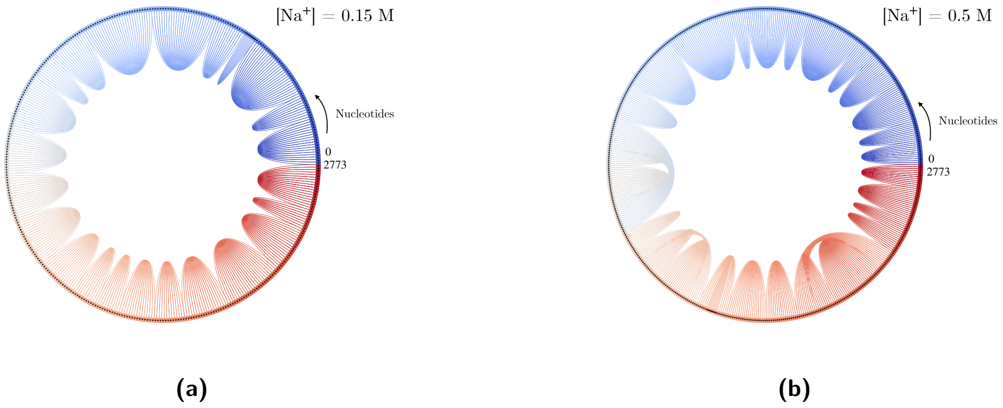


Figure 6.8: Chord diagrams representing the most persistent duplexes (at least 50%) in the simulations. We decided to use a color map, ranging from blue to red, to highlight the sequence identity of each nucleotide (from 0 to 2773).

A second kind of analysis we performed on the equilibrated part of the trajectories, based on the evolution of the secondary structures, involves the persistence of duplexes, *i.e.* groups of consecutive bases in sequence that participate to form a double stranded region of RNA. We made use of the function *duplex_finder()* provided by the oxDNA developers. As a consequence, the criterion used to detect HBs is the energy based one, which differs from that introduced in this work and used to generate the contact maps discussed before. Firstly, given all the duplexes for each time frame analyzed, we grouped those that are similar for at least 50% of the bases involved in the double strand. In this way, we wanted to remove some redundancy in the counting. Then, we counted the number of times each duplex occurs in the simulation and we retained those ones that had appeared 50% of the time at least. The obtained duplexes are reported in figures 6.8. The first thing to notice is that the persistent duplexes (the parabola-shaped objects connecting the nucleotides involved in their formation) involve nucleotides that are close in sequence ($|i-j| \lesssim 200$ for the 0.15 M and $|i-j| \lesssim 400$ for the 0.50 M). This could be the sign that even simulations with a duration comparable to ours might still be biased by the choice of the starting, rod-like conformation: with a non exhaustive sampling the energy barrier to overcome in order for those duplexes to break and reform others with new nucleotides that are far in sequence could be too high. This is not necessarily in contrast with the variability in contact maps highlighted by d_{KM}: the process of selection of the duplexes, in fact, assigns the same identity to duplexes that vary at most 50% in base pairs composition; however, those could manifest in very diverse contact maps. We can say, in other words, which the contact maps analysis is finer than the duplex one. Another thing that is worth noticing is the more extended nature of the bridges present in the 0.50 M simulation, with respect to the 0.15 M case. This was expected due to the higher shielding of the electrostatic repulsion between nucleotides, which allows for a higher conformational variability.

We also looked for for persistent pseudoknots formation: those would be easily detected in figure 6.8 as bridges that cross each other. As one can clearly see there is no such crossing in neither of the chord diagrams: this is curious, since the presence of pseudoknots in long ssRNA chains is expected (see *e.g.* [229, 230, 231]). We ascribe this lack of persistent pseudoknots to the short sampling: although pseudoknots might form in simulations performed with oxRNA [220], the rarity of their formation in a system as large as this viral RNA fragment poses an intrinsic limit.

Graph-based analysis of RNA secondary structures

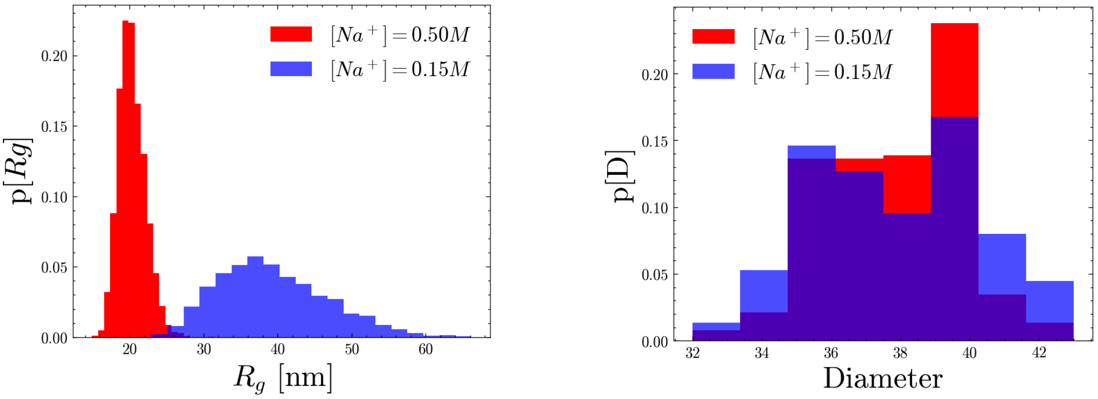
The last approach that we followed to investigate the variability of secondary structures in our simulations is based on the construction of a *dual graph* [232, 233, 222] for each secondary structure. The use of graph theory as a tool to characterize and classify secondary structures of ssRNA filaments in a topological way revealed useful not only for an *a posteriori* analysis but also to predict new RNA-like topologies [222]. In our work, the use of dual graphs to analyse secondary structures has multiple advantages: a) despite being developed for atomistic 3D RNA structures, it is directly compatible with a coarse-grained model of RNA; b) it is an automatable approach to give a general, quantitative description of secondary structures, independently on their complexity and the dimensionality; c) it highlights features that are potentially different to those highlighted by analyses based on real space positions.

We will hereby explain how a dual graph is built, given the contact maps representing the secondary structure of an RNA configuration. The steps are the following (see [222]):

1. each duplex (involving at least 3 base pairs) is mapped into a vertex
2. an at least 3-nucleotide-long sequence of consecutive nucleotides that are not involved in any pairing is mapped into an edge that connects the duplexes found at the far ends of it
3. hairpins are mapped into self-edges that point against the node related to the stem involved in the stem-loop motif
4. unpaired residues at the 5' and/or 3' ends of the RNA chain are not represented.

For each graph obtained in this way, one can construct three matrices: the adjacency matrix $[A_{ij}]$, the degree matrix $[D_{ij}]$ and the Laplacian matrix $[L_{ij}] := [D_{ij}] - [A_{ij}]$. $[A_{ij}]$ specifies the number of edges between vertices in the dual graph. Thus, the element A_{ij} is the number of edges between vertices i and j if they are connected, 0 otherwise; the diagonal element $A_{ii} = 2$ if a self-edge exists at vertex i, 0 otherwise. The diagonal element of $[D_{ij}]$ contains the number of edges incident on vertex i, or the row-wise sum of elements i in matrix $[A_{ij}]$; all off-diagonal elements of matrix $[D_{ij}]$ are zero. The Laplacian matrix can be diagonalized, and the lowest non-null eigenvalue λ_1 defines the algebraic connectivity (or Fiedler value) of the dual graph, and is a measure of the connectivity or compactness of the dual graph topology [222]. Other important features of the spectrum of $[L_{ij}]$ are the fact that for connected graphs (like those in our application) the eigenvalues are always non-negative and the minimum one is $\lambda_0 = 0$. We used this property to check that our graphs was always connected (no isolated groups of vertices). Isomorphic dual graphs (ignoring self-edges) have identical eigenvalue spectra for $[L_{ij}]$, but dual graphs with identical eigenvalue spectra are not necessarily isomorphic.

We calculated the values of λ_1 for each secondary structure built up on the frames of the equilibrated part of the trajectories. Moreover, we calculated the *diameter* of the graph, which is the length of the shortest-path between the most distant nodes (where the distance between two nodes is expressed in the minimum number of edges to connect them).



(a) **(b)**

Figure 6.9: Normalized histograms of the values of **(a)** R_g and **(b)** D for the equilibrated parts of the production runs.

In figure 6.9 we report the normalized histograms of the values of the radius of gyration (R_g) 6.9a and of the graph diameter (D) 6.9b. The most evident and interesting observation that can be extracted from the comparison of these plots is the fact that while there is a remarkable difference in the R_g distributions, the values of the graphs' diameters are very similar between the two scenarios. This suggests that, although conformationally different, the topologies of the graphs generated from the secondary structures can be similar. By comparing these values to the normalized histogram of the total number of nodes (figure 6.35a of Appendix A) one can see that the diameters involve roughly 1/3 of the total number of nodes in the 0.15 M case and less than 1/4 in the 0.50 M case. This indicates that the 0.15 M graphs' topologies are more rod-like, while the 0.50 M ones are more tree-like.

In figure 6.10a we report the normalized histogram of the values of λ_1 per each graph built on the equilibrated configurations. In order to extract a qualitative idea of their meaning, we compare these values to the analytical values of λ_1 calculated for 3 different kinds of simple "model" graphs (with a number of nodes equal to n): the path graph P_n [234], the cycle graph C_n [234] and the 2D squared lattice graph L_n [235] (examples of those graphs are reported in the appendix 6.34). While P_n and C_n are one-dimensional and linear in shape (C_n having constrained ends, unlike P_n), L_n is very well connected and tangled, with a complex yet regular topology. Similarities with the values of λ_1 derived from the spectra of these simple graphs is a good baseline to interpret the λ_1 values of our RNA graphs. The eigenvalues λ_1 of P_n, C_n and

L_n are given by [234, 235]:

$$\lambda_1(C_n) = 2 \cdot \left[1 - \cos\left(\frac{2\pi}{n}\right)\right], \quad \lambda_1(P_n) = 2 \cdot \left[1 - \cos\left(\frac{\pi}{n}\right)\right], \quad \lambda_1(L_n) = 4 \cdot \left[1 - \cos\left(\frac{\pi}{\sqrt{n}}\right)\right] \tag{6.4}$$

and are plotted in figure 6.10 within a range of n that is compatible to the values associated with the RNA2 secondary structures, as one can see from the normalized distribution of the number of nodes reported in figure 6.35a of the appendix.

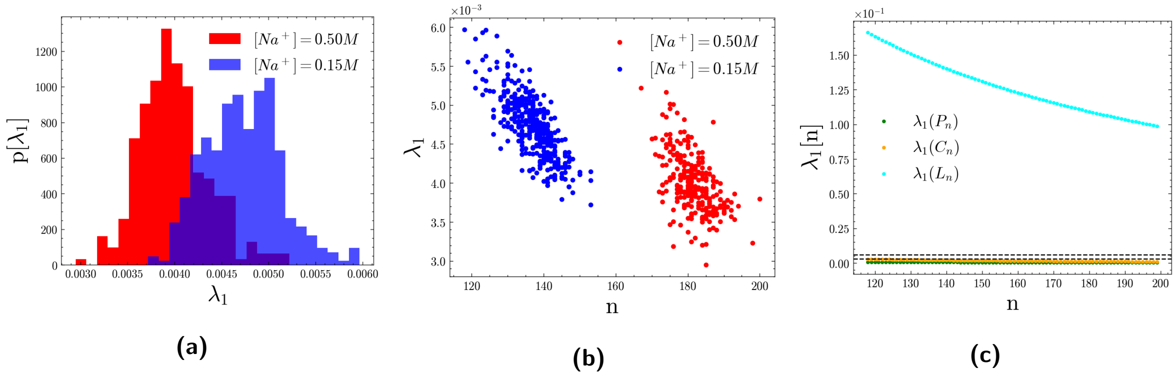


Figure 6.10: **(a)** Normalized histogram of the values of λ_1 of the equilibrated parts of the production runs; **(b)** the scatter plot of the values of n (number of nodes in each graph) and λ_1; **(c)** the analytical values of λ_1 for three simple classes of graph topologies, whose functional form are reported in equation (6.4): the horizontal dashed lines indicate the range of values of λ_1 shown in **(a)**.

The first consideration that can be done is that the values of λ_1 for both the simulations are more compatible with the C_n and P_n scenarios rather then L_n. This can be the signal of the presence of more rod-like topologies, instead of dense and much interconnected networks. It is still interesting to notice that the values from the simulations are more than twice those of C_n and P_n, indicating that their topologies are more interconnected than simple linear topologies.

One last consideration that we can make by comparing the histograms of λ_1 6.10a and of the number of nodes n 6.35a is that in the 0.15 M case the number of nodes is on average lower but the connectivity is higher (λ_1 is higher), while the opposite holds for the 0.50 M case. We can accordingly deduce that the network of duplexes is more packed and dense of connections in the 0.15 M case, even if the duplexes are less in number, while in the 0.50 M case a higher number of duplexes (possibly even composed by a lower number of base pairs) shows a lower inter-connectivity (as indicated by values of λ_1 more similar to those of P_n and C_n).

6.2.2 Out-of-equilibrium dynamics of RNA2 fragment under time-dependent spherical constraint

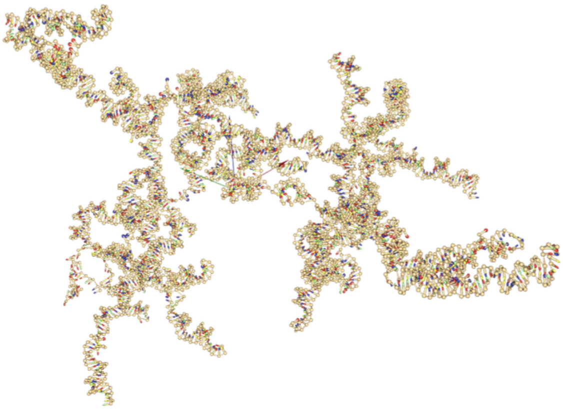


Figure 6.11: Three representative steps (3D configurations) of the packing process simulations, discussed in this subsection: on the left, the starting configuration (which coincides with the last, folded configuration of the runs discussed before); in the middle, an intermediate step; on the right, the last frame obtained before the simulation underwent interruption for numerical instability: it coincides with the most squeezed configuration possible before some of the bonds between the CG sites is broken due to van der Waals repulsion.

This subsection collects the results obtained by performing a non-equilibrium molecular dynamics simulations of the RNA2 fragment, adopting similar salt concentration conditions as before (0.15 M and 0.50 M). The purpose of this investigation is twofold:

1. on one hand, we wanted to squeeze the RNA2 fragment to occupy a very small spherical volume, to obtain a 3D structure of the fragment compatible with the available space inside the X-ray resolved 3D structure of the icosahedral capsid of the CCMV virus [219], such that a 3D model of full virion particle can be constructed and simulated

2. on the other hand, we wished to understand the process of self-assembly of the fragment [215, 216, 217], starting from the behaviour of the RNA (by forcing the packing via an outer external field, thus avoiding to include the explicit presence of the environment, *i.e.* the solvent and the capsid molecules)

Although the first purpose has been fulfilled, the second one was more ambitious; in fact, hereafter we will discuss some preliminary results that are far from being exhaustive to clarify the self-assembly process. However, our simulations show that the model is suitable for this application and that future simulations might be able to provide microscopic information *e.g.* regarding the energetics of the process, such as the work required for the external force to complete the packing.

After an overview of the setup, we will show and discuss the observables of interests: the total number of HB n_{HB} (based on the criteria introduced in section 6.2.1), the radius of gyration R_g, the values of λ_1 and the diameter D of the secondary structures' based graphs built as done before.

Simulation setup

One substantial difference with the freely folding runs is the use of LAMMPS [205] as molecular dynamics engine, instead of the native oxDNA software (provided with GPU implementation). Since we required a software with an already implemented spherical, time-dependent external force, we employed the *fix wall/lj93* command (see LAMMPS [205] (v2021) documentation for more details), whereby the external potential acting on a generic CG site i at position $\mathbf{r}_i$ is a time-dependent Lennard-Jones like potential with the following functional form:

$$V^{LJ}(\mathbf{r}_i, t) := \epsilon \left[\frac{2}{15} \left(\frac{\sigma}{\rho(t) - |\mathbf{r}_i - \mathbf{c}|} \right)^9 - \left(\frac{\sigma}{\rho(t) - |\mathbf{r}_i - \mathbf{c}|} \right)^3 \right], \quad \rho(t) - |\mathbf{r}_i - \mathbf{c}| < r_{cut} \tag{6.5}$$

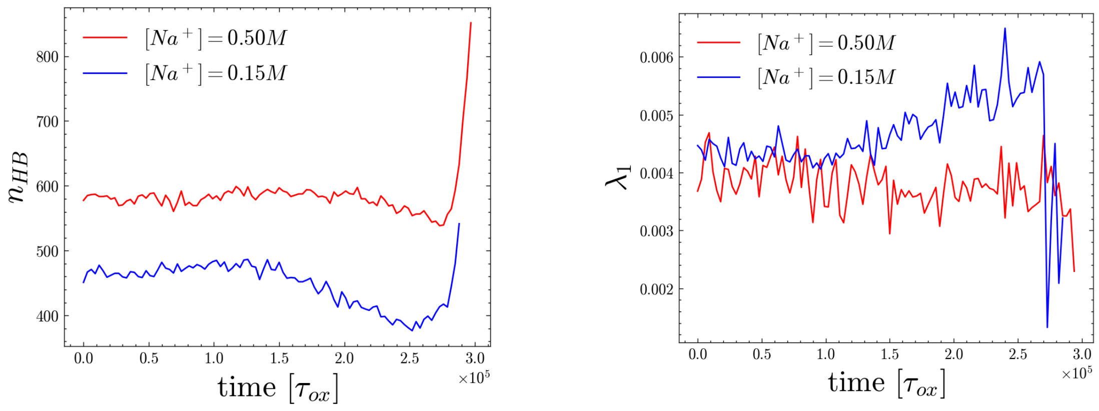
where $\mathbf{c}$ is the position vector of the center of the closing sphere and $\rho(t)$ is the radius of the sphere. This external sphere acts as a repulsive confinement; in fact, we set $r_{cut} = \sigma \cdot \sqrt[6]{\frac{2}{5}}$ so that the wall is ineffective if the distance between the CG site and the wall sphere is higher than the minimum of the Lennard-Jones like function. The radius of the sphere $\rho(t)$ varies linearly with time, as:

$$\rho(t) = \rho_0 + \frac{t}{T_{sim}} (\rho_f - \rho_0) \tag{6.6}$$

where $\rho_0 = 100 d_{ox}$ (see the table for unit conversion 6.2) and $\rho_f = 5 d_{ox}$. For these scenarios, however, we were limited to perform substantially shorter simulations with respect to the freely folding runs because of the lack of a stable GPU implementation of the MD package coupled with oxDNA. Given the volumes occupied by both initial configurations, the spherical walls start to effectively act about half-way of the simulation. Thus, the radius of the wall sphere starts from a value of ρ_0 and linearly decreases until the total simulation time is reached, where $\rho(T_{sim}) = \rho_f$.

We want to highlight the fact that there is no experimental evidence that the self-assembly process occurs by linearly packing the RNA (see *e.g.* [236] for experimental evidences on this process): our choice is again dictated by the simplicity of implementation and interpretability, yet the protocol can be easily adapted, for both the as functional forms employed for the potential and for the explicit time dependence of $\rho(t)$. The starting configurations for these packing simulations were taken from the last frames of the production runs for the free folding dynamics. The two simulations covered about $6 \cdot 10^7$ steps with a timestep of $\delta t = 5 \cdot 10^{-3}\tau_{ox}$, for a total simulation time of $T_{sim} \simeq 3 \cdot 10^5 \tau_{ox}$.

Hydrogen bonding and graph-based analysis



(a)

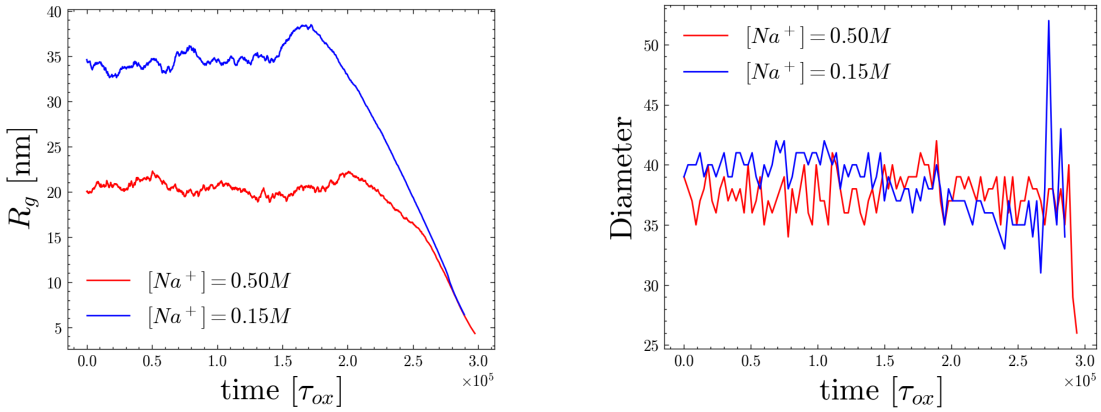
(b)

Figure 6.12: Results obtained by analysing the packing simulations. **(a)** Total number of HBs in time, as detected with the algorithm introduced above; **(b)** the values of λ_1 in time.

We first traced the number of HBs over time and plotted the results 6.12a. Probably due to the spatial proximity and extreme external pressure conditions during the later stages of the packing process, the number of nucleotide pairs satisfying HB conditions was unexpectedly high, leading to a rapid increase of the values of n_{HB}. We believe this growth to be somewhat artificial, likely revealing a weakness in the HB criteria adopted, as we would actually expect a greater breakage of the existing bonds with consequent decrease of the curves, given the strong instability of the system. In this concern, we hypothesize that an energy-based criterion for HB calculation (like the one suggested by the oxDNA developers) might reveal more robust and provide more insights into the system's behavior, under these unusual conditions.

As for the free folding case, we performed graph-based analyses. However, caution should be exerted while interpreting results obtained from these analyses, as they will in fact be similarly affected by the artifacts arising from the misconstruction of the contact maps, thereby impacting on the overall conclusions.

In 6.12b we report the values of λ_1 over time, which interestingly show a drastic decrease in graph connectivity during the final stages of packing. This observation suggests that significant changes in the structural properties of the system takes place during this phase.



(a)

(b)

Figure 6.13: **(a)** Values of R_g in time and **(b)** values of D in time.

The radius of gyration over time 6.13a overall exhibits an unsurprising trend, in spite of a moderate rise just before the monotonous decrease caused by the packing. One possible explanation for this phenomenon lies in the fact that, in the initial moments when the spherical wall exerts its action, the most exposed parts of the fragment adhere to it (to take the shape of a spherical shell). Perhaps even the parts of the filament that are further away are indirectly attracted to the compressing sphere (a kind of entropic effect). We also calculated the values of the diameter D for the graphs 6.13b, and the total number of nodes n per graph, in time 6.35b. The number of nodes and the total number of HB follow a similar trend: they start to decrease in the first stages of the packing and then they rapidly surge in the final steps. This correlation can be explained by assuming that the increase of n is caused by the formation of new, sparse duplexes (both the real and the fake ones), which is also in line with the increase of n_{HB}. Interestingly, the diameters follow different trends in the two simulations: while the 0.50 M value drops rapidly in the very last steps, the 0.15 M value, after a slight reduction, reaches a peak of around 50 and then it drops again close to the equilibrium values. The effect of the packing on the 0.15 M system can be interpreted as a drift of the topology of the graphs to a more elongated and stretched one which can be the symptom of a disruption of some interconnections between the nodes, while for the 0.50 M case the drop of diameter together with the rise of the number of nodes could be the signal of an increased connectivity and complexity of the topologies, although this observation is in contrast with the decrease of the values of λ_1 also in the 0.50 M. We are led to conclude that, as anticipated, the results obtained by the graph-based analyses in the last, critical stages of these simulations are vague and hard to be interpreted and this can be related to the unreliability of the contact maps created by using our algorithm for the HB detection.

6.3 All-atom simulation of the capsid and the virion

In this section we present the results of our all-atom molecular dynamics simulations of the CCMV virus capsid and virion models. The primary objectives of this sub-project are as follows:

1. To develop a reproducible and generalizable atomistic model of the viral capsid and virion using a well-defined algorithmic procedure.

2. To perform all-atom solvent-explicit dynamics simulations of the capsid and solvated virion, aiming to obtain high-resolution data for future comparison and validation of the CANVAS model.

3. To assess the stability of the CCMV capsid simulation with unstructured tails, a crucial feature that has not been explored in previous studies [237, 238].

4. To conduct MD simulations of an all-atom representation of the CCMV virion with its correct genetic content [219] (one of the three possible configurations), a novel investigation that has not been undertaken before for this virus.

5. To characterize the capsid-RNA contacts within the virion simulation.

To achieve these objectives, we performed all-atom molecular dynamics simulations in explicit solvent using GROMACS, leveraging GPU-accelerated computation for enhanced efficiency and performance. In this section, we briefly outline the steps involved in setting up the simulations, referring to the appendix for more detailed descriptions. Additionally, we present and discuss the analyses performed, including RMSD, radius of gyration , RMSF (all of these made using the

MDAnalysis python package [45]), RNA-capsid contact analysis using another (in-house modified) python package, *pynteraph* [239], and an in-depth examination of solvent-related artifacts.

6.3.1 Simulation setup and creation of the starting structures

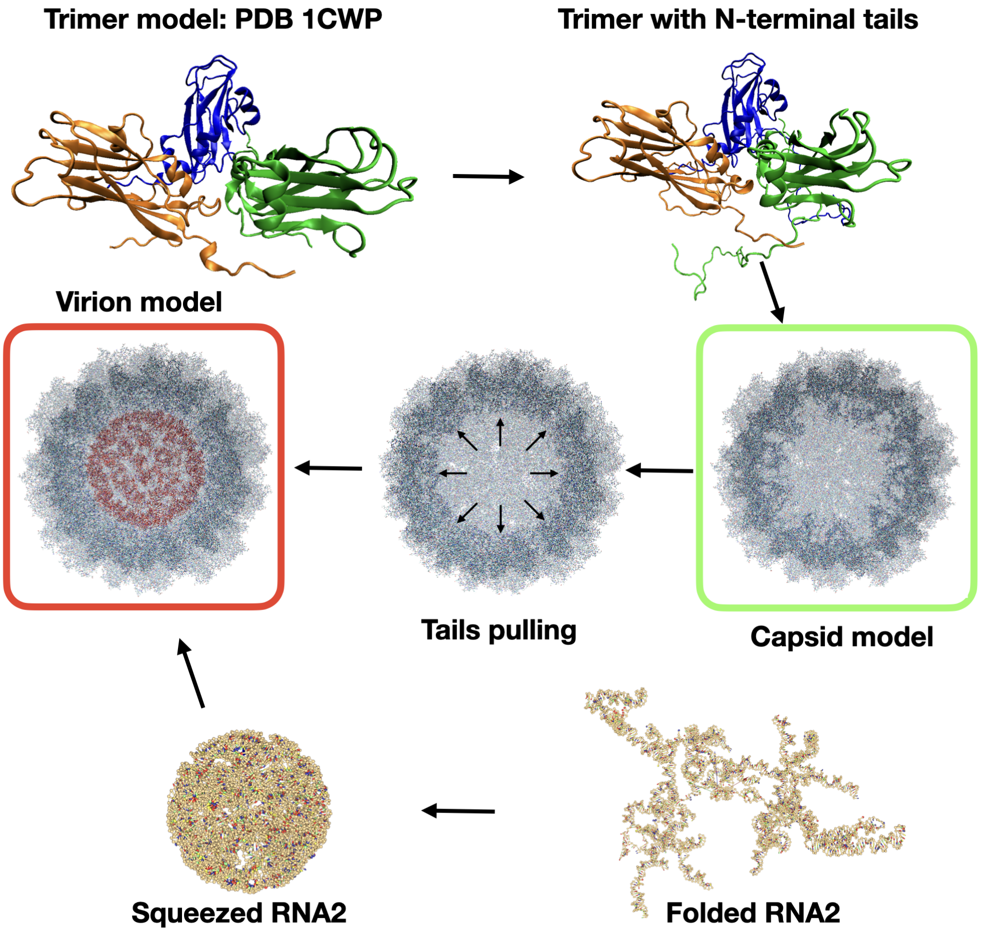


Figure 6.14: Schematic representation of the key steps followed to build the capsid and the virion starting atomistic models for MD simulations. A more exhaustive explanation is reported in appendix B of this chapter.

In this paragraph we briefly review the steps that we followed to model the starting structures. For a complete setup of the simulations involved in these steps, we redirect the reader to Appendix B of this chapter.

As shown in the infographic in figure 6.14, we took the structure of a trimer (one of the 60 different building blocks of the structure of the full capsid) from the PDB entry 1CWP [219]. We then used *MODELLER* [240] to create a model of N-terminal tails, which are disordered and as a consequence they cannot be resolved by means of X-ray crystallography. With it, we built a model of the full capsid, by reproducing the same modelled trimer 60 times and by taking advantage of its icosahedral symmetry: these steps produced the starting point for the simulations (equililbration and 200ns-long production run) performed on the empty capsid (no RNA inside), in explicit solvent. Then we performed a steered molecular dynamics on the last residues of the tails, in order to gain more free space inside the capsid. Finally, we inserted the structure obtained in the packing run of the RNA2 fragment with oxRNA (discussed before). In order to do that, we used the web-server TacoxRNA [241] to back-map the oxRNA structure into an atomistic one. The output of this insertion made the starting point for the simulations (equililbration and 200ns-long production run) performed on the virion particle, in explicit solvent.

In the next paragraph we discuss the results of the analyses performed on the trajectories obtained in the above-mentioned 200ns-long production runs.

6.3.2 Analysis of the trajectories

The first analyses performed on the frames extracted during the production runs of the capsid and the virion are: the calculation of the capsid protein-averaged RMSD with respect to the initial frame, in time (defined below); the radius of gyration in time; the capsid protein-averaged RMSF (defined below).

We defined this peculiar RMSD, averaged over the trajectories of all the capsid proteins, in order to take advantage of the huge amount of sampling the simulation of a full capsid provided, to extract an estimation of the deviations of these values. Moreover, in this case, we preferred to keep the calculations separated for the residues of the N-terminal tails (1 to 50) and the others. In particular, given the number of atoms in a single capsid protein ($N_{a,cp} = 2900$ in total, subdivided in tails $N_{a,t} \simeq 750$ and shell $N_{a,s} \simeq 2150$) and given the number of proteins in the capsid ($N_{cp} = 180$, for a total of 60 trimers), the averaged RMSD at each frame t and the relative standard deviation are defined as follow:

$$RMSD^{(i)}_{s/t}(t) := \sqrt{\frac{1}{N_{a,s/t}} \sum_{j=1}^{N_{a,s/t}} \|\mathbf{r}_j(t) - \mathbf{r}_j(0)\|^2} \qquad \langle RMSD_{s/t} \rangle_{cp}(t) := \frac{1}{N_{cp}} \sum_{i=1}^{N_{cp}} RMSD^{(i)}_{s/t}(t) \tag{6.7}$$

$$\sigma[RMSD_{s/t}](t) := \sqrt{\frac{1}{N_{cp}-1}\sum_{i=1}^{N_{cp}}\left(RMSD^{(i)}_{s/t}(t) - \left\langle RMSD_{s/t}\right\rangle_{cp}(t)\right)^2} \tag{6.8}$$

In this equations we do not explicitly report the minimization process involved in the calculation of $RMSD^{(i)}_{s/t}(t)$, which as already discussed in the Appendix of chapter 2.

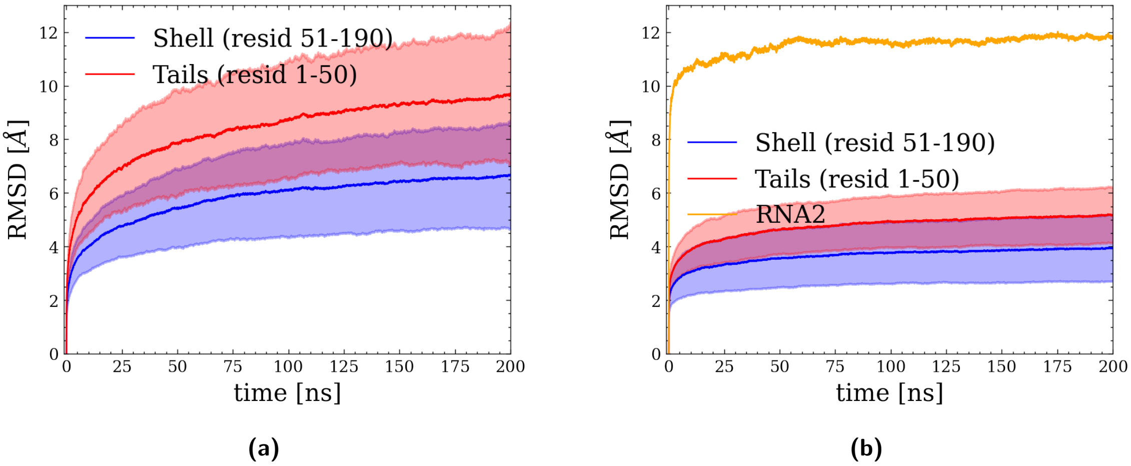
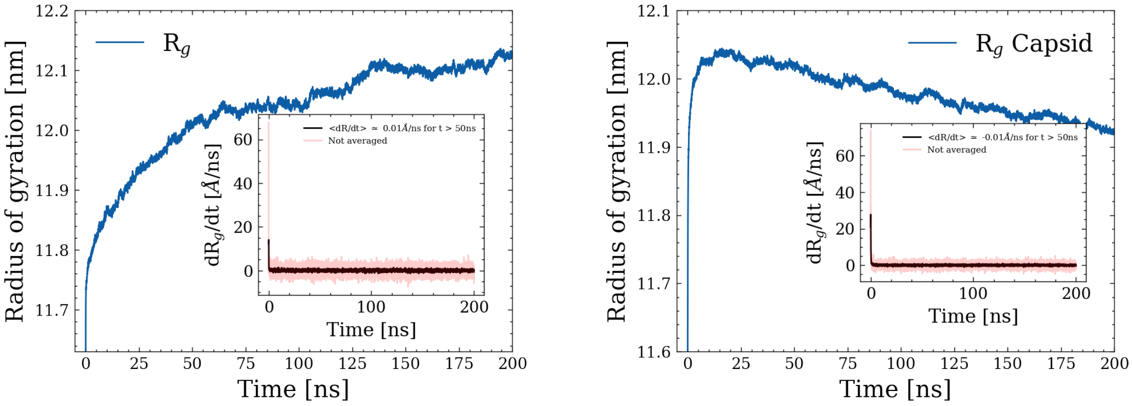


Figure 6.15: RMSD values of the **(a)** capsid and **(b)** virion production run, calculated as explained in the text, with respect to the initial frames.

The results are reported in figure 6.15. By looking at the behaviour of these curves, one can make some considerations about the equilibration of the trajectories along the 200ns. From these findings, it can be inferred that the capsid may not have reached convergence yet, while for the virion, convergence is more pronounced and occurs much more rapidly. Certainly, the interaction between RNA and tails plays a significant role, as they become less mobile after the initial stages of their attachment. This is in accordance with the RMSD value of the RNA2, reported in figure 6.15b: after a very rapid and intense growth, the value is strongly stabilized throughout the rest of the simulation. The fact that in the capsid run the values cannot be considered as converged can be the symptom that the system did not relaxed to equilibrium yet. This is not extraordinary, considering the size of the system itself (more than 500,000 atoms). This suggests that with our setup a longer run is required for the system to be considered at equilibrium, differently from other cases found in literature about molecular dynamics simulations of other viral capsids [242, 243].

The next quantity we calculated is the radius of gyration of the capsid and of the RNA2, in the virion run. The values obtained are reported in figure 6.16, accompanied by a subgraph displaying the numerical derivative, to demonstrate that both cases exhibit rapid and intense variation in the initial stages of the simulation. However, after the first growth the trend diverges: the capsid's Rg keeps increasing, and it is likely that 200 ns are insufficient to reach the equilibrium value. On the other hand, in the virion, the capsid's Rg decreases quasi-linearly, indicating that it also does not reach a plateau within the 200 ns time-frame. The observed difference is undoubtedly influenced by the presence of RNA2, but explaining the nearly linearly decreasing trend in the second case, after the initial rapid growth, is not straightforward. Notably, the Rg of RNA2, as reported in the appendix 6.36, follows the same trend as the capsid.



(a) (b)

Figure 6.16: Radius of gyration values of the (a) capsid and (b) virion production run. In the small box, a numerical derivative (in red) and its time-window averaged value (in black) of the radius is reported, showing that the most remarkable variation happens in the very first nanoseconds of simulations, in both cases.

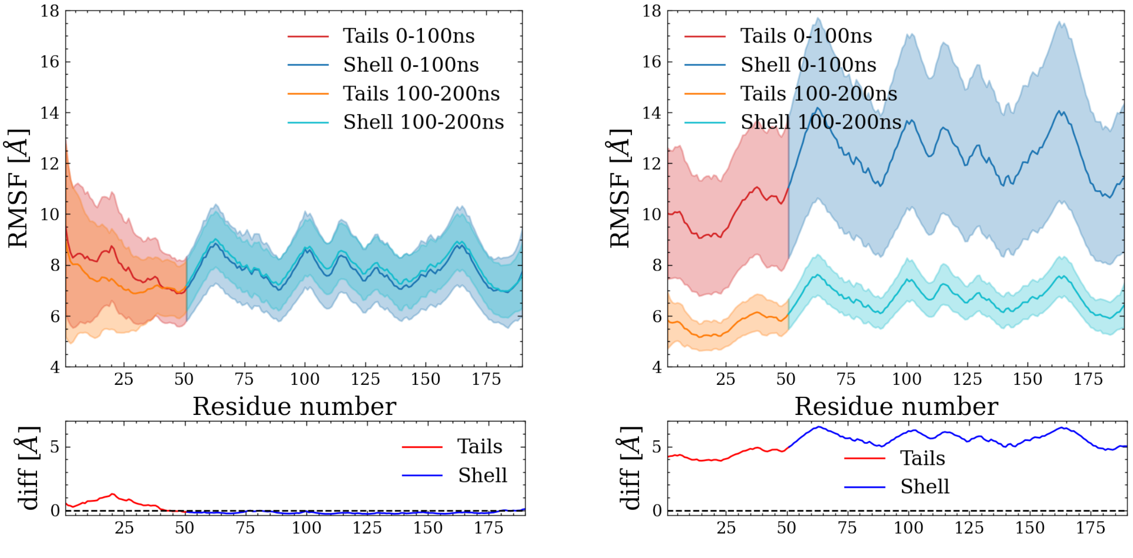
It is noteworthy to report that the literature is rich of observations of a swelling process that occurs to CCMV virions, if they are brought to physiological pH conditions from pH = 5 [244, 245]. It is curious to observe the behaviour of the simulated virion is opposite, showing a shrinkage after the rapid growth. In our simulations, however, although we chose the protonation states of charged amino acids to be coherent with a physiological pH condition, we do not control the pH value (and, in turn, possible changes in the protonation states) along the simulations. Future investigations, with a longer and statistically more relevant sampling, are required to assess whether the conditions of the simulations (pH and salt concentration) or other intrinsic properties of the choices made (force field used, here CHARMM36m, dimension of the simulation box...) are the cause of this discrepancy.

We also monitored the value of the relative fluctuations of the C_α atoms of the backbone of the residues in each capsid protein, by averaging over all the capsid proteins in the capsid (similarly to what we did for the RMSD). Specifically, the RMSF values and standard deviations are defined as follow:

$$RMSF^{(i)}[C_{\alpha j}] := \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left\|\mathbf{r}_j^{(i)}(t) - \left\langle \mathbf{r}_j^{(i)} \right\rangle\right\|^2} \qquad \langle RMSF[C_{\alpha j}]\rangle := \frac{1}{N_{cp}}\sum_{i=1}^{N_{cp}} RMSF^{(i)}[C_{\alpha j}] \tag{6.9}$$

$$\sigma[RMSF[C_{\alpha j}]] := \sqrt{\frac{1}{N_{cp}-1}\sum_{i=1}^{N_{cp}}\left(RMSF^{(i)}[C_{\alpha j}] - \langle RMSF[C_{\alpha j}]\rangle\right)^2} \tag{6.10}$$

where the $RMSF^{(i)}_{C_{\alpha j}}$ refers to the fluctuation relative to the mean position, along the trajectory, of the j-th C_α (or residue) taken from the i-th capsid protein.



(a) **(b)**

Figure 6.17: RMSF values of the **(a)** capsid and **(b)** virion production run, calculated as explained in the text. In the box reported below we show the differences of the mean values (the bold lines above).

The calculation was divided into two 100 ns blocks, and for each block, the RSMF represents the fluctuation relative to the mean position of the C_α data in that block. As for the RMSD, the alignment of the trajectory frames was performed with respect to the initial structure. In the case of the capsid, it is evident that the fluctuations are quite similar for residues within the shell, while for the tails, a more substantial difference is noticeable (as highlighted by the *diff* lower graph, which represents the difference between before and after). It can be inferred that the tails exhibit higher fluctuations around their mean positions in the initial phase, likely seeking their equilibrium. In the second phase, the fluctuations are generally comparable to those of the shell residues, though with a more pronounced deviation. This is consistent with their disordered nature, as further supported by the experimental fact that the starting crystalline structure does not include their atomic positions due to their high mobility. Regarding the virion simulation values, it is evident that the RMSF values in the first 100 ns block are substantially higher. In my opinion, these numbers indicate some pathological behavior, possibly related to the alignment process, which, not being optimal, resulted in a less representative average structure, thereby offsetting the numerical values of fluctuations relative to it. Despite this, an interesting observation can still be made, even considering the less "pathological" values in the second block. The fluctuations of the tails are even reduced (both in mean value and standard deviation) compared to the shell residues. From this deduction, supported by visual inspection of the simulations, it can be inferred that the tails attach to RNA2, forming a complex with a highly stable and less mobile structure. This observation aligns with the nearly constant RMSD value of RNA2 throughout the simulation after the initial rapid growth phase.

Capsid-RNA contacts

To investigate the network of interactions formed between the negatively charged phosphate groups of RNA2 nucleotides and the positively charged groups of protonated residues (atoms *NZ, HZ** for LYS, atoms *NE, HE, CZ, NH*, HH1*, and HH2** for ARG, atoms *N, H1, H2, H3* for N-terminal MET), we conducted a study of so-called *salt bridges*, following the nomenclature dictated by the analysis tool we used, namely *pyfferaph* (a modified version of the python package *pynteraph* [239]). For a given virion system configuration, a salt bridge between two charged residues (one acidic and the other basic or, equivalently for our system topologies, one with a negatively charged group and the other with a positively charged group) is conventionally established if at least one pair of atoms from each charged group is within a distance of 4.5 Å. Consequently, salt bridges under this definition are established by partial charge interactions and excluded volume effects of chemical environments.

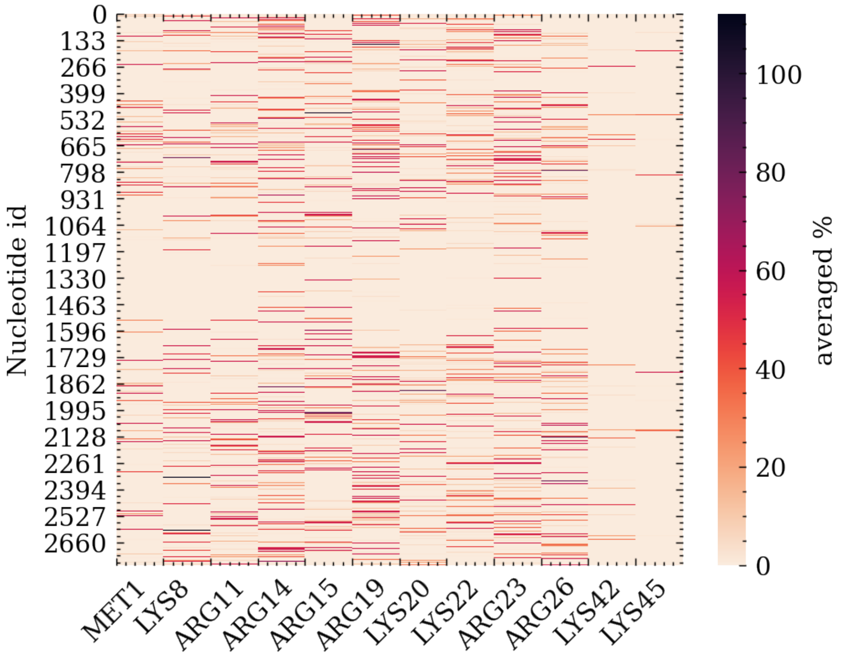
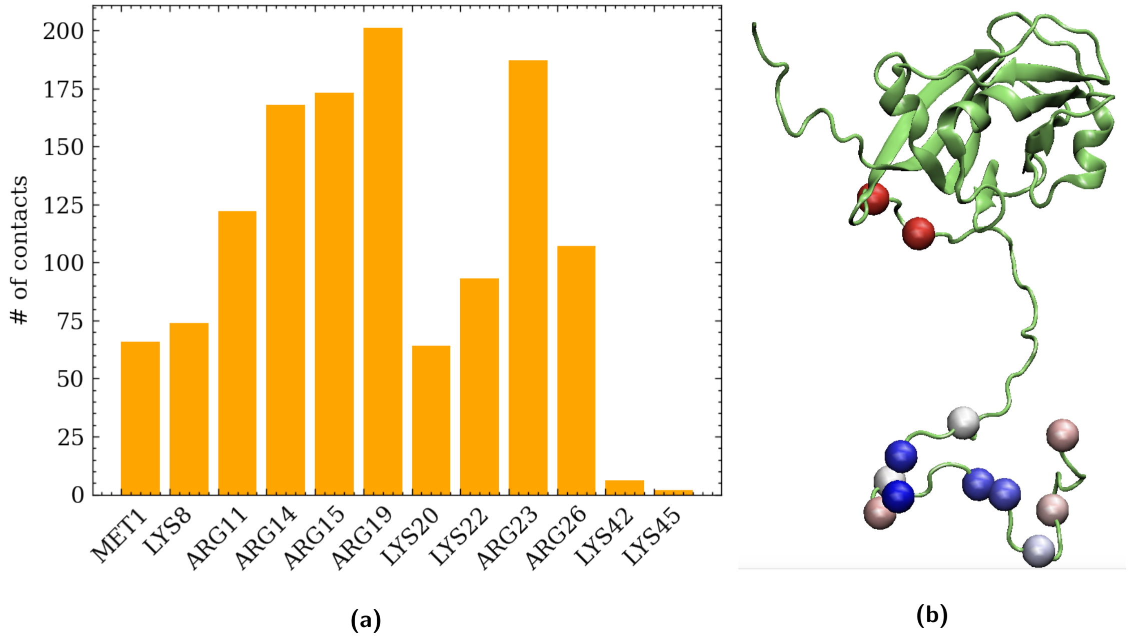


Figure 6.18: Contact persistence of the salt bridges formed between the charged group reported in the text: on the y axis the indices of the nucleotides of the RNA2 are reported, while on the x axis the names and indices of the amino acids containing the charged groups within each of the 180 capsid molecules are reported.

Similar to hydrogen bonds, these interactions emerge primarily from the non-bonded part of the force field. For a trajectory, pyfferaph's output (for our system) is a matrix of size $2774 \times (12 \cdot 180)$, as there are 2774 acidic groups (one phosphate group for each nucleotide), and $12 \cdot 180$ basic groups, representing 12 protonated residues for each of the 180 capsid molecules. The (i, j) element of the matrix indicates the total number of frames (200, one for each 1 ns of production run) where a salt bridge was observed between the i-th phosphate group and the j-th protonated amino acid group. To simplify the representation, the 180 blocks of the matrix (each with dimensions 2774×12) were averaged. The (i, k) element of the resulting averaged matrix represents the *average* persistence between the i-th phosphate group and the k-th amino acid class. This number can exceed 1.0 (or in percentage, >100%), indicating that, on average, the i-th nucleotide frequently interacts with more than one amino acid of the k-th class, specifically, amino acids with the same index but from different capsid molecules. Figure 6.18 depicts the discussed averaged matrix. Upon initial observation, it is apparent that nucleotides less involved in salt bridge formation are those more centrally positioned in the sequence. Another observation is the limited participation of lysines LYS42 and LYS45, likely due to their reduced accessibility to the RNA fragment, being closely positioned to the structured part of the capsid molecule. Applying an additional filter to this matrix allows counting the number of nucleotides involved in a salt bridge with a persistence (e.g., total number of times the bridge forms during the entire run) of at least 50% for each class of tail residues. This way, a value is obtained for each class of charged amino acids, indicating how frequently that specific class interacts with phosphate groups on average, thus identifying the residues most active in salt bridge formation.



(a)

(b)

Figure 6.19: (**a**): Histograms of the number of contacts found by filtering the matrix shown in 6.18 and summing up the number of filtered contacts per each amino acid class. (**b**): 3D ribbon representation of a single capsid molecule highlighting the position of the charged groups of the N-terminal tails involved in the salt bridges (colored spheres); the color scale is interpreted as follow: red indicates a very low number of contacts (< 10), white a medium number (~ 100) and blue a high number (~ 200).

Figure 6.19a presents the outcome of applying this secondary filter. In the plot, a value close to 180 indicates that many (if not all) amino acids of the respective class are, on average, involved in a 50% frame contact. Figure 6.19b illustrates the spatial distribution of C_α atoms for each of the 12 protonated amino acids along a representative capsid molecule chain. A clear observation from these data is the markedly greater involvement of arginines compared to lysines: a possible explanation of this fact can rely on the observation that the charged group in arginine occupies a broader volume with respect to lysine, and this makes the group in general more exposed also to salt bridges formation. Particularly, the most active arginines, namely ARG19 and ARG23, corresponding to the most intense blue spheres in Figure 6.19b, are situated in the central portion of the amino acid sequence of the tails, not the more terminal region (N-terminus). This observation challenges the simple expectation that charged groups spatially closer to the initial RNA structure create more persistent bonds during the simulation, and is both intriguing and non-trivial.

Solvent behaviour

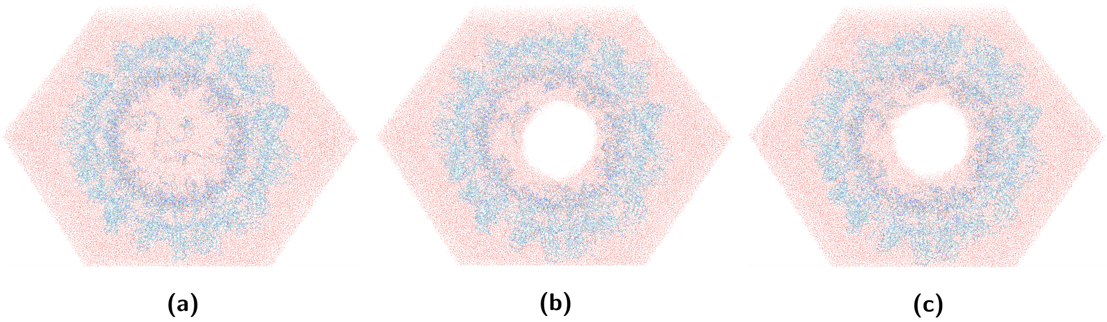


Figure 6.20: Snapshots of an hexagonal slice of 4nm of thickness, containing the center of the simulation box. The snapshots are taken **(a)** at t = 0ns, **(b)** at t = 1ns and **(c)** at t = 7ns of the production run of the virion simulation. A more general 3D visual inspection via VMD shows clearly that the region of vacuum formed inside the virion has a spherical shape.

A visual inspection of both the production runs, on VMD [39], shows a rapid initial unexpected behaviour of the solvent molecules (both the water molecules and the ions, Na^+ and Cl^-): the formation of an empty hole inside an almost spherical volume of about $5nm$ of radius, as partially documented with figure 6.20. We wanted to investigate more deeply this strange behaviour by performing two analyses: the calculation of the radial density of atomic/molecular species in the simulation box, at different instants of times; and the calculation of the total number of atoms belonging to the solvent atomic/molecular species in three different regions of the simulation box, every 1ns all along the production run.

In order to calculate the radial density of atoms (single ions or residues' representatives), we approximate the box to a sphere. Then, we divide the radius $R = 180$Å of this sphere into

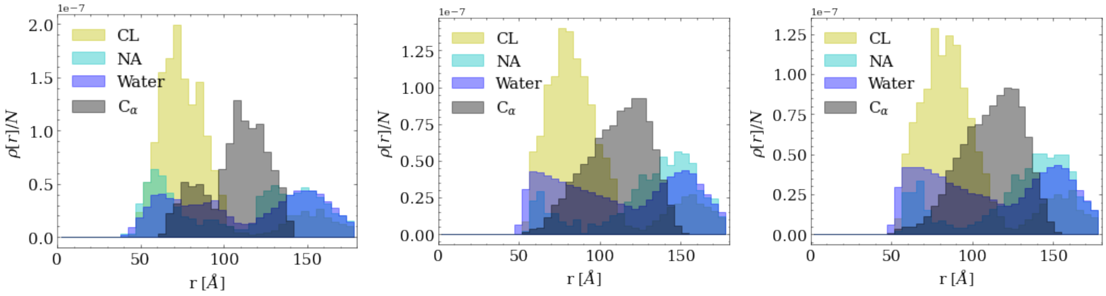
$N = 40$ spatially equivalent intervals I_i defined as follow:

$$r_i := \frac{R}{N} \cdot i \equiv \delta R \cdot i \qquad \Rightarrow \qquad I_i := [r_i, r_i + \delta R] \qquad \forall\, i = 0, ..., N-1 \tag{6.11}$$

We count the number of atoms (single ions or residues' representatives) $n_i = n(I_i)$ of a given species contained in the spherical crown with radii taken as the extremes of each I_i. We normalize each of these counters by the total number N of atoms of the given species, introducing $f_i = \frac{n_i}{N}$ (in order to be able to use the same scale for the distributions, independently on the species). Finally, we divide this numbers by the volume of the circular crown and we plot these values:

$$\rho_i := \frac{f_i}{\frac{4}{3}\pi\left((r_i + \delta R)^3 - (r_i)^3\right)} \equiv \frac{f_i}{\frac{4}{3}\pi \delta R\left(3r_i^2 + 3\delta R r_i + \delta R^2\right)} \tag{6.12}$$

The histograms representing the values of ρ_i for the capsid and the virion production runs at $t = 0\, ns$, $t = 100\, ns$ and $t = 200\, ns$ are reported in figures 6.21 and 6.22.



(a) **(b)** **(c)**

Figure 6.21: Histograms representing the (normalized) radial density of the atomic species in the simulation box (Na^+, Cl^-, water molecules and capsid C_αs). The histograms are taken **(a)** at t = 0 ns, **(b)** at t = 100 ns and **(c)** at t = 200 ns of the production run of the virion simulation.

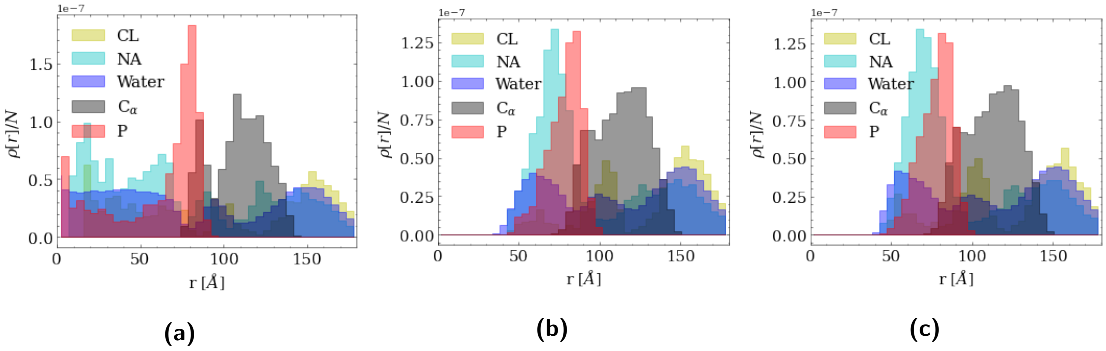


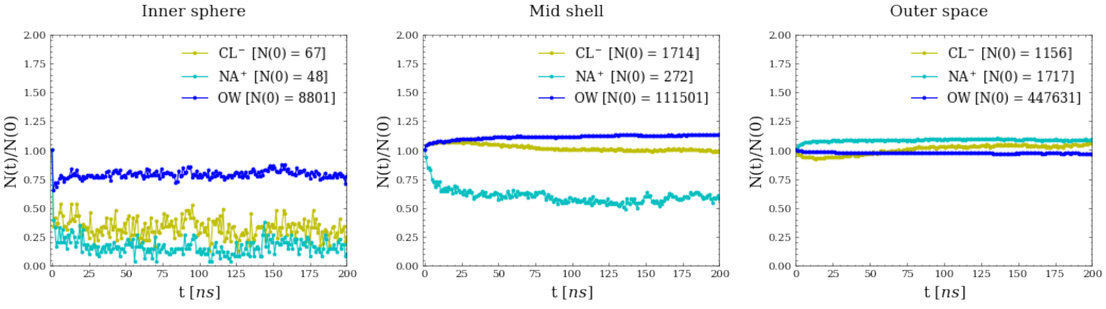
Figure 6.22: Histograms representing the (normalized) radial density of the atomic species in the simulation box (Na^+, Cl^-, water molecules, capsid C_αs and RNA2 backbone's phosphori). The histograms are taken **(a)** at t = 0 ns, **(b)** at t = 100 ns and **(c)** at t = 200 ns of the production run of the virion simulation.

In the case of the capsid, it is evident from the initial frame of the production run, in agreement with visual inspection, which there are virtually no atoms within the sphere centered at the box's midpoint with a radius of almost 5 nm. In fact, as also explained in Appendices B of this chapter, inspecting the frames preceding and following the initial equilibration process of the solvated system already shows the appearance of this empty void. Unfortunately, as we did not save the solvent-containing frames during the simulation, a more detailed analysis of the solvent behavior during this phase was not possible. Nevertheless, some insights can still be deduced from the graphs. At $t = 0\,ns$, the following observations can be made: two distinct peaks for C_α, indicating spatial separation between residues in the shell and tails; a spread peak of chlorine ions near the tails; two separate peaks of sodium ions, one at the edge of the void sphere and the other at the outer edge of the capsid; three peaks, two of which are close to the chlorine peak and one separated and external to the capsid, representing water molecules. At $t = 100\,ns$ and $t = 200\,ns$, the situation is slightly different. The observations are as follows: a single peak for C_α, indicating that the tail residues have approached the inner edge of the capsid, resulting in a more compact radial density; the single chlorine peak remains fairly unchanged; the two separate peaks of sodium ions persist but with different populations, as the external peak becomes more populated compared to the internal peak, contrary to the situation at $t = 0\,ns$; the three peaks of water molecules become two, and the first peak shows a decrease towards the outside, qualitatively corresponding to the growth of C_α density as we approach the core of the capsid shell. This histogram suggests that, starting from a uniform distribution (in the empty space) of water molecules and sodium ions, which should correspond to a mostly flat histogram, at least in the region between 0 Å and 75 Å approximately, the ions undergo substantial migration during the solvent equilibration phase before the production run. In particular, chlorine ions become denser near the tails, which are rich in positively charged LYS and ARG residues; in response to this, the water molecules that form the solvation shell move together with the ions, leaving the void behind. Initially, the sodium ions are carried by the flow of chlorine ions, but in a later phase they accumulate on the outer side, where some negatively charged residues (ASP and GLU) are present, to stay away from the tails. Therefore, we conclude that this phenomenon is mainly guided by electrostatic interactions. The capsid presents a strongly acidic environment internally (with a total charge, as calculated by GROMACS under physiological pH conditions, of approximately 1600e), and this environment, if not locally modified by potential charge exchanges with water molecules (which are impossible in a simulation without maintaining locally constant pH), causes an artifact in the solvent. we consider it highly improbable for such void conditions to occur in nature.

As anticipated, in order to deepen the understanding of the solvent behaviour, we also calculated the total number of atoms contained in three different regions of the simulation box in time (every 1 ns):

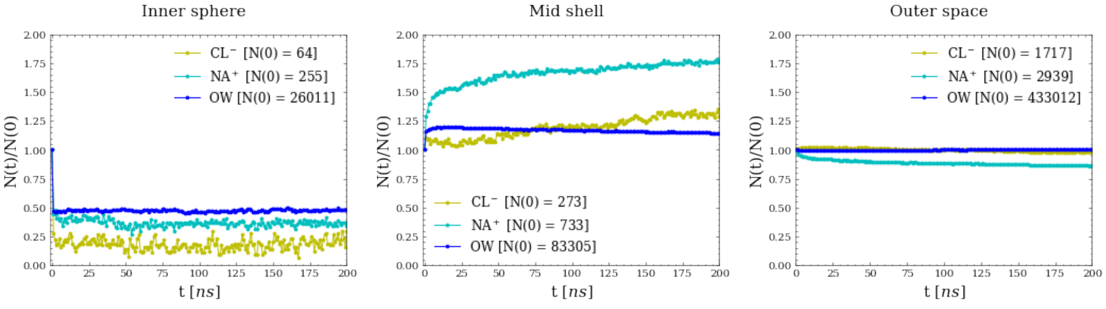
1. a sphere with radius 6 nm centered in the center of the box (named *Inner sphere*)
2. a spherical crown with inner radius 6 nm and outer radius 10 nm (named *Mid shell*)
3. the remaining space in the simulation box (named *Outer space*)

The values obtained by this analysis have been normalized with their initial value, for a better visualization of the time series: they are reported in figures 6.23 and 6.24.



(a) (b) (c)

Figure 6.23: Number of atomic/molecular species of the solvent in time along the production run of the capsid, normalized with respect to their initial value, in three different regions of the simulation box: **(a)** Inner sphere, **(b)** Mid shell and **(c)** Outer space.



(a) (b) (c)

Figure 6.24: Number of atomic/molecular species of the solvent in time along the production run of the virion, normalized with respect to their initial value, in three different regions of the simulation box: **(a)** Inner sphere, **(b)** Mid shell and **(c)** Outer space.

Regarding the two graphs related to the inner shell, the trend is evident and consistent between the virion and capsid simulations: all three solvent species rapidly escape from the center of the box within the first nanosecond. For the capsid, the escape is more pronounced for ions (compared to the initial number) than for water molecules (75% of which remain in the region). In the case of the virion, approximately 50% of water molecules escape, while ions show an even more marked escape. Notably, in the capsid run, chlorine ions exhibit less escape compared to the virion run. This phenomenon is likely due to the shielding effect induced by the negative charges of RNA on the positively charged residues present on the capsid tails.

In the mid shell, the behaviors are different. For the capsid, sodium ions (which also escape from the inner sphere) also escape from the mid shell, essentially populating the external space. On the other hand, chlorine ions show a more stable trend with an initial entry into this region and then a subsequent slight exit. In the case of the virion, sodium ions increase rapidly, remaining almost double in number compared to the initial instant. Chlorine ions, however, show a slight initial decrease, followed by an increase, surpassing the initial number by 25%.

It is again evident that the electrostatic contribution is the most relevant factor. For the capsid, positively charged residues in the tails play the main role, whereas in the virion the negative charges of RNA dominate this effect (since there are more of them, with a total charge of -2773e due to 2774 nucleotides, one of which is capped). Interestingly, in the mid shell of the virion, chlorine ions return to this region after the initial migration, likely being carried by the flow of sodium ions but with a slightly delayed escape kinetics.

In conclusion, we can claim that the artifact present in both capsid and virion simulations is caused by an electrostatic environment that, despite being modeled according to standard GROMACS criteria and being theoretically compatible with physiological pH conditions (as observed experimentally for stable virions [244]), evidently does not reflect the real environment that forms in the studied systems.

6.4 Multi-Resolution Simulations of the Trimer in Implicit Solvent

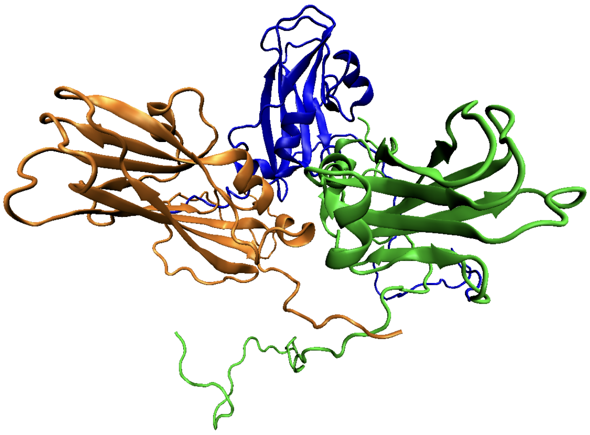


Figure 6.25: 3D Ribbon representation of the trimer model built and employed in this paragraph as starting configuration for multi-resolution MD simulations. The three colors highlight the 3 monomers composing the trimer, each one having the same amino acidic sequence.

In this section, we present the results of our all-atom and multi-resolution molecular dynamics simulations of a trimeric unit (three capsid molecules, an example show in 6.25) of the CCMV virus. The objectives of this sub-project are the following:

1. To assess the validity of the CANVAS model coupled with the implicit Debye-Hückel solvent implemented in LAMMPS [205], focusing on the fundamental building block of the CCMV capsid. The ultimate aim is to create a CANVAS model of the CCMV capsid/virion and simulate it with implicit solvent to reduce simulation times.

2. To estimate the simulation times required with LAMMPS and compare them with all-atom simulations in GROMACS. We investigate whether lower resolution simulations provide computational advantages in terms of speed.

To achieve these objectives, we employ molecular dynamics simulations at various levels of resolution, exploring both all-atom and multi-resolution approaches. The CANVAS model [93], in conjunction with implicit solvent representation, offers a lighter (in terms of the number of degrees of freedom) alternative to traditional all-atom simulations, potentially accelerating investigations of larger capsid and virion systems.

6.4.1 Simulation setup and creation of the starting structures

In order to test the validity of the CANVAS model, coupled with the Debye-Hückel model for implicit solvation (at an equivalent ionic concentration of 0.15M), we performed 5 different simulations of the trimer, each 100ns long, after a brief minimization protocol to avoid steric clashes and exploding Coulombic interactions. All the simulations have been done using LAMMPS, with the CHARMM36m force field [246]. Below we report the list of the models employed in the simulations (also shown in figure 6.26):

1. all-atom model of the trimer in explicit solvent (TIP3P water molecules)

2. all-atom model of the trimer in Debye-Hückel solvation (see figure 6.26a)

3. CANVAS model of the trimer (called CAN1, C1 or CG1 below) in Debye-Hückel solvation, obtained by treating the tails at atomistic resolution, a 1nm-thick layer of atoms in the junction of the tails to the structured part of the capsid molecules treated at medium-grained resolution, and the rest at the C_α low resolution (see figure 6.26b)

4. CANVAS model of the trimer (called CAN2, C2 or CG2 below) in Debye-Hückel solvation, obtained by treating the tails and a few residues in the central region of the structured part of the trimer (at the interface between monomers of the trimer itself) and at atomistic resolution, a 1nm-thick layer of atoms around the atomistic region treated at medium-grained resolution, and the rest at the C_α low resolution (see figure 6.26c)

5. CANVAS model of the trimer (called CAN3, C3 or CG3 below) in Debye-Hückel solvation, obtained by treating the tails and a few residues in the external region of the structured part of the trimer (at the interface between monomers of different trimers, which will be present in the future model of the full capsid) and at atomistic resolution, a 1nm-thick layer of atoms around the atomistic region treated at medium-grained resolution, and the rest at the C_α low resolution (see figure 6.26d)

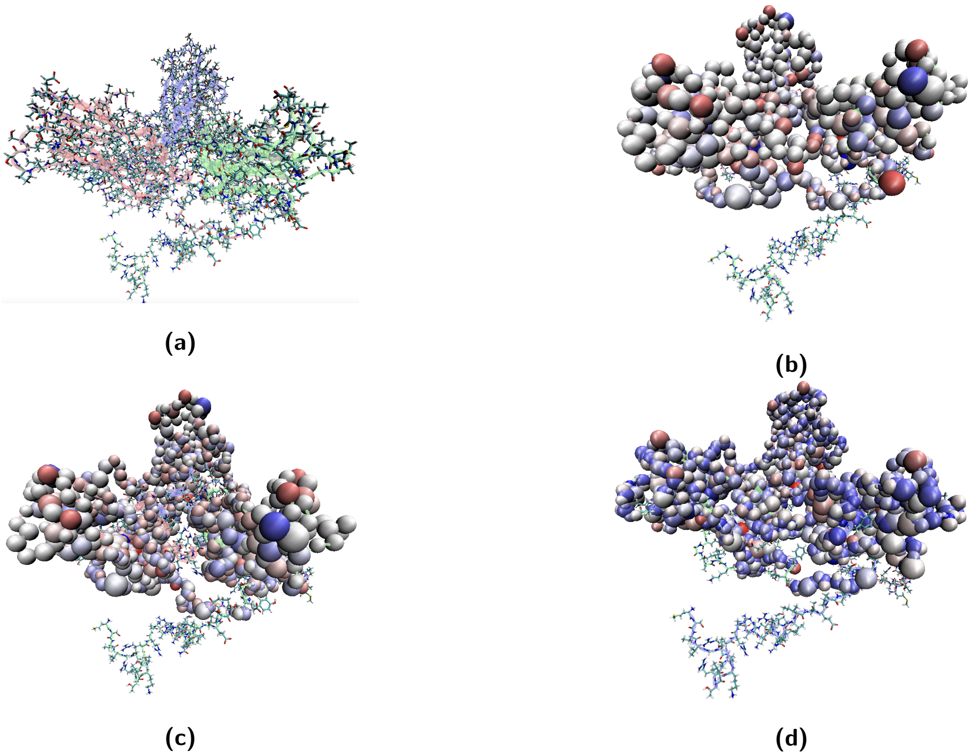


Figure 6.26: Starting configurations of the trimer simulations: **(a)** is the all-atom model; **(b)** is the CAN1 model; **(c)** is the CAN2 model. **(d)** is the CAN3 model. The figure shows the coarse-grained sites as van der Waals spheres, with radius given by the rules dictated by the CANVAS protocol for the topology construction. The color map used for the CG sites represents the charge (blue is positive and red is negative): unfortunately, every model has its own scale of colors and so it is not possible to make quantitative comparisons of the charges by looking at the figures.

The starting configuration for all the simulations was taken as the trimer model built to construct the all-atom model of the capsid, as explained in the Appendix B.

6.4.2 Analysis of the trajectories

We analysed the trajectories with the purpose of probing the validity of the CANVAS models employed, with respect to the all-atom explicit solvent model, here considered as the reference case. The analyses performed consist in: the calculation of the RMSF per residue (excluding the disordered tails) and a correlation study, quantified by the Pearson coefficient; the estimation of the stability in time of the secondary structures in all-atom explicit solvent and the all-atom implicit solvent, to test the goodness of the implicit solvent model to preserve possible secondary structures in the atomistic regions modelled by CANVAS; the calculation of the atom-wise values of the SASA in the atomistic regions (excluding the disordered tails), to see if those regions in the CANVAS models preserve the same solvent exposure to those in the all-atom explicit solvent model.

In figure 6.27 we report the values of the per residue RMSF (tails excluded) of the explicit solvent run (x axis, labelled as ES) plotted against the values obtained in the implicit solvent runs (y axis, labelled as DH, each one represented with a different color). The first observation that can be done is that the clouds of points relative to the all-atom DH and the C1 DH runs are the most scattered ones, while the C2 DH and the C3 DH appear very narrow and linear: this fact suggests a good correlation in the latter cases, while for the former a sub-optimal correlation is observed. Another notable fact is that the RMSFs are increasingly lower in amplitude, with respect to the AA ES simulation (the slope of the straight lines passing through the points is lower and lower, from AA DH to C1, C2 and C3 DH): this indicates that the CG sites in the core are somehow affected to the different modellization of the regions at the interface of the capsid molecules in the C2 case and at the border of the trimer in the C3 case, with an additional stiffening of the C_α behavior. The analysis confirms the ability of the CANVAS model to reproduce well the relative fluctuations among the residues [93], although it is not able to be quantitatively predictive in that. This can be a problem or not, depending on the feature or process that CANVAS is required to reproduce: for example, if we want to employ the model to describe the energetics of the disassembly of the capsid into trimers in certain conditions of pH and salt concentration, it might not be a problem that the fluctuations of the residues within the trimers themselves do not match quantitatively the absolute values found in the atomistic model.

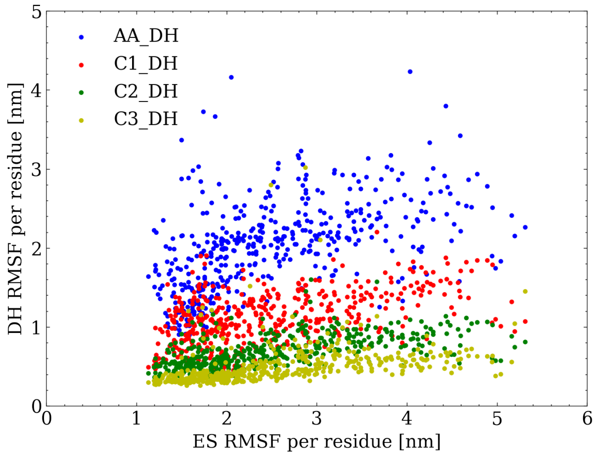


Figure 6.27: Scatter plot of the values of RMSF per residue in the structured region of the monomers (shell): on the x axis the values of the AA ES run are reported, while on the y axis the values of each run performed with the Debye-Hückel implicit solvation model are reported, as described in the legend.

The second, already mentioned analysis we performed consists in the calculation of the persistence of the secondary structures of the trimer in time, by comparing the AA ES simulation to the AA DH one. It is in fact known (as it was already discussed the topic in chapter 5 of this Thesis) that the implicit solvent models, coupled with atomistic force fields that are built in order to perform best with the presence of explicit water molecules, have the tendency to overestimate the formation of hydrogen bonding, affecting in turn the formation of secondary structures in the proteins. In figure 6.28 we reported a plot that associates to every residue, at every time frame, its co-participation (or not) to a secondary structure motif: the red color indicates the participation to a β sheet or β-like structures in general; the light-blue color indicates the participation to an α-helix, the dark blue color the participation to other helices and the white color represents unstructured/coiled pieces of the polymer. The y axes of the figures are divided in three regions for the three monomers of the trimer (each one made up by 190 residues). Although the figures are dense of information, there are some features that can be easily noticed. The first one consists in the fact that, differently from the expectations, the residues in the tails (residue ids 1 to 45/50, 191 to 235/240 and 381 to 425/430) remains unstructured in the implicit solvent simulation, while they show a tendency to form new secondary structures in the explicit solvent run.

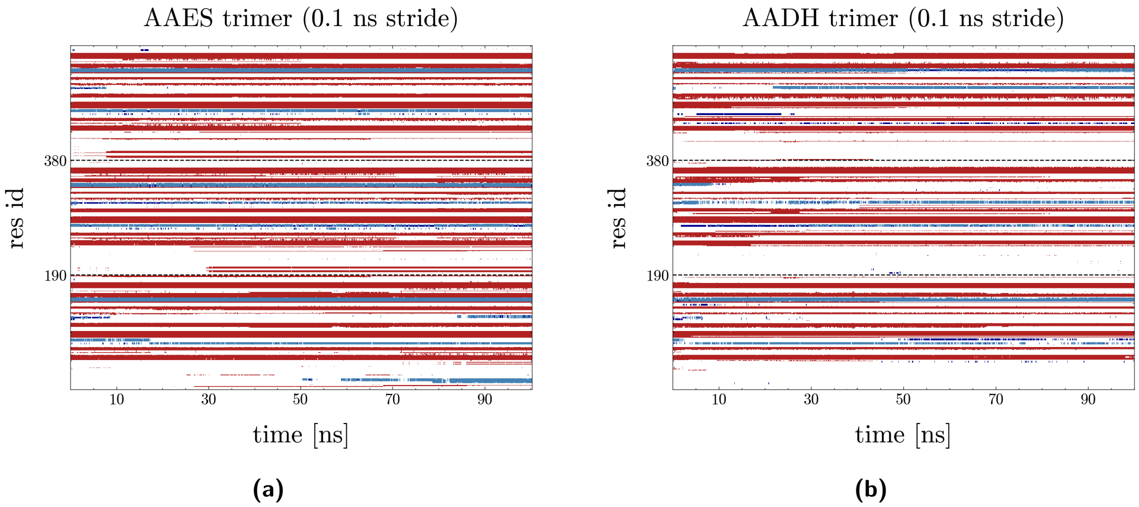
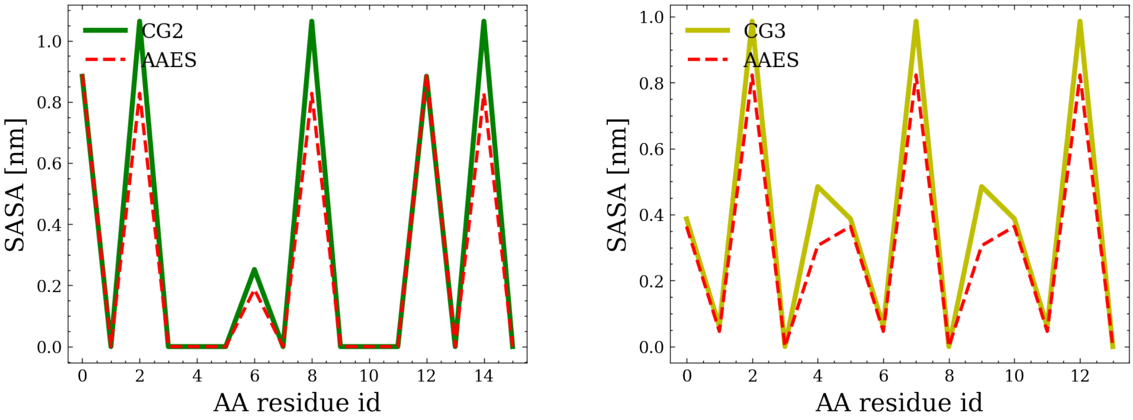


Figure 6.28: Plots that represent the secondary structure motif that each residue of the trimer is involved in, in time, as calculated by the VMD Timeline plug-in. **(a)** refers to the AA ES simulations, while **(b)** refers to the AA DH one.

In addition to this distinction, a general observation is that the implicit solvent tends to distort existing structures rather than overestimate them, albeit to a minor extent. Interestingly, the most remarkable differences are observed in the third monomer (residue ids 380 to 570), where a helical structure in the middle of the sequence converts to a β sheet in the DH simulation. Overall, however, we can infer that, within this limited sampling (which should certainly be extended for robust validation), the DH model performs well in this regard.

Calculation of the Solvent Accessible Surface Area (SASA) per residue, averaged over the entire run, was performed for the atoms preserved in the high-resolution part of the CANVAS model (excluding the disordered tails). This was conducted to investigate whether the exposure of the atomic portion to the solvent remains comparable to the all-atom explicit solvent (AAES) case.



(a) **(b)**

Figure 6.29: Solvent Accessible Surface Area calculated with the *gmx sasa* tool implemented in GROMACS, averaged over the trajectories. The values refer to the total averaged SASA per atomistic residue present in the **(a)** CAN2 and **(b)** CAN3 multi-resolution models.

The results, depicted in figure 6.29, exhibit excellent agreement between the explicit solvent simulation and the two CANVAS models, which involve additional atomic residues apart from the tails (CG2 and CG3). Notably, a slight tendency of the CANVAS model to overestimate SASA is observed, implying greater solvent exposure of atomic residues. However, this could potentially stem from an artifact of the van der Waals sphere model for low-resolution residues, which no longer align as closely as atomic atoms. Consequently, this leads to reduced shielding around sites, resulting in increased exposure even for atomic resolution sites.

6.5 Conclusions

In this section, we would like to wrap up the keys observations that came out from this broad work on the CCMV virus and its molecular constituents.

Regarding the RNA simulations, studying such a long RNA2 with the oxRNA model is feasible on GPUs, and the timescales, on the order of milliseconds (CG), seem to allow the system to relax to a state of equilibrium energy/volume occupancy. However, concerning secondary structures and their internal rearrangements, the analyses demonstrate that the contact maps' variability remains high (as expected for ssRNA [247]), but likely the simulated timescales here do not allow such a large system to undergo significant conformational changes in tertiary structure. At this point, it is unclear whether this limitation stems from the model or the duration of the sampling (also due to the fact that there are no applications of oxRNA to *ab initio* RNA folding in literature, given the relative youth of the model itself), although we think the model possesses all the attributes to facilitate the formation of more complex structures, such as pseudoknots or even true topological nodes. Graph-based analyses performed on free folding simulations show excellent potential for dimensionality reduction and more convenient handling and comparison of secondary structures produced by the simulations. Certainly, the results presented here can be further extended, for instance, studying other characteristics of the Laplacian spectrum. Note that the complexity of the graphs extracted from these simulations doesn't allow comparisons with the topologies studied in the context of dual graphs as a means of categorizing secondary structures of single-stranded RNA filaments. Regarding simulations of the out-of-equilibrium fragment, we can affirm that they have proven to be essential tools for constructing the initial structure of the virion, but they may not suffice to draw clear conclusions about the virion's self-assembly process solely from the RNA2 fragment perspective. One could also hardly try to rationalise the results obtained here in terms of nucleation theory [248], being the process strongly out-of-equilibrium. An extended study with different initial schemes and more realistic kinetic (see *e.g.* [236]) closure conditions could shed more light on the energetic aspects of the process, which we haven't explored here. Moreover, the choice of a perfectly symmetric, mean field spherical packing force is another weakness of our setup: a different choice, with for example the explicit use of capsomers-like units (pentamers and hexamers), could introduce less artifacts. Graph-based analyses, based on contact matrices that exhibit weaknesses in extreme regimes like these (when built using geometric criteria), have yielded results that can be hardly interpreted, except for the following. By looking at the behaviours of D and λ_1 in time (figures 6.13b and 6.12b), it seems that the 0.15 M case suffers substantially more the packing protocol, with respect to the 0.50 M. In fact, the trend of the above-mentioned observables is clearly affected in the 0.15 M case, while almost untouched in the 0.50 M one (excluding the last frames that probably contain artifacts, as already discussed). *We are led to conclude that the Coulombic repulsion could be one of the main players in the emergence of graph topologies and their connectivity, as the direct consequence of changing the Debye screening factor leads to opposite behaviours in our simulations.* As previously mentioned in the text, exploring the behavior of such topologies by constructing graphs based on energy criteria to establish hydrogen bond formation will be equally interesting.

Regarding the simulations of capsid and virion, the first thing we can infer is that a 200ns sampling is not sufficient to stabilize the value of the radius of gyration. Other works involving atomistic MD simulations of viral systems [249, 243, 250] demonstrate that longer sampling (on the order of microseconds at least) allows more pronounced convergence of observables. The high resolution of these simulations and the vast amount of information that can be obtained from them make them very useful for studying microscopic properties of contact between capsid tails and RNA, as partially shown in this study. A major open question on this topic, for instance, is the validity of the Hamiltonian Path Hypothesis proposed by Dykeman and coworkers [251]. See the next paragraph for a discussion on how simulations of this kind could be exploited to investigate its validity. My simulations show the occurrence of solvent instability inside the capsid, which might be related to a subsequent instability of the capsid structure itself. In short test simulations not reported in this Thesis, we attempted setting up and equilibrating solvent with a shell-only capsid simulation (without the tails), and the vacuum region did not form. This further confirms that the cause resides in the presence of the tails (and, in turn, in the high charge of the N-terminal tails) and the phosphate groups of RNA2 in the simulations mentioned above. The claim that we can draw from the capsid/virion simulations is that *plain MD simulations with a protonation state assigned at the beginning without the ability to update based on the local electrostatic environment (i.e., non-constant pH simulations) are likely not suited for systems with critical charge conditions, such as those studied here.*

The studies conducted here to test the applicability of the CANVAS model to the modeling of the trimer, potentially useful in creating a multi-resolution model of capsid and virion, have yielded satisfactory results. We can conclude that the model, coupled with the implicit solvent, is justified for extended studies of the dynamics of the aforementioned systems. The only drawback lies in computational efficiency: utilizing LAMMPS implemented on CPU for implicit solvent simulations drastically reduces efficiency to the point of risking being less efficient than the same simulation performed in GROMACS with implicit solvent, as shown in table 6.3. The observations made on the pathological solvent behavior in the all-atom simulations of capsid and virion would naturally vanish with an implicit solvent model: it will certainly be interesting to compare the behavior of the capsid with and without RNA2 under these new conditions, to determine whether the formation of solvent voids affects the solute dynamics and to what extent.

6.6 Perspectives

Here we report all the ideas that we came up with, during the planning, execution, revision and writing processes of the 3 projects discussed in this chapter. We tried to be exhaustive, realizing that it would probably take more than one lifetime to explore all of them.

- Expand the sampling of all simulations, both with different initial conditions and longer simulation times.

- Review some of the protocols: perform a slower RNA packing, improve the kinetic analysis by incorporating experimental information, study other components of the capsid instead of just the trimer (capsomers like pentamers or hexamers, as done in [238] for the salt-stable CCMV mutant).

- For RNA2: use a different criterion for calculating HB contacts; quantitatively compare the contacts obtained from simulations with contacts predicted by Vienna or other secondary structure prediction software.

- For RNA2: quantify the work done by the spherical shell to understand the extent of non-equilibrium processes and investigate the variation in free energy during packing. This could provide insights into the energetic of the self-assembly process (entropy vs. internal energy) and also shed light on the influence of the environment on this process. Address questions such as why the stable state during assembly is the virion, whereas during infection, the virion ruptures and releases all RNA into the new host cell. Determine if it is a matter of thermodynamic equilibrium between two states or if there's a change in the environment (e.g., pH, salt concentration) triggering the rupture.

- For the capsid and virion: study the asymmetry of motion, similar to what has been done for HBV [243], by comparing the RMSF (root mean square fluctuations) of individual CPs or dimers aligned with the trajectory, or with respect to the overall mediated structure, or with respect to the mediated structure of the same individual CP/dimer. A 5-fold symmetric behavior would exhibit RMSF symmetrically distributed over the structure, whereas in the case of HBV, it was observed that in regions where the RMSF should be the same due to symmetry, this is not the case.

- For the capsid and virion: investigate the acoustic properties of C_α atoms. Cluster the structures based on RMSD, determine the optimal number of clusters, find the medoid for each cluster, and calculate a kind of normalized RMSF with respect to the medoid for each C_α. Then, generate the temporal series of this RMSF-like data, perform FFT (fast Fourier transform) to identify frequencies with higher amplitudes.

- For the capsid and virion: thoroughly investigate the issue of solvent hole formation. Conduct alchemical transformations to understand how this effect correlates with the presence of an extremely charged environment (considering constant-pH MD for valuable insights). Calculate hydrodynamic pressure with the virial and perform short simulations with larger simulation boxes to determine if this affects the solvent hole formation (i.e., the water hole effect at the boundary).

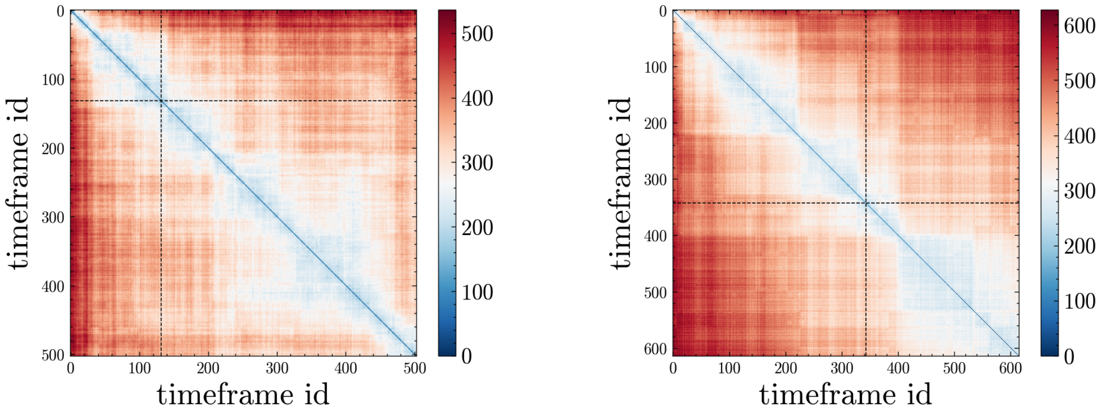
- For the capsid and virion: find a way how to use my simulations to investigate the above-mentioned Hamiltonian Path Hypothesis. It is a proposed concept that attempts to explain the contacts between viral RNA and the capsid in the context of virus assembly. This hypothesis suggests that the RNA molecules within a virus, particularly single-stranded RNA viruses, might adopt a path along the inner surface of the capsid that corresponds to a Hamiltonian path – a path that visits each vertex exactly once in a graph. In this case, the vertices represent distinct sites on the inner surface of the capsid, and the path would symbolize the physical route taken by the viral RNA as it interacts with the capsid. The hypothesis posits that the viral RNA adopts a specific conformation that allows it to interact with the inner surface of the capsid in a way that maximizes the formation of stabilizing interactions, such as hydrogen bonds and electrostatic interactions. This could contribute to the overall stability of the viral structure and aid in the process of self-assembly during viral replication. It suggests that the RNA sequence encodes not only the genetic information for viral replication but also a physical template for how the RNA should interact with the capsid during assembly. One idea would be to perform multiple MD simulation of the relaxation process alone of the RNA inside the capsid (as it happens in the very first nanoseconds of the virion run), by starting from initial configurations with different relative orientations of the RNA and the inner surface of the capsid: this way, it would be possible to collect statistics about the multiple ways in which RNA adapts and relax inside the capsid and by locating the disposition of the contacts we would be able test the hypothesis.

- For the capsid and virion: perform MARTINI simulations for both components. MARTINI simulations of the capsid without the tails have been already done [237], but for the virion, a model with fixed RNA2 secondary structure using an elastic network model can be used to simulate the virion stability for long times. In this regard, it would be extremely interesting to test the experimentally determined phase diagram for CCMV self-assembled structures reported in [252], which involves multiple combinations of pH and salt concentration conditions: also here, a constant-pH protocol would be required.

- For the capsid and virion: perform CANVAS simulations for the entire capsid and the entire virion, in order to test the applicability to the model to a huge and complex system.

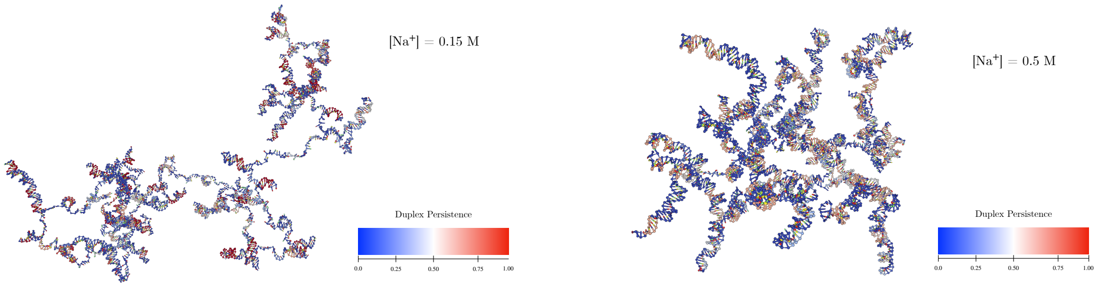
6.7 Appendix A: RNA2 simulations

Additional Figures



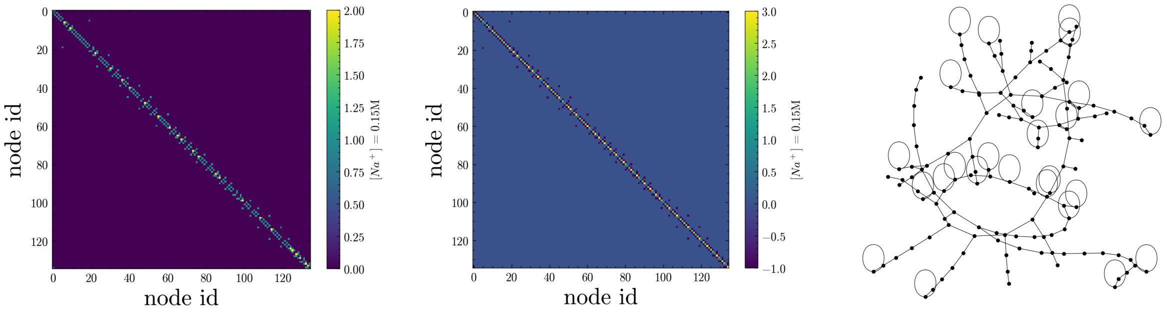
(a) **(b)**

Figure 6.30: Distances d_{KM} among every couples of the frames used to perform the hierarchical clustering.



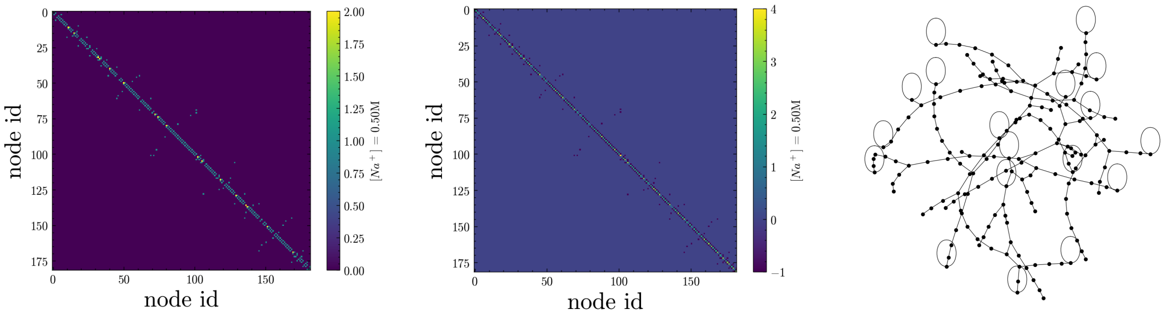
(a) **(b)**

Figure 6.31: Structures of the last configurations obtained in the free folding runs, both for the **(a)** 0.15 M and the **(b)** 0.50 M salt concentrations. The color map represents the nucleotides' affinity to participate to a duplex, as emerged from our simulations.



(a) **(b)** **(c)**

Figure 6.32: **(a)** Adiacency matrix, **(b)** Laplacian matrix and **(c)** a 2D pictorial representation of the graph corresponding to the structure shown in figure 6.31a.



(a) **(b)** **(c)**

Figure 6.33: **(a)** Adiacency matrix, **(b)** Laplacian matrix and **(c)** a 2D pictorial representation of the graph corresponding to the structure shown in figure 6.31b.

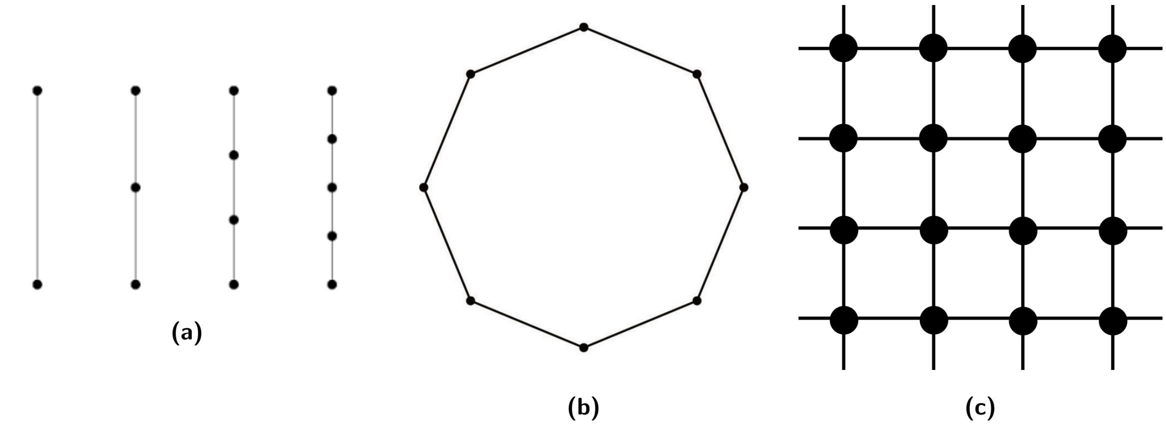


Figure 6.34: Examples of simple graph topologies: **(a)** 4 different path graphs P_n; **(b)** one circular graph C_n with $n = 8$ nodes; **(c)** one 2D lattice graph L_n with $n = 16$ nodes.

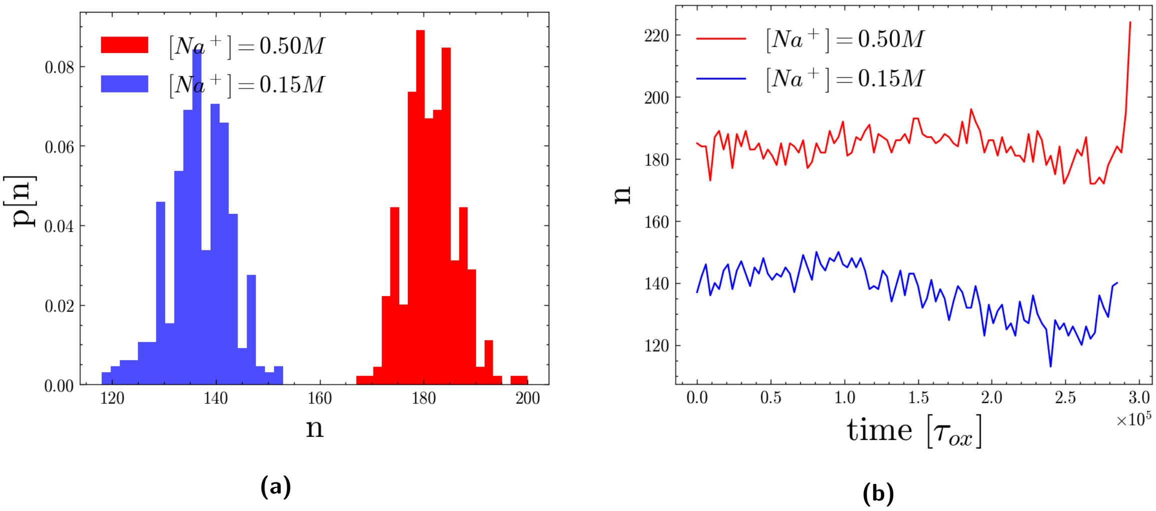


Figure 6.35: **(a)** Normalized histograms of the values of n (number of nodes) relative to the equilibrated parts of the free folding simulations; **(b)** values of n in time relative to the packing simulations.

Additional Tables

Folding (1 GPU)	**Packing (48 CPU cores)**
$1.56 \cdot 10^6 \tau_{ox}/\text{day}$	$2.86 \cdot 10^4 \tau_{ox}/\text{day}$

Table 6.1: Performances of the simulations by using respectively the native oxDNA software (Folding) and the LAMMPS implementation of the oxRNA2 model (Packing).

Simulation unit	**Physical unit**
1 unit of length (d_{ox})	8.4 Å
1 unit of time (τ_{ox})	3.06 ps

Table 6.2: Summary of the two units of measurement used in this work, from the oxRNA units to the IS units. Taken from the oxRNA wiki.

6.8 Appendix B: Capsid and virion simulations

Setup of the Capsid all-atom Simulation

Here we summarize the practical steps followed to setup the all-atom simulation of the viral capsid CCMV with GROMACS (v2018), for the sake of reproducibility of the results presented in the chapter.

Building the capsid structure from the trimer

1. We downloaded the PDB structure 1CWP [219]. **NB** on the PDB one can find both the PDB containing the structures of chain A, chain B and chain C along with a small piece of (presumed) RNA and the whole structure of the assembled capsid in its icosaedral geometry.

2. Starting from the **trimer** contained in the PDB, we removed the pre-existing RNA fragments and we used CHIMERA [253] as visual interface for MODELLER [164] to add the missing residues in the tails of the N-termini.

    - 26 missing for chain B and C and 42 missing for chain A
    - we used the MODELLER tool "Model/Refine Loops" from CHIMERA's interface

- we did not modify the residues already present in the chains
- we generated 5 models for the structure and kept the one with higher score values

3. We created a box and we performed energy minimization of the structure inside the box (in vacuum, *i.e.* protein only, constraining everything BUT the N-termini), as suggested in Bonvin's website, in order to relax the tails.

4. we took the minimized structure, solvated it and added salt (0.15M of NaCl).

5. we performed another energy minimization (restraining everything BUT the N-termini).

6. we performed an NVT equilibration run of 1ns (restraining everything BUT the N-termini).

7. we performed an NPT equilibration run of 200ps (restraining everything BUT the N-termini).

8. we took the frame with the smallest Radius of Gyration from the NPT trajectory and we performed another ∼10ns-long NVT run, with the same conditions.

9. From the last simulation, we extracted one of the 3 monomers that appeared to be the one with the "most reasonable" N-terminus tail. Reasonable means with a tail's volume/structure that does not risk to interfere with neighbor capsomers.

10. **we used this capsomer to build the whole capsid**:

    - with the "MatchMaker" function included in CHIMERA, we superimposed three copies of the extracted monomer to each monomer contained in the trimer's structure of PDB 1CWP
    - we built a trimer with these 3 copies
    - we built the full capsid starting from the 60 trimers contained in the same PDB file 1CWP (without the N-terminal tails)
    - with the "MatchMaker" function included in CHIMERA, we superimposed each trimer contained in the full capsid's structure of PDB 1CWP, to the trimer built in the previous point

11. we inizialized the system's topology using CHARMM36m force field both for proteins and water (TIP3P suggested model)

Energy minimizations of the whole capsid

1. we created a box of dodecahedral shape, with $d = 1.2nm \equiv r_{cut}$ for the non-bonded interactions
2. we performed three energy minimizations (algorithm: steepest decent), with $|\mathbf{F}_{max}| \leq F_{tol} = 5000kJ/(mol \cdot nm)$, $|\mathbf{F}_{max}| \leq F_{tol} = 1000kJ/(mol \cdot nm)$ and then $|\mathbf{F}_{max}| \leq F_{tol} = 100kJ/(mol \cdot nm)$
3. we added solvent molecules
4. **Ionization**: we divided the solvated system into two regions: an inner sphere of radius $11nm$ and the rest of the simulation box. we manually inserted ions to neutralize these two regions separately and then we added randomly in the box a number of remaining atoms needed to reach the desired concentration (Na$^+$ and Cl$^-$ at 0.15M) and neutrality.
5. we performed another three energy minimizations, with the same parameters, onto the full solvated system

Restrained Molecular Dynamics

Given the dodecahedral box, filled with both the capsid and the solvent structures minimized, the next step consists in heating up the system to the desired temperature ($T_{md} = 300\,K$ in our case). To do so, we chose to couple water and the capsid with two different thermostats and we performed an annealing procedure, bringing the system from $0\,K$ to $300\,K$ in $10\,ps$.

1. we heated the system (restraining the capsid) from $0\,K$ to $300\,K$ via the annealing protocol in GROMACS
   - **No strange empty holes in the center of the system up to here**
2. we performed a restrained MD simulation of $100ps$ in the NVT ensemble, to equilibrate the water at the desired temperature, using $(k_x, k_y, k_z) = (1000, 1000, 1000)\dfrac{kJ}{mol \cdot nm^2}$
   - **$\sim$5nm-radius sphere of empty space generates at the end of this run. We hypothesize that the migration of ions due to the electrostatic interactions with the tails of the capsid drags also solvation's shells**
3. we performed a restrained MD simulation of $500ps$ in the NPT ensemble, to equilibrate the water at the desired pressure, using $(k_x, k_y, k_z) = (1000, 1000, 1000)\dfrac{kJ}{mol \cdot nm^2}$

- **Same hole as before is present at the end of this run**

4. we relaxed the retrains with a restrained MD simulation of $500ps$ in the NPT ensemble, using $(k_x, k_y, k_z) = (100, 100, 100)\frac{kJ}{mol \cdot nm^2}$

After the equilibration of the solvent, we performed a 200ns-long unrestrained molecular dynamics simulation of the whole system.

Setup of the Virion all-atom Simulation

In order to construct the initial configuration for the virion simulations, we took as starting step the capsid structure (capsid molecules with the N terminals) as obtained from the procedure described in the above paragraph "Building the capsid structure from the trimer". Then the following steps are followed, in order to insert the RNA2 structure:

1. we made a 1ns-long steered molecular dynamics run to pull the center of mass of the last amino acid of each N-terminal tail of the capsid molecules towards the inner surface of the capsid shell. This step was required to create more free space inside the capsid, in order to setup the subsequent manual insertion of the atomistic structure of the squeezed RNA2 fragment: otherwise, no back-mapped structures obtained from the packing procedure revealed to be compatible with the original free space in the capsid.

2. we took the final structure obtained by the packing process of the RNA2 fragment with oxRNA (described in the second section of this chapter) made at $[Na^+] = 0.15M$ and back-mapped it into its DNA-equivalent all-atom version, by using a webtool called *TacoxDNA* [241]

3. we used the software HiRE-RNA [254] to perform the conversion from atomistic DNA to atomistic RNA

4. we relaxed the dihedral angles of the structure by using the YASARA minimization webserver [255]

5. we manually inserted the structure so obtained into the capsid with the pulled tails by using the VMD software

After that, we followed the same steps described for the capsid in the paragraphs "Energy Minimizations of the whole capsid" and "Restrained Molecular Dynamics", by restraining also the RNA2 fragment together with the capsid molecules.

NB: for all the simulations (both of the capsid and the virion) we used PME for the electrostatics and a cut-off of 1.2nm for the Lennard-Jones term.

Additional Figures

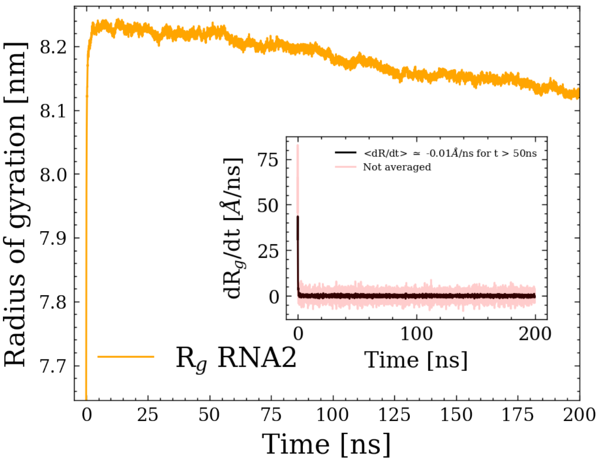


Figure 6.36: Radius of gyration in time calculated for the RNA2 fragment in the virion run.

6.9 Appendix C: Trimer simulations

Additional Figures

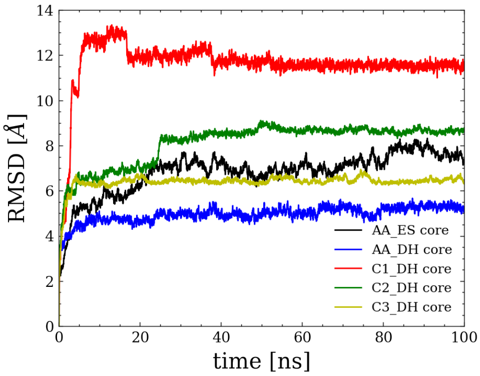
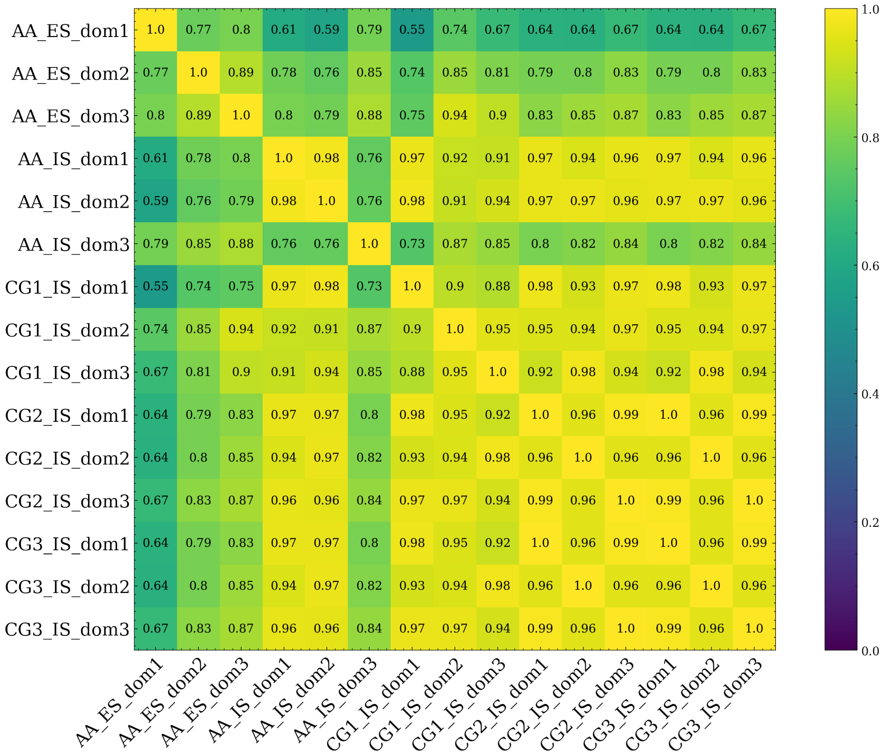


Figure 6.37: RMSD values calculated for the C_α of the residues in the core part of the capsid molecules (tails excluded), with respect to the first frame of the production run.



	AA_ES_dom1	AA_ES_dom2	AA_ES_dom3	AA_IS_dom1	AA_IS_dom2	AA_IS_dom3	CG1_IS_dom1	CG1_IS_dom2	CG1_IS_dom3	CG2_IS_dom1	CG2_IS_dom2	CG2_IS_dom3	CG3_IS_dom1	CG3_IS_dom2	CG3_IS_dom3
AA_ES_dom1	1.0	0.77	0.8	0.61	0.59	0.79	0.55	0.74	0.67	0.64	0.64	0.67	0.64	0.64	0.67
AA_ES_dom2	0.77	1.0	0.89	0.78	0.76	0.85	0.74	0.85	0.81	0.79	0.8	0.83	0.79	0.8	0.83
AA_ES_dom3	0.8	0.89	1.0	0.8	0.79	0.88	0.75	0.94	0.9	0.83	0.85	0.87	0.83	0.85	0.87
AA_IS_dom1	0.61	0.78	0.8	1.0	0.98	0.76	0.97	0.92	0.91	0.97	0.94	0.96	0.97	0.94	0.96
AA_IS_dom2	0.59	0.76	0.79	0.98	1.0	0.76	0.98	0.91	0.94	0.97	0.97	0.96	0.97	0.97	0.96
AA_IS_dom3	0.79	0.85	0.88	0.76	0.76	1.0	0.73	0.87	0.85	0.8	0.82	0.84	0.8	0.82	0.84
CG1_IS_dom1	0.55	0.74	0.75	0.97	0.98	0.73	1.0	0.9	0.88	0.98	0.93	0.97	0.98	0.93	0.97
CG1_IS_dom2	0.74	0.85	0.94	0.92	0.91	0.87	0.9	1.0	0.95	0.95	0.94	0.97	0.95	0.94	0.97
CG1_IS_dom3	0.67	0.81	0.9	0.91	0.94	0.85	0.88	0.95	1.0	0.92	0.98	0.94	0.92	0.98	0.94
CG2_IS_dom1	0.64	0.79	0.83	0.97	0.97	0.8	0.98	0.95	0.92	1.0	0.96	0.99	1.0	0.96	0.99
CG2_IS_dom2	0.64	0.8	0.85	0.94	0.97	0.82	0.93	0.94	0.98	0.96	1.0	0.96	0.96	1.0	0.96
CG2_IS_dom3	0.67	0.83	0.87	0.96	0.96	0.84	0.97	0.97	0.94	0.99	0.96	1.0	0.99	0.96	1.0
CG3_IS_dom1	0.64	0.79	0.83	0.97	0.97	0.8	0.98	0.95	0.92	1.0	0.96	0.99	1.0	0.96	0.99
CG3_IS_dom2	0.64	0.8	0.85	0.94	0.97	0.82	0.93	0.94	0.98	0.96	1.0	0.96	0.96	1.0	0.96
CG3_IS_dom3	0.67	0.83	0.87	0.96	0.96	0.84	0.97	0.97	0.94	0.99	0.96	1.0	0.99	0.96	1.0

Figure 6.38: Mutual Pearson coefficients calculated by comparing the RMSF arrays relative to each monomer (capsid molecule, tails excluded) of each of the 5 simulations performed.

Additional Table

System	MD Engine	Performance (48 cores)
AA ES	GROMACS	25.2 ns/day
AA ES	LAMMPS	4.4 ns/day
AA DH	LAMMPS	25.4 ns/day
C1 DH	LAMMPS	14.2 ns/day
C2 DH	LAMMPS	7.2 ns/day
C3 DH	LAMMPS	8.4 ns/day

Table 6.3: Performances of simulations of the trimer system with (ES) and without (DH) the solvent, by using GROMACS and/or LAMMPS simulation packages. The reduced efficiency of the CANVAS models is probably due to the higher and higher presence of long-range harmonic springs connecting the CG sites in the elastic network model part, as dictated by the CANVAS procedure for model building.

Chapter 7

Conclusions

In this chapter, we will reflect on the advantages and limitations of employing atomistic and multi-resolution molecular dynamics simulations in the study of biomolecules. We will draw upon the conclusions from the various chapters of this thesis to provide a comprehensive overview.

Atomistic Molecular Dynamics Simulations:
Pros:

1. Detailed Insights: Atomistic simulations provide a high level of detail, allowing us to observe molecular interactions, structural changes, and dynamical behaviors at the atomic level. This level of granularity is essential for understanding the fine-grained mechanisms of biomolecular processes.

2. Accurate Energetics: Atomistic simulations offer precise energetic information, enabling us to calculate thermodynamic properties, binding affinities, and reaction pathways accurately. This information is valuable for rational drug design and other applications.

Cons:

1. Computational Intensity: Atomistic simulations are computationally demanding, especially for large biomolecular systems and long timescales. This can limit the feasibility of exploring certain phenomena, such as rare events or long-term conformational changes.

2. Sampling Challenges: Achieving adequate conformational sampling in atomistic simulations can be challenging, particularly for biomolecules with rugged energy landscapes. The

limited timescales accessible with current computational resources may lead to incomplete exploration of phase space.

Multi-Resolution Molecular Dynamics Simulations:
Pros:

1. Enhanced Efficiency: Multi-resolution simulations, which utilize coarse-grained models or other simplifications, offer substantial computational efficiency. They allow the exploration of longer timescales and larger systems, making them suitable for studying complex biological processes.

2. Exploring Large Systems: Multi-resolution approaches enable the investigation of large biomolecular complexes, such as viral capsids or ribosomes, which are often beyond the reach of fully atomistic simulations. This is critical for understanding complex biological systems.

Cons:

1. Loss of Detail: Coarse-grained and multi-resolution models sacrifice some level of structural and energetic detail, which may limit the accuracy of certain analyses. This can be a drawback when precision is essential, as in drug discovery.

2. Model Dependence: The choice of simplification or coarse-graining strategy is critical and must be made carefully. The accuracy of multi-resolution simulations is highly dependent on the quality of the chosen model, and inaccuracies can lead to erroneous conclusions.

Overall, the choice between atomistic and multi-resolution molecular dynamics simulations should be guided by the specific research goals and system characteristics. In our work, we have demonstrated the power of atomistic simulations in revealing detailed molecular insights, such as the behavior of pathogenic mutations in SBDS. These simulations allowed us to understand the impact of these mutations on structural dynamics and binding affinities. However, we also encountered limitations in terms of computational intensity and sampling challenges, particularly when dealing with large-scale conformational changes of the system.

In contrast, our study of chymotrypsin-related proteases showcased the advantages of multi-resolution modeling that identifies common conformational motions across a diverse protein family. The potential efficiency of these simulations will allow to explore a broader range of protein dynamics. However, we acknowledge the trade-off in detail and precision.

In the study of the CCMV virus, we employed multi-resolution techniques to explore large systems. These simulations were instrumental in gaining insights into viral assembly processes, but they also revealed the need for longer timescales to achieve convergence, regarding both the all-atom (capsid and virion) and the coarse-grained (viral genome) simulations.

In conclusion, both atomistic and multi-resolution molecular dynamics simulations are indispensable tools in the study of biomolecular systems. They complement each other and offer a spectrum of insights depending on the research objectives. Future research should continue to harness the strengths of both approaches while addressing their respective limitations, ultimately advancing our understanding of the complex world of biomolecules.

Bibliography

[1] Francis Crick. “Central Dogma of Molecular Biology”. In: *Nature* (1970). DOI: doi.org/10.1038/227561a0.

[2] Bruce Alberts. *Molecular biology of the cell*. Garland science, 2017.

[3] Richard L Schowen. *Principles of biochemistry 2nd ed.(Lehninger, Albert L.; Nelson, David L.; Cox, Michael M.)* 1993.

[4] Bernhard Rupp. *Biomolecular crystallography: principles, practice, and application to structural biology*. Garland Science, 2009.

[5] J. C. Kendrew and et al. “A Three-Dimensional Model of the Myoglobin Molecule Obtained by X-Ray Analysis”. In: *Nature* (1958). DOI: doi.org/10.1038/181662a0.

[6] Kresten Lindorff-Larsen et al. “How fast-folding proteins fold”. In: *Science* 334.6055 (2011), pp. 517–520.

[7] Noora Aho et al. “Scalable constant pH molecular dynamics in GROMACS”. In: *Journal of Chemical Theory and Computation* 18.10 (2022), pp. 6148–6160.

[8] John Jumper and *et al.* “Highly accurate protein structure prediction with AlphaFold”. In: *Nature* (2021). DOI: doi.org/10.1038/s41586-021-03819-2.

[9] Marco De Vivo et al. “Role of Molecular Dynamics and Related Methods in Drug Discovery”. In: *J. Med. Chem.* (2016). DOI: doi.org/10.1021/acs.jmedchem.5b01684.

[10] N. Liguori et al. “Molecular dynamics simulations in photosynthesis”. In: *Photosynth Res* (2020). DOI: doi.org/10.1007/s11120-020-00741-y.

[11] Daan Frenkel and Berend Smit. *Understanding molecular simulation: from algorithms to applications*. Vol. 1. Elsevier, 2001.

[12] Dominik Marx and Jürg Hutter. *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods*. Cambridge University Press, 2009. DOI: 10.1017/CBO9780511609633.

[13] Cliff Peter Burgess. *Introduction to effective field theory*. Cambridge University Press, 2020.

[14] R. Car and M. Parrinello. "Unified Approach for Molecular Dynamics and Density-Functional Theory". In: *Phys. Rev. Lett.* 55 (22 Nov. 1985), pp. 2471–2474. DOI: 10.1103/PhysRevLett.55.2471. URL: https://link.aps.org/doi/10.1103/PhysRevLett.55.2471.

[15] P. Hohenberg and W. Kohn. "Inhomogeneous Electron Gas". In: *Phys. Rev.* 136 (3B Nov. 1964), B864–B871. DOI: 10.1103/PhysRev.136.B864. URL: https://link.aps.org/doi/10.1103/PhysRev.136.B864.

[16] Wendy D Cornell et al. "A second generation force field for the simulation of proteins, nucleic acids, and organic molecules". In: *Journal of the American Chemical Society* 117.19 (1995), pp. 5179–5197.

[17] Marco De Vivo et al. "Role of molecular dynamics and related methods in drug discovery". In: *Journal of medicinal chemistry* 59.9 (2016), pp. 4035–4061.

[18] Luis A. Marqués et al. "Molecular dynamics study of the configurational and energetic properties of the silicon self-interstitial". In: *Phys. Rev. B* 71 (8 Feb. 2005), p. 085204. DOI: 10.1103/PhysRevB.71.085204. URL: https://link.aps.org/doi/10.1103/PhysRevB.71.085204.

[19] A.O. Caldeira and A.J. Leggett. "Path integral approach to quantum Brownian motion". In: *Physica A: Statistical Mechanics and its Applications* 121.3 (1983), pp. 587–616. ISSN: 0378-4371. DOI: https://doi.org/10.1016/0378-4371(83)90013-4.

[20] T. Schneider and E. Stoll. "Molecular-dynamics study of a three-dimensional one-component model for distortive phase transitions". In: *Phys. Rev. B* 17 (3 Feb. 1978), pp. 1302–1322. DOI: 10.1103/PhysRevB.17.1302. URL: https://link.aps.org/doi/10.1103/PhysRevB.17.1302.

[21] Giovanni Bussi, Davide Donadio, and Michele Parrinello. "Canonical sampling through velocity rescaling". In: *The Journal of Chemical Physics* 126.1 (Jan. 2007). ISSN: 0021-9606. DOI: 10.1063/1.2408420. URL: https://doi.org/10.1063/1.2408420.

[22] Randall B Shirts, Scott R Burt, and Aaron M Johnson. "Periodic boundary condition induced breakdown of the equipartition principle and other kinetic effects of finite sample size in classical hard-sphere molecular dynamics simulation". In: *The Journal of chemical physics* 125.16 (2006), p. 164102.

[23] In-Chul Yeh and Gerhard Hummer. "System-size dependence of diffusion coefficients and viscosities from molecular dynamics simulations with periodic boundary conditions". In: *The Journal of Physical Chemistry B* 108.40 (2004), pp. 15873–15879.

[24] Tom Darden, Darrin York, and Lee Pedersen. "Particle mesh Ewald: An Nlog(N) method for Ewald sums in large systems". In: (1993). DOI: doi.org/10.1063/1.464397.

[25] Juliana Ferreira de Oliveira et al. "Structure, dynamics, and RNA interaction analysis of the human SBDS protein". In: *Journal of molecular biology* 396.4 (2010), pp. 1053–1069.

[26] Valentino Bezzerri and Marco Cipolli. "Shwachman-Diamond syndrome: molecular mechanisms and current perspectives". In: *Molecular Diagnosis & Therapy* 23.2 (2019), pp. 281–290.

[27] Camille Shammas et al. "Structural and mutational analysis of the SBDS protein family: insight into the leukemia-associated Shwachman-Diamond syndrome". In: *Journal of Biological Chemistry* 280.19 (2005), pp. 19221–19229.

[28] Abril Gijsbers et al. "Interaction of the GTPase Elongation Factor Like-1 with the Shwachman-Diamond syndrome protein and its missense mutations". In: *International Journal of Molecular Sciences* 19.12 (2018), p. 4012.

[29] Félix Weis et al. "Mechanism of eIF6 release from the nascent 60S ribosomal subunit". In: *Nature structural & molecular biology* 22.11 (2015), pp. 914–919.

[30] Andrew J Finch et al. "Uncoupling of GTP hydrolysis from eIF6 release on the ribosome causes Shwachman-Diamond syndrome". In: *Genes & development* 25.9 (2011), pp. 917–929.

[31] Jarrod A Smith. "MOLMOL: A free biomolecular graphics/analysis package". In: *Genome Biology* 1.2 (2000), pp. 1–4.

[32] Berk Hess. "Convergence of sampling in protein simulations". In: *Phys. Rev. E* 65 (3 Mar. 2002), p. 031910. DOI: 10.1103/PhysRevE.65.031910. URL: https://link.aps.org/doi/10.1103/PhysRevE.65.031910.

[33] Elena Spinetti et al. "A comparative molecular dynamics study of selected point mutations in the Shwachman–Bodian–Diamond syndrome protein SBDS". In: *International Journal of Molecular Sciences* 23.14 (2022), p. 7938.

[34] Jianhua Lin. "Divergence measures based on the Shannon entropy". In: *IEEE Transactions on Information theory* 37.1 (1991), pp. 145–151.

[35] Ziv Bar-Joseph, David K Gifford, and Tommi S Jaakkola. "Fast optimal leaf ordering for hierarchical clustering". In: *Bioinformatics* 17.suppl_1 (2001), S22–S29.

[36] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.

[37] Bernhard Knapp, Luis Ospina, and Charlotte M Deane. "Avoiding false positive conclusions in molecular simulation: the importance of replicas". In: *Journal of Chemical Theory and Computation* 14.12 (2018), pp. 6127–6138.

[38] Shunzhou Wan et al. "Rapid, precise, and reproducible prediction of peptide–MHC binding affinities from molecular dynamics that correlate well with experiment". In: *Journal of chemical theory and computation* 11.7 (2015), pp. 3346–3356.

[39] William Humphrey, Andrew Dalke, and Klaus Schulten. "VMD: Visual molecular dynamics". In: *Journal of Molecular Graphics* 14.1 (1996), pp. 33–38. ISSN: 0263-7855. DOI: https://doi.org/10.1016/0263-7855(96)00018-5. URL: https://www.sciencedirect.com/science/article/pii/0263785596000185.

[40] Mark James Abraham et al. "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers". In: *SoftwareX* 1 (2015), pp. 19–25.

[41] Kresten Lindorff-Larsen et al. "Improved side-chain torsion potentials for the Amber ff99SB protein force field". In: *Proteins: Structure, Function, and Bioinformatics* 78.8 (2010), pp. 1950–1958.

[42] William L Jorgensen et al. "Comparison of simple potential functions for simulating liquid water". In: *The Journal of chemical physics* 79.2 (1983), pp. 926–935.

[43] Michele Parrinello and Aneesur Rahman. "Polymorphic transitions in single crystals: A new molecular dynamics method". In: *Journal of Applied physics* 52.12 (1981), pp. 7182–7190.

[44] Martin K. Scherer et al. "PyEMMA 2: A Software Package for Estimation, Validation, and Analysis of Markov Models". In: *Journal of Chemical Theory and Computation* 11 (Oct. 2015), pp. 5525–5542. ISSN: 1549-9618. DOI: 10.1021/acs.jctc.5b00743. URL: http://dx.doi.org/10.1021/acs.jctc.5b00743 (visited on 10/19/2015).

[45] Naveen Michaud-Agrawal et al. "MDAnalysis: A toolkit for the analysis of molecular dynamics simulations". In: *Journal of Computational Chemistry* 32.10 (2011), pp. 2319–2327. DOI: https://doi.org/10.1002/jcc.21787. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.21787.

[46] Joshua A Anderson, Chris D Lorenz, and Alex Travesset. "General purpose molecular dynamics simulations fully implemented on graphics processing units". In: *Journal of computational physics* 227.10 (2008), pp. 5342–5359.

[47] Peter H Colberg and Felix Höfling. "Highly accelerated simulations of glassy dynamics using GPUs: Caveats on limited floating-point precision". In: *Computer Physics Communications* 182.5 (2011), pp. 1120–1129.

[48] Peter L Freddolino et al. "Challenges in protein-folding simulations". In: *Nature physics* 6.10 (2010), pp. 751–758.

[49] Ajit Singh and Harwant Singh. "Time-scale and nature of radiation-biological damage: approaches to radiation protection and post-irradiation therapy". In: *Progress in biophysics and molecular biology* 39 (1982), pp. 69–107.

[50] Michael Tsabar et al. "Connecting timescales in biology: can early dynamical measurements predict long-term outcomes?" In: *Trends in cancer* 7.4 (2021), pp. 301–308.

[51] Curtis G Callan Jr. "Broken scale invariance in scalar field theory". In: *Physical Review D* 2.8 (1970), p. 1541.

[52] Leo P Kadanoff. "Scaling laws for Ising models near T c". In: *Physics Physique Fizika* 2.6 (1966), p. 263.

[53] Tomoyuki Kinjo and Shi-aki Hyodo. "Equation of motion for coarse-grained simulation based on microscopic description". In: *Phys. Rev. E* 75 (5 May 2007), p. 051109. doi: 10.1103/PhysRevE.75.051109. url: https://link.aps.org/doi/10.1103/PhysRevE.75.051109.

[54] R. Zwanzig. *Nonequilibrium Statistical Mechanics 3rd ed.* Oxford University Press, 2001.

[55] W. G. Noid et al. "The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models". In: *The Journal of Chemical Physics* 128.24 (June 2008). issn: 0021-9606. doi: 10.1063/1.2938860. url: https://doi.org/10.1063/1.2938860.

[56] W. G. Noid. "Perspective: Coarse-grained models for biomolecular systems". In: *The Journal of Chemical Physics* 139.9 (Sept. 2013). issn: 0021-9606. doi: 10.1063/1.4818908. url: https://doi.org/10.1063/1.4818908.

[57] Robert Zwanzig. "Nonlinear generalized Langevin equations". In: *Journal of Statistical Physics* 9.3 (1973), pp. 215–220.

[58] Kari Gaalswyk, Ernest Awoonor-Williams, and Christopher N Rowley. "Generalized Langevin methods for calculating transmembrane diffusivity". In: *Journal of Chemical Theory and Computation* 12.11 (2016), pp. 5609–5619.

[59] Jing-Tao Lü et al. "Semi-classical generalized Langevin equation for equilibrium and nonequilibrium molecular dynamics simulation". In: *Progress in Surface Science* 94.1 (2019), pp. 21–40.

[60] Roberto Menichetti and Andrea Pelissetto. "Comparing different coarse-grained potentials for star polymers". In: *The Journal of Chemical Physics* 138.12 (2013).

[61] Roberto Menichetti et al. "Integral equation analysis of single-site coarse-grained models for polymer–colloid mixtures". In: *Molecular Physics* 113.17-18 (2015), pp. 2629–2642.

[62] Patrick Diggins IV et al. "Optimal coarse-grained site selection in elastic network models of biomolecules". In: *Journal of chemical theory and computation* 15.1 (2018), pp. 648–664.

[63] Marco Giulini et al. "An Information-Theory-Based Approach for Optimal Model Reduction of Biomolecules". In: *J. Chem. Theory Comput.* (2020). DOI: doi.org/10.1021/acs.jctc.0c00676.

[64] M. Scott Shell. "The relative entropy is fundamental to multiscale and inverse thermodynamic problems". In: *The Journal of Chemical Physics* 129.14 (Oct. 2008). 144108. ISSN: 0021-9606. DOI: 10.1063/1.2992060. URL: https://doi.org/10.1063/1.2992060.

[65] Jiang Wang et al. "Machine learning of coarse-grained molecular dynamics force fields". In: *ACS central science* 5.5 (2019), pp. 755–767.

[66] Siewert J Marrink et al. "The MARTINI force field: coarse grained model for biomolecular simulations". In: *The journal of physical chemistry B* 111.27 (2007), pp. 7812–7824.

[67] Saeed Najafi and Raffaello Potestio. "Folding of small knotted proteins: Insights from a mean field coarse-grained model". In: *The Journal of chemical physics* 143.24 (2015), 12B606_1.

[68] Gregory L Dignon et al. "Temperature-controlled liquid–liquid phase separation of disordered proteins". In: *ACS central science* 5.5 (2019), pp. 821–830.

[69] Sebastian Kmiecik et al. "Coarse-grained protein models and their applications". In: *Chemical reviews* 116.14 (2016), pp. 7898–7936.

[70] Sergei Izvekov and Gregory A. Voth. "A Multiscale Coarse-Graining Method for Biomolecular Systems". In: *J. Phys. Chem. B* (2005). DOI: doi.org/10.1021/jp044629q.

[71] Izvekov S et al. "Effective force fields for condensed phase systems from ab initio molecular dynamics simulation: a new method for force-matching." In: *J Chem Phys.* (2004). DOI: 10.1063/1.1739396.

[72] Solomon Kullback and Richard A Leibler. "On information and sufficiency". In: *The annals of mathematical statistics* 22.1 (1951), pp. 79–86.

[73] Joseph F. Rudzinski and W. G. Noid. "Coarse-graining entropy, forces, and structures". In: *J. Chem. Phys.* (2011). DOI: 10.1063/1.3663709.

[74] W. Tscho et al. “Simulation of polymer melts I. Coarse-graining procedure for polycarbonates”. In: *Acta Polymerica* (1998). DOI: 10.1002/(SICI)1521-4044(199802).

[75] Florian Muller-Plathe Priv.-Doz. Dr. “Coarse-Graining in Polymer Simulation: From the Atomistic to the Mesoscopic Scale and Back”. In: *ChemPhysChem* (2002).

[76] Alexander P. Lyubartsev and Aatto Laaksonen. “Calculation of effective interaction potentials from radial distribution functions: A reverse Monte Carlo approach”. In: *Phys. Rev. E* 52 (4 Oct. 1995), pp. 3730–3737. DOI: 10.1103/PhysRevE.52.3730. URL: https://link.aps.org/doi/10.1103/PhysRevE.52.3730.

[77] Taketomi Hiroshi, Ueda Yuzo, and Nobuhiro. “Studies on protein folding, unfolding and fluctuations by computer simulation”. In: *Int. J. Peptide Protein Res.* (1975).

[78] P. Faccioli et al. “Dominant Pathways in Protein Folding”. In: *Physical Review Letters* (2006).

[79] Shoji Takada. “Gō-ing for the prediction of protein folding mechanisms”. In: *Proceedings of the National Academy of Sciences* 96.21 (1999), pp. 11698–11700.

[80] Ronald D Hills Jr and Charles L Brooks III. “Insights from coarse-grained Gō models for protein folding and dynamics”. In: *International journal of molecular sciences* 10.3 (2009), pp. 889–905.

[81] Shoji Takada. “Gō model revisited”. In: *Biophysics and physicobiology* 16 (2019), pp. 248–255.

[82] Monique M. Tirion. “Large Amplitude Elastic Motions in Proteins from a Single-Parameter, Atomic Analysis”. In: *Phys. Rev. Lett.* 77 (9 Aug. 1996), pp. 1905–1908.

[83] Ivet Bahar, Ali Rana Atilgan, and Burak Erman. “Direct evaluation of thermal fluctuations in proteins using a single-parameter harmonic potential”. In: *Folding and Design* 2.3 (1997), pp. 173–181. ISSN: 1359-0278. DOI: https://doi.org/10.1016/S1359-0278(97)00024-2. URL: https://www.sciencedirect.com/science/article/pii/S1359027897000242.

[84] Xavier Periole and Siewert-Jan Marrink. “The Martini coarse-grained force field”. In: *Biomolecular simulations: methods and protocols* (2013), pp. 533–565.

[85] Hualin Li and Alemayehu A Gorfe. "Aggregation of lipid-anchored full-length H-Ras in lipid bilayers: simulations with the MARTINI force field". In: *PloS one* 8.7 (2013), e71018.

[86] Clément et al. Arnarez. "Dry Martini, a Coarse-Grained Force Field for Lipid Membrane Simulations with Implicit Solvent". In: *J. Chem. Theory Comput.* (2015).

[87] Benedict EK Snodin et al. "Introducing improved structural properties and salt dependence into a coarse-grained model of DNA". In: *The Journal of chemical physics* 142.23 (2015), 06B613_1.

[88] Christian Matek et al. "Coarse-grained modelling of supercoiled RNA". In: *The Journal of chemical physics* 143.24 (2015), p. 243122.

[89] Swarup Dey et al. "DNA origami". In: *Nature Reviews Methods Primers* 1.1 (2021), p. 13.

[90] Marco Giulini et al. "From system modeling to system analysis: The impact of resolution level and resolution distribution in the computer-aided investigation of biomolecules". In: *Frontiers in Molecular Biosciences* 8 (2021).

[91] Arieh Warshel and Michael Levitt. "Theoretical studies of enzymic reactions: dielectric, electrostatic and steric stabilization of the carbonium ion in the reaction of lysozyme". In: *Journal of molecular biology* 103.2 (1976), pp. 227–249.

[92] Hai Lin and Donald G Truhlar. "QM/MM: what have we learned, where are we, and where do we go from here?" In: *Theoretical Chemistry Accounts* 117 (2007), pp. 185–199.

[93] Raffaele Fiorentini, Thomas Tarenzi, and Raffaello Potestio. "Fast, Accurate, and System-Specific Variable-Resolution Modeling of Proteins". In: *Journal of Chemical Information and Modeling* 63.4 (2023), pp. 1260–1275.

[94] Astrid F Brandner et al. "Modelling lipid systems in fluid with Lattice Boltzmann Molecular Dynamics simulations and hydrodynamics". In: *Scientific reports* 9.1 (2019), pp. 1–14.

[95] Matej Praprotnik, Luigi Delle Site, and Kurt Kremer. "Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly". In: *The Journal of chemical physics* 123.22 (2005), p. 224106.

[96] Matej Praprotnik, Luigi Delle Site, and Kurt Kremer. “Adaptive resolution scheme for efficient hybrid atomistic-mesoscale molecular dynamics simulations of dense liquids”. In: *Physical Review E* 73.6 (2006), p. 066701.

[97] Raffaello Potestio et al. “Hamiltonian Adaptive Resolution Simulation for Molecular Liquids”. In: *Phys. Rev. Lett.* 110 (10 Mar. 2013), p. 108301. doi: 10.1103/PhysRevLett.110.108301. url: https://link.aps.org/doi/10.1103/PhysRevLett.110.108301.

[98] Karsten Kreis et al. “Advantages and challenges in coupling an ideal gas to atomistic models in adaptive resolution simulations”. In: *The European Physical Journal Special Topics* 224 (2015), pp. 2289–2304.

[99] Raffaello Potestio et al. “Monte Carlo Adaptive Resolution Simulation of Multicomponent Molecular Liquids”. In: *Phys. Rev. Lett.* 111 (6 Aug. 2013), p. 060601. doi: 10.1103/PhysRevLett.111.060601. url: https://link.aps.org/doi/10.1103/PhysRevLett.111.060601.

[100] CC Wang, JY Tan, and LH Liu. “Hamiltonian adaptive resolution molecular dynamics simulation of infrared dielectric functions of liquids”. In: *Journal of Applied Physics* 123.20 (2018), p. 205103.

[101] Thomas Tarenzi et al. “Open boundary simulations of proteins and their hydration shells by Hamiltonian adaptive resolution scheme”. In: *Journal of chemical theory and computation* 13.11 (2017), pp. 5647–5657.

[102] Alexey Onufriev. “Implicit solvent models in molecular dynamics simulations: A brief overview”. In: *Annual Reports in Computational Chemistry* 4 (2008), pp. 125–137.

[103] Alexey V Onufriev and David A Case. “Generalized Born implicit solvent models for biomolecules”. In: *Annual review of biophysics* 48 (2019), pp. 275–296.

[104] Alexey Onufriev. “The generalized Born model: its foundation, applications, and limitations”. In: *Departments of Computer Science and Physics, Blacksburg, Virginia, USA* (2010).

[105] Thomas Tarenzi et al. “In search of a dynamical vocabulary: a pipeline to construct a basis of shared traits in large-scale motions of proteins”. In: *Applied Sciences* 12.14 (2022), p. 7157.

[106] Herman JC Berendsen and Steven Hayward. “Collective protein dynamics in relation to function”. In: *Current opinion in structural biology* 10.2 (2000), pp. 165–169.

[107] Katherine A Henzler-Wildman et al. “A hierarchy of timescales in protein dynamics is linked to enzyme catalysis”. In: *Nature* 450.7171 (2007), pp. 913–916.

[108] Chitra Narayanan et al. “Applications of NMR and computational methodologies to study protein dynamics”. In: *Archives of biochemistry and biophysics* 628 (2017), pp. 71–80.

[109] Buyong Ma et al. “Folding funnels and binding mechanisms”. In: *Protein engineering* 12.9 (1999), pp. 713–720.

[110] Ruth Nussinov and Buyong Ma. “Protein dynamics and conformational selection in bidirectional signal transduction”. In: *BMC biology* 10.1 (2012), pp. 1–5.

[111] DE Koshland Jr. “Application of a theory of enzyme specificity to protein synthesis”. In: *Proceedings of the National Academy of Sciences of the United States of America* 44.2 (1958), p. 98.

[112] Fabian Paul and Thomas R Weikl. “How to distinguish conformational selection and induced fit based on chemical relaxation rates”. In: *PLOS Computational Biology* 12.9 (2016), e1005067.

[113] Li-Quan Yang et al. “Protein dynamics and motions in relation to their functions: several case studies and the underlying mechanisms”. In: *Journal of Biomolecular Structure and Dynamics* 32.3 (2014), pp. 372–393.

[114] Ulf Hensen et al. “Exploring protein dynamics space: the dynasome as the missing link between protein structure and function”. In: *PloS one* 7.5 (2012), e33931.

[115] Dorothee Kern and Erik RP Zuiderweg. “The role of dynamics in allosteric regulation”. In: *Current opinion in structural biology* 13.6 (2003), pp. 748–757.

[116] Yan Zhang et al. “Intrinsic dynamics is evolutionarily optimized to enable allosteric behavior”. In: *Current opinion in structural biology* 62 (2020), pp. 14–21.

[117] Zhongjie Liang, Gennady M Verkhivker, and Guang Hu. “Integration of network models and evolutionary analysis into high-throughput modeling of protein dynamics and allosteric regulation: theory, tools and applications”. In: *Briefings in Bioinformatics* 21.3 (2020), pp. 815–835.

[118] Manel A Balsera et al. “Principal component analysis and long time protein dynamics”. In: *The Journal of Physical Chemistry* 100.7 (1996), pp. 2567–2572.

[119] Sarah A Mueller Stein et al. “Principal components analysis: a review of its application on molecular dynamics data”. In: *Annual Reports in Computational Chemistry* 2 (2006), pp. 233–261.

[120] Sebastian Kmiecik et al. “Modeling of protein structural flexibility and large-scale dynamics: Coarse-grained simulations and elastic network models”. In: *International journal of molecular sciences* 19.11 (2018), p. 3496.

[121] Joseph A Marsh and Sarah A Teichmann. “Parallel dynamics and evolution: Protein conformational fluctuations and assembly reflect evolutionary changes in sequence and structure”. In: *BioEssays* 36.2 (2014), pp. 209–218.

[122] Taisong Zou et al. “Evolution of conformational dynamics determines the conversion of a promiscuous generalist into a specialist enzyme”. In: *Molecular biology and evolution* 32.1 (2015), pp. 132–143.

[123] Chitra Narayanan et al. “Conservation of dynamics associated with biological function in an enzyme superfamily”. In: *Structure* 26.3 (2018), pp. 426–436.

[124] She Zhang et al. “Shared signature dynamics tempered by local fluctuations enables fold adaptability and specificity”. In: *Molecular biology and evolution* 36.9 (2019), pp. 2053–2068.

[125] Karolina Mikulska-Ruminska et al. “Characterization of differential dynamics, specificity, and allostery of lipoxygenase family members”. In: *Journal of chemical information and modeling* 59.5 (2019), pp. 2496–2508.

[126] Neeraj K Gaur et al. “Evolutionary conservation of protein dynamics: insights from all-atom molecular dynamics simulations of ‘peptidase’domain of Spt16”. In: *Journal of Biomolecular Structure and Dynamics* (2021), pp. 1–13.

[127] Sandra Maguid, Sebastian Fernandez-Alberti, and Julian Echave. “Evolutionary conservation of protein vibrational dynamics”. In: *Gene* 422.1-2 (2008), pp. 7–13.

[128] Javier A Velázquez-Muriel et al. “Comparison of molecular dynamics and superfamily spaces of protein domain deformation”. In: *BMC structural biology* 9.1 (2009), pp. 1–14.

[129] Frances Pearl et al. “The CATH Domain Structure Database and related resources Gene3D and DHS provide comprehensive domain family information for genome analysis”. In: *Nucleic acids research* 33.suppl_1 (2005), pp. D247–D251.

[130] Michael Levitt, Christian Sander, and Peter S. Stern. “Protein normal-mode dynamics: Trypsin inhibitor, crambin, ribonuclease and lysozyme”. In: *Journal of Molecular Biology* 181.3 (1985), pp. 423–447. issn: 0022-2836.

[131] Charles C. David and Donald J. Jacobs. “Principal Component Analysis: A Method for Determining the Essential Dynamics of Proteins”. In: *Protein Dynamics: Methods and Protocols*. Totowa, NJ: Humana Press, 2014, pp. 193–226.

[132] Wenjun Zheng. “Anharmonic normal mode analysis of elastic network model improves the modeling of atomic fluctuations in protein crystal structures”. In: *Biophysical journal* 98.12 (2010), pp. 3025–3034.

[133] Sara E. Dobbins, Victor I. Lesk, and Michael J. E. Sternberg. “Insights into protein flexibility: The relationship between normal modes and conformational change upon protein-protein docking”. In: *Proceedings of the National Academy of Sciences* 105.30 (2008), pp. 10390–10395.

[134] M Delarue and Y.-H Sanejouand. “Simplified Normal Mode Analysis of Conformational Transitions in DNA-dependent Polymerases: the Elastic Network Model”. In: *Journal of Molecular Biology* 320.5 (2002), pp. 1011–1024. issn: 0022-2836.

[135] Alexander E Gorbalenya et al. “Cysteine proteases of positive strand RNA viruses and chymotrypsin-like serine proteases: a distinct protein superfamily with a common structural fold”. In: *FEBS letters* 243.2 (1989), pp. 103–114.

[136] Enrico Di Cera. “Serine proteases”. In: *IUBMB life* 61.5 (2009), pp. 510–515.

[137] Aparna Laskar et al. “Modeling and structural analysis of PA clan serine proteases”. In: *BMC research notes* 5.1 (2012), pp. 1–11.

[138] Heli AM Mönttinen, Janne J Ravantti, and Minna M Poranen. “Structural comparison strengthens the higher-order classification of proteases related to chymotrypsin”. In: *PloS one* 14.5 (2019), e0216659.

[139] Wenzhe Ma, Chao Tang, and Luhua Lai. “Specificity of trypsin and chymotrypsin: loop-motion-controlled dynamic correlation as a determinant”. In: *Biophysical journal* 89.2 (2005), pp. 1183–1193.

[140] Ricardo J Sola and Kai Griebenow. “Influence of modulated structural dynamics on the kinetics of α-chymotrypsin catalysis: Insights through chemical glycosylation, molecular dynamics and domain motion analysis”. In: *The FEBS journal* 273.23 (2006), pp. 5303–5319.

[141] Pnina Dauber-Osguthorpe et al. “Low frequency motion in proteins: comparison of normal mode and molecular dynamics of streptomyces griseus protease A”. In: *Journal of computational physics* 151.1 (1999), pp. 169–189.

[142] Cristian Micheletti, Paolo Carloni, and Amos Maritan. “Accurate and efficient description of protein vibrational dynamics: comparing molecular dynamics and Gaussian models”. In: *Proteins: Structure, Function, and Bioinformatics* 55.3 (2004), pp. 635–645.

[143] Weizhong Li et al. “Ultrafast clustering algorithms for metagenomic sequence analysis”. In: *Briefings in bioinformatics* 13.6 (2012), pp. 656–668.

[144] Felix Gabler et al. “Protein sequence analysis using the MPI bioinformatics toolkit”. In: *Current Protocols in Bioinformatics* 72.1 (2020), e108.

[145] Liisa Holm and Chris Sander. “The FSSP database: fold classification based on structure-structure alignment of proteins”. In: *Nucleic Acids Research* 24.1 (1996), pp. 206–209.

[146] Janne Ravantti, Dennis Bamford, and David I Stuart. “Automatic comparison and classification of protein structures”. In: *Journal of structural biology* 183.1 (2013), pp. 47–56.

[147] Liisa Holm. “DALI and the persistence of protein shape”. In: *Protein Science* 29.1 (2020), pp. 128–140.

[148] Gregory D Friedland and Tanja Kortemme. “Designing ensembles in conformational and sequence space to characterize and engineer proteins”. In: *Current opinion in structural biology* 20.3 (2010), pp. 377–384.

[149] Eleanor Campbell et al. “The role of protein dynamics in the evolution of new enzyme function”. In: *Nature chemical biology* 12.11 (2016), pp. 944–950.

[150] Marilisa Neri et al. “Coarse-Grained Model of Proteins Incorporating Atomistic Detail of the Active Site”. In: *Phys. Rev. Lett.* 95 (21 Nov. 2005), p. 218102.

[151] Thomas Tarenzi et al. “Open-Boundary Molecular Mechanics/Coarse-Grained Framework for Simulations of Low-Resolution G-Protein-Coupled Receptor–Ligand Complexes”. In: *Journal of Chemical Theory and Computation* 15.3 (2019), pp. 2101–2109.

[152] Aoife C. Fogarty, Raffaello Potestio, and Kurt Kremer. “A multi-resolution model to capture both global fluctuations of an enzyme and molecular recognition in the ligand-binding site”. In: *Proteins: Structure, Function, and Bioinformatics* 84.12 (2016), pp. 1902–1913.

[153] Raffaele Fiorentini, Kurt Kremer, and Raffaello Potestio. “Ligand-protein interactions in lysozyme investigated through a dual-resolution model”. In: *Proteins: Structure, Function, and Bioinformatics* 88.10 (2020), pp. 1351–1360.

[154] Raffaello Potestio et al. “ALADYN: a web server for aligning proteins by matching their large-scale motion”. In: *Nucleic acids research* 38.suppl_2 (2010), W41–W45.

[155] D. Defays. “An efficient algorithm for a complete link method”. In: *The Computer Journal* 20.4 (Jan. 1977), pp. 364–366. ISSN: 0010-4620.

[156] Matteo Marsili, Iacopo Mastromatteo, and Yasser Roudi. “On sampling and modeling complex systems”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2013.09 (2013), P09003.

[157] Ryan John Cubero et al. “Statistical criticality arises in most informative representations”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2019.6 (2019), p. 063402.

[158] Matteo Marsili and Yasser Roudi. “Quantifying relevance in learning and inference”. In: *Physics Reports* 963 (2022), pp. 1–43.

[159] Margherita Mele, Roberto Covino, and Raffaello Potestio. “Information-theoretical measures identify accurate low-resolution representations of protein configurational space”. In: *arXiv preprint arXiv:2205.08437* (2022).

[160] Roi Holtzman, Marco Giulini, and Raffaello Potestio. “Making sense of complex systems through resolution, relevance, and mapping entropy”. In: *arXiv preprint arXiv:2203.00100* (2022).

[161] Ward Cheney and David Kincaid. “Linear algebra: Theory and applications”. In: *The Australian Mathematical Society* 110 (2009), pp. 544–550.

[162] Joost Schymkowitz et al. “The FoldX web server: an online force field”. In: *Nucleic acids research* 33.suppl_2 (2005), W382–W388.

[163] András Fiser, Richard Kinh Gian Do, and Andrej Šali. “Modeling of loops in protein structures”. In: *Protein science* 9.9 (2000), pp. 1753–1773.

[164] Benjamin Webb and Andrej Sali. “Comparative protein structure modeling using MODELLER”. In: *Current protocols in bioinformatics* 54.1 (2016), pp. 5–6.

[165] Giovanni Bussi, Davide Donadio, and Michele Parrinello. “Canonical sampling through velocity rescaling”. In: *The Journal of chemical physics* 126.1 (2007), p. 014101.

[166] Berk Hess et al. “LINCS: a linear constraint solver for molecular simulations”. In: *Journal of computational chemistry* 18.12 (1997), pp. 1463–1472.

[167] Tom Darden, Darrin York, and Lee Pedersen. “Particle mesh Ewald: An N log (N) method for Ewald sums in large systems”. In: *The Journal of chemical physics* 98.12 (1993), pp. 10089–10092.

[168] Robert T McGibbon et al. “MDTraj: a modern open library for the analysis of molecular dynamics trajectories”. In: *Biophysical journal* 109.8 (2015), pp. 1528–1532.

[169] William Humphrey, Andrew Dalke, and Klaus Schulten. “VMD: visual molecular dynamics”. In: *Journal of molecular graphics* 14.1 (1996), pp. 33–38.

[170] Hans Neurath, Kenneth A Walsh, and William P Winter. “Evolution of Structure and Function of Proteases: Amino acid sequences of proteolytic enzymes reflect phylogenetic relationships.” In: *Science* 158.3809 (1967), pp. 1638–1644.

[171] Carlos López-Otín and Judith S Bond. “Proteases: multifunctional enzymes in life and disease”. In: *Journal of Biological Chemistry* 283.45 (2008), pp. 30433–30437.

[172] Lizbeth Hedstrom. “Serine protease mechanism and specificity”. In: *Chemical reviews* 102.12 (2002), pp. 4501–4524.

[173] Sonia Verma, Rajnikant Dixit, and Kailash C Pandey. “Cysteine proteases: modes of activation and future prospects as pharmacological targets”. In: *Frontiers in pharmacology* 7 (2016), p. 107.

[174] Neil D Rawlings, Dominic P Tolle, and Alan J Barrett. "MEROPS: the peptidase database". In: *Nucleic acids research* 32.suppl_1 (2004), pp. D160–D164.

[175] Neil D Rawlings, Alan J Barrett, and Robert Finn. "Twenty years of the MEROPS database of proteolytic enzymes, their substrates and inhibitors". In: *Nucleic acids research* 44.D1 (2016), pp. D343–D350.

[176] Sandra Maguid et al. "Exploring the common dynamics of homologous proteins. Application to the globin family". In: *Biophysical journal* 89.1 (2005), pp. 3–13.

[177] Yi He et al. "Sequence-, structure-, and dynamics-based comparisons of structurally homologous CheY-like proteins". In: *Proceedings of the National Academy of Sciences* 114.7 (2017), pp. 1578–1583.

[178] P Gayathri et al. "Crystal structure of the serine protease domain of Sesbania mosaic virus polyprotein and mutational analysis of residues forming the S1-binding pocket". In: *Virology* 346.2 (2006), pp. 440–451.

[179] Shekeb Khan et al. "Crystal structure of the passenger domain of the Escherichia coli autotransporter EspP". In: *Journal of molecular biology* 413.5 (2011), pp. 985–1000.

[180] Di Sun et al. "Roles of the picornaviral 3C proteinase in the viral life cycle and host cells". In: *Viruses* 8.3 (2016), p. 82.

[181] Hok-Kin Choi et al. "Structural analysis of Sindbis virus capsid mutants involving assembly and catalysis". In: *Journal of molecular biology* 262.2 (1996), pp. 151–167.

[182] Charles C David and Donald J Jacobs. "Characterizing protein motions from structure". In: *Journal of Molecular Graphics and Modelling* 31 (2011), pp. 41–56.

[183] Deshun Lu et al. "Crystal structure of enteropeptidase light chain complexed with an analog of the trypsinogen activation peptide". In: *Journal of molecular biology* 292.2 (1999), pp. 361–373.

[184] Tadaaki Kishi et al. "Crystal structure of neuropsin, a hippocampal protease involved in kindling epileptogenesis". In: *Journal of Biological Chemistry* 274.7 (1999), pp. 4220–4224.

[185] Sarah E St John et al. "Targeting zoonotic viruses: Structure-based inhibition of the 3C-like protease from bat coronavirus HKU4—The likely reservoir host to the human coronavirus that causes Middle East Respiratory Syndrome (MERS)". In: *Bioorganic & medicinal chemistry* 23.17 (2015), pp. 6036–6048.

[186] Jin Xu et al. "Structure basis for the unique specificity of medaka enteropeptidase light chain". In: *Protein & cell* 5.3 (2014), pp. 178–181.

[187] Wolfgang Kabsch and Christian Sander. "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features". In: *Biopolymers: Original Research on Biomolecules* 22.12 (1983), pp. 2577–2637.

[188] Wouter G Touw et al. "A series of PDB-related databanks for everyday needs". In: *Nucleic acids research* 43.D1 (2015), pp. D364–D368.

[189] Jörg Behler and Michele Parrinello. "Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces". In: *Phys. Rev. Lett.* 98 (14 2007), p. 146401. doi: 10.1103/PhysRevLett.98.146401. url: https://link.aps.org/doi/10.1103/PhysRevLett.98.146401.

[190] Yaoyi Chen et al. "Machine learning implicit solvation for molecular dynamics". In: *The Journal of Chemical Physics* 155.8 (2021), p. 084101. doi: 10.1063/5.0059915. url: https://doi.org/10.1063/5.0059915.

[191] Kathryn M. Lebold and W. G. Noid. "Dual approach for effective potentials that accurately model structure and energetics". In: *The Journal of Chemical Physics* 150.23 (2019), p. 234107. doi: 10.1063/1.5094330. eprint: https://doi.org/10.1063/1.5094330. url: https://doi.org/10.1063/1.5094330.

[192] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. isbn: 1441412697.

[193] TensorFlow Developers. "TensorFlow". In: *Zenodo* (2022).

[194] Paul E Smith. "The alanine dipeptide free energy surface in solution". In: *The Journal of chemical physics* 111.12 (1999), pp. 5568–5579.

[195] SG Kalko, E Guardia, and JA Padró. “Molecular dynamics simulation of the hydration of the alanine dipeptide”. In: *The Journal of Physical Chemistry B* 103.19 (1999), pp. 3935–3941.

[196] Hyunbum Jang and Thomas B Woolf. “Multiple pathways in conformational transitions of the alanine dipeptide: an application of dynamic importance sampling”. In: *Journal of computational chemistry* 27.11 (2006), pp. 1136–1141.

[197] Marcello Sega et al. “Quantitative protein dynamics from dominant folding pathways”. In: *Physical review letters* 99.11 (2007), p. 118102.

[198] Didier Devaurs et al. “A multi-tree approach to compute transition paths on energy landscapes”. In: (2013).

[199] Peter J. Steinbach and Bernard R. Brooks. “New spherical-cutoff methods for long-range forces in macromolecular simulation”. In: *Journal of Computational Chemistry* 15.7 (1994), pp. 667–683. DOI: https://doi.org/10.1002/jcc.540150702. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jcc.540150702. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.540150702.

[200] Emilio Gallicchio, Kristina Paris, and Ronald M Levy. “The AGBNP2 implicit solvation model”. In: *Journal of chemical theory and computation* 5.9 (2009), pp. 2544–2564.

[201] Rocco Caliandro, Giulia Rossetti, and Paolo Carloni. “Local Fluctuations and Conformational Transitions in Proteins”. In: *Journal of Chemical Theory and Computation* 8.11 (2012). PMID: 26605630, pp. 4775–4785. DOI: 10.1021/ct300610y. eprint: https://doi.org/10.1021/ct300610y. URL: https://doi.org/10.1021/ct300610y.

[202] Ronny Lorenz et al. “ViennaRNA Package 2.0”. In: *Algorithms for molecular biology* 6 (2011), pp. 1–14.

[203] Sandro Bottaro et al. “Barnaba: software for analysis of nucleic acid structures and trajectories”. In: *RNA* 25.2 (2019), pp. 219–231.

[204] KA Olszewski, L Piela, and HA Scheraga. “Mean field theory as a tool for intramolecular conformational optimization. 2. Tests on the homopolypeptides decaglycine and icosalanine”. In: *The Journal of Physical Chemistry* 97.1 (1993), pp. 260–266.

[205] Steve Plimpton. “Fast Parallel Algorithms for Short-Range Molecular Dynamics”. In: *Journal of Computational Physics* 117.1 (1995), pp. 1–19. issn: 0021-9991. doi: https://doi.org/10.1006/jcph.1995.1039.

[206] Mart M Lamers and Bart L Haagmans. “SARS-CoV-2 pathogenesis”. In: *Nature reviews microbiology* 20.5 (2022), pp. 270–284.

[207] Wang-Shick Ryu. “Virus life cycle”. In: *Molecular virology of human pathogenic viruses* (2017), p. 31.

[208] Alaa AA Aljabali et al. “The viral capsid as novel nanomaterials for drug delivery”. In: *Future science OA* 7.9 (2021), FSO744.

[209] Andrew J Pollard and Else M Bijker. “A guide to vaccinology: from basic principles to new developments”. In: *Nature Reviews Immunology* 21.2 (2021), pp. 83–100.

[210] Stefania Castelletto and Alberto Boretti. “Viral particle imaging by super-resolution fluorescence microscopy”. In: *Chemical Physics Impact* 2 (2021), p. 100013.

[211] Yu Zhang et al. “Application of plant viruses as a biotemplate for nanomaterial fabrication”. In: *Molecules* 23.9 (2018), p. 2311.

[212] Joshua W Wilkerson et al. “Nanoreactors: Strategies to encapsulate enzyme biocatalysts in virus-like particles”. In: *New biotechnology* 44 (2018), pp. 59–63.

[213] Kai Li et al. “Viruses and their potential in bioimaging and biosensing applications”. In: *Analyst* 135.1 (2010), pp. 21–27.

[214] Maria Luisa Izaguirre-Mayoral et al. “Silicon and nitrate differentially modulate the symbiotic performances of healthy and virus-infected Bradyrhizobium-nodulated cowpea (Vigna unguiculata), yardlong bean (V. unguiculata subsp. sesquipedalis) and mung bean (V. radiata)”. In: *Plants* 6.3 (2017), p. 40.

[215] JB Bancroft, GJ Hills, and Rou Markham. “A study of the self-assembly process in a small spherical virus formation of organized structures from protein subunits in vitro”. In: *Virology* 31.2 (1967), pp. 354–379.

[216] Ruben D Cadena-Nava et al. “Self-assembly of viral capsid protein and RNA molecules of different sizes: requirement for a specific high protein/RNA mass ratio”. In: *Journal of virology* 86.6 (2012), pp. 3318–3326.

[217] Rees F Garmann et al. “The assembly pathway of an icosahedral single-stranded RNA virus depends on the strength of inter-subunit attractions”. In: *Journal of molecular biology* 426.5 (2014), pp. 1050–1060.

[218] Rees F Garmann et al. “Physical principles in the self-assembly of a simple spherical virus”. In: *Accounts of chemical research* 49.1 (2016), pp. 48–55.

[219] Jeffrey A Speir et al. “Structures of the native and swollen forms of cowpea chlorotic mottle virus determined by X-ray crystallography and cryo-electron microscopy”. In: *Structure* 3.1 (1995), pp. 63–78.

[220] Petr Šulc et al. “A nucleotide-level coarse-grained model of RNA”. In: *The Journal of Chemical Physics* 140.23 (2014), p. 235102. doi: 10.1063/1.4881424. eprint: https://doi.org/10.1063/1.4881424. url: https://doi.org/10.1063/1.4881424.

[221] Rees F. Garmann, Aaron M. Goldfain, and Vinothan N. Manoharan. “Measurements of the self-assembly kinetics of individual viral capsids around their RNA genome”. In: *Proceedings of the National Academy of Sciences* 116.45 (2019), pp. 22485–22490. doi: 10.1073/pnas.1909223116. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1909223116. url: https://www.pnas.org/doi/abs/10.1073/pnas.1909223116.

[222] Swati Jain et al. “An extended dual graph library and partitioning algorithm applicable to pseudoknotted RNA structures”. In: *Methods* 162 (2019), pp. 74–84.

[223] Lorenzo Rovigatti et al. “A comparison between parallelization approaches in molecular dynamics simulations on GPUs”. In: *Journal of computational chemistry* 36.1 (2015), pp. 1–8.

[224] Erik Poppleton et al. “OxDNA. org: a public webserver for coarse-grained simulations of DNA and RNA nanostructures”. In: *Nucleic acids research* 49.W1 (2021), W491–W498.

[225] Erik Poppleton et al. “oxDNA: coarse-grained simulations of nucleic acids made simple”. In: *Journal of Open Source Software* 8.81 (2023), p. 4693.

[226] A Sengar et al. “ˇSulc P (2021) A Primer on the oxDNA Model of DNA: When to Use it, How to Simulate it and How to Interpret the Results”. In: *Front. Mol. Biosci. 8: 693710. doi: 10.3389/fmolb* (2021).

[227] Pauli Virtanen et al. “SciPy 1.0: fundamental algorithms for scientific computing in Python”. In: *Nature methods* 17.3 (2020), pp. 261–272.

[228] Philip M Morse. “Diatomic molecules according to the wave mechanics. II. Vibrational levels”. In: *Physical review* 34.1 (1929), p. 57.

[229] Birgit ALM Deiman and Cornelis WA Pleij. “Pseudoknots: A vital feature in viral RNA”. In: *Seminars in Virology*. Vol. 8. 3. Elsevier. 1997, pp. 166–175.

[230] Namhee Kim et al. “Candidates for novel RNA topologies”. In: *Journal of molecular biology* 341.5 (2004), pp. 1129–1144.

[231] Ian Brierley, Simon Pennell, and Robert JC Gilbert. “Viral RNA pseudoknots: versatile motifs in gene expression and replication”. In: *Nature Reviews Microbiology* 5.8 (2007), pp. 598–610.

[232] Hin Hark Gan, Samuela Pasquali, and Tamar Schlick. “Exploring the repertoire of RNA secondary motifs using graph theory; implications for RNA design”. In: *Nucleic acids research* 31.11 (2003), pp. 2926–2943.

[233] Samuela Pasquali, Hin Hark Gan, and Tamar Schlick. “Modular RNA architecture revealed by computational analysis of existing pseudoknots and ribosomal RNAs”. In: *Nucleic acids research* 33.4 (2005), pp. 1384–1398.

[234] Jiaqi Jiang. *An introduction to spectral graph theory*. 2012.

[235] Yuji Nakatsukasa, Naoki Saito, and Ernest Woei. “Mysteries around the graph Laplacian eigenvalue 4”. In: *Linear Algebra and its Applications* 438.8 (2013), pp. 3231–3246.

[236] Rees F Garmann, Aaron M Goldfain, and Vinothan N Manoharan. “Measurements of the self-assembly kinetics of individual viral capsids around their RNA genome”. In: *Proceedings of the National Academy of Sciences* 116.45 (2019), pp. 22485–22490.

[237] Jingzhi Chen, Yves Lansac, and Guillaume Tresset. “Interactions between the molecular components of the cowpea chlorotic mottle virus investigated by molecular dynamics simulations”. In: *The Journal of Physical Chemistry B* 122.41 (2018), pp. 9490–9498.

[238] Janos Szoverfi and Szilard N Fejer. “Dynamic stability of salt stable cowpea chlorotic mottle virus capsid protein dimers and pentamers of dimers”. In: *Scientific Reports* 12.1 (2022), p. 14251.

[239] Matteo Tiberti et al. “PyInteraph: a framework for the analysis of interaction networks in structural ensembles of proteins”. In: *Journal of chemical information and modeling* 54.5 (2014), pp. 1537–1551.

[240] András Fiser and Andrej Šali. “Modeller: generation and refinement of homology-based protein structure models”. In: *Methods in enzymology.* Vol. 374. Elsevier, 2003, pp. 461–491.

[241] Antonio Suma et al. “Tacoxdna: A user-friendly web server for simulations of complex DNA structures, from single strands to origami”. In: *Journal of computational chemistry* 40.29 (2019), pp. 2586–2595.

[242] Elvira Tarasova et al. “All-atom molecular dynamics simulations of entire virus capsid reveal the role of ion distribution in capsid’s stability”. In: *The journal of physical chemistry letters* 8.4 (2017), pp. 779–784.

[243] Jodi A Hadden et al. “All-atom molecular dynamics of the HBV capsid reveals insights into biological function and cryo-EM resolution limits”. In: *Elife* 7 (2018), e32478.

[244] James M Fox et al. “Analysis of a salt stable mutant of cowpea chlorotic mottle virus”. In: *Virology* 222.1 (1996), pp. 115–122.

[245] JAMES M FOX et al. “Characterization of a disassembly deficient mutant of cowpea chlorotic mottle virus”. In: *Virology* 227.1 (1997), pp. 229–233.

[246] Jing Huang et al. “CHARMM36m: an improved force field for folded and intrinsically disordered proteins”. In: *Nature methods* 14.1 (2017), pp. 71–73.

[247] Tamar Schlick and Anna Pyle. “RNA structural variability and functional versatility challenge RNA structural modeling and design”. In: *Biophysical journal* 113.2 (2017), E1–E2.

[248] Richard P Sear. “Nucleation: theory and applications to protein solutions and colloidal suspensions”. In: *Journal of Physics: Condensed Matter* 19.3 (2007), p. 033101.

[249] Juan R Perilla and Klaus Schulten. “Physical properties of the HIV-1 capsid from all-atom molecular dynamics simulations”. In: *Nature communications* 8.1 (2017), p. 15959.

[250] Diane L Lynch et al. “Understanding Virus Structure and Dynamics through Molecular Simulations”. In: *Journal of Chemical Theory and Computation* (2023).

[251] Eric Charles Dykeman et al. “Simple rules for efficient assembly predict the layout of a packaged viral RNA”. In: *Journal of molecular biology* 408.3 (2011), pp. 399–407.

[252] L Lavelle et al. “Phase diagram of self-assembled viral capsid protein polymorphs”. In: *The Journal of Physical Chemistry B* 113.12 (2009), pp. 3813–3819.

[253] Eric F Pettersen et al. “UCSF Chimera—a visualization system for exploratory research and analysis”. In: *Journal of computational chemistry* 25.13 (2004), pp. 1605–1612.

[254] Samuela Pasquali and Philippe Derreumaux. “HiRE-RNA: a high resolution coarse-grained energy model for RNA”. In: *The journal of physical chemistry B* 114.37 (2010), pp. 11957–11966.

[255] Elmar Krieger et al. “Improving physical realism, stereochemistry, and side-chain accuracy in homology modeling: four approaches that performed well in CASP8”. In: *Proteins: Structure, Function, and Bioinformatics* 77.S9 (2009), pp. 114–122.