Università
Ca'Foscari
Venezia

Master's Degree Thesis

in Science and Technology of Bio and Nanomaterials

Insights to the function of ion channels through an integrated in-house computational toolbox

Academic Year 2021-2022

Supervisors:
Dr. Achille Giacometti
Dr.ssa Marta Simeoni

Graduand:
Jacopo Moi, 847329

Abstract

Using Molecular Dynamics, Graph and Network Theory as well as Machine Learning techniques, we develop an integrated pipeline that can be used to study the function-structure relationship of ion-channels. We apply this concepts to the analysis of variants in sodium channel Nav1.7 subunit found in clinical studies of painful syndromes

Acknowledgement

I would first like to give my most heartfelt thanks to my thesis advisors Professor Simeoni Marta of the Faculty of Computer Science at Ca' Foscari and Professor Giacometti Achille of the Faculty of Science and Technology of Bio and Nanomaterials at Ca' Foscari, Venice. Most of the achievements in this work had light from the freedom to improve aspects and solve problems that emerged from the constant and patient analysis of the results in progress in a teamwork fashion. In addition, I greatly rejoiced in the participation of the various meetings at ECLT in which the papers presented, always projected toward future applications, made me realize the importance of finding a solid path in the study of ”complex” topics, and that the best way to achieve an awareness of a problem often lies in group work without which results could not have been achieved. Especially on this latter aspect, they patiently built up an interdisciplinary team made of computer scientists, physicists and chemists, and i personally believe that achieve a functional team is one of the most difficult task in the universe. Another great special thanks must be given to Cedrix: thanks to his kindness, advice and access on CINECA GALILEO cluster gave me the opportunity to perform simulations and data-analysis requiring state-of-the-art computational power. I would also like to thanks the extraordinary Computer Science team leaded by Alvise Spanò and composed by Davide Pizzolato, Lorenzo Fanton, Albebrto Pesce and Francesco Vinci. I hope in the future to collaborate with them many more times. Thank you!

Contents

Chapter 1

Introduction

Nowadays neuropathic diseases are still poorly understood and unresolved disorders that continue to attract the interest of the scientific community. Within this class of diseases, Neuropathic Pain Disorders represent an even more challenging issue and the key to their action seems to be mediated by a class of membrane proteins, the voltage-gated sodium channels [13]. Recently, a computational Pipeline devoted to this problem has been developed at Ca' Foscari University [51], with a focus on Painful Neuropathies expressed by the NaV1.7 protein belonging to this class. The key point addressed was the development of a reliable in silico tool able to classify experimentally performed point-mutations in pathogenic (leading to a disease) and non-pathogenic (not leading to any disease). The proposed method, which combined homology models, graph theory and machine learning techniques, was tested against a set of 85 mutations that have experimentally studied before. While successful, the original implementation displayed several drawbacks including computational speed, lack of proper tests of each single element of the pipeline, as well as an overall difficulty in extending it to other case studies. In addition, a systematic computational study on the structural effects of these point-mutations was missing. This thesis is devoted to address both points, to improve the computational protocol on one hand and to perform dedicated molecular dynamics studies on the structural effects of a representative sample of the point mutations, in order to gain more insight on their pathogenic pathway. NaV1.7 ionic channel is of extreme interest for medical applications because it is highly expressed in the peripheral nervous system and it has been speculated that gain-of-function mutations are in direct connection with the onset of painful neuropathies [13]. On the other hand, this is a complex protein made of 1659 residues grouped in two long intracellular-coils and four voltage-sensing domains arranged around a central aqueous channel formed by the pore domain [43].

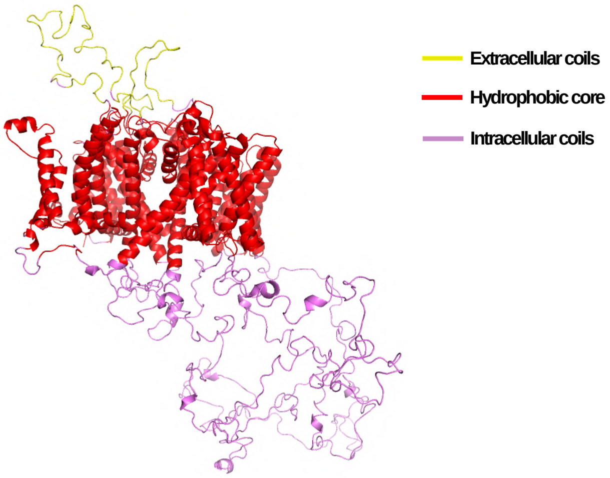


Figure 1.1: MOESM3 Wild-Type

It is worth emphasizing that, because of the size and fragility of NaV1.7 structure, it is difficult to refine and purify it by in-vitro techniques, and this is one of the main advantage of a preliminary in-silico approach in this particular case.

The original pipeline involved first the use of homology-modeling [28] to obtain the mutated structures starting from three different templates of the NaV1.7 protein (MOESM3 for *Acrobacter Bultzieri*, 6a90 for *Periplaneta Americana* and 6j8j for *Homo Sapiens*), and then an energy minimization of the obtained structures using the Fragment-Guided Molecular Dynamics [56] protocol. Subsequently to this first phase, graphs of the non-covalent interactions (RINs) of the obtained structures were generated and then compared with each other using Graph Kernel methods [34]. This allows the analysis and extraction of properties within a reduced dimensionality and a corresponding gain in speed a reliability. Finally, machine learning techniques were implemented on the result to perform a binary classification between pathogenic and non-pathogenic mutations. As the authors noted in the original article [51], this task is only possible if the refined structures have achieved a high degree of differentiation among themselves and a lack of control of this requirement could invalidate the whole process.

The first part of this thesis then focuses mainly on the in-house implementation of two tools of the original pipeline: a new Residue Interaction Network generator was already developed by a computer science team at the Ca' Foscari DAIS Department and needed to be tested with a suitable set of molecules, and a new protocol Fragment-Guided Molecular-Dynamics (FG-MD) has been implemented since the already employed FG-MD tool [56] was one of the main shortcomings of the original Pipeline. Before their inclusion in the new Pipeline, both these tools were tested separately by creating an ad- hoc environment for each of them. After that, the new Pipeline was tested step by step in order to validate the results. Encouraged by the results, the Pipeline was then applied in full production. The thesis is organized as follows: Chapter 2 is devoted to introducing the concepts of Residue Interaction Network and Kernel method on graphs. Chapter 3 presents the used Molecular Dynamics techniques, with a specific focus on the Fragment-Guided refining protocol, which will be then successively implemented and tested. In Chapter 4 the tools and workflow of the original version of the Pipeline is briefly reminded, and its shortcomings and drawbacks highlighted. In Chapter 5, the new implementation of the Pipeline is then presented, including all performed tests on the in-house implementation of the FG-MD protocol and the RIN generator RINmaker. In Chapter 6 the new Pipeline is tested with the same set of structures of the original pipeline, thus providing an independent double-check of the original results. This is followed by the application of the new Pipeline in a full-fledged way, with a critical assessment of the strengths of this improved pipeline. Finally, Chapter 7 reports the results of free energy landscape (FEL) studies first performed a on Trp-Cage protein as benchmark and then on a set of point-mutated structures related derived from the MOESM3 template. This study has a two-fold objective. Firstly, it provides an overview of the effect of the FGMD protocol on the specific point-mutated studied structures. Secondly, it paves the way to a full all-atom MD study of the whole NaV1.7.

Chapter 2

Residue interaction network

2.1 Graph theory

2.1.1 Elements of graph theory

In its simplest form, a graph is a collection of vertices that can be connected to each other by means of edges. In particular, each edge of a graph joins exactly two vertices. Using a formal notation, a graph is defined as follows [47]:

Definition 2.1.1 (Undirected simple Graph) *An undirected simple graph G consists of a set V of nodes and a set of edges $E = \{\{u, v\} \mid u, v \in V, u \neq v\}$, for which we write $G = (V, E)$. Each edge $\{u, v\} \in E$ is said to join two nodes u, v, which are called its end points.*

A useful way to fully describe a graph in matrix form is by its adjacency matrix $Adj(G)$ [6]. This matrix is defined as :

Definition 2.1.2 (Adjacency matrix) *Let a_{ij} be the element in the i-th row and j-th column of a matrix Adj(G). Then, the Adj(G) adjacency matrix of a graph G(V,E) can be defined as follows:*

$$a_{ij} = \begin{cases} 1 \quad if \quad v, u \in E(G), \\ 0 \qquad otherwise. \end{cases} \tag{2.1}$$

Definition 2.1.1 gives the foundations for understanding the graph object (See Fig. 2.1(a)). Since the E set is thought as an unordered set of V pairs, this definition does not allow to have more than one edge connection two nodes u, v. This definition can be extended in order to have an undirected multigraph [4]:

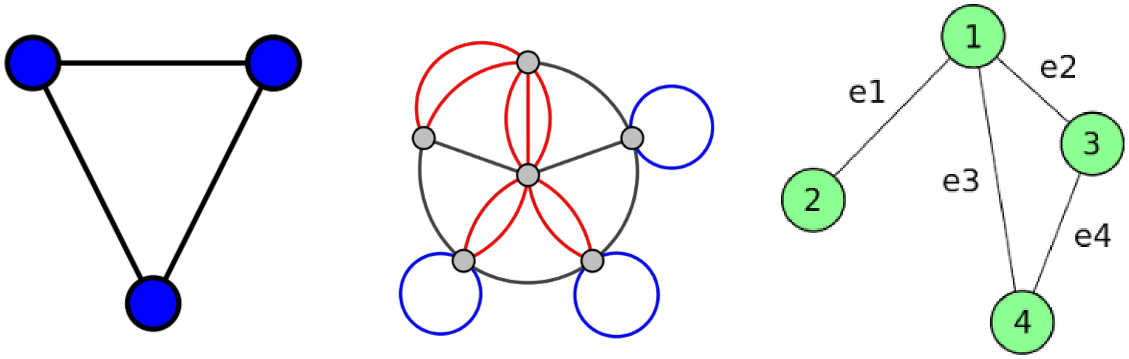
Definition 2.1.3 (Undirected Multigraph) *An undirected multigraph $G = (V, E)$ consists of a set V of nodes and a multiset of edges $E = \{\{u, v\} \mid u, v \in V, u \neq v\}$.*

This definition extend the notion of Undirected Simple Graphs and allow to have more edges for a given unordered pair of vertices (See Fig. 2.1(b)). To complete the description is important to introduce the notion of labeled graph:

Definition 2.1.4 (Labeled Graph) *A labeled graph is a graph which has labels associated with each edge and/or each vertex.*

From the Definition 2.1.4 is possible to add labels to an Undirected Multigraph (See Fig. 2.1(c)).

Network will become prominent in the next section because it's one of the most useful representation for the non-covalent interactions between residues in a protein.



(a) Undirected graph (b) Undirected multigraph (c) Labeled graph

Figure 2.1: Different type of graphs according to the definitions [9]

An important concept also used in this work is the subgraph[47]:

Definition 2.1.5 (Subgraph) *A graph H is a subgraph of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$ such that for all $e \in E(H)$ with $e = e(u, v)$ we have that $u, v \in V(H)$.*

2.1.2 Residue interaction network (RIN)

Protein structures can be represented as networks (graphs) where amino acid residues are nodes and their interactions are edges. This approach was used to study various protein aspects, including protein structure flexibility, folding of protein domains, structural patterns, key residues in folding, residue fluctuation, and side-chain clusters [2]. In a typical Residue Interaction Network, every node is uniquely labeled with its position along the protein backbone, while the edges are labeled with the specific type of non-covalent interaction that occour between its end points. Its form is clearly an undirected multigraph (See 2.1(b)) since:

- A specific interaction that occour between two residues in a protein exists without a specified direction (undirected)
- Different non-covalent interactions can occour between two residues (multigraph)

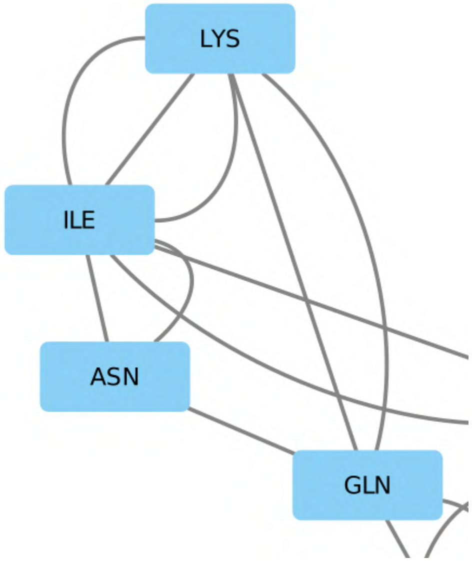


Figure 2.2: A small detail of the RIN of the Trp-Cage miniprotein made with RINmaker and visualized with Cytoscape

2.2 Functions and parameters defined on graphs

2.2.1 Metrics on graph: centrality

An important metric for network analysis is deciding on whether there are any vertices “more important” than others. The importance of a vertex is, of course, dependent on what a graph is actually modeling [47]. There exists multiple and qualitatively different centrality measures, here is reported the ones used in this work [47]:

Definition 2.2.1 (Betweenness Centrality) *Given a graph G=(V,E) where E$\neq \emptyset$, the betweenness centrality $c_B(u)$ of vertex u is defined as:*

$$c_B = \sum_{x \neq y} \frac{|S(x,u,y)|}{|S(x,y)|} \tag{2.2}$$

where $S(x, y)$ is the set of all the shortest paths between two nodes $x, y \in V(G)$ and $S(x, u, y) \subseteq S(x, y)$ the ones that pass through vertex $u \in V(G)$.

2.2.2 Graph comparison: Kernel functions on graphs

In the context of graph representations of protein interaction, one prominent question is how to measure how much two graphs are similar. Within the protein realm, for example, one might want to know whether or not a given protein is an enzyme or be able to predict if point-mutations are pain-related or not [52]. Kernel methods offer a natural framework to study these questions. Starting directly from the definition of Kernel:

Definition 2.2.2 (Kernel function) *A kernel is a function k that for all $\boldsymbol{x}, \boldsymbol{z} \in X$ satisfies:*

$$k(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle \tag{2.3}$$

Here

$$\begin{aligned} \phi : \mathcal{X} &\to \mathcal{F} \\ \mathbf{x} &\mapsto \phi(\mathbf{x}) \in \mathcal{F} \end{aligned} \tag{2.4}$$

is a map between a vector $\mathbf{x} \in \mathcal{X}$ to a Hilbert space $\phi(\mathbf{x}) \in \mathcal{F}$

The kernel function k is positive-definite and essentially compute the inner-product between two given vectors which are mapped by ϕ to a feature space F, which is an abstract inner-product vector space. The F is the space of features related to some properties of the graphs. This mathematical statement is important since it is not possible to define directly an inner-product between graphs but it is possible to map them to a feature space, where the features form a basis. The kernel can be normalized in order to have a value $[0, 1]$. More this value approach 1 more the assessed graphs are similar, respect the chosen kernel. Below are defined some important kernels used during this work, they also represent a general framework for deriving other kernels [34].

Vertex-Histogram kernel The vertex histogram kernel is a basic linear kernel on vertex label histograms. The kernel assumes node-labeled graphs.

Definition 2.2.3 (Vertex-Histogram kernel) *Let $\mathcal{G}$ be a collection of graphs, a set of node labels $\mathcal{L}$ and a function $\ell : \mathcal{V} \rightarrow \mathcal{L}$ which assigns labels to the vertices of the graphs, the vertex label histogram of a graph $G = (V, E)$ is a vector $\mathbf{f} = (f_1, f_2, ..., f_d)$ such that $f_i = |\{v \in \mathcal{V} : \ell(v) = i\}|$ for each $i \in \mathcal{L}$.*

Given two graphs G_1, G_2, the vertex histogram kernel is defined as:

$$k(G, G') = \langle \mathbf{f}, \mathbf{f}' \rangle \tag{2.5}$$

In other words, given a labelling function ℓ, the vertex-histogram first assign labels to vertex $\mathcal{V}$ by the function ℓ for each input graphs; it counts how many vertex have the same label i in each graph and it assign the result to a vector $\mathbf{f}$, so the resulting vectors $\mathbf{f}$,$\mathbf{f}'$ are multiplied by the dot-product rule (See Fig. 2.3(a)). The result of the dot-product is the kernel function output. This is a fast kernel and the complexity is linear, suitable for computational application.

Similarly to Definition 2.2.3, is possible to define the feature vectors $\mathbf{f}$ and $\mathbf{f}'$ as an histogram on edge labels where now $\ell : \mathcal{E} \rightarrow \mathcal{L}$ is a function that assigns labels from an edge label set $\mathcal{L}$ to the edges collection $\mathcal{E}$ of the graphs. In this case the kernel is called Edge Histogram (See Fig. 2.3(c)).

Subgraph-Matching kernel The subgraph matching kernel counts the number of matchings between subgraphs of bounded size in two graphs.

The kernel is very general since it can be applied to graphs that contain node labels, edge labels, node attributes or edge attributes.

Definition 2.2.4 (Subgraph-Matching kernel) *Given two graphs $G = (V, E)$, $G^{'} = (V^{'}, E^{'})$, the set of all bijections $\mathscr{B}$ between $S \subseteq V$ and $S^{'} \subseteq V^{'}$ and a wight function $\lambda : \mathscr{B}(G, G^{'}) \to \mathbb{R}$, The subgraph matching kernel is defined as:*

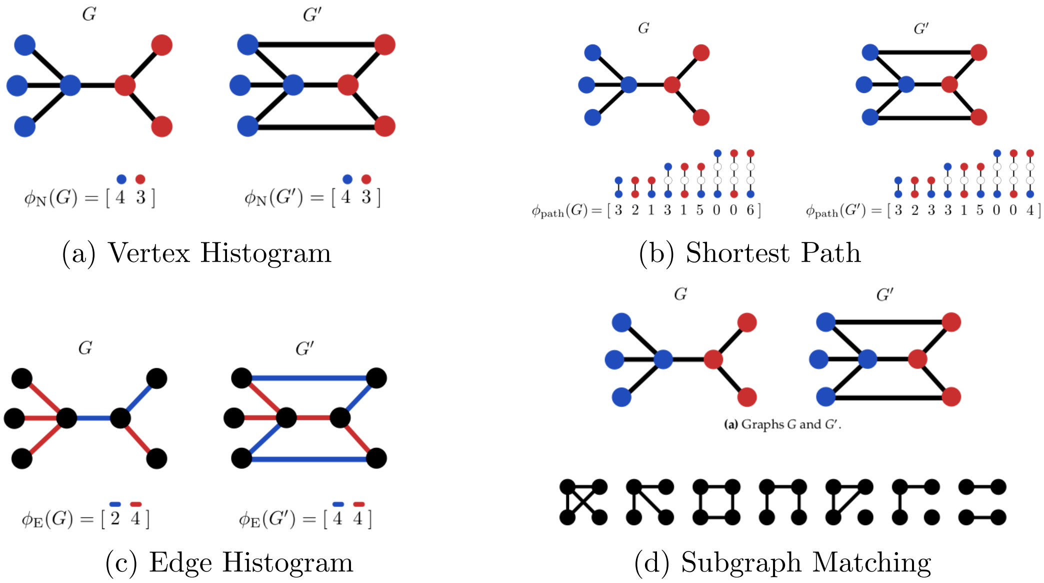
$$k(G, G^{'}) = \sum_{\phi \in \mathscr{B}} \phi(\lambda) \prod_{v \in \phi} k_V(v, \phi(v)) \prod_{e \in \phi_v \times \phi_v} k_E(e, \phi(e)) \tag{2.6}$$

More intuitively, the bijection set $\mathscr{B}$ in this case is made by the subgraphs of G and $G^{'}$ which $V(G) \to V(G^{'})$. For constructing this set, in first instance the graph product $G \times G^{'}$ is made; every nodes and edges of the graph product is weighted by a set of rules, in order to obtain a weighted graph product; from this graph product the algorithm enumerate every cliques for constructing the $\mathscr{B}(G, G^{'})$ set. the kernel compare every nodes and edges with desired k_V, k_E for every bijection ϕ selected from $\mathscr{B}(G, G^{'})$.

Weishfer-Lehman Kernel The key idea of the Weisfeiler–Lehman algorithm is to replace the label of each vertex with a multiset label consisting of the original label of the vertex and the sorted set of labels of its neighbors. Te resultant multiset is then compressed into a new, short label. Such new label refects the knowledge of the node and its neighborhood. This relabeling process is then repeated for h iterations. By performing this procedure simultaneously on all input graphs, it follows that two vertices from diferent graphs will get identical new labels if and only if they have identical multiset labels. The kernel function in this case compare the node labels of the graphs resulting after each iteration and summarizes the comparison with a real number. It can be shown that this is equivalent to comparing the number of shared subtrees between the two input graphs (the kernel considers all subtrees up to height h)[51].

The output of a kernel can be organized in a matrix containing the evaluation of the kernel function on all pairs of graphs. This matrix is said Gram-matrix:

$$G_{i,j} = \langle \phi(G_i), \phi(G_j) \rangle \tag{2.7}$$


G
G'
$\phi_{\mathrm{N}}(G) = [\ 4\ \ 3\ ]$
$\phi_{\mathrm{N}}(G') = [\ 4\ \ 3\ ]$
G
G'
$\phi_{\mathrm{path}}(G) = [\ 3\ \ 2\ \ 1\ \ 3\ \ 1\ \ 5\ \ 0\ \ 0\ \ 6\ ]$
$\phi_{\mathrm{path}}(G') = [\ 3\ \ 2\ \ 3\ \ 3\ \ 1\ \ 5\ \ 0\ \ 0\ \ 4\ ]$
G
G'
$\phi_{\mathrm{E}}(G) = [\ 2\ \ 4\ ]$
$\phi_{\mathrm{E}}(G') = [\ 4\ \ 4\ ]$
G
G'
(a) Graphs G and G'.

(a) Vertex Histogram

(b) Shortest Path

(c) Edge Histogram

(d) Subgraph Matching

Figure 2.3: Representation of the obtained feature vectors $\phi(G)$ and $\phi(G')$ for the selected Kernel functions.

Chapter 3

Molecular dynamics

Molecular dynamics is one of the main methods for exploring the conformational space of large molecules such as proteins. The simulation of the motion is realized by the numerical solution of the classical Newtonian dynamic equations. Usually, Newton's equations of motion are used to capture the trajectories of particles in the system where the forces applied to the particles composing the system derives from a description of the potential energy, called forcefield. The simulation procedure is usually constructed as follows:

1. Initialize the system in the desired ensemble.
2. Compute the potential energy from the topology of the system according to a specific forcefield.
3. Compute the forces for each particle.
4. Integrate Newton's equation of motion.
5. Repeat steps 3 and 4 for a desired length of time.

3.1 Forcefield

How described above, the MD simulation needs a description of the potential energy respects all of the particles and bonds present in the system. The basic definition for a forcefield has the standard form:

$$V(\mathbf{r}^N) = v_b(l_i) + v_\theta(\theta_i) + v_\omega(\omega) + v_{LJ}(\mathbf{r}_{ij}) + v_C(\mathbf{r}_{ij}) \tag{3.1}$$

Writing every terms in detail:

$$\begin{aligned} V = & \sum_{\text{bonds}} K_r(r_i - r_0)^2 + \sum_{\text{angles}} K_\theta(\theta - \theta_{eq})^2 + \\ & + \sum_{\text{dihedrals}} \frac{V_n}{2}[1 + \cos(n\phi - \gamma)] + \sum_{i<j} \left[\frac{A_{ij}}{R_{ij}^{10}} - \frac{B_{ij}}{R_{ij}^{6}} \right] + \\ & + \sum_{\text{H}-\text{bonds}} \left[\frac{A}{r^{12}} - \frac{B}{r^6} \right] + \sum_{i<j} \frac{q_i q_j}{\epsilon R_{ij}} \end{aligned} \tag{3.2}$$

In Table 3.1 a detailed description of every term is reported:

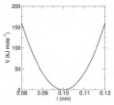
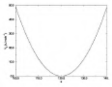
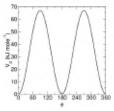
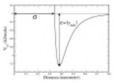
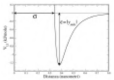
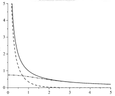
Type	Formula	f graph	Description
Harmonic potential	$\sum_{\text{bonds}} K_r(r_i - r_0)^2$		Harmonic potential for bond length, fixed at r_0
Bending potential	$\sum_{\text{angles}} K_\theta(\theta - \theta_{eq})^2$		Harmonic potential for bond angle vibration, fixed at θ_{eq}
Torsional potential	$\sum_{\text{dihedrals}} \frac{V_n}{2}[1 + \cos(n\phi - \gamma)]$		Periodic potential for dihedral angles, where V_n is the amplitude and γ the phase factor
H-Bond potential	$\sum_{i<j} \left[\frac{A_{ij}}{R_{ij}^{10}} - \frac{B_{ij}}{R_{ij}^{6}}\right]$		L-J potential for H-Bonds modelled as 10-6. A_{ij} and B_{ij} are characteristic length of atoms referred to the minimum of the function (potential well)
VdW potential	$\sum_{\text{H-bonds}} \left[\frac{A}{r^{12}} - \frac{B}{r^{6}}\right]$		L-J potential for VdW modelled as 12-6. A and B are characteristic length.
Coulomb potential	$\sum_{i<j} \frac{q_i q_j}{\epsilon R_{ij}}$		Coulomb potential where q_i and q_j refers to partial charges defined in the chosen forcefield.

Table 3.1: Standard forcefield terms description (AMBER) [39]

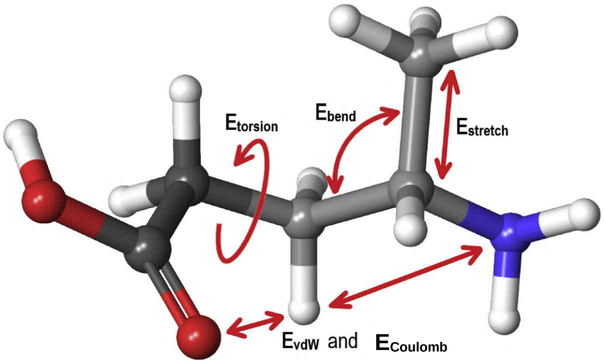


Figure 3.1: Illustration of the potential energy terms that make up the force field expression and the interactions they correspond to.[29]

The equilibrium parameters r_{eq} and θ_{eq},partial charges q and LJ characteristic length A and B as well as other parameters, are specified for every type of atoms and interaction in the forcefield description.

In this work the used forcefield were CHARMM36[21] and AMBER-14[30], which suits well for membrane and globular protein.

3.2 Integrators

Newton equations of motion are clearly continuous respect time but discrete in the context of simulation, since there is the discretization induced by the machine. Integrators allow to discretize the Newton's equation and to recursively update velocity and position of particles subjected to forces. In this work two integrators were mainly used: Verlet integration which use a deterministic approach for calculate numerically equation of motion and Langevin integrator, which is an extension of Verlet in the context of system accounting for the solvent effect.

3.2.1 Velocity Verlet integration

This algorithm allows to approximate Newton's equation of motion in order to update simultaneously position and velocity of a particle at consecutive times. This latter aspect is possible since in a classical system is safe to state that:

$$\dot{\mathbf{x}}(t) = \mathbf{v}(t) \qquad \ddot{\mathbf{x}}(t) = \frac{\mathbf{F}(\mathbf{x}(t))}{m} \tag{3.3}$$

Velocity Verlet integration algorithm is, in first instance, obtained from the tylor expansion of $\mathbf{r}(t+\Delta t)$, $\mathbf{v}(t+\Delta t)$, $\dot{\mathbf{v}}(t+\Delta t)$ around t:

$$\mathbf{r}(t+\Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2}\frac{\mathbf{F}(\mathbf{r}(t))}{m}\Delta t^2 + \mathcal{O}(\Delta t^3) \tag{3.4}$$

$$\mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \frac{1}{2}\dot{\mathbf{v}}(t)\Delta t + \frac{1}{2}\ddot{\mathbf{v}}(t)\Delta t^2 + \mathcal{O}(\Delta t^3) \tag{3.5}$$

Substituting in Eq.(3.5) to the $\dot{\mathbf{v}}(t)$ term the rearranged Tylor expansion of $\dot{\mathbf{v}}(t+\Delta t)$, in order to approximate $\ddot{\mathbf{v}}(t)$:

$$\mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \frac{\Delta t}{2m}(\mathbf{F}(t) + \mathbf{F}(t+\Delta t)) + O(\Delta t^4) \tag{3.6}$$

Eq.(3.4) and (3.6) are now ready for a recursive calculation of $\mathbf{r}(t)$ and $\mathbf{v}(t)$ at consecutive step: Given $\mathbf{x}_k$, $\mathbf{v}_k$, Δt and a forcefield for evaluate $\mathbf{F}$:

1. Calculate $\mathbf{x}_{k+1}$=$\mathbf{x}_k + \Delta t \mathbf{v}_k + \Delta t^2 \frac{F(x_k)}{2m}$
2. Evaluate $\mathbf{F}(\mathbf{x}_{k+1})$ (forcefield)
3. calculate $\mathbf{v}_{k+1}$
4. Go to step 1

Clearly this algorithm can be extended to all of the atoms present in a system. In this case, the system is represented by a matrix, where every entry is the position and velocity of the particle-i. The Δt is called the timestep. Generally, the timestep chosen in a simulation is of 1 or 2 femtosecond, which is sufficient smaller of the solvent relaxation time ($\sim 2\,\mathrm{ps}$). Higher timestep value could add a too much error, since the Verlet method is approximated.

3.2.2 Langevin integrator

Langevin equation of motion is a stochastic differential equation of the form:

$$m_i \frac{\mathrm{d}^2 \mathbf{x}(t)}{\mathrm{d}t^2} = \mathbf{F}_i - \gamma_i \frac{\mathrm{d}\mathbf{r}_i}{\mathrm{d}t} m_i + \eta_i(t) \tag{3.7}$$

Eq. (3.7) is the Newton's equation of motion equipped with two additional term: the friction coefficient γ_i and the noise term $\vec{\eta}_i(t)$

taken from a Gaussian distribution $\mathcal{N}(0,1)$ that describe random fluctuations due to solvent-solute collision. The integration is carried out in the same framework of the Verlet integration, with an additional random force $\vec{\eta}(t)$ obtained from the distribution and added to i-th particle.

The Langevin and Verlet integrators "simply" resolve the equation of motions for a system of forces, so naturally they run over time toward a system in quiet, that correspond to the minimum of the energy at a given set of parameters (T,P,T_{bath},etc).

3.2.3 Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (L-BFGS)

Langevin and Verlet integrators resolve the time-dependent equation of motions. Another class of integrator not-dependent on time are the Quasi-Newtonian methods. This class of algorithm, instead of resolving the equation of motions respect time, are designed to search for the minimum energy conformation between the possible states given a starting conformation. L-BFGS is the most used in molecular dynamics applications. More precisely, gradient minimization is the problem of finding the set of positions which is the minimum of the gradient respect the potential:

$$\min \nabla V(\mathbf{r}_{ij}) \tag{3.8}$$

This work is basically done searching iteratively for position $\mathbf{r}_{ij}$ which minimize the potential function. This is done in first instance expanding $\nabla V(\mathbf{r} + \Delta\mathbf{r})$ around $\Delta t = 0$:

$$\nabla V(\mathbf{r}_k + \Delta\mathbf{r}_k) = \nabla V(\mathbf{r}_k) + \mathbf{H}\Delta\mathbf{r} \tag{3.9}$$

Where $\mathbf{H}$ is the Hessian matrix of the potential. Setting the gradient $\nabla V(\mathbf{r}_k + \Delta\mathbf{r}) = 0$ and writing $\Delta\mathbf{r} = \mathbf{r}_{k+1} - \mathbf{r}_k$ an update rule is obtained for estimating $\mathbf{r}_{k+1}$:

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \mathbf{H}^{-1}\Delta V(\mathbf{r}_k) \tag{3.10}$$

This algorithm is mostly used for removing steric clashes and searching conformation which minimize locals interaction. It can be also used for simulated annealing simulations but it does not guarantee to find the global minimum.

3.3 System parameters

The system where the simulation is carried out heavily influence the outcome and must be balanced between a detailed approximation of a real life model and the available computational power. In protein simulation the most important aspects are:

- Presence or not of boundary conditions (BC)
- Solvent description (lipid or water)
- Coupling of pressure and temperature respect the chosen ensemble
- Description of non-bonded interaction

3.3.1 Periodic boundary conditions

Periodic boundary conditions (PBCs) are a set of boundary conditions which are often chosen for approximating a large system by using a small part called a unit cell. The box enclosing the system is surrounded by other boxes containing the same system (for example, in the case of a 2D system, the number of adjacent boxes is eight (See Fig. 3.2). It's normal to use this kind of approximation for obtaining bulk properties of liquid or solid in a simulation. In the case of protein they are mostly used for calculating long-range interactions with

Particle Mesh Ewald (PME) methods and for applying algorithm for maintaining constant pressure (Monte Carlo barostat).

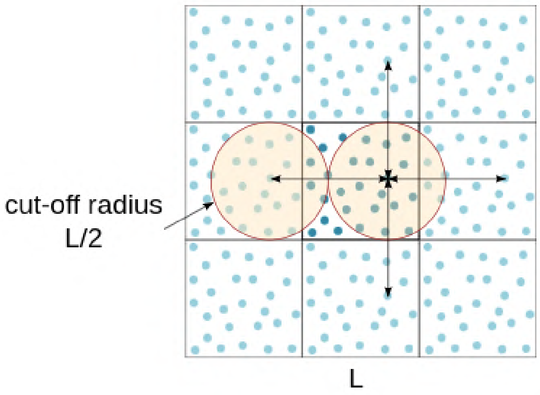


Figure 3.2: Boundary conditions as repetition of a cubic box

3.3.2 Water models

Most molecular dynamics simulations are carried out with the solute surrounded by a droplet or periodic box of explicit water molecules. In a typical case, water molecules will account for over 80% of the particles in the simulation. Water–water interactions dominate the computational cost of such simulations, so the model used to describe the water needs to be fast as well as accurate. Many different water models exists nowadays, but in this work the TIP3P explicit water model is used. The original TIP3P site model has positive charges on the hydrogens and a negative charge on oxygen. The potential considered here involve a rigid water monomer that is represented by three interaction sites, and is described by an energy functions composed by a Coulomb potential term and a Lennard-Jones potential term [23]:

$$V_{\text{TIP3P}} = \sum_{i}^{\text{on } m} \sum_{j}^{\text{on } n} \frac{k_C q_i q_j}{r_{ij}} + \frac{A}{r_{OO}^{12}} - \frac{B}{r_{OO}^{6}} \tag{3.11}$$

Where k_C is the electrostatic constant equal to $332.1\,\frac{\text{Å}\cdot\text{kJ}}{\text{mol}\cdot\text{eV}^2}$, q_i, q_j the partial charges of the involved atoms, r_{ij} the distance between two atoms or charged sites and A,B the LJ-parameters. The sum extends to atoms of the monomers m and n.

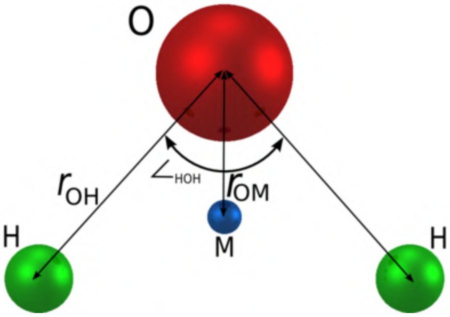


Figure 3.3: Water model

In Tab. 3.2 a comparison of forcefield parameters for common water models is provided.

Model	O sigma (Å)	O epsilon (kcal/mol)	O charge	H charge	O-H bond	H-O-H angle
SPC	3.166	0.15535	-0.82	0.41	1.0	109.466667
TIP3P	3.15061	0.1521	-0.834	0.417	0.9572	104.52
TIPS3P	3.1506	0.1521	-0.834	0.417	0.9572	104.52

Table 3.2: Comparison of forcefield parameters for three point charge common water models

3.3.3 Temperature coupling

To meet the requirement of having constant temperature in the chosen ensemble a statistical strategy is to deploy a thermostat function. A thermostat function couple a fictitious bath T_f with the absolute T_{abs} of the system, which is function of the total kinetic energy of every particle:

$$T_{abs} = \frac{1}{k_B N_{\text{df}}} \sum_{i=1}^{N} m_i v_i^2 \tag{3.12}$$

where N_{df} are the degrees of freedom of the system. From Eq. (3.12) is clear that imposing a condition $T_{abs} = T_{bath}$ needs an expression $v_{bath}(t)$ for modify velocity (here momentum) of the particles. This task is left to the chosen thermostat.

Andersen thermostat Andersen is the simplest but powerful thermostat for coupling the system to a heat bath in the canonical ensemble. This function add stochastic forces, sampled by the Maxwell-Boltzmann distribution, that modify the kinetic energy of the atoms or molecules [11]:

$$P(v_{x,i}) = \left(\frac{\beta m_i}{2\pi}\right)^{\frac{1}{2}} \exp\left(-\frac{m_i \beta v_{x,i}^2}{2}\right) \tag{3.13}$$

Where $v_{x,i}$ is $x-$component of the velocity of particle i. Since this method has to mimic particle collisions, the probability that this event happen is sampled from a Poisson distribution:

$$P(t) = \nu e^{-\nu t} \tag{3.14}$$

Where $\nu = \frac{1}{\tau}$ is the frequency of collisions. Most of the times the input parameter is τ. In this work, this thermostat was used especially in the simulation regarding canonical ensemble, where the Verlet integrator is used.

Langevin thermostat Langevin equation 3.7 has a temperature control just embedded in his form, since the $\vec{\eta}(t)$ factor is chosen by a Gaussian distribution, which can be scaled with variance equal to:

$$\sigma^2 = 2 m_i \gamma_i k_B T_{\text{bath}} / \Delta t \tag{3.15}$$

Where T_{bath} is the chosen bath temperature in Kelvin.

3.3.4 Monte Carlo barostat

Simulations in the NPT ensemble need to maintain a constant fixed pressure P^*. In the NPT ensemble, the relevant probability function is:

$$p_{\text{NPT}} = \exp\big(-\beta U(\{\vec{r}, \vec{r}\}) - \beta P^* V(\{\vec{r}\})\big) \tag{3.16}$$

So for maintaining the simulation in the correct ensemble the sampled states must be constrained to have P^*V be constant. One algorithm is the so called volume rescaling [14]. During a simulation with timestep Δt, the volume $V(t+\Delta t)$ is scaled toward the pressure $P(t)$ according to:

$$V(t+\Delta t) = \tau_p^{-1} V(t)\Big(\frac{P(t) - P^*}{P^*}\Big)\Delta t \tag{3.17}$$

Where τ_p is the characteristic time for this process. The smaller the relaxation time τ_p , the more closely the instantaneous pressure is tied to the target, and the stronger the disturbance of the actual dynamics by individual rescaling operations. In molecular dynamics, τ_p must be large enough to produce meaningful data

3.3.5 Approximation of non-bonded interaction

The largest computational cost s from identifying and calculates non-bonded interactions. Both Lennard-Jones and Electrostatic interactions are pairwise, since they involve the computation of potential energy on two different particle. This means that for a system formed by N particles, the computational cost is $\mathcal{O}(N^2)$, that become inaccessible for large systems. However, is possible to control this cost as follows: (See Eq. (3.1)).

- Lennard-Jones potential after 2.5σ can be considered neglectible
- Coulombic potential can be summed both in real space and fourier space

Lennard-Jones Cutoff The first approximation is modyfing LJ potential introducing a cutoff $r_C^{LJ} \geq 2.5\sigma$, above which the functions can be approximated to 0.

$$v_C'^{LJ} = \begin{cases} v_C^{LJ}(r) & r < r_C^{LJ} \\ 0 & r > r_C^{LJ} \end{cases} \tag{3.18}$$

This approximation clearly needs to be refined in the case of simulations in the NVE ensembles but in the case of this work the simulations were carried in NPT and NVT ensembles, where energy conservation is not strictly required. Furthermore, the simulated system were composed by a single protein, where the great majority of LJ interactions distances are in the order of σ.

Columbic potential- Reaction Field Using Explicit solvation for big protein can be computationally expensive. Reaction field is a mathematical approximation where, the solvent effects on coulombic interaction is approximated by introducing a correction to the Coulombic potential [41] in vacuo:

$$E = \frac{q_1 q_2}{4\pi\epsilon_0}\left(\frac{1}{r} + k_{rf} r^2 - c_{rf}\right) \tag{3.19}$$

$$k_{rf} = \left(\frac{1}{{r_{cutoff}}^3}\right)\left(\frac{\epsilon_{solvent} - 1}{2\epsilon_{solvent} + 1}\right) \tag{3.20}$$

$$c_{rf} = \left(\frac{1}{r_{cutoff}}\right)\left(\frac{3\epsilon_{solvent}}{2\epsilon_{solvent} + 1}\right) \tag{3.21}$$

Where $\epsilon_{solvent}$ refers to the dielectric constant of the solvent and k_{rf} and c_{rf} the constant used for the approximation.

Coulombic potential- Particle Mesh Ewald Generally a cutoff r_C^q is also introduced for Coulombic potential. However, the truncation with a cutoff $r_C^q > r_C^{LJ}$ of the Coulombic potential alone is not sufficient to approximate long-range electrostatic interactions since it vanishes slowly respect the LJ-potential. For improving this aspect the Particle Mesh Ewald (PME) method is used.

In the PME method, the electrostatic interaction are split as a sum of a term relating interactions in real space and interactions in Fourier space [16]. First point the Coulomb potential is written as a sum of two terms:

$$V(\mathbf{r}) = V_{\mathrm{sr}}(\mathbf{r}) + V_{\mathrm{lr}}(\mathbf{r}) \tag{3.22}$$

Here $V(\mathbf{r})$ is the Coulomb potential. That can be regarded as a sum of two terms where the first is the contribution of short-range interaction while the second means contribution from long-range interaction. The basic idea is to write the long-range contribution in Fourier space while the short-range interaction remains in the real space:

$$V(\mathbf{r}) = \sum_{ij} V_{sr}(\mathbf{r}_i - \mathbf{r}_j) + \sum_i q_i \phi^k(\mathbf{r}_i) \tag{3.23}$$

$$\phi^k(\mathbf{r}) = \frac{1}{V} \sum_k \tilde{\rho}(\mathbf{k}) e^{i\mathbf{k}\mathbf{r}_i} \tag{3.24}$$

Where ϕ^k is the recovered Fourier transform of the potential as sum on a lattice of k points, where $\tilde{\rho}(\mathbf{k})$ is the Fourier transform of charge density. In Eq.(3.23) the summation is performed on the charge q_i, in order to recover the potential value. It's important to fix the two main requisites of this method:

- The system must be periodic, since Fourier transform implicitly assume periodicity.
- The system must be in neutral charge condition, so must be add counter-ions (Na^+) in the case of charged proteins.

3.3.6 Simulation routine

Before starting MD, the system parameters must be chosen according to the kind of objects that one want to simulate. In the case of this work different parameters were used according to the simulation context and aim, so here the common setups are reported . A molecular dynamics simulation is divided in multiple step:

System preparation In a first instance, a forcefield needs to be chosen in order to describe the topology of the systen, which is used for calculate the energy between atoms.

Energy minimization (See Section 3.2.3) In a 3D molecular structure can occur unphysical situation like distance between atoms smaller than the VdW radius or improper angles. Most of the time these defects generate enormous forces; for example if in the topology are present two atoms with distance smaller than the VdW radius, the calculated potential from LJ potential fell before the characteristic length, giving an enormous potential and so a big force. Integration of this forces tend to generate unphysical velocities which destabilize the system, in jargon "explode". For suppressing these defects, the first step is a minimization via L-BFGS.

Simulation (start) At the beginning of the simulation the timestep Δt, the bath temperature T_{bath} and the bath pressure P_{bath} and a forcefield are chosen. The initial velocity are sampled from the Maxwell-Boltzmann distribution according to a fixed T_{bath}. The thermostat and the barostat are eventually fixed to the desired temperature and pressure, according to the chosen ensemble where the simulation is carried out.

Simulation (production) During the simulation the integrator update forces and position at every amount of steps, the thermostat and the barostat update also the position and velocities of the particles in order to maintain constant temperature and pressure.

End After the desired amount of timesteps are integrated, the trajectory of the particles is analyzed in order to extrapolate ensemble data. Since in nature proteins naturally tend to stay in the most favourable energetic conformation, there is the need to find the minimum of the potential function respect the simulated system. For this scope a variation to the procedure called Simulated Annealing is used. In this case the T_{bath} is slowly reduced; since the $E_{system} = E_{kin} + E_{pot}$ is constant at a fixed T, reducing the T_{bath} allows to gradually remove the kinetic energy, in order to find the minimum potential energy.

3.4 Observables from MD simulations

In principle, macroscopic properties of a system can be estimated from the trajectory of molecular dynamic simulation, where time averages on these represent thermodynamic properties of a system at the macroscopic scale [35]. Indicating with $\Gamma = \{\mathbf{r}(t), \mathbf{p}(t)\}$ the phase space where $\mathbf{r}_i$ and $\mathbf{p}_i$ are respectively position and momentum, the observable $\langle A \rangle$ can be calculated as:

$$\langle A \rangle_t = \lim_{T \to \infty} \frac{1}{T} \int_0^T A(\mathbf{r}^N(t), \mathbf{p}^N(t))dt \tag{3.25}$$

Clearly, is not possible to solve this integral for a molecular system, given the extraordinary large number of position and momentum degrees of freedom. However, under the ergodic hypothesis, time average of a property can be computed from the ensemble average:

$$\langle A(\mathbf{r},\mathbf{p})\rangle_t = \langle A(\Gamma(t)\rangle_\rho \tag{3.26}$$

And so

$$\lim_{t\to\infty}\frac{1}{T}\int_0^T A(\mathbf{r}^N,\mathbf{p}^N)dt = \int \rho(\Gamma)A(\Gamma)d\Gamma \tag{3.27}$$

Where $\rho(\Gamma)$ is the probability density in the chosen ensemble. This hypothesis allows essentially to calculate a time average of a property directly from the explored ensemble of microstates:

$$\langle A(\mathbf{r},\mathbf{p})\rangle_t = \frac{1}{M}\sum_{i=1}^{N} A(\mathbf{p}^N,\mathbf{r}^N) \tag{3.28}$$

With this approach, also the probability density ρ respect microstates i can be sampled. This will become important in the next subsection, where it will be managed for obtaining the free energy surface (FES).

3.4.1 Free energy surface

An important observable used in this work is the free energy defined from a microscopic description as:

$$F = -k_BT\ln Z \tag{3.29}$$

Since from the partition function $Z(N,V,T)$ only the accessed states are known, the F can be described as a probability distribution where $Z(q_i)$ is the region of the partition function described by a parameter q_i, called collective variable or reaction coordinates:

$$F(q_i) = -k_BT\ln\frac{Z(q_i)}{Z} \tag{3.30}$$

$$Z(q_i) = \int dr^N dp^N e^{-\beta H(r^N,p^N)}\delta(Q(r^N,p^N)-q_i) \tag{3.31}$$

where k_B is the Boltzmann constant, T the temperature of the system and Z the partition function in the canonical ensemble (N,V,T).

In essence, q_i act as a parameter for partitioning all the available microstates $Q(r^N, p^N)$ into a set of macrostates parametrized by q_i. The integral in Eq. (3.31) entails a delta function that filters for only those Boltzmann factors for configurations with the specified q_i. This allow to recover $p(q_i)$:

$$p(q_i) = \frac{Z(q_i)}{Z} = e^{-\beta F(q_i)} \tag{3.32}$$

And so the free energy contributions for every macrostate q_i can be recovered by the use the inverse Boltzmann law in Eq.(3.30):

$$-\ln p(q_i) = \frac{F(q_i)}{k_B T} \tag{3.33}$$

What is obtained at the end of the simulation is a set of q_i values taken at different times. For compute $p(q_i)$ a histogram technique is exploited: the q_i values obtained are divided in a number of bins $[q_i^0, q_i^j], ..., [q_i^{max-j}, q_i^{max}]$. For each bin the occurence of q_i is counted and the values are scaled in order to have $\int_{-\infty}^{+\infty} p(q_i) dq_i = 1$. As collective variables are usually coupled, a slightly different procedure is used.

Let q_1 and q_2 be to reaction coordinates extracted from the trajectory, is possible to describe the surface free energy as:

$$\frac{F(q_1, q_2)}{k_B T} = -\ln p(q_1, q_2) \tag{3.34}$$

Where $p(q_1, q_2)$ is obtained by extracting the joint distribution (2D-histogram): in practice, every event $E_{q_1,q_2} = p(q_1, q_2)$ is counted in a binned grid:

$$[q_1^0, q_1^j], ..., [q_1^{max-j}, q_1^{max}] \times [q_2^0, q_2^j], ..., [q_2^{max-j}, q_2^{max}] \tag{3.35}$$

After the grid is constructed, the free energy contribution of every 2D-bin is calculated from Eq. (3.34).

3.4.2 Reaction coordinates

In molecular dynamics, a reaction coordinate (collective variable) is a 1-dimensional abstract coordinate which in principle, correspond to any parameter that could be measured throughout the simulation [22]. These coordinates can be geometric quantities (Radius of Gyration, RMSD, etc.), chemo-physical quantities (Accessible Surface Area, polarization, etc.) and also more abstract parameters. In this thesis, the Kernel similarity from the Shortest-Path kernel was also used as reaction coordinate. In the following, some of these reaction coordinates are listed. Choosing the right collective variable is not a trivial task because it needs a pre-knwoledge respect important parameter of the examined system. This complication worsen in the case this variables are used as basis for describing the free energy surface because they needs to be uncorrelated. The best choice in this case is Rg and Fraction of native contact.

Radius of gyration Radius of gyration is one of the most common reaction coordinates, is defined as:

$$R_g = \left(\frac{\sum_j \|\mathbf{r}_i\|^2 m_i}{\sum_i m_i} \right)^{\frac{1}{2}} \tag{3.36}$$

Where $\mathbf{r}_j$ is the position vector of the C_α carbon in the backbone in the simulation frame i. Rg is essentially the square root of the weighted average respect positions of a chosen set of atoms.

Fraction of native contact The fraction of native contacts for a conformation can be calculated in different ways. The basic definition involve first to select a reference group in the protein (Sidechains, C_α, charged residues, atoms, etc.) and count as a native contact when distance d_{ij} between element-i and j is below some distance radius. Defining as Nat the ensemble of all of this native contact, the fraction is defined as:

$$\rho = \frac{|\{Nat\}|_i}{|\{Nat\}|_0} \tag{3.37}$$

Where i and 0 are from the i-th and 0-th frames of the trajectory respectively.

In this work, native contacts are defined from the group comprising C_α separated at least by 4 residues in the backbone, with a cutoff radius equal to 6.5 Å[36]. Although crude, this approach was chosen because of the large size of the proteins in question (1639 residues) and the large number of frames obtained in the molecular dynamics trajectories (25000 frames), in which a calculation on the full-atom model was too computationally time-consuming, moreover there was not enough information on any important specific atomic subset to choose as reference group.

RMSD Root mean-square standard deviation (RMSD) is defined as:

$$\mathrm{RMSD}(t_1, t_2) \;=\; \left[\frac{1}{M}\sum_{i=1}^{N} m_i \|\mathbf{r}_i(t_1) - \mathbf{r}_i(t_2)\|^2\right]^{\frac{1}{2}} \tag{3.38}$$

In this form is defined as the measure of the average distance between the atoms of the same structure at moment t_1 and t_2.

Other important scoring functions

Surface-Area Solvent Accessibility (SASA) The solvent accessible surface area (SASA) describes the area over which contact between protein and solvent can occur [42]. Essentially, taken an atom (or a residue) of interest, with an approximated method is possible to estimate how much surface of the probed object s theoretically available to the solvent. This is clearly dependent on the type of solvent used in the system, neighborhood and quality of the residue of interest; a residue with a low SASA are meant as ”buried”, which it means that there are no contacts with the surrounding solvent.

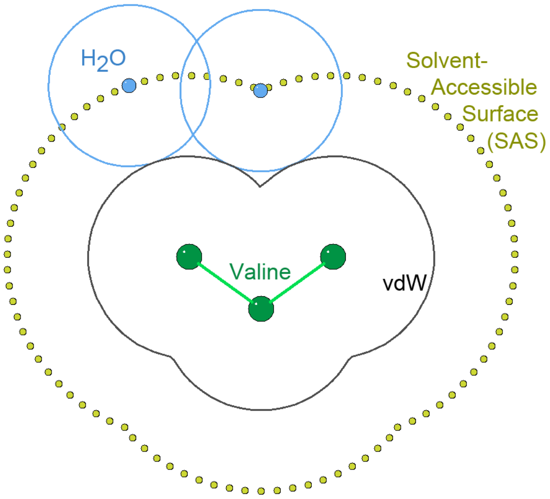


Figure 3.4: Example of SASA calculation

Shake-Rupley is most used algorithm for calculating SASA of a given atom or residue [42].

Kernel similarity Given the desire to find faster and finer computation than the fraction of native contacts described above, a novel score defined as the normalized dot-product from the kernel computation on graphs (See Section 2.2.2). Specifically, the score is defined as:

$$\text{Kernel Similarity} = \frac{k(G_0, G_i)}{\sqrt{k(G_0, G_0)k(G_i, G_i)}} \tag{3.39}$$

Where G_0 and G_i stands for the Residue Interaction Network at 0-*th* and i-*th* frames of the trajectory respectively.

3.5 FG-MD protocol

In the following, the FG-MD protocol is described.

3.5.1 Intro

Taken a protein of interest, the native structure correspond to the unique geometric conformation that sit on the global minimum of the free energy. The most common way for getting this native state is using Molecular Dynamics simulations, that allow to explore the free energy landscape. Despite the key role played by these methods, discovering the native structure remains an open problem for the reasons listed below:

1. Exploring the energy landscape needs the overcome of high energy barriers between them, and this is extremely demanding from the computational point of view.
2. A complete description of the systems (and so his real energy landscape) is essentially impossible.

While point 1 can be tackled by tuning the simulation parameters and setting the right simulation environment, point 2 hinge upon the intrinsic uncertainty given by the approximations embedded in the forcefield. Most of these models, called physical-based forcefields, use precomputed values obtained by quantum-mechanical calculations and fit well for describing local interactions in the system. Another class of models are the knowledge-based forcefield, where the values are extrapolated from a large dataset of structures. These two classes have in some way mutually exclusive performances: the physical one seems to have a better convergence but most of the time it drives the structure away from the native state, the knowledge-based instead seems to have in general poor performance because is highly-specific and so is not also suitable for protein close to the class from which the forcefield where obtained.

So, the FG-MD[56] protocol proposed by Zhang and collaborators, take advantage of this performance difference to derive his own forcefield, where the distance-maps from structures close to the initial one and a knowledge H-Bond potential appears in forces computed as a "correction" to AMBER99[40]. This seems to work well for refining near-native structure obtained by homology modelling.

3.5.2 Assumptions

In order to obtain such term and given its unknown nature, the description must rely on some assumptions listed below. Some of these are of general validity.

1. The backbone H-Bond network greatly contribute to maintain the protein secondary structures in folded state.

2. The long-range interactions reflects mostly on the C_α protein local topology, maintaining properly folded the secondary structures and giving the final tertiary structure.
3. If the initial model is near-native (this assessed by the use of a bunch of scoring functions, explained in the 1.4 section) the topology (and so the energy landscape) can be improved with a good accuracy.

The 1^{st} and the 2^{st} points are the basis for constructing the FG-MD forcefield:

The backbone H-Bond network is defined as the ensemble of all H-Bonds between the carboxyl group and the amino group of every residue main carbon. The angles α_{N-H-O} and β_{C-H-O} and the distance d_{O-H} characterize the secondary structure. In the FG-MD protocol these important values are forced to averages, pre-computed from a set of experimentally high-definition structure.

3.5.3 Alpha-carbon structure and his role in refining structure

The C_α structure is the chain formed by the main backbone carbons each carrying a sidechain group. The Distance-map of C_α can be regarded as a matrix where each entry is the distance r_{ij} from every C_{ij}. This is a very important object because it binds the structure of the protein to its folding state, while the non-bonded interactions between different C_α sidechains are responsible for the quality of the folding. So, proteins with similar C_α distance-map are expected to be in the same fold. In fact, all of the most famous scores like TM-score and

GDT-HA use the C_α distance as metrics for measuring how much two structures are similar, that it are in the same fold. In the FG-MD protocol, TM-score is deployed for searching through a non-redundant PDB database structures which their C_{alpha} distance map is close to the protein of interest, and use the distance-map for guiding the simulation toward the native state. TM-score, H-Bond score as GDT-HA are described in Section 3.4.2 and Kernel similarity is introduced in the next paragraph.

3.5.4 Original score

TM-score TM-score [58] is one of the most famous and important scoring function: given two structures, the initial model and second as the native state, it first finds the best superposition between the two structures (aligning a random set of carbon iteratively) and then counts the C_α from the initial and from the native template whose distance between them is less than 5 Å. The score is calculated as:

$$\text{TM-score} = \max\left[\frac{1}{L_N}\sum_{i=1}^{L_c}\frac{1}{1+(\frac{d}{d_0})^2}\right] \tag{3.40}$$

where L_N is the length of the amino acid sequence of the target protein that is interested, L_c the number of residues that commonly appear on the template and target structures, d_i is the distance between the i-th pair of residues between the template and the target structures and d_0 is a distance scale that normalizes distances. This score is based on the assumption that sequence-gap between two amminoacid sequences does not affect the recognition of the folding degree between two structures. In fact this is a sequence-independent score and it use both in the global sequence-dependent alignment (See Sectoin 3.5.5) and on the sequence-independent alignment TM-Align[57], that is the basic tool used in this protocol.

GDT-HA The Global Distance Test-High Accuracy (GDT-HA) is the main scores used in CASP competition [55]. The GDT score is calculated as the largest set of amino acid residues' alpha carbon atoms in the model structure falling within a defined distance cutoff (0.5 Å,1 Å,..) of their position in the experimental structure, after iteratively superimposing the two structures. It is very strict because it war originally devised for guiding the X-Ray refinement of protein. In fact, the MD based refining improve it just of a slight amount.

HB-score In order to measure the similarity between the H-Bonds network of the native and an alternative model, Zhang et al.[56] defined a score based on the number of consensus hydrogen bonds between the two structures:

$$\text{HB-score} = \frac{\text{N}^\circ \text{ of consensus hydrogen backbone bonds}}{\text{N}^\circ \text{ of hydrogen backbone bonds present in the native model}} \tag{3.41}$$

This score is calculated using HBPLUS[31], a software which gives in output the H-Bond network of a given protein. Note that in the original publication the authors didn't give information on what kind of H-Bond interactions (MC-MC and MC-SC) are taken in account in this score. In the present thesis, a modified algorithm is proposed within the FG-MD framework, and instead of counting the H-Bonds and mathching pairwise, the H-Bond network is first obtained by the auxilium of the RINmaker (See Section 5.1.1) and after compared pairwise with the native one. This method is explained in Section 5.2.1.

3.5.5 TM-Score software and TM-Align

TM-Score software Despite the sequence-independent nature of the TM-Score function, the first software made with this score was a sequence-dependent structure aligner called TM-Score.

It finds the best superimposition of two structures (initial vs. native), counts the C_α with $d < 5\,\text{Å}$, after this set is found it show the TM-Score, the RMSD, GDT-HA and the alignment of the two structures. Since it's sequence-dependent, the difference of two residues in the sequence is counted as a -1 penalty. So, from two completely different structures, the TM-Score of the superposition probably will be close to 0, conversely it will be 1 for two closely matching structures.

TM-Align This tool is of prominent interest in bio-informatics area. The main difference between TM-Score software is on the algorithm used for the superposition: it doesn't compare two structures only globally but also locally, in a sequence-independent fashion. Essentially, given two structures the TM-Score doesn't find only the best axis between them, but also the best local-superposition on the C_α with $d < 5\,\text{Å}$. In fact, it discriminates better between folded or non-folded structure. And it it commonly assured that a TM-Score > 0.5 indicates nearly identical folds. A useful function of this program is the *Cross-comparison* mode, in which given a set of PDBs as input, it provides a matrix of the calculated TM-Score for each pair in the set.

With the knowledge of the tools and of the assumptions discussed above in mind, the FG-MD protocol can be now discussed.

3.5.6 The modified-AMBER99 FG-MD Forcefield

Basically the ultimate goal of a forcefield is to provide a way to express the potential energy between entities in a system, so the improved forcefield used by FG-MD can be rewritten as:

$$E = E_{\text{AMBER99}} + E_{\text{FG-MD}} \tag{3.42}$$

While AMBER99 is expressed by Eq. (3.2), the FG-MD term is made as a sum of several terms, Each one discussed in the following paragraphs.

$$E_{\text{FG-MD}} = E_{C_\alpha} + E_{HB} + E_{C_\alpha - \text{CLASH}} \tag{3.43}$$

Knowledge-based H-Bond potential -E_{HB} This terms compute energy between donor and acceptor (N-H-O,C-O-H) groups from the main chain. The energy is defined as:

$$E_{\text{HB}} = k_1(d_{ij} - d_0)^2 + k_2(\alpha - \alpha_0)^2 + k_3(\beta - \beta_0)^2 \qquad r_{ij} < 3\,\text{Å} \tag{3.44}$$

Where d_{ij} is the distance between O-H, α is N-H-O angle and β is the H-O-C angle.The equilibrium distance and angle d_0, α_0, β_0 are respectively 1.95 Å,150° and 160° respectively. These averages were computed by Zhang et al.[56] from a reference database of high-resolution structures.

C_α **clash potential -**$E_{C_{\alpha-\mathbf{clash}}}$ In order to speed-up the convergence a clash potential was added for relaxing the $C_{\alpha-ij}$ with serious clash between them. It's an important term because it allows to disorder the system when all of the other forces tends to constraints. This terms is continuously evaluated during the annealing, and prevents apparition of local clash.

$$E_{C_{\alpha-\text{CLASH}}}(r_{ij}) = k(3.6 - r_{ij}) \qquad r_{ij} < 3.6\,\text{Å} \tag{3.45}$$

Where the force constant k is $200\,\frac{\text{kcal}}{\text{mol}}$ and r_{ij} is the distance in Å between the $C_{\alpha-ij}$ of the initial model. The cutoff guarantees the application only on local clash.

C_α **distance restraints-**E_{C_α} This is the heart of the protocol: the distance restraints added from global and fragmental template. The process is explained after the equation is presented:

$$EC_\alpha(r_{ij}) = k_1(r_{ij} - r_{ij}^{(1)})^2 + k_2(r_{ij} - r_{ij}^{(2)})^2 + k_3(r_{ij} - r_{ij}^{(3)})^2 \qquad r_{ij} < 15\,\text{Å} \tag{3.46}$$

Where r_{ij} is the distance in Å between the $C_{\alpha-ij}$ of the initial model, and $r_{ij}^{(2)}$,$r_{ij}^{(3)}$ the corresponding distance taken from $C_{\alpha-ij}$ of the global templates and fragmental templates. The $r_{ij}^{(1)}$ distances from the initial model are added in order to avoid unfolding that occour during simulated annealing at high temperature,and it allows also to speed-up the simulated annealing that is carried out at high temperature. The $r_{ij}^{(2)}$ and $r_{ij}^{(3)}$ distances are collected from PDB structures searched by TM-Align software against a database of high-resolution structures (Xray resolution $< 1.5\,\text{Å}$).The 20 templates with highest TM-Score are selected for collecting the distance-maps. An outline is given in the next subsection.

The distance cutoff of $r_{ij} < 15\,\text{Å}$ was chosen according to the observation which highlights the fact that long-range retraints have a bad performance in terms of refinement.

The importance of this terms emerges from the fact that exists more accurate fragments rather than global templates, in fact the application of this potential and the distance-restraint from the initial model is sufficient to achive a good refining of the model of interest.

3.5.7 Fragmental routine and Simulated Annealing

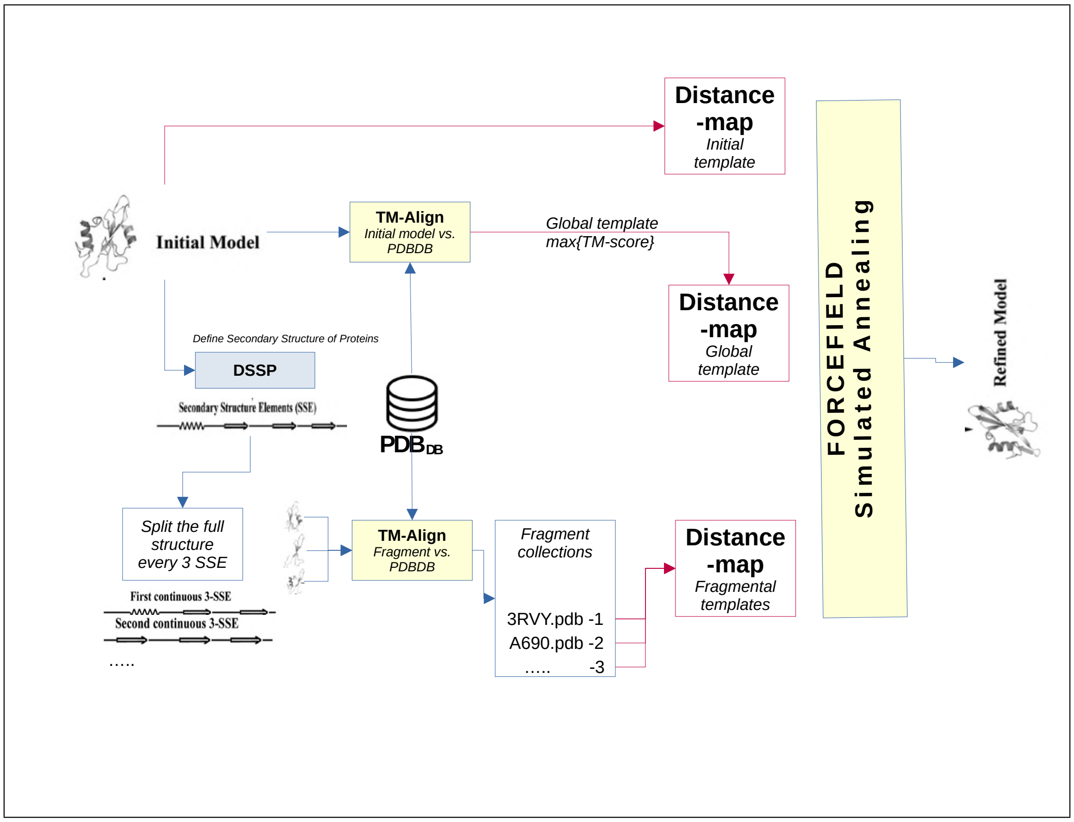


Figure 3.5: FG-MD protocol with detailed description of distance-map sampling routine

The FG-MD protocol shown in the Fig. 3.5 can be divided into three main parts, which are Distance-maps sampling for the forcefield construction and simulated annealing. Each of these steps is described below:

Distance-map sampling

The values that will form the modified part of the forcefield in Eq. (3.43) are, as mentioned above, obtained from the Distance-map from the initial model, global template and for every discovered fragment in the local PDB database (See Eq.(3.46)) . Since a TM-score > 0.5 means an high grade of folding, both fragmental and global templates are chosen with a cutoff of 0.5. All Distance-maps are sampled with respect to the cutoff in Eq.(3.43).

Distance-map from the initial model At this stage, the Distance-map is derived directly from the initial model.

Distance-map from the Global template Before collecting the Distance-map, TM-Align with the initial model in input is run against the local PDB database and the structure (i.e. Global template) corresponding to the highest TM-Score is collected. After this step, the Distance-map is obtained from the selected Global-template.

Distance-map from Fragments At first, a DSSP algorithm assigns secondary structures along the backbone of the initial model. After this, the initial model is divided into all possible fragments formed by three consecutive secondary structures. Each of the obtained fragments is run against the local PDB database with TM-Align to search for similar fragments (i.e. with an high TM-Score). Fragments with the highest TM-Score found by TM-Align are collected for every evaluated initial fragment. Lastly, Distance-maps for every collected fragment are sampled.

Simulated Annealing After all of the Distance-maps are obtained, the modified-AMBER99 is constructed and applied to the initial model for the Simulated Annealing.

3.5.8 Original implementation

This protocol is only available in a web version[15] and the authors have not coded any distribution to implement locally. For the needs of this work, a local version was implemented and consequently integrated into the new implementation of the Pipeline (See Section 5.2.1).

3.5.9 Problems and future improvements

Despite the proven reliability on refining, this protocol rely greatly on some aspects:

- It's highly dependent on the number of available PDB data and on their experimentally resolution. There are some classes of protein (such as the membrane proteins) for which the experimental resolution, it is not yet sufficient for refining with this protocol. For the MOESM3 example, the only structures with appreciable global TM-Score are the 6A90, that has a resolution of 2.9 Å and 3RVY with a 1.5 Å resolution but the latter has less residues than MOESM3.
- The H-Bond potential essentially drives the backbone H-Bond toward pre-computed averages, used as equilibrium parameter. These averages don't account for the specific type of secondary structure were the potential is applied and do not discriminate

the class which the initial model belongs to. For this point an improvement can be carried out by RIN graph analysis: after searching with TM-Align, the graph of these fragment contains the equilibrium distances of the backbone H-Bond. Applying this distance as a contrain to the correspondent H-Bond in the inital model, a faster and more accurate convergence could be probably obtained.

- Although Zhang's energy landscape of near-native model obtained by FG-MD potentials reassembles a funnel-like landscape, when the TM-Score is less than 0.4 an high energy barrier appear, making the global minimum unaccessible.

Chapter 4

Old Computational Pipeline

4.1 Introduction

In this pipeline, homology modelling, graph theory and machine learning are combined in order to investigate and find similarities between different point mutations of a given protein, where the mutations are pre-known and labeled with respect to whether they are pathogenic or non-pathogenic [51]. Given a Wild-Type and a set of known mutation, the workflow can be summed up as follow:

1. Production of point-mutated structures from the Wild-Type by SWISS-MODEL [54], a powerful homology modelling tool.

2. Energy minimization of the obtained structures using Zhang's FG-MD implementation.

3. Production of Residue Interaction Networks by RING2.0[38] tool on the obtained refined structures.

4. Cross comparison of the obtained RINs by the means of graph kernel methods.
5. Supervised learning of the obtained Gram-matrix in point 4 by the use of support vector machine (SVM).
6. Clustering of the Gram-matrix by the use of Dominant Set (DS) [37].

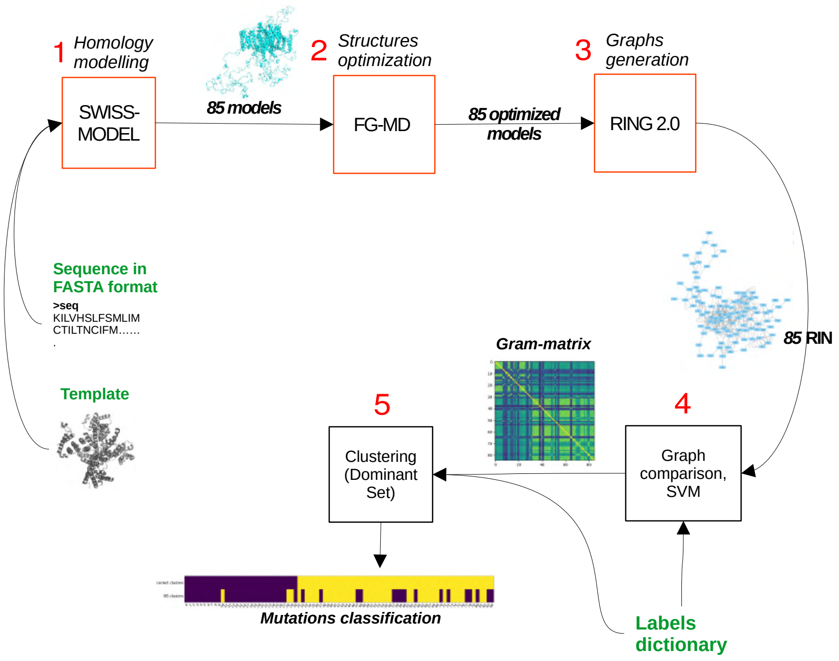


Figure 4.1: Flowchart of the old implementation. Boxes in red refers to web-tools, boxes in black refers to in-house scripts and in green the user inputs.

The aim of the workflow depicted in Fig. 4.1 is to discern pathogenic or non-pathogenic mutation. This task is done by feeding the SVM and the Dominant Set Clustering algorithm with the results from the graph kernels applied on graphs, which is the Gram-matrix (See Eq. (2.7)). What is expected to find in this matrix is a number of clusters grouping similar topology from mutants with the same label. The pre-known labels for the mutations allows to assess if the pipeline is able to discriminate between pathogenic or non-pathogenic. The passage from refined structures to the topology reflects the crucial assumption of the whole process: the input refined structures must represent a geometry where the mutation on the starting structure has achieved its effect, this means to assume that a point mutation have an effect on the overall topology, especially for globular and membrane proteins[1].

4.2 Tools of the pipeline

In this section we illustrate the various computational tools involved in the various steps of the pipeline.

4.2.1 Homology modelling

How it works The ultimate goal of protein modeling is to predict a structure from its sequence with an accuracy that is comparable to the best results achieved experimentally [7]. Homology modelling has matured into an important technique in structural biology, significantly contributing to narrowing the gap between known protein sequences and experimentally determined structures [54]. This technique is based on the observations that the structure of a protein is uniquely determined by its amino acid sequence and that similar sequences adopt practically identical structures, and distantly related sequences still fold into similar structures [7]. Thanks to these observation, given a mutated sequence of a wild-type, the mutated 3D structures can be inferred from the conserved sequence/geometry of a selected template respect the provided mutated sequence. The engine used in this work is SWISS-MODEL [54] and the general workflow of this method can be described in these steps:

1. User provide a sequence in FASTA format and a related template.
2. The provided sequence in step 1 serve as a query to search for evolutionary related protein structures
3. For the selected template, a 3D protein model is automatically generated by transferring conserved atom coordinates as defined by the target-template alignment
4. On the resulting structure, residue coordinates corresponding to insertions/deletions in the alignment are generated by loop modelling (ProMod3[49])
5. Side-chains are modelled on the full backbone of the resulting structure (SCWRL4[25]).
6. The final model quality is estimated by the scoring function QMEAN and QMEANbrane[48].

During the step 4-5, the backbone is constructed and optimized piece by piece by Monte Carlo minimization with CHARMM22 forcefield.

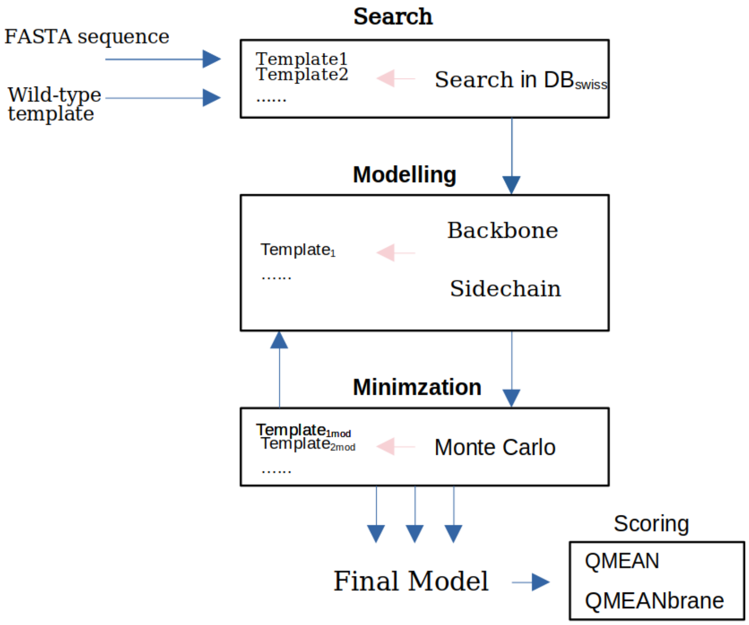


Figure 4.2: SWISS-MODEL algorithm

Two important instruments of this tool needs a description: ProMod3 and SCWRL4.

ProMod3 ProMod3 [49] is a versatile homology modelling toolbox. The main characteristic is the highly efficient algorithm used for building backbone residues in the modelled structures. The decision on which residue to build and at what angle (ϕ, ψ) is performed with the auxilium of 12 scoring functions, which take in accounts local characteristic (Secondary structure, solvent accessibility, clash, ecc ...) and global characteristic (All atoms interactions, packing, ecc..)[3, 10].

SCWRL4 SCWRL4 [25] is a sidechain builder that assigns sidechain structures to the final model, in the case are not present in the initial template. It rely on a discrete rotamers library as data source and on a fast graph-tree decomposition algorithm. The crucial point is that this algorithm guarantee to find the sidechain conformation sitting and the global minimum with respect all the possibile rotamers in the given backbone conformation.

Molecular dynamics / Refining In this stage is important to choose the tools according to the aim of the project:

- If the aim is to bring the structures closed to the native state, the best strategy is to employ protocols for refining.
- If the aim is to assess the structural change given by a point-mutation, the strategy is more focused on molecular dynamics.

Nowadays, there are a lot of refining protocols and software. The one used in this was Zhang's implementation of the FG-MD[56] protocol described in 3.5.

Residue Interaction Network generation In this stage, Residue Interaction Networks (See Section 2.1.2) were produced with the RING2.0[38] web version.

Support vector machine (SVM) Support vector machine is a supervised learning approach used to analyze a given data set and to build a model that separates data into a desired and distinct number of classes [8]. In the 2-dimensional case, from a set of (x_n, y_n) where x_n is the input vector and y_n is the associated label, the SVM try to find the optimal hyperplane between these points which can separate the data according to the given labels. In the pipeline case, x_n are the obtained kernel value ($k \in [0, 1]$) and y_n are the labels (PAT or NEUTRAL) associated with the pathogenic behaviour of the mutations and from the output is considered only the accuracy of the prediction.

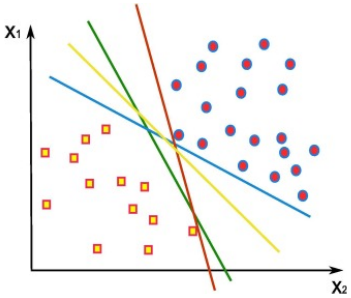


Figure 4.3: Example of separation hyperplanes

Dominant Set (DS) The unsupervised partitioning of data (or clustering) (See Fig. 4.3) is a problem that pervades computer vision research[37]. In this case, the main effort of clustering algorithms is to find similarity and discriminate between pathogenic and non-pathogenic mutations from the Gram-matrix obtain from Kernel method on graphs. In this work the Dominant Set [37] is used. What is expected to obtain in this final stage, is a classification of the mutations given in the step 1 between pathogenic or not-pathogenic (PAT or NEUTRAL) .

4.2.2 Workflow of the old pipeline

As shown in Fig. 4.1, what the input pipeline needs are:

- A Label Dictionary as a text file where every point mutation of the Wild-Type was reported with the following coding (Native residue)-(Backbone position)-(Mutated res),(Label) where:
    1. Native residue is the residue type found in the Backbone position on the native model
    2. Backbone position is a number pointing to the Backbone position that needs to be mutated
    3. Mutated residue is the residue type to mutate in the Backbone position
    4. Label is a number equal to 1 or 0 respectively if the mutation is pathogenic or not.
- An evolutionary related template.
- The full FASTA sequences of the point-mutated Wild-Type sequences reported in the Label Dictionary.

Below every step of the aforementioned implementation is listed from an operational point of view:

1. User provide to SWISS-MODEL Expasy Server the 85 query sequences in FASTA and an evolutionary related template. Each provided sequence represents the point-mutated Wild-Type sequence according to the Label Dictionary.
2. The obtained 85 models from point 1 are evaluated by QMEANbrane (See Section 4.2.1) and manually uploaded (one by one) to the FG-MD server through the graphical interface.
3. Refined model obtained at point 2 are uploaded (one by one) to the RING2.0 web server through the graphical interface.

4. The obtained Residue Interaction Networks at the point 2 are evaluated by the Kernel scripts.

5. The Gram-matrix from point 4 (See Eq.(2.7)) and Label Dictionary are fed to the Support Vector Machine script.

6. The Gram-matrix from point 4 and the Label Dictionary are fed to the Dominant Set Clustering script. The output is a classification of predicted labels from Gram-matrix compared with the known labels.

Although the described implementation has shown remarkable results when applied to refined structures derived from homology modelling, the composition as a sequence of disconnected web-tools makes it difficult, if not impossible, to automate the process for future application in a big-data context. Moreover, the Zhang's FG-MD web-version shown a response latency of 24-48 hours for every structure in addition to the poor control which a user has on this tool. The RING2.0[38] Residue Interaction Networks generator also has also shown poor performances in terms of computational time, since the elaboration of one of the considered structures in this work requires between 15-20 min; moreover, even the latter tool has, in our opinion, a crude description of the parameters (energy, donor-acceptor angles, etc.) related to non-covalent interactions. These observations prompted the search for a new, faster and more flexible architecture, as described in the following chapter.

Chapter 5

New Computational Pipeline

5.1 New implementation

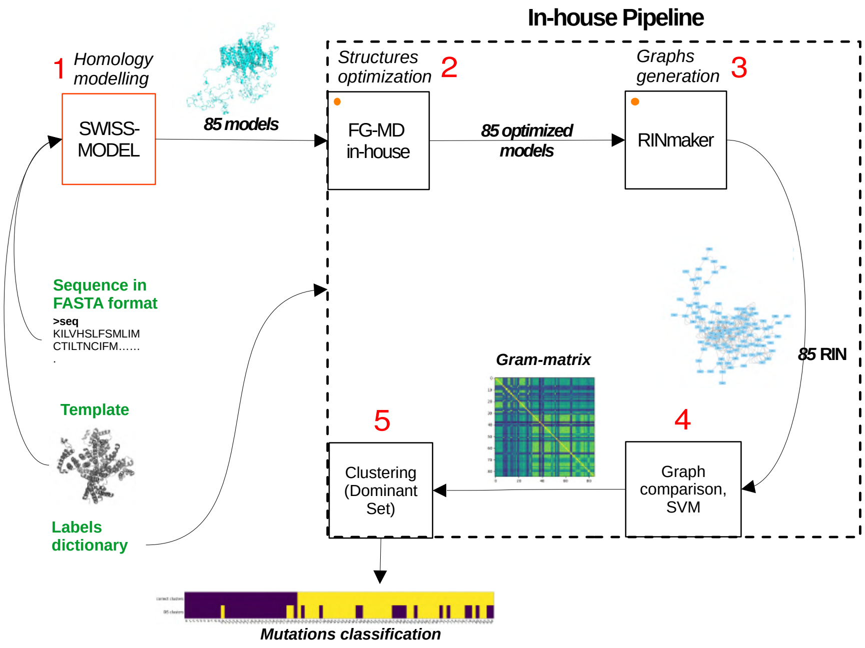


Figure 5.1: Flowchart of the new implementation; dotted line indicates incorporation into a single tool.

In Fig. 5.1 the new implementation is presented. Most of the tools explained in Section 4.1 are now embedded in a local and flexible architecture described in Fig. 5.2. Specifically, the two major improvements respect the previous implementation are:

- Substitution of the FG-MD web version with a local in-house implementation GPU accelerated (See Section 3.5).
- Substitution of RING2.0 web server with the in-house implemented tool RINmaker.

The Pipeline is coded in Python 3. The architecture can be regarded as a state machine. Every step of the pipeline modify the global state and update the various data structure.

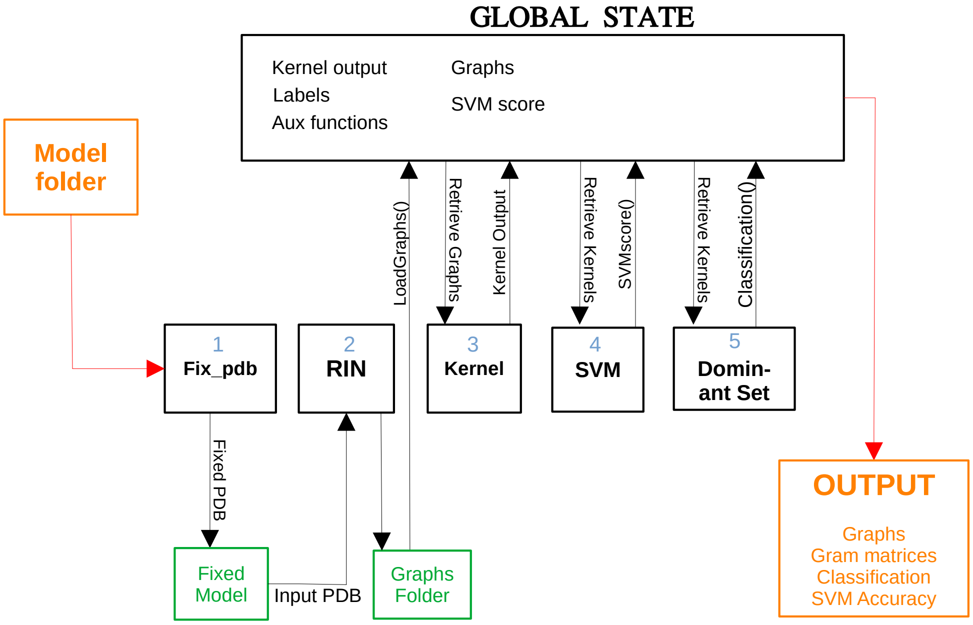


Figure 5.2: Architecture of the pipeline script

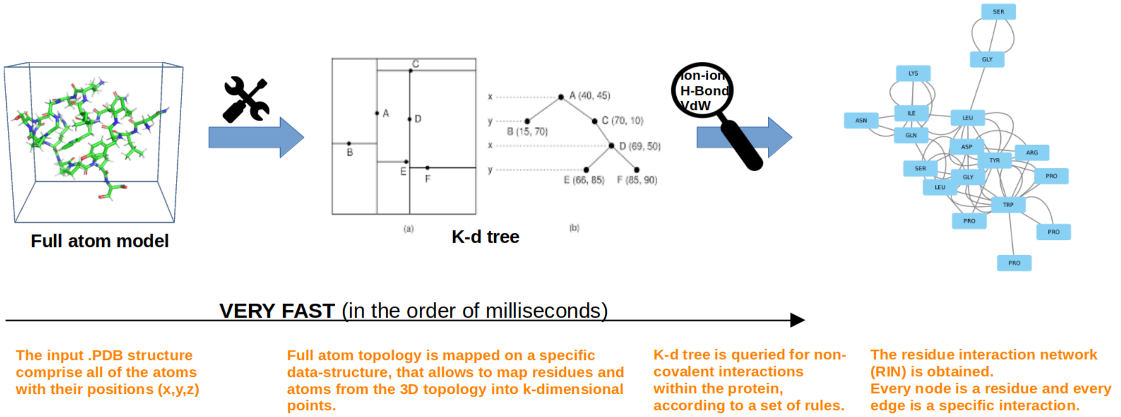
The architecture depicted in Fig.5.2 was chosen in order to give maximum flexibility to the modification of the pipeline. A description of each step i given below:

1. **PDBfix** It's a series of functions comprising Reduce, structures checker and a renumbering function for the residues, in order to achieve a secure uniqueness for every residues in the chain.
2. **RINmaker**: In this step RINmaker is called for generating the graphs from the Fixed model folder. The user can decide the parameters for generate the RINs. The generated graphs are loaded into memory with the LoadGraphs() function.
3. **Kernel**: At this stage the user can choose between a set of kernels, and can compute one as many as needed. The supported kernels are Vertex-Histogram, Edge-Histogram, Subgraph Matching, Pyramid Match, Shortest-Path and Neighborhood Hash. The output from this step is stored in the global context.
4. **SVM**: From the stored Gram-matrices the SVM try to separate clusters and gives in output the obtained accuracy.
5. **Dominant Set**: Dominant Set takes in input the stored Gram-Matrices and produce the clustering. After every step the output is stored in a folder where it contains the Graphs and subfolders which belong to every kernels; they contains Gram-matrices,SVM accuracy and the classification.

As depicted in the flowchart of the pipeline (See Fig. 5.3), the obtained graphs from RINmaker are compared by the use of graph kernels, described in Section 2.2.2. Before the production use of the aforementioned pipeline, the FG-MD in house version and the RINmaker software were tested as described in the following sections.

5.1.1 RINmaker

RINmaker is the tool used in this pipeline for generating residue interaction networks (RIN), it has been developed by Ca' Foscari Computer Science team. It takes the full atom structure in .PDB format as input and returns a RIN in the .GraphML format, the wokflow is shown in 5.3. This software is based on a data structure called k-dimensional tree, which allows to organize atoms in the inserted topology in a binary tree where every node is a k-dimensional point. The power of this data-structure resides on the performance achieved when the tree is queried for finding the nearest neighbours of a given point with respect to a specific distance treshold (*cutoff*).



The input .PDB structure comprise all of the atoms with their positions (x,y,z)

Full atom topology is mapped on a specific data-structure, that allows to map residues and atoms from the 3D topology into k-dimensional points.

K-d tree is queried for non-covalent interactions within the protein, according to a set of rules.

The residue interaction network (RIN) is obtained.
Every node is a residue and every edge is a specific interaction.

Figure 5.3: RINmaker workfow

The non-covalent interactions are selected according to a set of rules involving geometric factor respect the quality of atoms involved in supposed bonds[18] . Every nodes of the obtained RIN is labelled with the position on the backbone and the type of residue in this form:

(Chain):(Position)_(Residue). Since in the RIN there is at most one node for each amino-acid, it follows that each node has a different label.

If there exists an interaction between two nodes, an edge is created, with the following informations:

- First node in (Chain):(Position)_(Residue) form
- Second node in (Chain):(Position)_(Residue) form
- Kind of interaction in the form (Interaction):(Chain or Side) where Chain or Side account for MC_ MC for Main-Chain interactions or SC_ SC for Side-Chain interactions. Also MC-SC and SC-MC is marked.

The edge representing bonds are also enriched with attributes (energy, atoms involved, distance, ecc.).

Like other generators of Residue Interaction Networks [38], RINmaker allows the user to customize the *cutoff* parameter for each type of interaction and to return the graph with interactions respecting a given global rule in the input parameter *interaction-type= SELECTION*. With respect to the latter:

- *all*: The edges ensemble is formed by all the interactions found among the candidates.
- *multiple*: For each type of interaction only the best one is selected from each pair of residues.
- *one*: For each pair of residues only the best interaction is selected.

- *hbond-realistic (optional)*: For each pairs of residues only the best HBOND:MC-MC interaction is selected.

Interaction type	**Cutoff**
H-Bond	3.5 Å
VdW	0.5 Å
Ion-ion	4 Å
π-cation	5 Å
π-π	6.5 Å

Table 5.1: *Default* cutoff parameters

5.1.2 Test for the RINmaker software

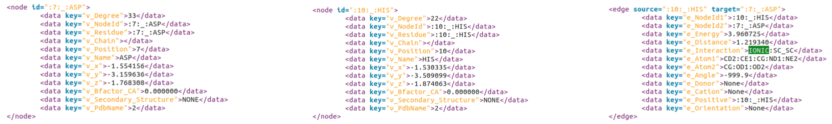
At the beginning of this thesis, RINmaker was already completed. Only the test phase was missing. An ad-hoc environment was created for this specific work from the Ca' Foscari Computer Science Department [46].

Test-cases

Most of the tests where built with PyMol with the auxilium of some Python script. For every test the input parameter and the expected output must be strictly in accordance. Every input parameter was checked, labeled and evaluated manually before the test run.

Example test case fully explained (test no°1) In this test the ionic bond formed by two residues of type Histidine and Asparagine is assessed. The two residues were bring in close promixity, with center of mass

$d = 1.22\,\text{Å}$ that is less than the threshold value of $4\,\text{Å}$. The charged atom of Histidine and Asparagine are respectively, in PDB code, OD2(-) and NE2(+), so they are candidates for a ion-ion bond. Since the two rules are respected this test is positive and a ionic bond is effectively added to the graph.



```
<node id=":7:_:ASP">
        <data key="v_Degree">33</data>
        <data key="v_NodeId">:7:_:ASP</data>
        <data key="v_Residue">:7:_:ASP</data>
        <data key="v_Chain"></data>
        <data key="v_Position">7</data>
        <data key="v_Name">ASP</data>
        <data key="v_x">-1.554156</data>
        <data key="v_y">-3.159636</data>
        <data key="v_z">-1.768308</data>
        <data key="v_Bfactor_CA">0.000000</data>
        <data key="v_Secondary_Structure">NONE</data>
        <data key="v_PdbName">2</data>
</node>
```

(a) Node1 tag

```
<node id=":10:_:HIS">
        <data key="v_Degree">22</data>
        <data key="v_NodeId">:10:_:HIS</data>
        <data key="v_Residue">:10:_:HIS</data>
        <data key="v_Chain"></data>
        <data key="v_Position">10</data>
        <data key="v_Name">HIS</data>
        <data key="v_x">-1.530335</data>
        <data key="v_y">-3.509099</data>
        <data key="v_z">-1.874063</data>
        <data key="v_Bfactor_CA">0.000000</data>
        <data key="v_Secondary_Structure">NONE</data>
        <data key="v_PdbName">2</data>
</node>
```

(b) Node2 tag

```
<edge source=":10:_:HIS" target=":7:_:ASP">
        <data key="e_NodeId1">:10:_:HIS</data>
        <data key="e_NodeId2">:7:_:ASP</data>
        <data key="e_Energy">3.960725</data>
        <data key="e_Distance">1.219340</data>
        <data key="e_Interaction">IONIC:SC_SC</data>
        <data key="e_Atom1">CD2:CE1:CG:ND1:NE2</data>
        <data key="e_Atom2">CG:OD1:OD2</data>
        <data key="e_Angle">-999.9</data>
        <data key="e_Donor">None</data>
        <data key="e_Cation">None</data>
        <data key="e_Positive">:10:_:HIS</data>
        <data key="e_Orientation">None</data>
</edge>
```

(c) Edge tag

Figure 5.4: Reported tags in the .GraphML output

At present since the number of tests designed for all types of bonds is very high, it is not possible to show them here. A partial list of the tests performed is presented in Table C.3. Although this, we can guarantee the proper functioning of the RINmaker software, by virtue of the fact that the results of new implemented pipeline are strictly in accordance when providing as input the structures obtained from FG-MD web server for the previous pipeline, as we will see in Section 6.2.

5.2 An in-house version of the FG-MD protocol

In this section we present the in-house implementation of the FG-MD protocol. From now on it will be referred as in-house FG-MD.

Fig. 5.5 shows the workflow of the in-house FG-MD. Notice that the difference with respect to the Zhang FG-MD workflow (See Fig.3.5) is the software used for the secondary structure prediction. More precisely we use STRIDE[17] where the Zhang software is unknown. Moreover the toolkit for the in-house FG-MD molecular simulation is OpenMM[12] while in the Zhang FG-MD it is LAMMPS[50].

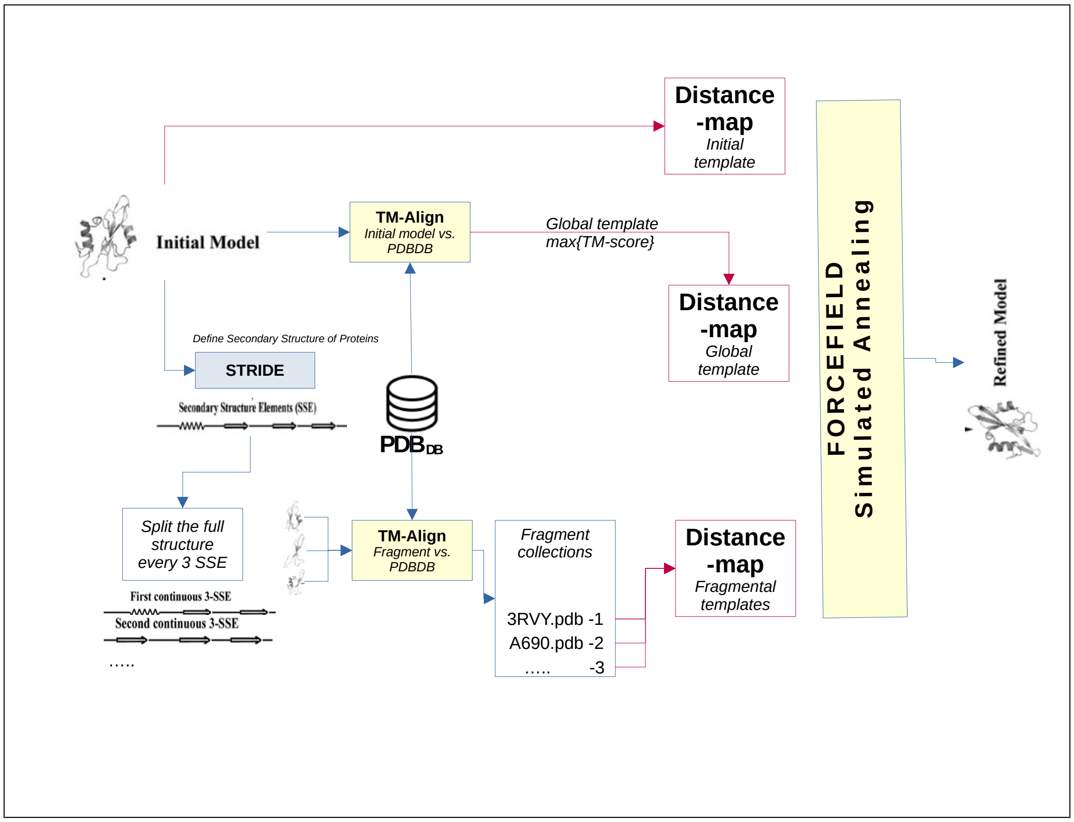


Figure 5.5: FG-MD protocol with detailed description of distance-map sampling routine

This version of FG-MD was coded in Python3, and use the above-mentioned OpenMM as engine for the simulation and OpenStructure[5] package for protein modelling and manipulation. Is fully automated. Regarding the effiency OpenMM[12] is GPU accelerated, so it's more faster than the CPU version of LAMMPS. Anyway, the real bottleneck of the process is the querying routine by TM-Align on the PDB database: empirically the query time for the 1eqm (200 residues) is about 20 minutes on a 8-core Intel i7 and increase to 90 minutes for the NaV1.7 Wild-Type full structure (1650 residues). This also highlights the urgency of find a more efficient organization of the database. The first version is public available at GitHub repository `https://github.com/jacopomoi/FGMD.git`.

5.2.1 Calibration test for the in-house FG-MD implementation

For the refinement tests the MOESM3 refined Wild-Type and two structures provided by Zhang were selected. The results were strictly monitored during the simulated annealing by the use of scoring functions both introduced in the original paper and newly defined. Since the authors gave just little hints regard the simulation parameters, the protocol was tested many times with different parameters (bath temperature, integration time, H-Bond contraints ecc..). The best simulations were chosen for evaluation. When the capability on refining was assessed, the protocol was also tested against the MOESM3 and 6a90 variants.

Methods

All of the simulation where carried in explicit solvent TIP3P water model with Langevin integrator whose timestep was set to 2 ps. The system was set to be periodic with a cell size manually set respect the dimension of the target potein and the long-range interaction were taken in account with the particle mesh Ewald method. The force-fields used were AMBER99 for the gorund-truth simulation and the modified-AMBER99 (See Eq. (3.43)) for testing the in-house FG-MD implementation. The database used for conformation sampling was constructed with PISCES [53] server, and includes all of the known structure with $R_{factor} \leq 2.5$. The simulation were performed on the Ca' Foscari Turing cluster, using the embedded GPUs as accelerators.

The authors of the original study[56] gave me some hints on what parameters are used in their protocol. Such parameters are listed in Table 5.2 together with the parameters of the in-house version.

Parameter	**Zhang (LAMMPS)**	**In-house (OpenMM)**
Integrator	Verlet	Verlet
Thermostat	Nose-Hoover	Andersen
Cutoff	1.0 nm	1.0 nm
Periodic	yes	yes
Solvent	unknown	TIP3P
T_{max}	100K	100K (can be set)
T_{min}	1K	1K
T_{step}	1K	1K
Tot. Steps	10^4	10^5

Table 5.2: Comparison between parameters of Zhang and in-house implementation

In the original paper, the goodness of refining eas evaluated by measuring the difference in structural scores (TM-Score, HB-Score,etc.) between the initial model and the final refined. The same approach was followed in these tests, moreover by also evaluating them at each step of the simulation.

Scoring functions

Four scores where taken in account during the simulations: the RMSD of common C_α carbon, GDT-HA, TM-Score and Kernel-Similarity of the Subgraph Matching Kernel applied to the H-Bond network (without the MC-MC edges, that are taken in account in the HB-Score). Each of this scores has values between [0,1].

H-Bond score Given two graphs G_{nat} and G_k where G_{nat} refers to the RIN of the native model and G_k to the k-th RIN graph obtained at every steps of the simulation, the HB-Score is calculated as it follows:

First, the H-Bond main-chain (HBOND:MC-MC in RIN) subgraph is extracted with a cutoff of $d < 3.5\,\text{Å}$, in order to select only the H-Bond belonging to the secondary structures. In this case the subgraph is an undirected vertex labeled graph and the edge can be considered unlabeled since they refers to the same interaction type.

After this step:

$$\text{HBscore} = \frac{|E(SG_{nat}) \cap E(SG_k)|}{|E(SG_{nat})|} \tag{5.1}$$

Where SG refers to the correspnding H-Bond main-chain subgraph.

Or in matrices notation where a_{ij} refers to the adjacency matrix (See Eq.(2.1.2)):

$$\text{HBScore} = \frac{\sum_{ij}(a_{ij} \cdot {a^{(k)}}_{ij})}{\sum_{ij}\ a_{ij}^{(nat)}} \tag{5.2}$$

Where $\cdot$ represent the pairwise multiplication. In other words, the edges of the $G^{nat} \cap G^k$ are counted and compared with the G^{nat} edges number.

Kernel Similarity on side-chain network For assessing the similarity of the H-Bond networks involving interaction of the type MC-SC and SC-SC, kernels between graphs were deployed, especially the Subgraph Matching Kernel. Note that this score is not present in the original publication, but seems necessary because the HB-Score does not take in account the whole network, but only the specific H-Bond that passes between two residues. For assessing the Kernel Similarity (See Section 3.4.2) of the H-Bond network, first a graph at every step of the simulation was generated by RINmaker. After this, subgraphs comprising only SC-SC and SC-MC interactions were extracted from every graph. The nodes of the subgraphs were labeled by the position number along the backbone while the edges where labeled according to the type of interaction.

Summarizing the used scores functions respect structures in PDB notation:

- TM-Score compares the folds measuring CA locally.
- GDT-HA compares the folds measuring CA globally.
- HB-Score compare edges pairwise of H-Bond MC-MC networks formed by C and N.
- Kernel Similarity compares SC-SC and MC-SC bonds networks.

Simulations without FG-MD forcefield

As a preliminary result, we first run simulations in the absence of the FG-MD constraints. This simulations were produced for obtaining a ground truth in order to subsequently test the effectiveness of the FG-MD protocol in-house version. The values obtained from the collective variables were used for comparing the relative refined structures.

1eqmA and 1ewl bacterial enzymes

The models chosen for the assessment were two small bacterial enzymes PDB 1eqmA and 1ewl, made of 158 and 215 residues respectively. These non-refined structures and the corresponding native structures were provided by Zhang's team in the standard test package for their FG-MD web-based version, and were chosen in this test because the results and scores obtained in the original publication[56] were deeply analyzed by the authors.

Score	1eqmA	1ewl
TM-Score	0.791	0.885
GDT-HA	0.573	0.686
Similarity	0.37	0.701
HBScore	0.39	0.46

Table 5.3: Scores values between initial models vs. native structures.

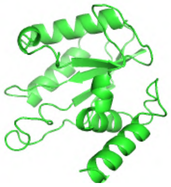


Figure 5.6: 1eqmA

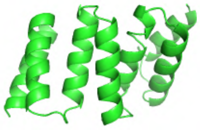
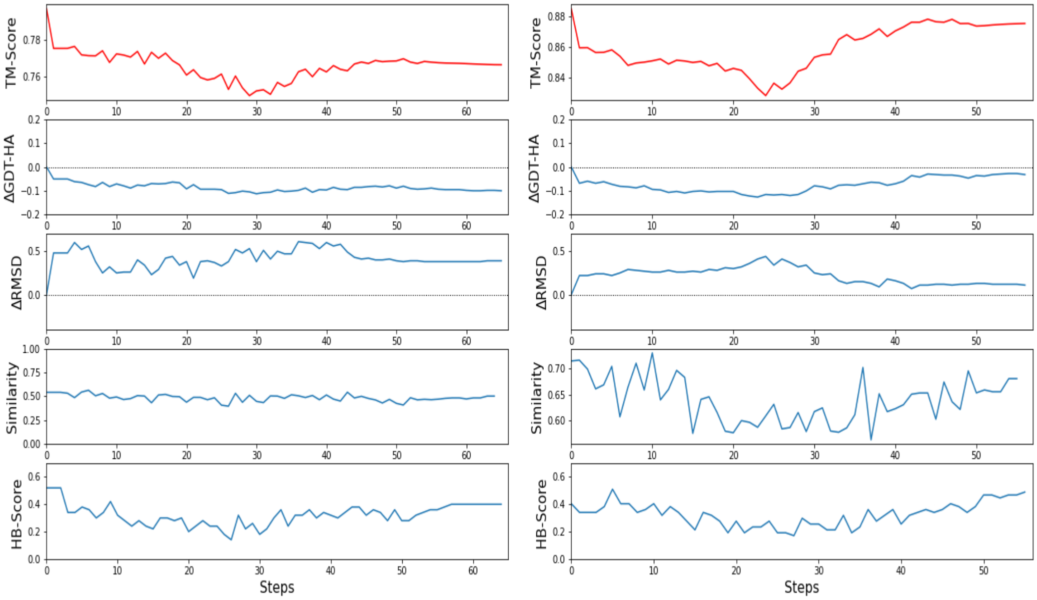


Figure 5.7: 1ewl



(a) 1eqmA at $T = 273\,\mathrm{K}$ (b) 1ewl at $T = 273\,\mathrm{K}$

Figure 5.8: Simulation in the new implemented framework only with AMBER99 (i.e. no FG-MD forcefield)

From this analysis is clear that AMBER99 forcefield alone cannot refine a near-native protein. All of the scores got worse during the anneal: The TM-Score rapidly decreased during the first steps of the simulation and got worse during the annealing. The final TM-Score (0.766 and 0.87) are smaller respect the starting one (0.791 and 0.88). In the case of 1eqmA RMSD and GDT-HA worse during all of the process while in the 1ewl returned near the initial value at the end of the process. This because the protein probably does not unfold but also does not return to its native state either. It should be stressed that since GDT-HA is a very hard score, a slight change on it remark a great change on the C_α structure, so a even small decrease (-0.1) of its value translate in a quality worsening of the model. The HB-Score dramatically decreased for 1eqmA while returned to the initial value for the 1ewl. The Kernel similarity of 1eqmA remained stable around 0.50, indicating probably a non complete unfolding, while for 1ewl increased of a slight amount (from 0.4 to 0.48).

Score (no FG-MD)	**1eqmA**	**1ewl**
ΔTM-Score	-0.25	-0.01
ΔGDT-HA	-0.1	-0.01
ΔK.-Similarity	0.0	+0.08
ΔHB-Score	-0.18	0

Table 5.4: Difference between the reported initial scores in Table 5.3 values between initial models and those obtained at the end of the simulation without FG-MD forcefield.

Simulations with FG-MD forcefield

The two selected models in the previous section are now studied with the FG-MD forcefield.

1eqmA

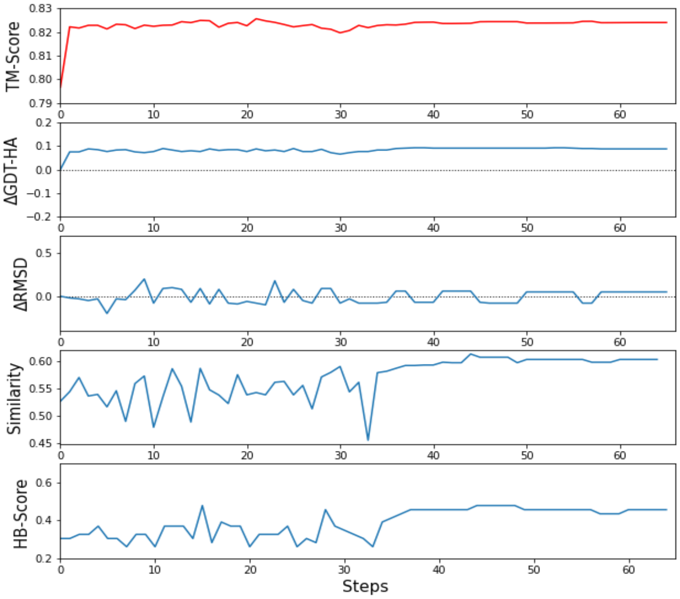
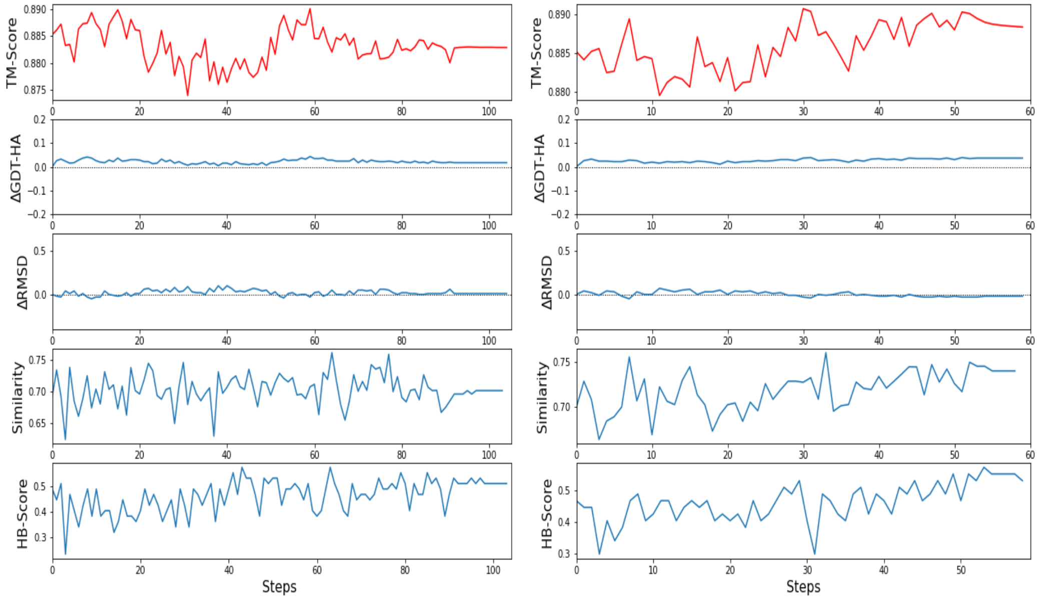


Figure 5.9: 1eqmA $T_{bath} = 330\,\mathrm{K}$

As reported in Fig. 5.9 of 1eqmA, the scores dramatically increased during the first steps of the simulations. The restraints added for C_α, the C_α clash force and the H-Bond potential drew the simulation toward the native state. This is in accordance with the observation of Zhang with respect to the landscape of near native models: the energy landscape in this case is more close to a funnel type and the added potentials make the simulation to converge very fast to the global minimum.

1ewl



(a) 1ewl $T_{bath} = 273\,\mathrm{K}$

(b) 1ewl $T_{bath} = 330\,\mathrm{K}$

Figure 5.10: Simulated annealing at different temperature for PDB 1ewl

As can be seen from Fig. 5.10 the improvement was most on the HB-Score: in the (a) case the HB-Score passed from 0.46 to 0.512 and in the case (b) from 0.46 to 0.553. The Kernel Similarity passed from 0.701 to 0.749 in the second case (b) while remained 0.701 for the case (a). The TM-Score clearly had a little improvement from the initial (0.885) to 0.888 for the second case while interestingly a little worsening for the first case (0.883). The RMSD remained substantially unchanged while GDT-HA had a good improvement in the second case (0.722 vs 0.685 of the initial model). Interestingly the highest TM-score where not obtained in the cooling stage but earlier. The GDT-HA and HB-score reported are in accordance with the ones reported by Zhang except for the TM-Score, which they report an higher one (0.891 vs 0.888). This analysis also suggests a dependence of 1ewl global minimum from the temperature.

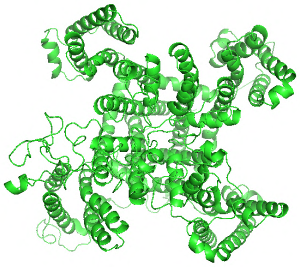
	In-house		Zhang	
Score (with FG-MD)	**1eqmA**	**1ewl**	**1eqmA**	**1ewl**
ΔTM-Score	+0.04	-0.003	+0.04	-0.01
ΔGDT-HA	+0.05	+0.04	+0.03	+0.04
ΔHB-Score	+0.05	+0.12	+0.04	+0.04
ΔKernel-Similarity	+0.3	+0.05	-	-

Table 5.5: Difference between the reported initial scores in Table 5.3 values between initial models and those obtained at the end of the simulation with FG-MD forcefield in the original publication (Zhang) and in the in-house version..

NaV1.7 from MOESM3 template

In this simulation the FGMD forcefield was tested against the core part of the MOESM3-NaV wildtype obtained from the SWISS-MODEL pipeline. In this case the core was directly taken from the WildType after removing the coils.

Score	value
TM-Score	0.998
GDT-HA	0.263
Similarity	0.815
HBScore	1



For these simulations few remarks are in order:

- Since this is a membrane protein, it lives on three different ambient: the coil live in the intracellular phase, the hydrophobic core in the lipid membrane and the coil upward the core lives in the extracellular phase. This setup is not simple to achieve, also because the lipid membrane simulation are very computational demanding, and these coils seems have a marginal function respect the core. So in this setup the coil pointing inward the intracellular phase were removed while these pointing outward where left since they are short.

- Here the native states is intended as the output from the FGMD Server, so the TM-Score are calculated against not a known native state but against the refined protein.

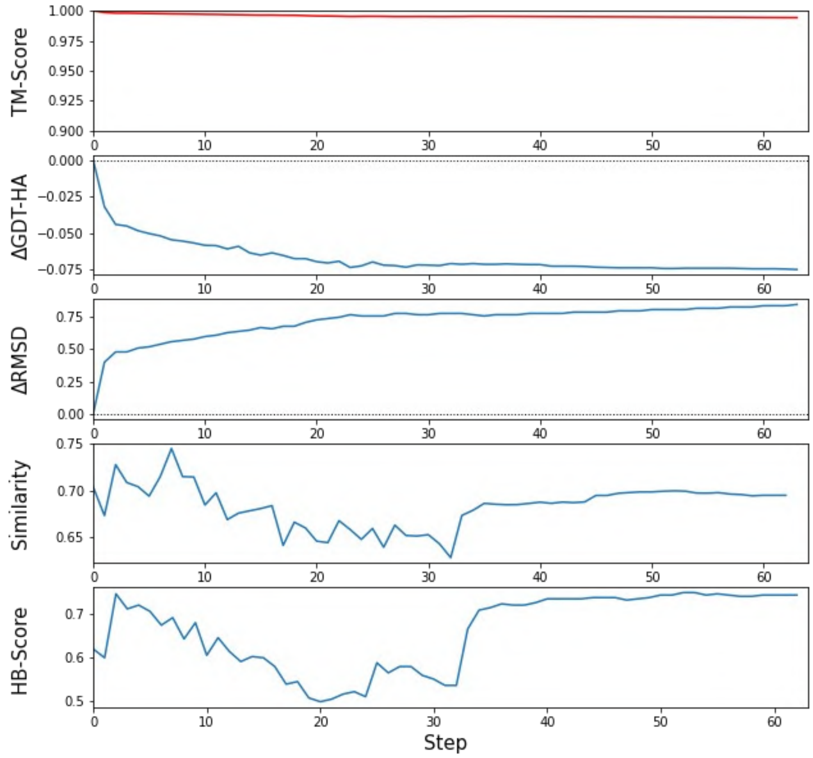


Figure 5.11: MOESM3 Wild-Type without intracellular-coil

As seen by the scoring functions in Fig. 5.11, the TM-Score of the core part is very high (0.998), meaning a quasi-perfect match between the two structures, so here the FG-MD terms do only a little job on refining and act most as constraint for the structure instead as a driving force. This arise from the fact that homology modeling structures are obtained from experimental data, so in the big database comprising all of the known resolved proteins obviously the fragment-searching routine select the one with the highest TM-Score, that in this case is the 6A90. The HB-score is 1, indicating a perfect H-Bond main-chain network match between the two structures. The GDT-HA is very low, probably because of the upward coil. The HB-Score had a drastic worsening during the simulation, so this justify the long time response of the server-based version, since it gives an answer on 20 try, selecting the annealing with all of the scores (TM-Score, GDT-HA, HBscore) improved. Here the Kernel Similarity shown some difference. As an hypothesis, the fact that Kernel Similarity don't improve but worsen indicates two different conformations of the sidechains network between the assessed model and the native structure. Since the native Wild-Type used here is the FG-MD Wild-Type to which, before the score assisted comparation, the intracellular coils were removed, it is highly probable that this score indicates an influence of the intracellular-coil on the sidechain network of the native model, while in the assessed model this effect is not achieved since it is refined without. As will be shown in Section 6.4, this fact will become prominent on cluster analysis of point-mutated models.

A complete comparation of TM-Score between FG-MD in-house and FG-MD web version of the refined model is depicted in Table C.1 for 6a90 based models and Table C.2 for MOESM3 based models.

Chapter 6

Application of the new Pipeline to the NaV1.7 sodium channel

6.1 Expected results from the Pipeline

As reported in Fig. 6.1, the results of the algorithm can be monitored in human-readable way, by visually assessing the presence of clusters on the Gram-matrix, valuating the Support Vector Machine accuracy and comparing the output of the Mutations classification. In the golden case, Gram-matrix, SVM accuracy and classification have to appear as follows:

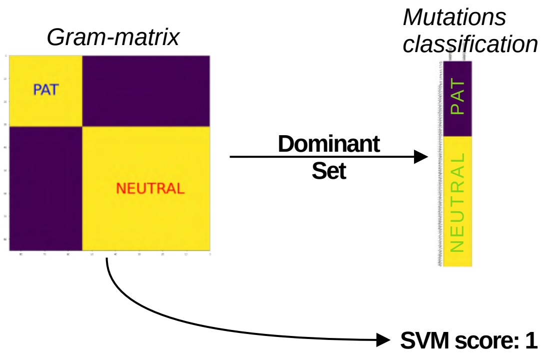


Figure 6.1: Results from pipeline in the golden case; in yellow 0-30 pathogenic mutations (PAT), in darkblue 31-85 non-pathogenic mutations (NEUTRAL)

6.1.1 RINmaker parameters

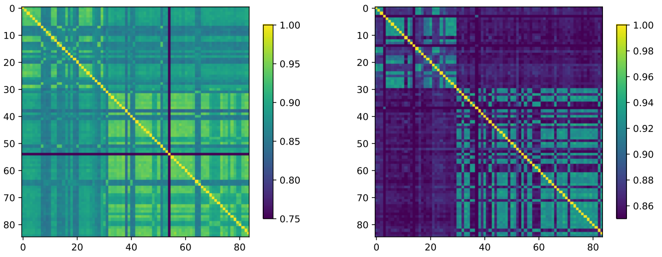
The parameters chosen for the RINmaker were *Default* (See Table 5.1) and *interaction-type=multiple*, *h-bond-realistic* (See Section 2.1.2). from now on these parameters will be the ones used in every analysis from this chapter onward inclusive.

6.2 Results after Zhang FG-MD refinement

In this section, a comparison of the results between the new and old implementation is presented, consequently results from the pipeline in full-production are discussed. The purpose of this first analysis is to use the results previously obtained from the old Pipeline[51] with Zhang FG-MD refined structures as test-case for the new Pipeline.

6.2.1 Pipeline test from Zhang FG-MD refined structures

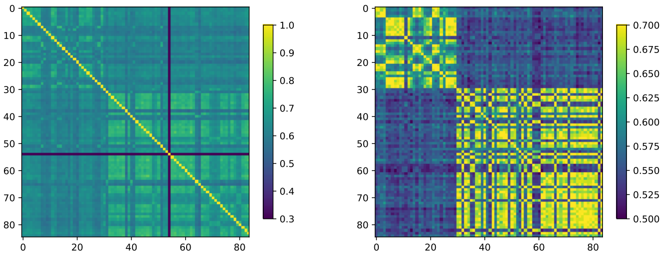
Here results from the new Pipeline are compared with results of the previous implementation (See Section 5.1), where the inputs are refined structures obtained for the previous work [51]. Specifically, the structures refers to the FG-MD web refined 85 point-mutated structures in .PDB obtained from homology-modelling on MOESM3 and 6a90 templates. These structures were injected in the step 2 of the aforementioned new Pipeline. Gram-matrices, SVM accuracies and classification are compared with those obtained from the old Pipeline[51].



(a) MOESM3

(b) 6a90

Figure 6.2: Gram-matrices results from the new Pipeline of Vertex-Histogram kernel applied on networks from 6a90 and MOESM3 based models obtained in the previous work.

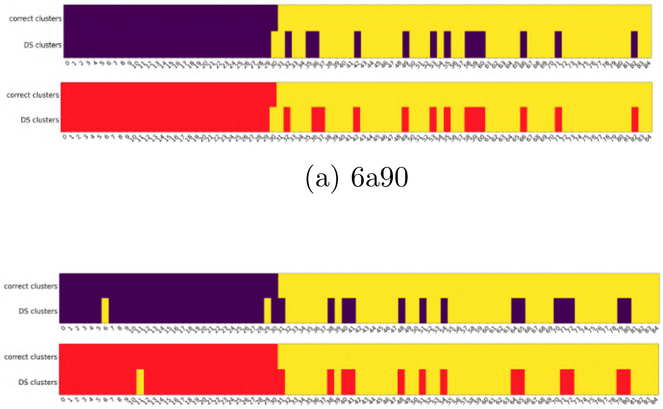


(a) MOESM3

(b) 6a90

Figure 6.3: Gram-matrices results from the new Pipeline of Weisfeiler-Lehman kernel applied on networks 6a90 and MOESM3 based models obtained in the previous work.

As reported in the previous work [51] there is a clear differentiation in both Vertex-Histogram and Weisfeiler-Lehman Gram-matrices, where clusters appear in two different areas grouping mutations according to the pathogenicity. From the results shown in Fig. 6.2,6.3 it is possible to identify both the same reported clusters, grouping the 31 PAT mutations in the upper-left corner and the 54 NEUTRAL in the lower-right corner. The dissimilarity of the mutation n°54 (L1267V) mutation turned out to be caused by an incorrect residue numbering within the starting PDB, which prompted the insertion of a routine for fixing PDBs before entering in the step 2 of the Pipeline (See Section 5.1).



(a) 6a90

(b) MOESM3

Figure 6.4: Comparison of Mutations classification between previous obtained (violet) [51] and the one obtained with the new pipeline (red) on Weisfeiler-Lehman kernel.

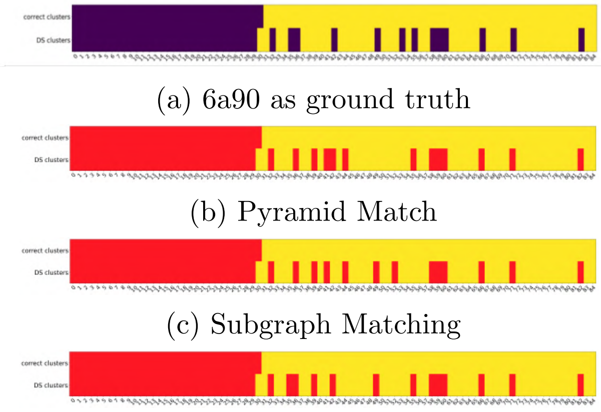
The classification shown in the Fig. 6.4 for the WL-Kernel accentuates a perfect match between the results, with a slight improvement for the MOESM3 in 6.4(b), where a previously erroneous classification of the mutation 29 became correctly predicted by the new pipeline. It should be noted from the value scale in Fig. 6.2, 6.3 that in the Gram matrices the difference between the clustering and non-clustering zone is on the order of 10^{-2}, which indicates, respect the clustering zone, a not too pronounced differentiation. This fact will be analyzed in Section 6.4. As can be seen from Table 6.1, SVM accuracy of the kernels applied to interaction subgraphs are close to 1 and that indicates a good linear separation of cluster, in accordance with what reported in the previous work [51].

Interaction	**6a90**	**MOESM3**
ALL	0.953	0.923
IONIC	0.953	0.989
PICATION	0.883	0.776
PIPISTACK	0.789	0.647
HBOND	0.952	0.929

Table 6.1: Summary of the obtained SVM accuracy on Gram-matrices from subgraphs correspnding to the listed interactions.

6.2.2 More results from the structures

Because the new pipeline allows multiple kernels to be applied in the same run, the same dataset used in Subsection 6.2.1 was analyzed with the auxilium of Subgraph Matching Kernel (See Definition 2.1.5), Pyramid Match Kernel[19] and Edge Histogram Kernel (See Definition 2.3c). In these results, every edges belonging to the input graphs were labeled as (Node1):(Node2)_(Interaction type):(MC-MC or MC-SC or SC-SC) where Source and Target refers to the endpoints of the edge and the remaining to the type of interaction involved.



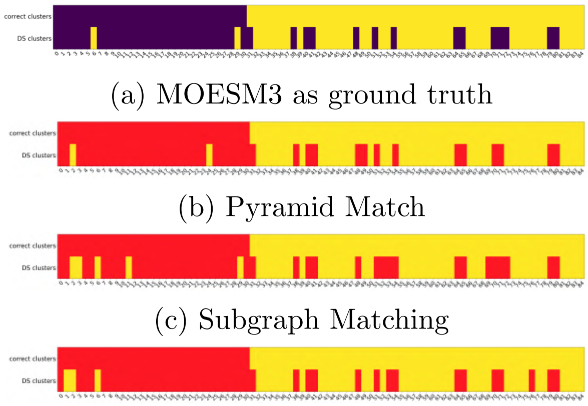
(a) 6a90 as ground truth

(b) Pyramid Match

(c) Subgraph Matching

(d) Edge Histogram

Figure 6.5



(a) MOESM3 as ground truth

(b) Pyramid Match

(c) Subgraph Matching

(d) Edge Histogram

Figure 6.6

Comparison of Mutations classification between previous obtained for 6a90 and MOESM3 (violet)[51] and those obtained with the new pipeline (red) on different kernels.

Kernel	6a90	MOESM3
Pyramid Match	0	0
Subgraph Matching	-1	-6
Edge Histogram	0	-2

Table 6.2: Net difference in misclassified(-) / improved(+) mutations respect the ground truth according to the kernel used.

As can be seen from Table 6.2, the use of finer kernels and a more expressive edge labeling does not seem to improve mutations classification; in the case of the 6a90, the kernel choice does not seem to affect classification, unlike the MOESM3 where, on the other hand, Subgraph Matching failed 6 classification. Among all the assessed Kernels, both Weisfeiler-Lehman and Vertex/Edge-Histogram outperforms all others tested, even in terms of computational cost. It can also be noted from Fig. 6.5,6.6 that some classifications are "weak," and the Dominant Set does not always classify them in the same way using different kernels.

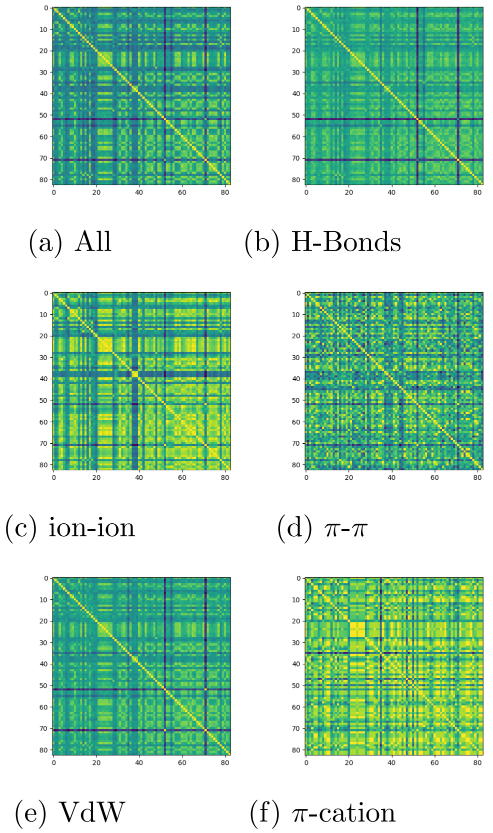
6.3 Results after in-house FG-MD refinement

In this sections the full in-house pipeline is fully applied to models derived from the homology-modelling step. At this stage, results and some critical issues that emerged will be explained.

6.3.1 Pipeline input and parameters

At each run of the pipeline the input structures matched the 85 mutated structures obtained by SWISS-MODEL on one of the two MOESM3 or 6a90 templates. The FG-MD simulations were carried in TIP3P solvent or with Reaction Field approximation when specified with the modified-AMBER99 forcefield.

6.3.2 Results from full-models in TIP3P solvent

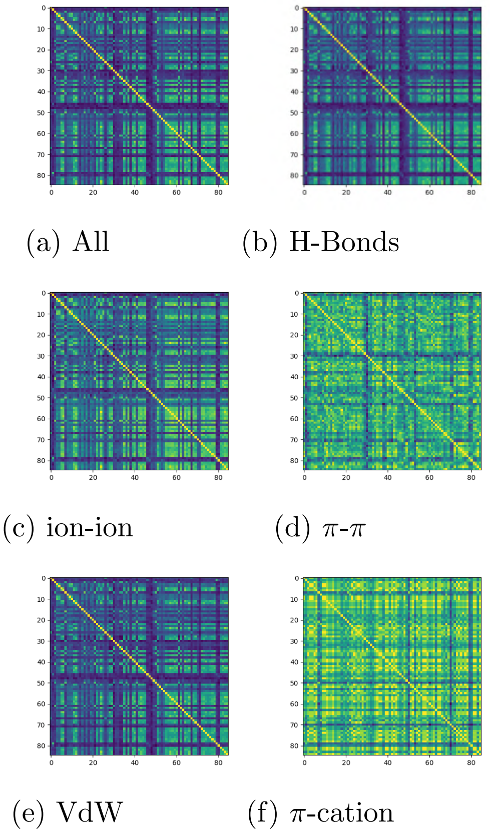


(a) All (b) H-Bonds

(c) ion-ion (d) π-π

(e) VdW (f) π-cation

Figure 6.7: 6a90



(a) All (b) H-Bonds

(c) ion-ion (d) π-π

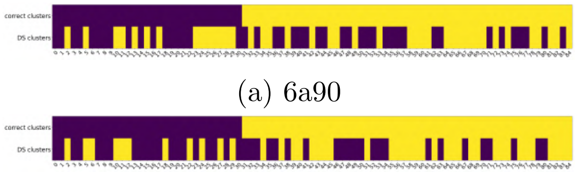
(e) VdW (f) π-cation

Figure 6.8: MOESM3

Gram-matrices of subgraphs respect all interactions (a), H-Bonds (b), ion-ion (c), π-π (d), VdW (e) and π-cation (f) in MOESM3 and 6a90 derived full-models

Interaction	6a90	MOESM3
ALL	0.625	0.635
IONIC	0.625	0.635
PICATION	0.532	0.426
PIPISTACK	0.789	0.561
HBOND	0.625	0.635

Table 6.3



(a) 6a90

(b) MOESM3

Figure 6.9

Summary of the obtained SVM scores on Gram-matrices from subgraphs correspnding to the listed interactions (Table 6.1); Mutation classification-Dominant Set (Fig. 6.9)

As can be seen from the Fig. 6.7,6.8, Pipeline did not produce cluster differentiation. The SVM Scores reported in Table 6.3 are considerably lower than those reported in Table 6.1 when the input were structures previously FG-MD refined; interestingly ALL, IONIC, and HBOND SVM accuracy are in perfect agreement, indicating non-differentiation of the clusters from the interaction subgraphs. From the Dominant Set output reported in Fig. 6.9, is clear an incorrect classification of the mutations with an error rate of 50% for 6a90 and 41% for MOESM3. Although this problem may at first glance deflect its true nature, as explained in the Section 6.4 it is traced to a particular characteristic of the analyzed structures .

6.3.3 Results from the hydrophobic-core part of the models

In an attempt to improve the results obtained in Section 6.3.1, the Pipeline was applied only to the hydrophobic core of each structure. To this, it was assumed that probably not the entire structure of the proteins, and thus the resulting Residue Interaction Network, is physically relevant for the purpose of detecting the pathogenicity of a point-mutation.

Method

The geometry of a structure considered in this work can be split into 3 parts: the core, which is the most conserved region, the intracellular and extracellular coils. According to the subdivision shown in Table 6.3.3, from each of the 85 models was extracted the core extended with the extracellular coil. Each obtained submodel was fed into the Pipeline. The resulting Gram-matrices are shown in Fig. 6.11,6.12 and SVM results are shown in Table 6.6.

MOESM3	**N° residues**
Extracellular	50
Core	782
Intracellular	545
6a90	**N° residues**
Extracellular	72
Core	782
Intracellular	505

Table 6.4: Subdivisions

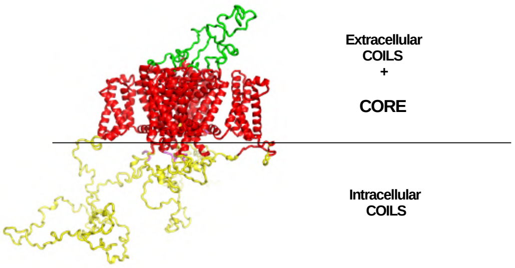
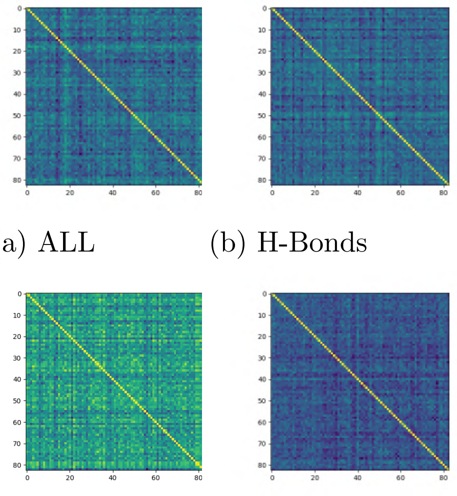


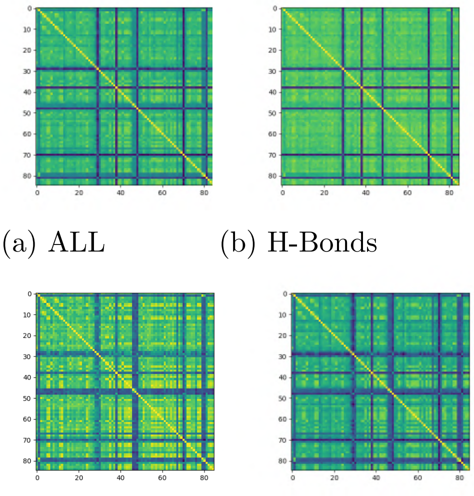
Figure 6.10: Subdivision shown on a structure



(a) ALL (b) H-Bonds

(c) ion-ion (d) VdW

Figure 6.11: 6a90



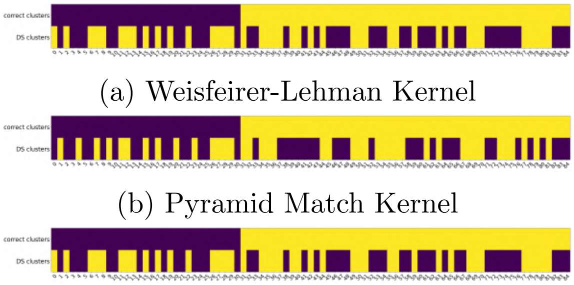
(a) ALL (b) H-Bonds

(c) ion-ion (d) VdW

Figure 6.12: MOESM3

Selection of Gram-matrices of subgraphs respect all interactions (a), H-Bonds (b), ion-ion (c), VdW (d) in refined 6a90 (Fig. 6.11) and MOESM3 (Fig. 6.12) based models after refining them with the in-house FG-MD implementation with TIP3P as solvent.

6a90 and MOESM3 with TIP3P solvent

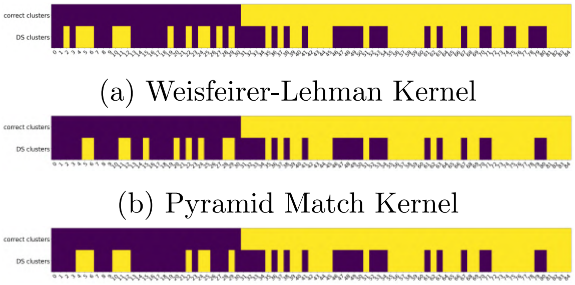


(a) Weisfeirer-Lehman Kernel

(b) Pyramid Match Kernel

(c) Edge-Histogram Kernel

Figure 6.13: 6a90 based models



(a) Weisfeirer-Lehman Kernel

(b) Pyramid Match Kernel

(c) Edge-Histogram Kernel

Figure 6.14: MOESM3 based models

Mutations classification from the considered models

Interaction	6a90	MOESM3
ALL	0.625	0.635
IONIC	0.625	0.635
PICATION	0.532	0.594
PIPISTACK	0.789	0.610
HBOND	0.625	0.635

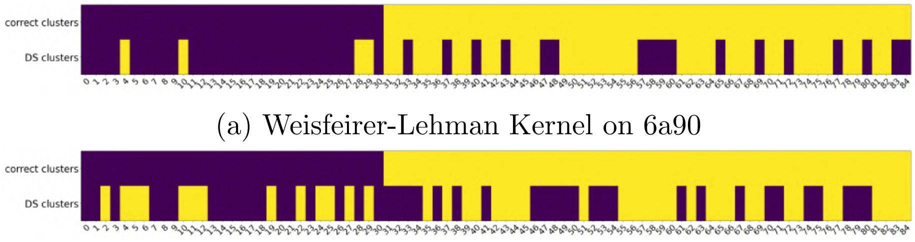
Table 6.5: SVM Scores for Weisfeirer-Lehman Kernel

Kernel	6a90	MOESM3
Weisfeirer-Lehman	41%	44%
Pyramid Match	48%	37%
Edge Histogram	52%	27%

Table 6.6: Percentage of failed classifications

As observed for the full-model run in Section 6.3.1, the classification of mutations fails for more than one third of the total number of models. It should be noted, however, that the percentage in Table 6.6 is relatively lower for the models derived from MOESM3, especially via the Edge-Histogram kernel; in these analyzed cases, a solvent effect cannot be ruled out. In any case, the worse performance of the full-models in Section 6.3.1 than those considered in this analysis can be attributed to an effect due to the presence of intracellular-coils, which will be hypothesized in Section 6.4.

6a90 and MOESM3 based models in vacuo (Reaction Field approximation)



(a) Weisfeirer-Lehman Kernel on 6a90

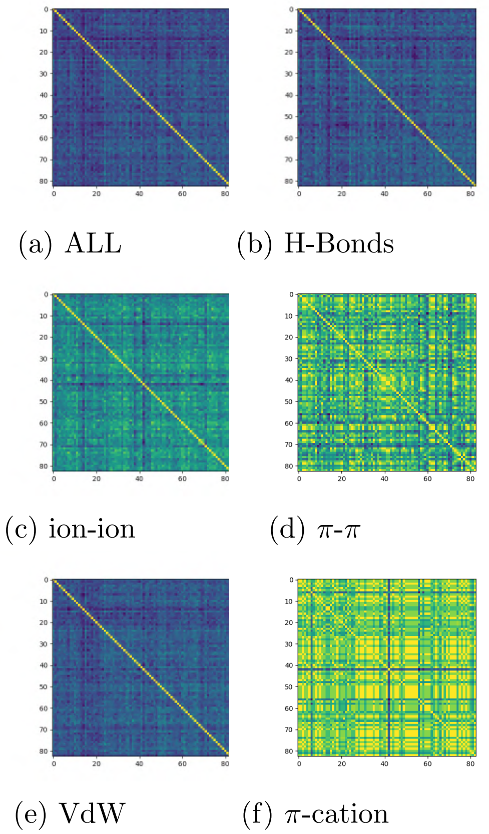
(b) Weisfeirer-Lehman Kernel on MOESM3

Figure 6.17: Classifications of mutations on 6a90(a) and MOESM3(b) based models in vacuo

Interaction	6a90	MOESM3
ALL	0.625	0.635
IONIC	0.625	0.635
PICATION	0.532	0.594
PIPISTACK	0.789	0.610
HBOND	0.625	0.635

Table 6.7: SVM Accuracy

Contrary to expectation, in the runs in which the solvent was replaced with the Reaction Field approximation (See Section 3.3.5) a weak but clear appearance of clusters can be seen from Fig. 6.15 relative to the 6a90 models: in this case there is an improvement in SVM accuracy and failed predictions drop to 21%, which is considerably lower than previous cases. With regard to the models derived from MOESM3 the results obtained are, on the other hand, considerably lower than those obtained in explicit solvent.

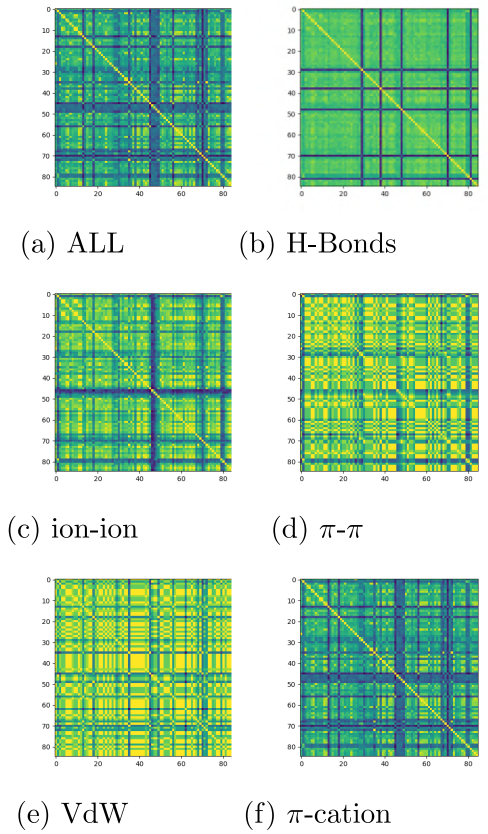


(a) ALL (b) H-Bonds

(c) ion-ion (d) π-π

(e) VdW (f) π-cation

Figure 6.15: 6a90



(a) ALL (b) H-Bonds

(c) ion-ion (d) π-π

(e) VdW (f) π-cation

Figure 6.16: MOESM3

Gram-matrices from Vertex-Histogram kernel of subgraphs respect all interactions (a), H-Bonds (b), ion-ion (c), π-π (d), VdW (e) and π-cation (f) in refined 6a90 (Fig. 6.11) and MOESM3 (Fig. 6.12) based models after refining them with the in-house FG-MD implementation with Reaction Field approximation.

6.4 Insight from the discrepancy on the results

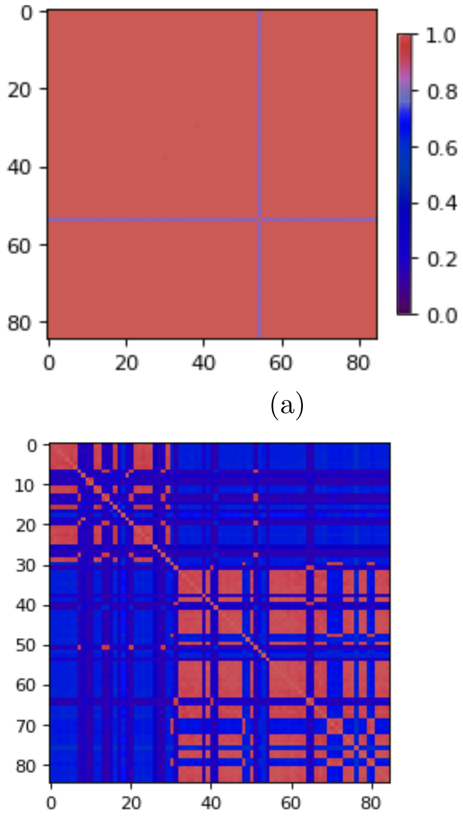
To investigate the reasons for the discrepancy in the results reported from the in-house Pipeline in Section 6.3.3, the models used for the Pipeline test are analyzed in detail in this section, moreover a partial solution is proposed.

6.4.1 Method

At the beginning the same subdivision used in Section 6.3.3 was performed to the models used for the Pipeline test, according to Table 6.3.3. Each of the two obtained set of submodel was cross compared by TM-Align (See Section 3.5.5) for assessing the folding score, and the results were organized in a matrix where every entry it's the TM-Score between two submodel; after this was done, each set is analyzed with the Pipeline from the RIN generation stage.

6.4.2 Observations and discussion

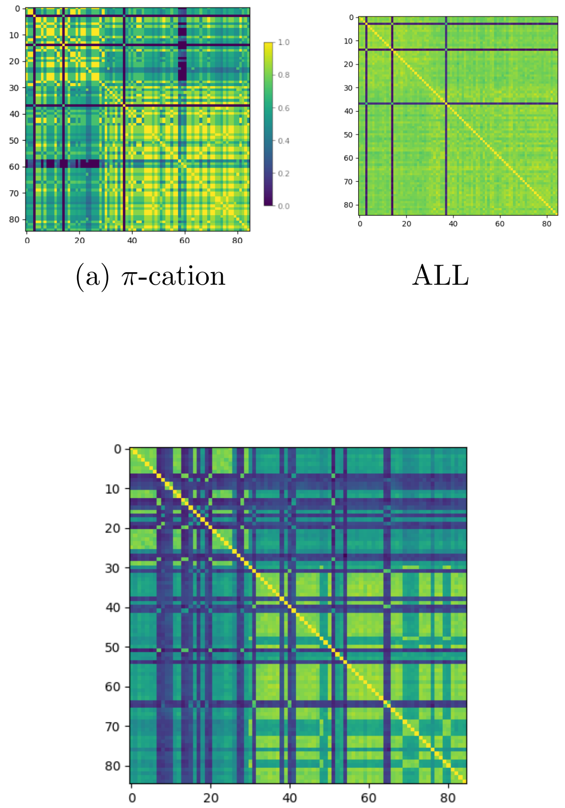
As a start, the procedure described above is applied to each Zhang FG-MD web structure used for new Pipeline test in Section 6.2.



(a)

(b)

Figure 6.18: TM-Score matrices



(a) π-cation

ALL

(b)

Figure 6.19: Gram-matrices

TM-Score (Fig. 6.18) and Gram-matrices from Vertex-Histogram kernel from different subgraphs (Fig. 6.19) of core (a) and coils (b) substructures of 6a90 based models after Zhang FG-MD refinement.

Recalling that the TM-Score is a measure of similarity between the 3D backbones of two proteins while the kernel is a measure of similarity between networks, the analysis can be based on the following observations:

1. TM-Score matrix of core From Fig. 6.18(a) does not show any

remarkable structural differentiation.

2. Generally, the difference between clustered and unclustered zones in Gram-matrices, as observed in Subsection 6.2.1, is in the order of 10^{-2}.

3. From a visual analysis,the TM-Score matrix of coils in Fig. 6.18(b) closely resemble the coils Gram-matrix in Fig. 6.19(b) with two clear clusters.

4. The Gram-matrix in Fig. 6.19(a) of the π-cation subgraph highlight a well-defined clustering with a difference between clustered and non-clustered zone of about 0.4.

Point 1 can be explained as a consequence of the concerted action of SWISS-MODEL[54] and FG-MD algorithm (See Fig. 3.5): since the amino acids sequence of a point-mutation differs for only a residue from the Wild-Type sequence, and given that the hydrophobic core is the most conserved part, models obtained from SWISS-MODEL will share the same equal geometry for the hydrophobic core. Bearing this fact in mind, when the obtained models will be subjected to the FG-MD stage, the querying routine will sample same contact-maps for every hydrophobic-core, which will be applied during the FG-MD refining as identical C_α restraints for every model. Moreover, since the hydrophobic core is constructed from experimentally resolved structure, the contact-maps will be sampled exactly from the same experimentally resolved found in the reference database (notably the 3RVY which is the building block of MOESM3 and 6a90), since there is obviously the highest degree of folding between them. Because of this, the restraints won't add new information to the backbone during the refining and they will constrain the ϕ and ψ angles (See Section B.3) of the hydrophobic core toward same values, leaving the remaining degrees of freedom to refer only to possible rotamers conformations respect the X_i angles of the sidechains ensemble. In other words, C_α backbone geometry of the hydrophobic core will be the same at both the beginning and end of refining (See Fig. 6.18(a)), with the exception of sidechains rotamers. Having said this, it can be stated that the clusters obtained in the Gram-matrices will refer only to final rearrangements of the sidechains networks which, at the beginning of the refining, will initially be very similar to each other since every sampled contact-maps won't rearrange the backbone of the hydrophobic-core. A proof of what has just been said can be reported from the distribution of torsional angles difference (See Appendix B.1) in Fig. 6.20 of the core part, where is possible to see a narrow distribution centered in 0° (and 360°) for the ϕ and θ angles while little and wider peaks centered around 110° and 250° for the sidechain angles distributions; it's highly probable that these weak re-arrangement of the sidechains indicates the formation of interactions, thus justifying the observation in point 2 and validating kernels as a method sensitive and able to highlight small variations in a reduced size dataset.

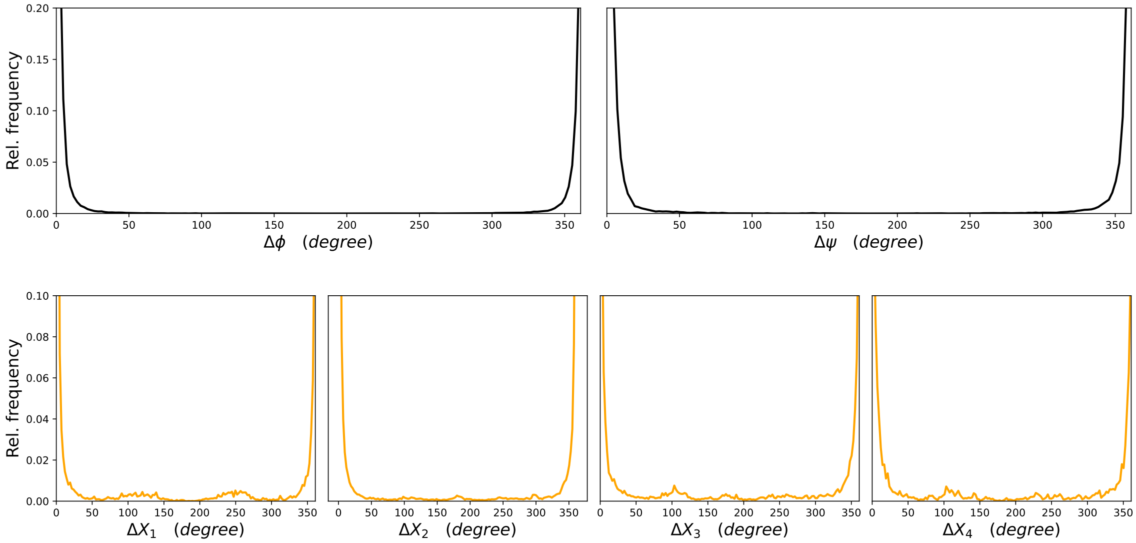


Figure 6.20: Core

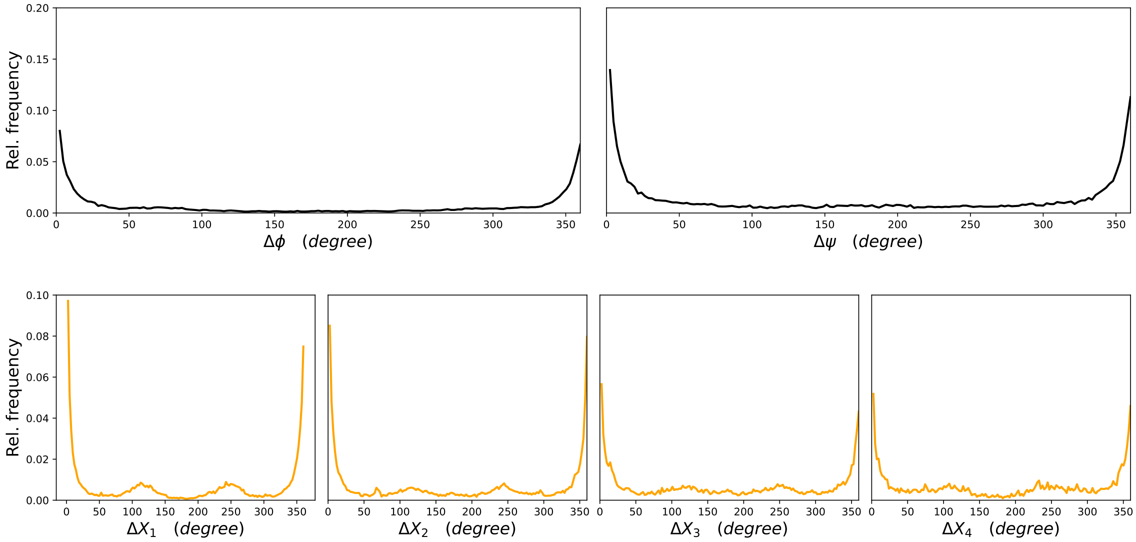


Figure 6.21: Intracellular coil

Distribution of torsional angles difference (See Appendix B.2 for method) between the Wild-Type 6a90 and the investigated substructures.

Concerning point 3, to explain the clustering in both TM-Score matrix in Fig. 6.18(b) and Gram-matrix in Fig. 6.19(b), one must consider how SWISS-MODEL generates the coils in the less conserved parts of the protein (See Section 4.2.1): it relies on a pre-computed rotamers library, which is used to predict backbone and sidechain conformations during the score-assisted backbone building. Probably one of the employed scores used is sensitive to the hydrophobicity of the mutated residue and since the pathogenicity of a mutation is strongly correlated to the hydrophobicity, it could be that this latter influence the backbone construction algorithm toward a set of preferred conformations (See Fig. A.1) according to the pathogenicity, instead of the expected random behaviour. When these models are obtained in this way and they enter in the FG-MD querying routine, the coils are matched with some available fragments in an unpredictable way, according to the Secondary Structure Elements which the algorithm finds on this part of the backbone. This behaviour indicates a grey area in this protocol: since coils have random structures they cannot have fragments which fold is similar and so a matched fragment is more a random fact than a physical one, moreover, this behavior is not reproducible in the in-house version due to the lack of detailed information regarding the original implementation. To close the circle, observation 4 summarises all the other points discussed by providing a direct evidence and pointing a way forward to a future improvement which will be discussed in the next section.

6.4.3 Extracting important interactions

Since $\pi-$cation interactions are generally rare and the subgraph in Fig. 6.19(a) is well clustered, the test Dataset was analyzed by means of an histogram showing the relative frequency of appearance of the edges related to this type of interaction. The purpose of this analysis was to understand if the clusters, and in which grade, are consequences of the presence or absence of specific edges (i.e. if they are due to a structural rearrangement of the sidechains net). First, all of the edges of every subgraphs were collected in a list; then, the frequency of appearance of every listed edge was obtained and the data organized in two different histograms according to the pathogenic label for each considered mutation (See Section B.2 for a more formal explanation). The most interesting of these histogram is the one referring to the frequency of the edges appearing commonly in both pathogenic and non-pathogenic graphs in fig 6.22:

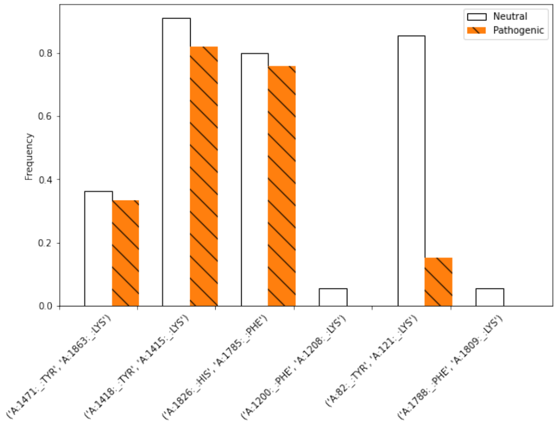
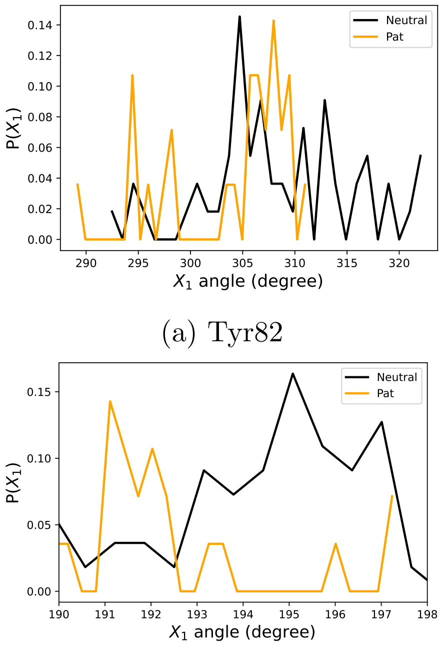


Figure 6.22: Relative Frequency of π-cation edges appearing in both pathogenic and non-pathogenic $\pi-$cations subgraphs



(a) Tyr82

(b) Lys121

Figure 6.23: X_1 angle distributions of residues Tyr82 (a) and Lys121 (b)

From Fig.6.22 it is clear that the edge between Tyr62 and Lys121 appears with different frequencies among pathogenic and non-pathogenic graphs, indicating a stable bond in the case of non-pathogenic while a more instable one for pathogenic. Moreover, the histograms in Fig. 6.23 show how Lys121, which is a charged residue, reaches two different conformations according to the pathogenicity behavior of the mutations involved, probably rearranging the charged group toward the ring of the Tyr82, as shown in the following figure.

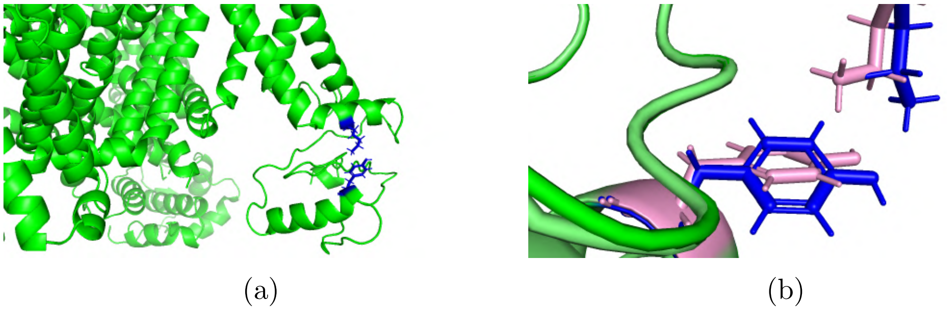
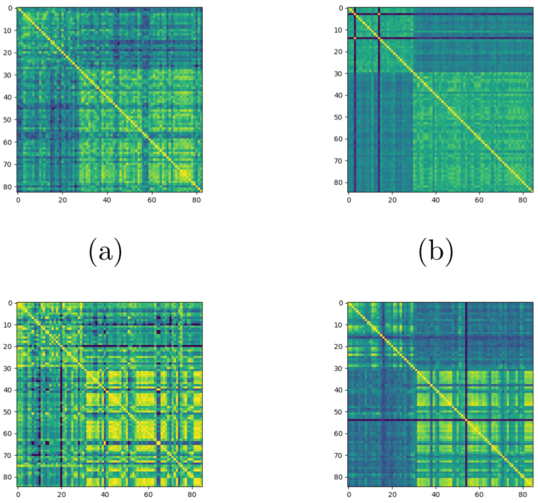


Figure 6.24: Detail of the $\pi-$cation bond 62:121 (a) for M1532I pathogenic in blue and A766T non-pathogenic in pink (b)

From Fig.6.24(b) it can be noticed that the two different rotamers have different conformations in the pathogenic and non-pathogenic cases, in fact the A766T related (pink) has the angle characteristic of a $\pi-$cation bond that M1532I (blue) has not. Notice moreover in Fig.6.24(a) notice that the two residues reside in a peripheral position of these with respect to the core. Recalling the discussion in Section 6.4.2, this bond is clearly influenced by the specific coil conformation sampled from FG-MD, that it is influencing the sidechain network according to the pathogenicity. This fact also enforce the observation on the similarity of Gram-matrix and TM-Score matrix regarding the intracellular coil struture. Taking advantage of this fact, the sensibility of the method can be enhanced by applying the aforementioned procedure as a filter stage before the kernel computation (See Method B.2). This stage improve Gram-matrices clustering beacuse of two main reason:

- It allows to remove noise that comes from fictitious bonds seen by RIN after Molecular Dynamics
- It allows to compare feature vectors projected on a sort of "principal components" basis, where the components are selected according to the common edges ensemble whose difference in frequency between pathogenic and non-pathogenic is above a fixed treshold.

How can be seen from Fig. 6.25, applying this further step to the weak clustered matrix from the hydrophobic core (See Fig. 6.19(a)) allows for highlighting clusters that are also presents in the ion-ion and VdW subgraphs both from MOESM and 6a90 based models, with an improved difference between clustered and non-clustered zone:



(a)

(b)

(c)

(d)

Figure 6.25: Gram-matrix of subgraphs respect ion-ion (a), VdW (b) interactions in 6a90 based models and ion-ion (c) and VdW (d) interactions in the MOESM3 based models

Here the extracted interactions in 6.25 are highlighted into the respective structures:

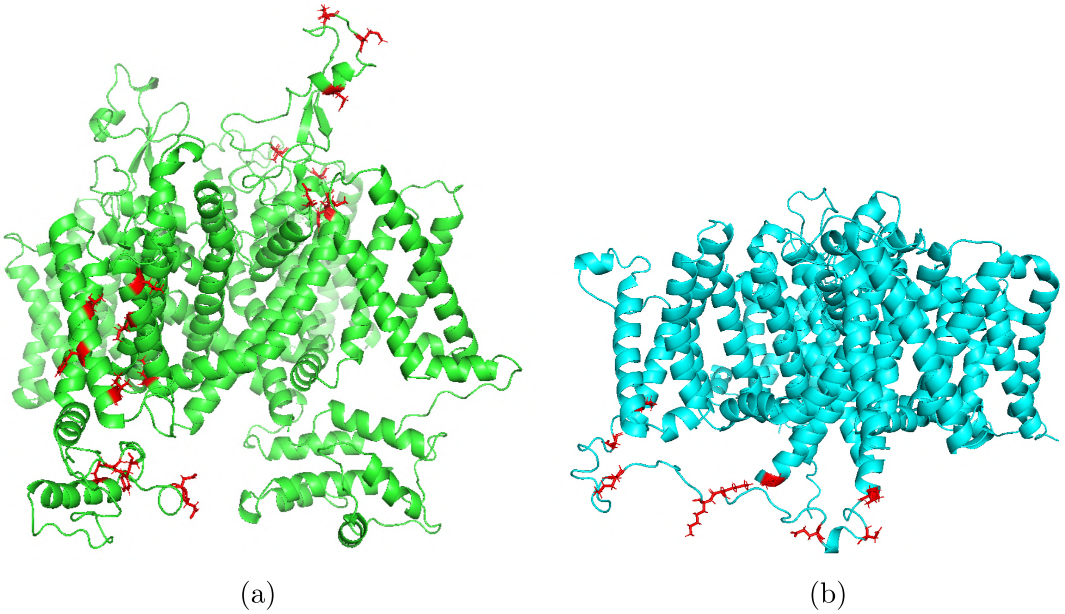


Figure 6.26: Ion-ion principal interactions in 6a90 (a) and MOESM3 (b)

From Fig. 6.26 it is possible to see that in 6a90 (a) these important bonds are distributed between residues in the core and in the "voltage sensor" where most of the point-mutations analyzed in this work are placed, while in the MOESM (b) are mostly in the activation gate. As a little confirmation, it is clear that MOESM clusters are given by interactions of the coils with the activation-gate, since they are spatially very close, while in the 6a90 probably the mechanism is more complex.

6.5 Results from the application of the filter stage

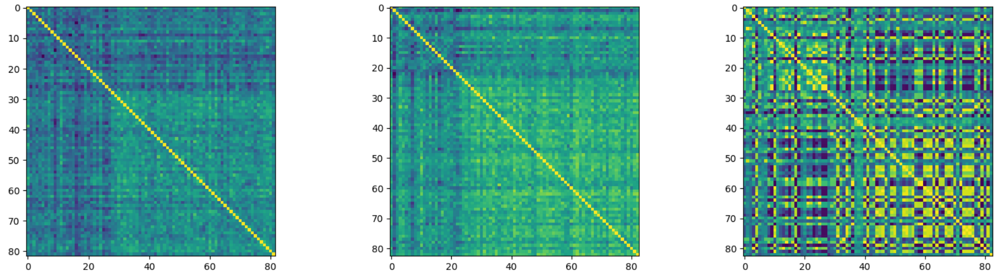
In this Section, the selection procedure described in B.2 is now applied to the Pipeline in when inputs are homology-modelling structures. This results in an improvement of the classification and prediction analyses.

Method

The filter described in B.2 was coded and added to the Pipeline between the Graph comparison and Classification stage. The T_{dif} threshold was manually optimized during the run. The intracellular-coils belonging to the full structures were cleaved out before the analysis because the unpredictable way in which FG-MD protocol sample them, as explained in Section 6.4.

6.5.1 6a90 based models

Recalling the analysis in Fig.6.11 where there was any clear cluster separating pathogenic and non-pathogenic mutations, the same models subjected to the Pipeline with the filtering stage produced an improved clustering, as can be seen from the following selected matrices in fig 6.27:

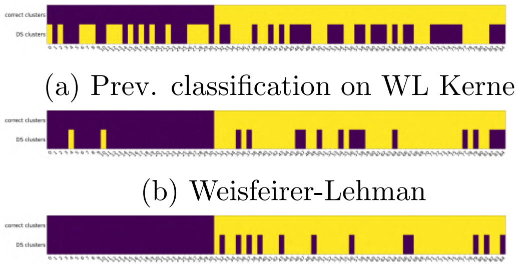


(a) Core-TIP3P

(b) Core-Vacuo

(c) Full models

Figure 6.27: Selected Gram-matrix of filtered subgraphs respect VdW interactions on 6a90 models

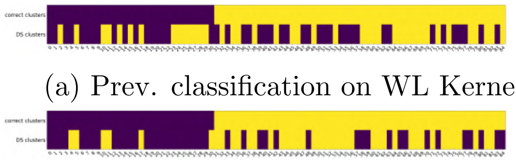


(a) Prev. classification on WL Kernel

(b) Weisfeirer-Lehman

(c) Edge-Histogram

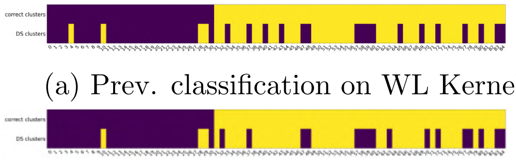
Figure 6.28: 6a90core-TIP3P



(a) Prev. classification on WL Kernel

(b) Weisferher-Lehman

Figure 6.29: 6a90full-models - TIP3P



(a) Prev. classification on WL Kernel

(b) Edge-Histogram

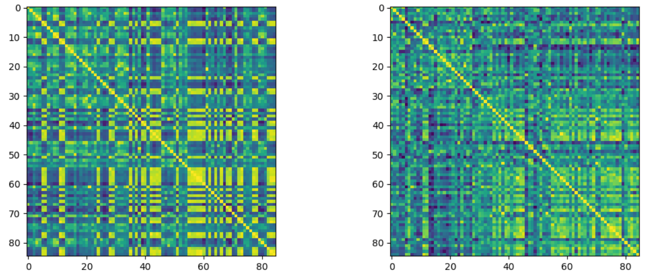
Figure 6.30: 6a90core-vacuo

Dominant-set clustering applied to VdW Gram-matrix from Weisfeirer-Lehman (a) and Edge-Histogram (b) Kernels from 6a90 hydophobic core.

Apart for π-cation, π-π and ion-ion interactions, cluster appearance was more pronounced in the H-Bonds and VdW matrix, the latter with a better accuracy. Starting from the classification respect the solvated core in Fig. 6.29, is possible to note a drastic improvement with a precentage of failed classifications both equal to 2% for the Weisfherer-Lehman and Edge-Histogram kernels, which represents the lowest errors obtained in this work, with a gap of 30% respect previously obtained (See Section 6.3.3). Notably, Dominant Set Algorithm was able to classify the whole set of pathogenic mutation when applied to Edge-Histogram kernel. The classification of the cores in vacuo in Fig. 6.30, shown a little improvement respect the previous classification (a) when performed on Edge-Histogram kernel, failing for 22% of the mutation vs. 26% of the previous and thus indicating an already good structure differentiation achieved after refining stage. Regarding the full structures the classification was performed only on H-Bonds matrix in Fig. 6.27(c) because was the best clustered, failing in fact for 29% of the mutations compared to the previous of 50% (See Table 6.6). Notably, the Gram-matrix in Fig. 6.7 resemble very closely the related TM-Score matrix in Fig. A.2 indeed remarking what observed in Section 6.4 about the cluttering behavior of the intracellular-coils on this type of analysis.

6.5.2 MOESM3 based models

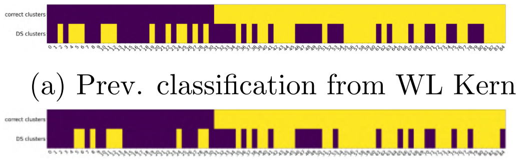
In this set of results, the inputs provided to the Pipeline were models refined with reaction field approximation (See Section 3.3.5).



(a) Core-in vacuo (b) Full model- TIP3P

Figure 6.31: Selected Gram-matrix of filtered subgraphs respect VdW interactions of MOESM3 based models

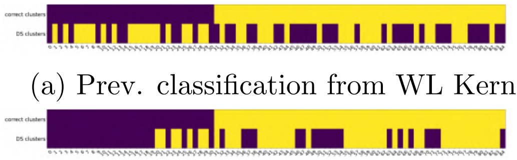
As seen for previous cases, Kernels on VdW and H-Bonds interactions subgraphs gives the best output in terms of clustering and classification:



(a) Prev. classification from WL Kernel

(b) WL-Kernel

Figure 6.32: Core-vacuo



(a) Prev. classification from WL Kernel

(b) WL-Kernel

Figure 6.33: Core-TIP3P

Dominant-set clustering applied to VdW Gram-matrix from Weisfeirer-Lehman (a) and Edge-Histogram (b) Kernels on MOESM3 hydrophobic core.

in Fig. 6.32,6.33 are presented the improved classifications with the filter for the core in vacuo and solvated. About the core models, is possible to see a slight improvement in classification over the previous, with a percentage of misclassification equal to 35% vs. 45%. Solvated cores, on the other hand, were better ranked on Weisfehrer-Lehman kernel with an improvement of 31 vs. 44%. The full model is not presented because the classification did not improve and the variation between kernels was too high, confirming once again the difficulty of discerning the structure from the effect that coils have on the Residue Interaction Network.

Chapter 7

Molecular dynamical study

In the previous work [51], one aspect that urgently needed to be investigated was to obtain information about the point-mutated models in order to assay the differences between pathogenic and non-pathogenic behavior. Prior to this, the method used was applied to the Trp-Cage system because it is well known, and therefore useful for testing the use of collective variables (See Section 3.4.2) both canonical and non-canonical.

7.1 Trp-Cage as test for method validation

In order to understand the typical form of a Free Energy Landscape, simulation and consecutive construction of the free energy surface were carried out of a small protein, the Trp-Cage structure. Trp-cage (1L2Y.pdb) is a 20-residue miniprotein, which is believed to be the fastest folder known so far. It contains a short α-helix in residues 2–9, a 310-helix in residues 11–14, and a C-terminal polyproline II helix to pack against the central tryptophan (Trp-6) [59]. The latter residue seems to be involved in the formation/breaking of side-chain bonds during the various folding-unfolding processes, because of its preferential position in the core region[27]. The important aspect of this protein is that, when solvated, it exhibits a well-defined transition structure at specific temperatures. Normally the folding temperature is found around 400K [33], but this value depends on the used forcefield in the molecular dynamics simulation. For example with CHARMM36[21] the folding is seen at temperature 400K, higher than in other forcefields [36]. Previous studies of this protein were carried out with replica exchange method [59], because allows to explore more conformations at different temperatures. Here unrestrained Molecular Dynamics were chosen as our aim was the analysis of the local minima, and this tool is sufficient to this aim. The analysis was performed using the method described in 3.4, by histogramming two reaction coordinates at different temperatures.

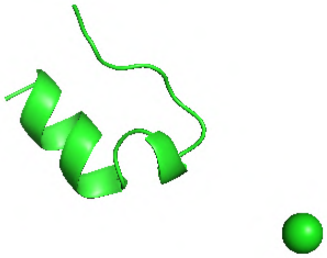


Figure 7.1: Native state of Trp-Cage miniprotein with a Cl^{-} as counter ion, Model 1 of the 1L2Y.pdb

7.1.1 Simulation Method

The simulation were carried out in explicit water TIP3P (See Section 3.3) with CHARMM36[21] forcefield. The integrator chosen was

Langevin Integrator with timestep set to 2 ps. The production runs were performed in NVT for a time length of 45 ns at different temperatures equal to 330 K,400 K,450 K, 460 K,470 K,480 K,490 K,500 K; the selected temperature was kept constant with the Langevin Thermostat (See Section 3.3.3). This set of temperatures were chosen in agreement with a study conducted on the same protein in TIP3P [36] where $T = 450\,\mathrm{K}$ was identified to be the melting temperature for this protein when common forcefields, as CHARMM36, are used. Before every production run the system was equilibrated for 5 ns in NPT ensemble with pressure equal to 1 atm, which was kept constant with a Monte Carlo Barostat (See Section 3.3.4). All of the simulations were performed with OpenMM[12] python package on CINECA GALILEO100.

7.1.2 Data Analysis Method

Trajectories from molecular dynamics were used to obtain the Free Energy Surface (See Section 3.4) at each of the selected temperatures. Mainly Radius of Gyration, Fraction of native contact and Kernel Similarity (See Section 3.4.2) were used as reaction coordinates for this purpose. The Kernel-Similarity (See Eq.3.4.2) was obtained by the use of the Vertex-Histogram Kernel, after obtaining the Residue Interaction Network for every frame in the trajectory. Radius of Gyration and Fraction of native contact were performed with MDTraj[32] python package while Kernel-Similarity was calculated by an in-house script based on GraKel[44] package.

7.1.3 Results

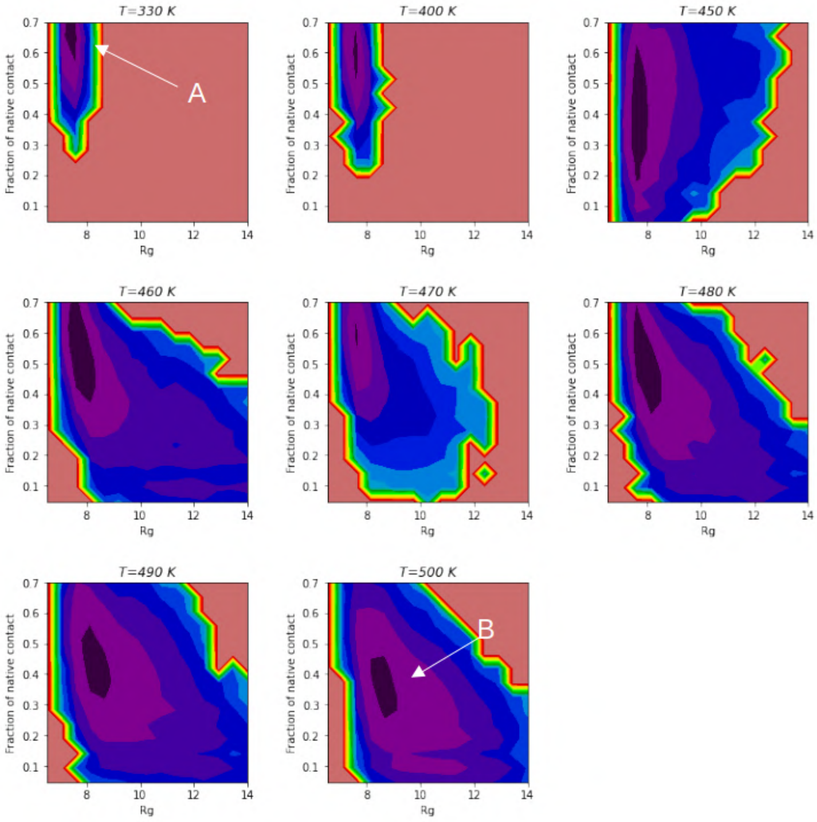
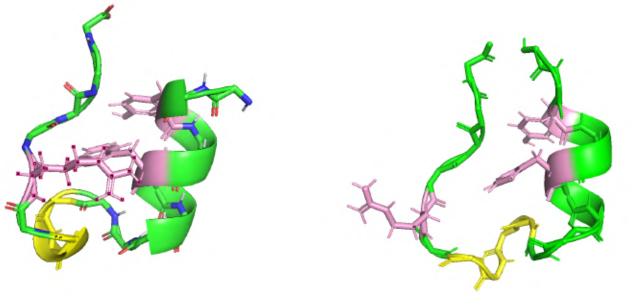


Figure 7.2: Free energy landscape of Trp-Cage at different temperature.



(a) Native state A (b) Denaturated state B

Figure 7.3: 3D models of Trp-Cage at the two identified minima with the three important residues Gly11, Arg16, Trp6 highlighted (pink); in (b) is possible to note the denaturated α-Helix (yellow)

The selected pair of reaction coordinates (Radius of Gyration, Fraction of native contact) is the most used in literature thanks to the reciprocal low correlation. As reported in previous studies[36], the system is trapped in a local minimum A with Rg=7 Å for temperature below 400K. At 450K the structure pass from the first native minimum A to a second minimum B, as depicted in 7.2, which refers to a partially unfolded state of the protein (See Fig. 7.3(b)). At 500K the equilibrium is mainly focused on this latter point B. Note that in this small protein the number of native contacts is tipically small and the fast nature of the process involving the formation of bonds can be difficult to capture only by counting. For this reason, the Kernel-Similarity was also studied as coordinate instead of number of native contact (See Fig. 7.4). the Free Energy Surfaces were obtained with Radius of Gyration and Kernel Similarity in order to study the goodness of graph comparison as reaction coordinate.

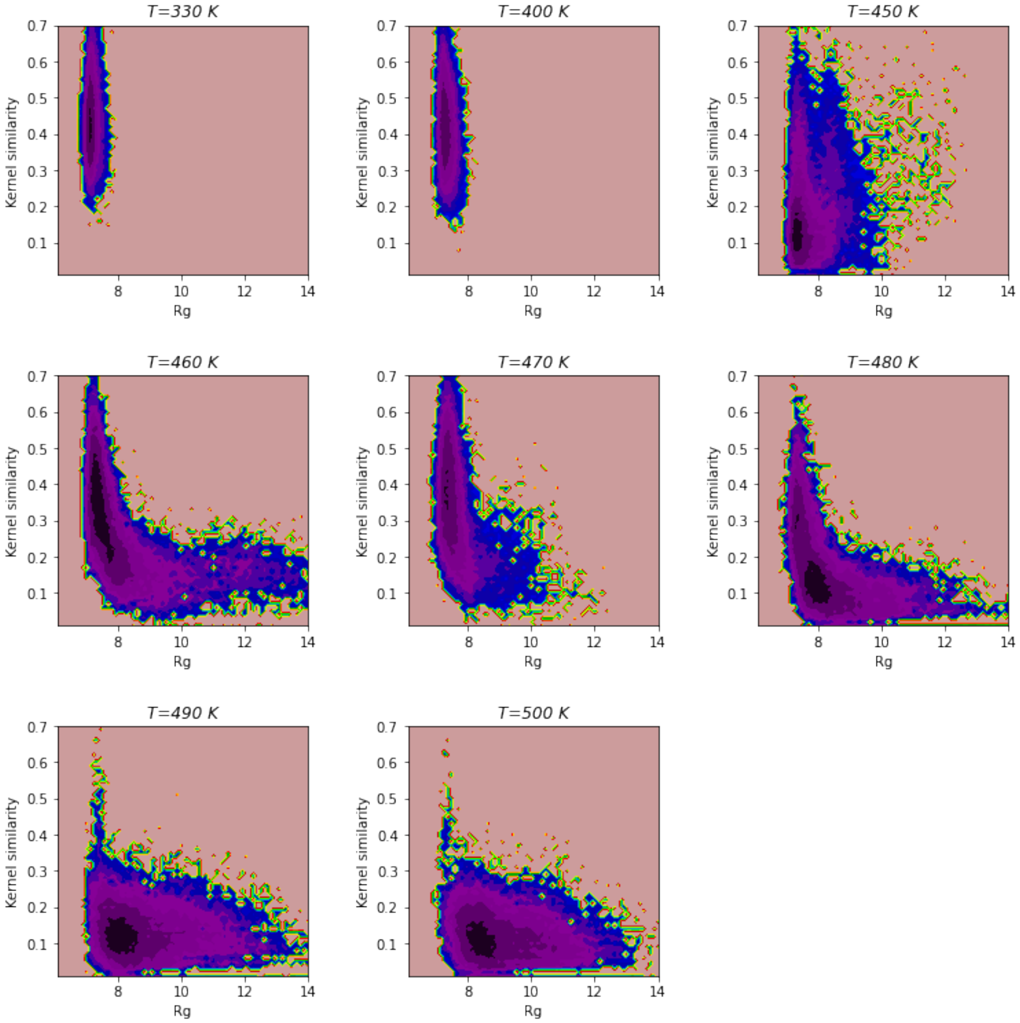


Figure 7.4: Free energy surface of Trp-Cage in function of Radius of Gyration and Kernel similarity.

While these new results agree with previous ones, they are more transparent. This can probably be ascribed to the fact that networks encode much more information than the number of native sites, and the comparison with kernel function gives continuous rather than discrete coordinate. This also improves the bin size in the 2D-histogram when constructing the free energy landscape and allows to have more overlap between a large number of bins (150 from Kernel-Similarity compared to the 11 bins of the fraction of native contact). To get further insight, the betweenness centrality of the Trp6 residue (i.e. TRP:6 node) from trajectory at different temperatures was analyzed. This residue appears to be involved in the formation/breaking of side-chain bonds during the various folding-unfolding processes, probably because its preferential position in the core region of the Trp-Cage protein [59]. This is reported in Fig. 7.5.

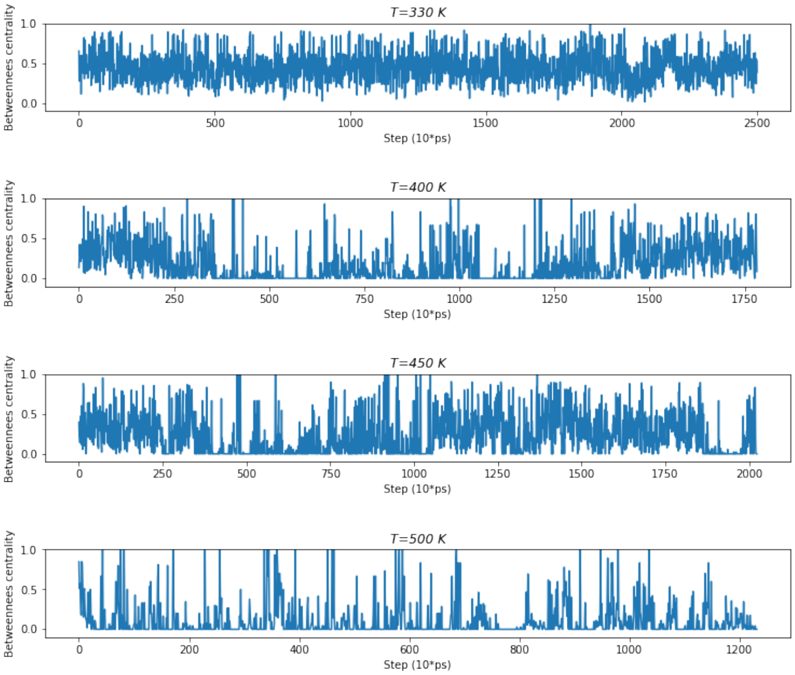
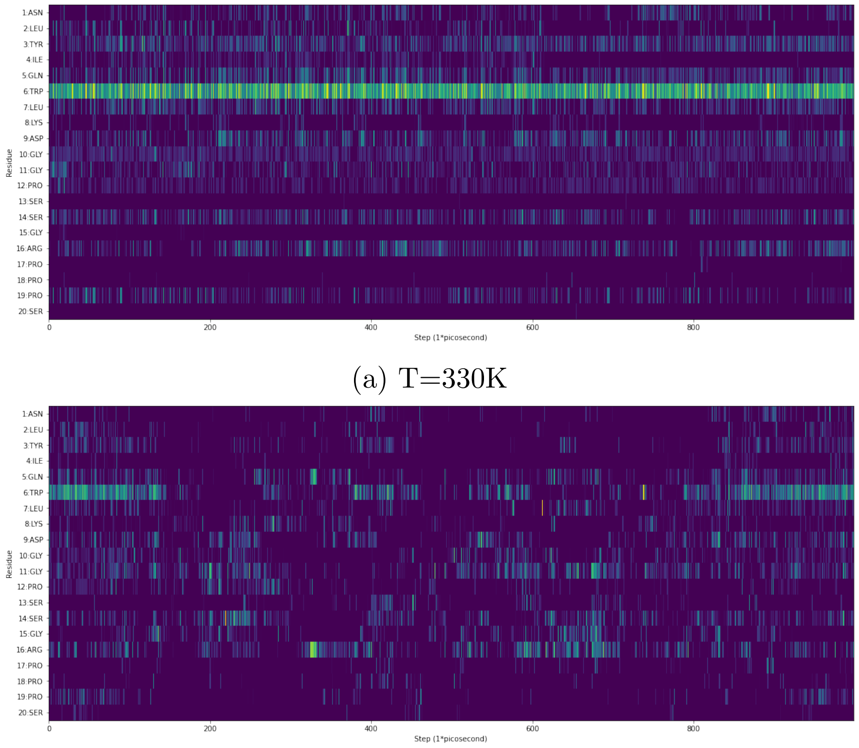


Figure 7.5: Betweenness centrality for Trp6 residue node in Trp-Cage graph series, at different temperature.

At 330K the centrality of the node remains stable, indicating, in agreement with the energy landscape analysis, an equilibrium localized in the first minimum A. At 400K, as depicted in Fig. 7.2, the equilibrium is shared between two conformations and a shift in centrality can be seen from the initial stage of the process (0-350) to the last stage (1750-2500). This can be ascribed to the salt-bridges and side-chain H-Bonds that this residue forms/breaks with the neighbour side-chains which shift the protein conformation between A and B [59]. At 500K the conformation is stable in the B state and the centrality is low, indicating a weakly-bonding Trp6. The various spike that can be seen in the process refers mostly on the solvent noise, so to apparent bonds found by RINmaker (See Section 5.1) when building the Residue Interaction Network.

Extending this analysis to all residues (See Fig.7.6, it's possible to identify the most important with respect to the minima depicted in Fig. 7.4:



(a) T=330K

(b) T=480K

Figure 7.6: Betweenness centrality of all the residues of Trp-Cage at different temperature

From Fig. 7.6(a) there is a clear dominance of the Trp6 residue, so the system remains in the identified minimum A during the simulation. At 480K, after the first 150 steps, The importance of Arg16 and Gly11 kick in. In agreement with previous studies[27], at a certain simulation time the distances between these residues starts to drop, indicating breaking and reforming of bonds and a shift to another minimum. In Fig. 7.6 (b) this process is partially captured, as the increase of betweenness centrality for these set of residues, clearly indicates the formation of new interactions among them. While constructive, the present energy landscape analysis on Trp-Cage cannot clearly guarantee a similar scenario for other, more complex, protein such as the MOESM.

7.2 Free Energy Landscape from Molecular Dynamics on variants derived from MOESM3

In order to gain preliminary insights on the MOESM3 structural changes given by point mutation, a set of unbiased Molecular Dynamics simulations were obtained and analyzed by appropriate reaction coordinates.

7.2.1 Simulation

For this simulations, 5 point-mutated models previously obtained with SWISS-MODEL were selected according to the pathogenic or non-pathogenic behaviour. In the corresponding structures the intra-cellular coils were removes and the obtained structure simuated with unbiased Molecular Dynamics for 50 ns at different temperatures in a range from 270 K-330 K. The simulations were carried out in explicit solvent TIP3P (See Section 3.3) with a ionic force of $0.150mM$(NaCl) and CHARMM36[21] forcefield. Verlet integration were coupled with an Andersen thermostat set to the desired temperature. The non-bonded interactions cutoff was 1 nm for both direct and long-range interactions, the latter accounted with Particle Mesh Ewald method. Production runs were performed in NVT for a time length of 45 ns and before a run the system was equilibrated for 5 ns in NPT ensemble with a pressure equal to 1 atm, which was kept constant by a Monte Carlo Barostat (See Section 3.3.4). The choice of this ionic force was made in agreement with previous studies on the Sodium flow through the channel pore[45], where the Na^+ ions appear to interact with the selectivity filter.

All simulations were performed with OpenMM[12] python package on

CINECA GALILEO100, with the GPUs as accelerators.

Mutation	Label
I136V	PAT
R185H	PAT
I228M	PAT
I1399D	NEUTRAL
V795I	NEUTRAL
Wild-Type	NEUTRAL

Table 7.1: Selected mutations

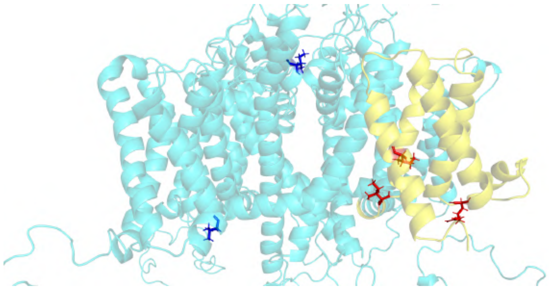
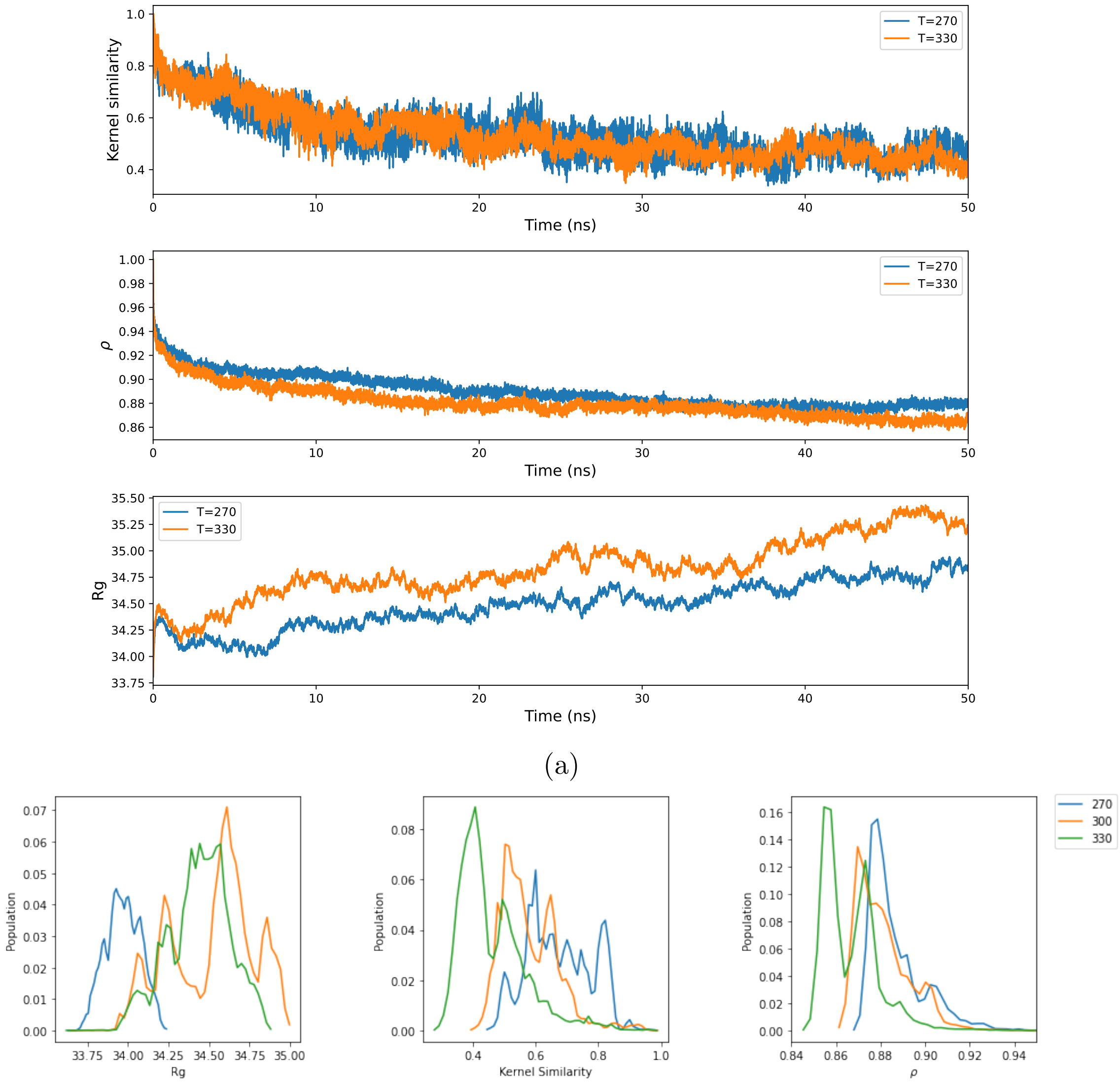


Figure 7.7: Wildtype with highlighted mutated pathogenic (red) and non-pathogenic (blue) residues

7.2.2 Data Analysis Method

The obtained trajectories at different temperatures were analyzed using some reaction coordinates, the Kernel-Similarity, the Radius of Gyration Rg, the Fraction of native contacts ρ (See Sec.3.4.2). The Kernel-Similarity (See Eq.(3.4.2)) was obtained by the use of the Vertex-Histogram Kernel, after obtaining the Residue Interaction Network for every frame in the trajectory. Radius of Gyration and Fraction of native contact were performed with MDTraj[32] python package while Kernel-Similarity was calculated by an in-house script based on GraKel[44] package. All data analysis was performed on CINECA GALILEO100. Interestingly, both kernels similarity and density of native contacts decreased with time, indicating a system moving farther from equilibrium.



(a)

(b)

Figure 7.8: Trajectory (a) and histogram (b) of the selected reaction coordinates at different temperatures for the MOESM3 Wild-Type.

7.2.3 Free Energy Landscape of MOESM3 Wild-Type

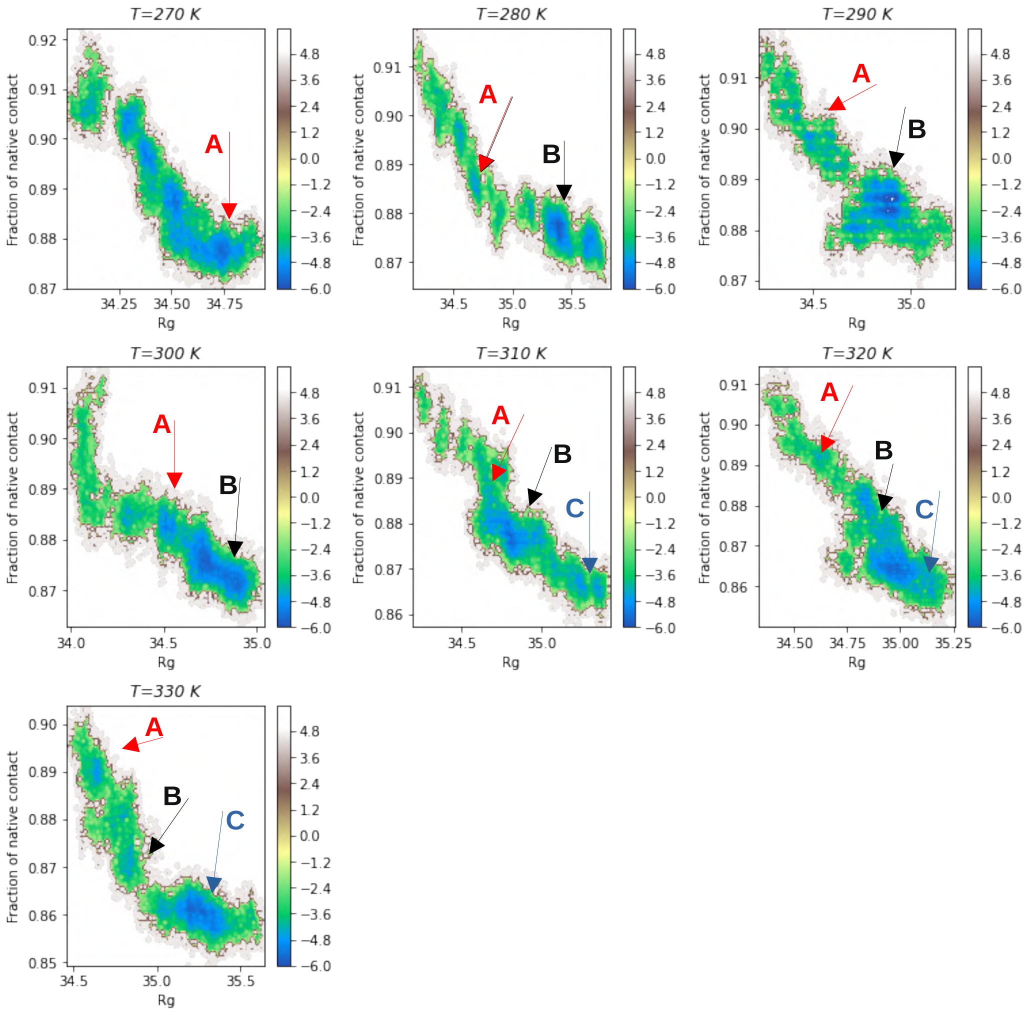
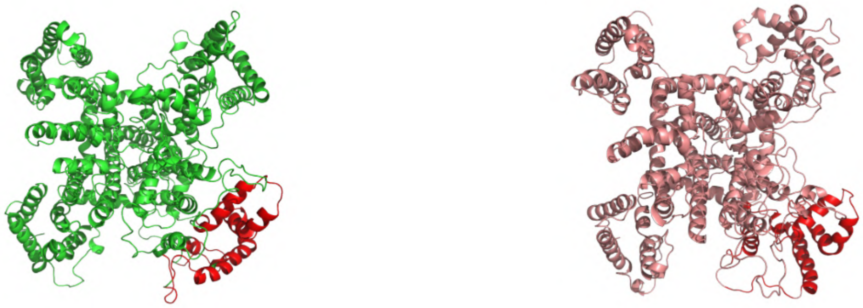


Figure 7.9: Free Energy Landscape $\frac{E}{KbT}$ of MOESM3-Wild Type core part (Rg vs Fraction of native contact); A,B,C refer to the three identified minima.

In the Fig. 7.9 the Free Energy Landscape $\frac{E}{k_BT}$ respect Radius of Gyration and Fraction of native contact is presented. From this analysis is possible in the first instance to notice a complex mechanism, involving different minima at different temperatures. Generally, the system is trapped in the minimum A at $T = 270\,\mathrm{K}$. At $T = 300\,\mathrm{K}$ and $T = 320\,\mathrm{K}$, which are close to the physiological temperature, the protein jump to another conformation C, which become clear at $T = 330\,\mathrm{K}$. Since the transition between B and C appears around the physiological temperature it can be assumed that it is indicative of some kind of characteristic process of the moesm core. Below are highlighted the respective structures at these two found minima:



(a) State B

(b) State B

Figure 7.10: Structures corresponding to minima obtained from the FEL marked as B(a) and C(b) of the Wild-Type

In Fig. 7.10 is possible to see a clear rearrangement of the Voltage-Gain of Domain IV. The state B also carry a partial unfolded $\alpha-$helix belonging to the selectivity filter. The state C instead has a more orderly structure, with the upward-coil bending toward the core. The partial unfolded state B and the bending in state C suggest a probable interactions of the selectivity filter with the Na^+ cation in solution, which is better expressed as temperature increases. These observations on the structural rearrangement of the backbone, especially for the IV Domain, shows a marked sensitivity of this backbone part for variations involving solvent (since its real environment is a lipid-bilayer) and temperature. Given the improved accuracy showed for Trp-Cage (See

Section 7.1), the Free Energy Surfaces were also obtained with Radius of Gyration vs. Kernel Similarity as in Fig. 7.11, where is possible to see the same minimum found for the Fraction of native contact:

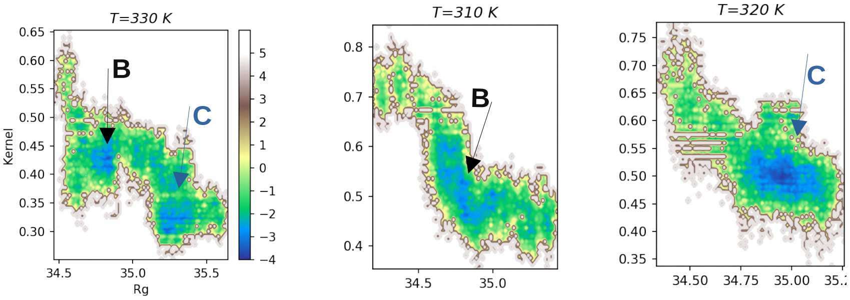


Figure 7.11: Details of B and C minima obtained from FEL of Kernel-Similarity vs Rg.

As for the Trp-Cage case, the Kernel Similarity as collective variable with the Radius of gyrations provided a more precise and differentiated representation of the FEL, especially the characterization of those minima which belong to a rearrangement of the side-chain net: In Fig. 7.11 the found minimums A and B are much more definite with respect Fraction of native contact counterparts. From the figure is clear a dominance of the minimum C at $T = 320\,\mathrm{K}$, while at $T = 310\,\mathrm{K}$ the minimum B is dominant. Interestingly, at $T = 330\,\mathrm{K}$ both minima B and C are visible, indicating that conformations are in equilibrium, and the energy gap between them is on the order of k_BT. As 330 K is slightly above the physiological temperature, they can be explored by a biological process.

7.2.4 Results from mutants

The same approach was also applied to the mutants of the Wild-Type. From the previous analysis seems that $T = 310\,\mathrm{K}$ and $T = 320\,\mathrm{K}$ are key temperatures where the structures converges to the two identified minima.

Rg vs Fraction of native contacts

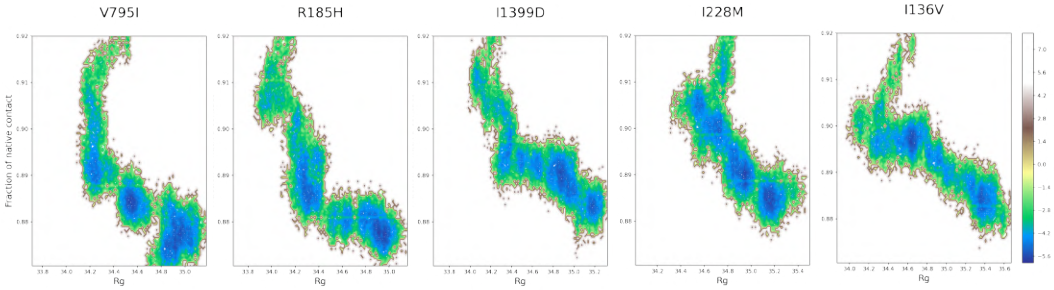


Figure 7.12: FES from mutants at $T = 320K$

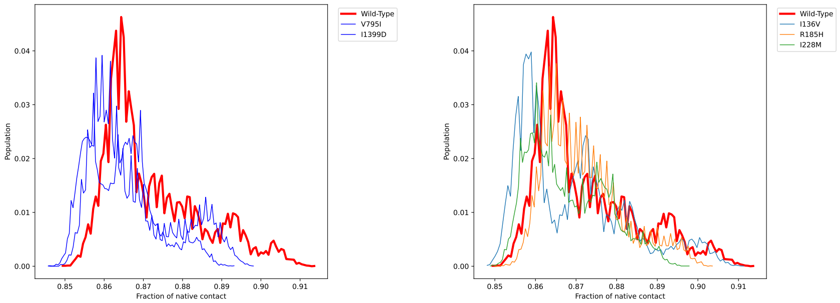


Figure 7.13: Histogram of fraction of native contact as collective variable for mutants at $T = 320\,\mathrm{K}$

Apart for the pathogenic I136V mutation, this set of collective variables seems unable to capture the different minima belonging to pathogenic and non-pathogenic mutations. Probably the description of Fraction of native contact based on distances between C_α is too rough for describing minimum and it is too much dependent on the position coordinate of the C_α ensemble. So is is clear that if the backbone is not heavily affected by the mutations, the rearrangement of the sidechain net is not captured.

FEL described with Rg vs. Kernel-Similarity

Free Energy Surface described with Radius of Gyration and Kernel-Similairty is shown:

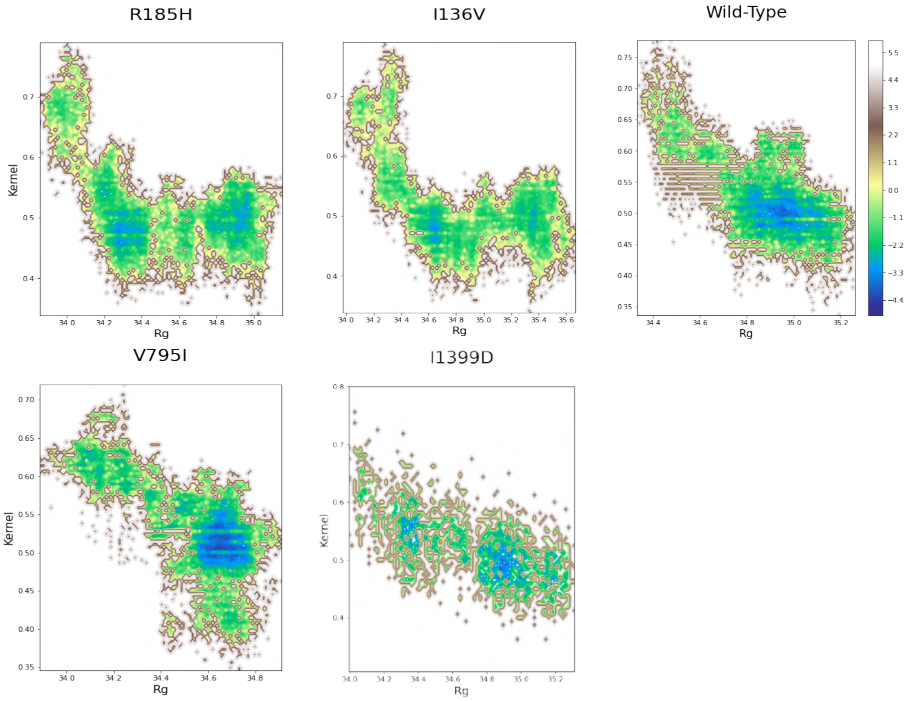


Figure 7.14: FES from mutants at $T = 320\,\mathrm{K}$

From Fig. 7.14 is possible to note two dominant minimus in V795I and I1399D neutral mutations while there are not dominant minimums in R185H and I136V pathogenic mutations. These are encouraging results that clearly show the effect of pathogenic mutations. The histogram graph shows a differentiation between pathogenic and non-pathogenic:

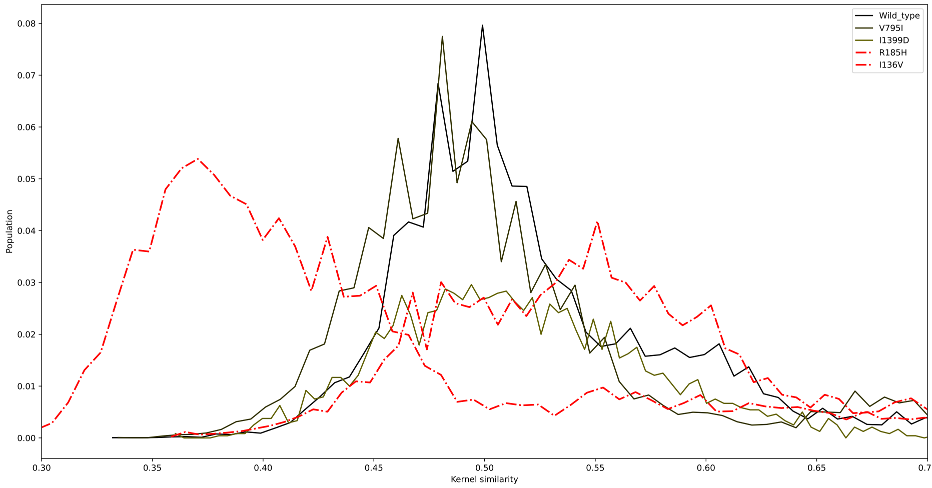


Figure 7.15: Histogram of Kernel similarity values for selected pathogenic and non-pathogenic mutations

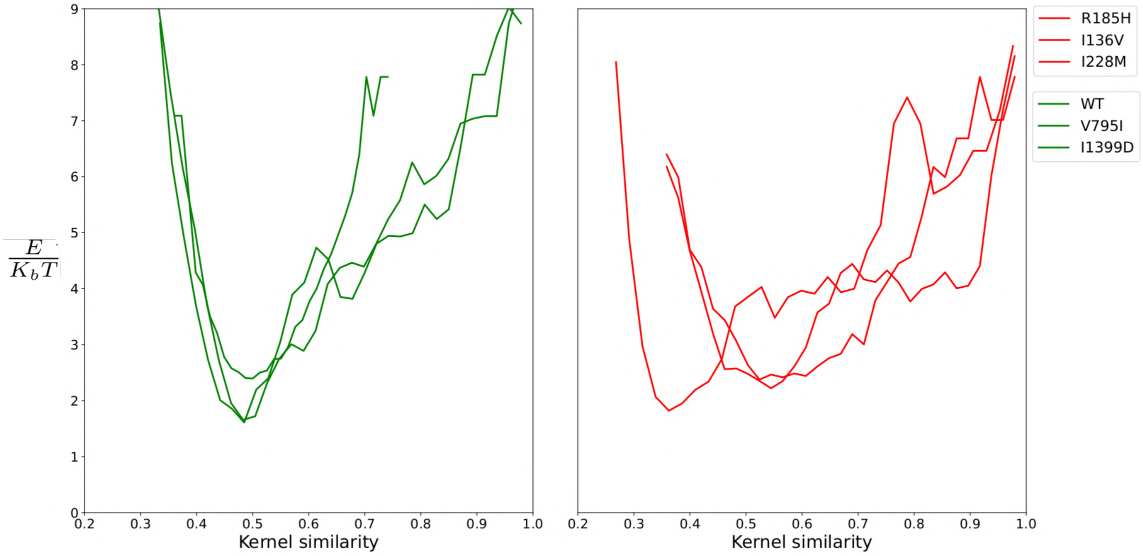


Figure 7.16: Free Energy vs Kernel similarity histogram in Fig. 7.15, divided according to pathogenic (red) and non-pathogenic (green) behavior.

The corresponding minimum depicted in Fig. 7.16 is clearly identified from Kernel-Similarity and it shows a preferential minimum as 0.5 for the non-pathogenic, while the landscape is more complex for pathogenic mutation. This results that in the pipeline essentially compared the minima, since every RIN correspond to a unique arrangement of the non-covalent interactions, and the minimum of the free energy is the result of the simulated-annealing. The appearance of clusters in cross-comparison with Graph-kernel in Fig. 7.17 is a consequence of the fact that RIN correspond to a "snapshot" of the protein.

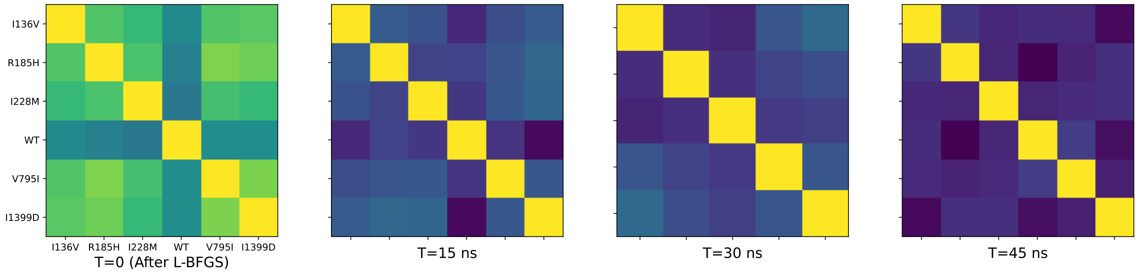
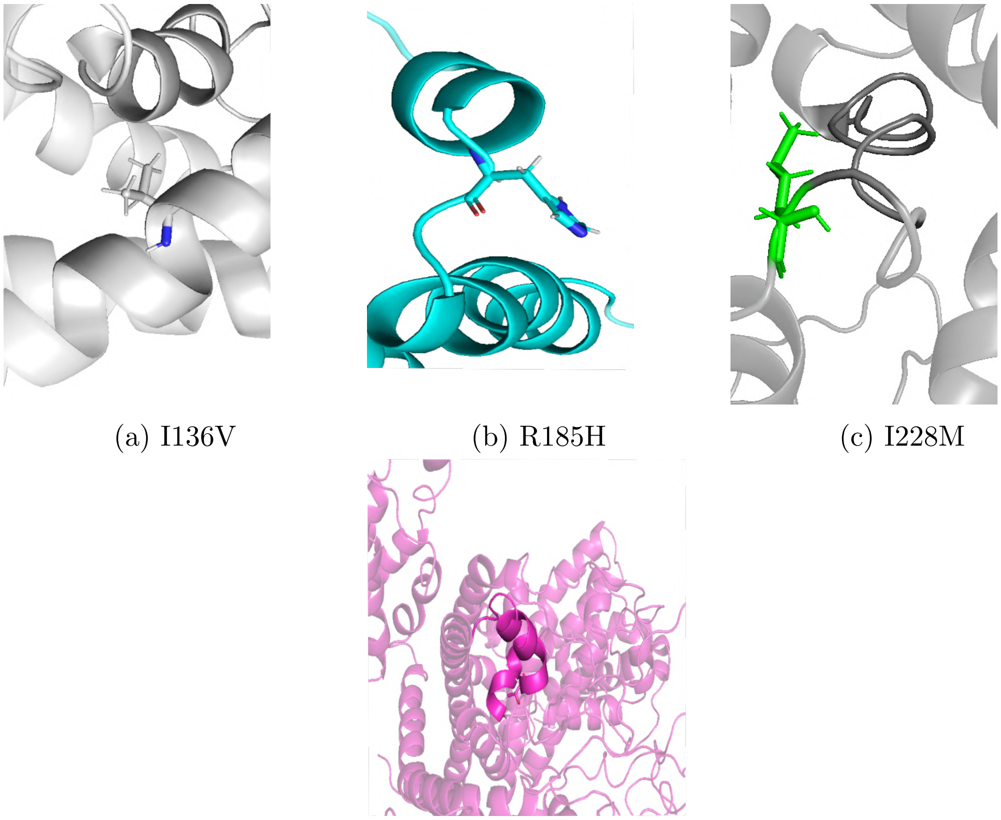


Figure 7.17: Gram-matrices of Vertex histogram kernel applied to graphs of all interactions taken at different simulation time. The Node and Edges are not filtered.

Note that this analysis is to be carried out at different times for a full picture.

Insights into structural changes along the backbone

In the following are depicted the backbone unfolding relative to pathogenic mutation, where the structure correspond to C minimum at $T = 320\,\mathrm{K}$.



(a) I136V

(b) R185H

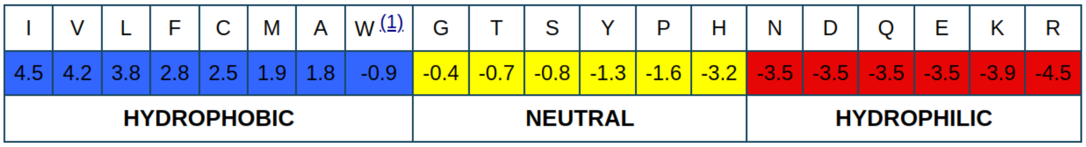
(c) I228M

(d) I1399D

Figure 7.18: Local misfolding due to point mutation

7.2.5 Solvent-Accessible Surface Area of MOESM3 based model

Further support to this scenario is provided by the analysis of SASA of every residues during the trajectory of the simulation. This analysis was limited to $T = 320\,\mathrm{K}$. In this case the SASA is calculated in ”residue” mode, where the Shake-Rupley algorithm construct the probing ball from the C_α as center.



I	V	L	F	C	M	A	W(1)	G	T	S	Y	P	H	N	D	Q	E	K	R
4.5	4.2	3.8	2.8	2.5	1.9	1.8	-0.9	-0.4	-0.7	-0.8	-1.3	-1.6	-3.2	-3.5	-3.5	-3.5	-3.5	-3.9	-4.5
HYDROPHOBIC								NEUTRAL						HYDROPHILIC					

Figure 7.19: Relative hydrophobicity of the 20 aminoacids, accordin to the Kyte-Doolittle scale [26].

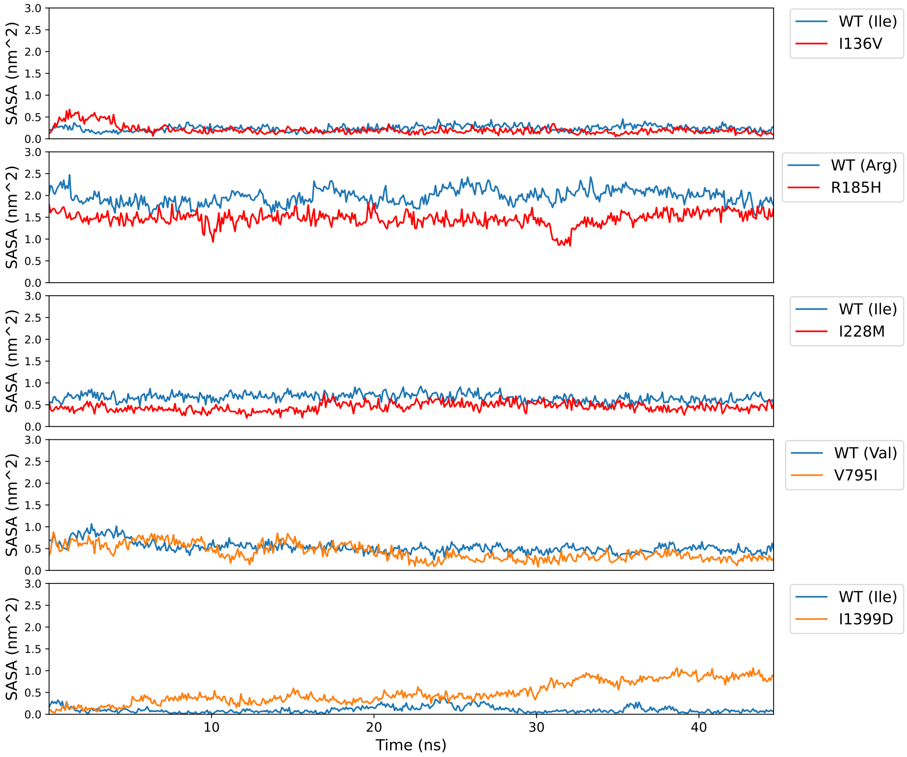
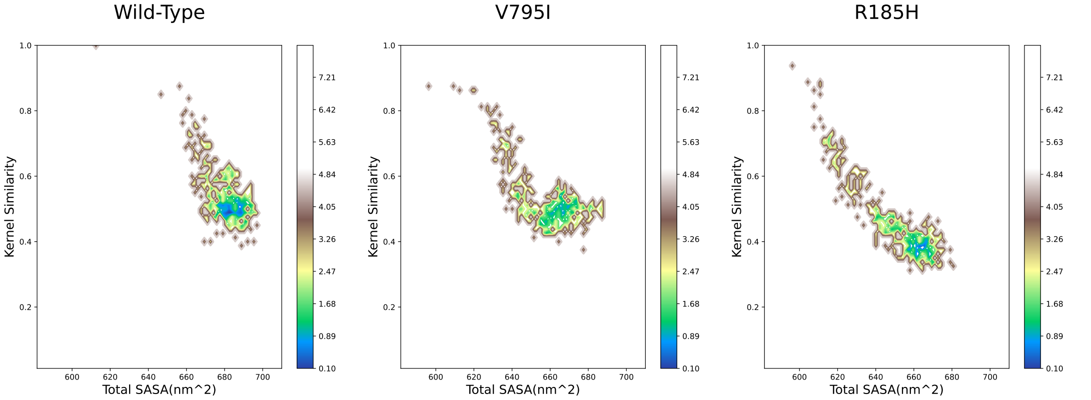


Figure 7.20: Comparison of SASA between the mutated residue on pathogenic mutants (red) and non-pathogenic (orange) on the respective Wild-Type backbone position.

In Fig. 7.20 the time evolution of mutated-residue (SASA) respect time is reported. Is clear that the he relative hydrophobicity or hydrophilicity of residues plays a prominent role in this analysis. For the I136V mutation, SASA clearly show the similar behavior between the mutated residue and the corresponding one in the Wild-Type: since both isoleucine and valine are both hydrophobic residues, the mutation shares a low and constant SASA, both cases indicating that this residue remains mostly buried during the process and achieves a conformation that minimizes interactions with water. From R185H instead the situation is much more complicated because both Arginine and Histidine are negative charged residue and a strong interaction with the solvent is expected; indeed Wild-Type Arginine, which is believed to be the most hydrophilic residue in membrane proteins[20], shows marked instability compared to the histidine in the mutant, indicating that arginine in the Wild-Type switches from one rotamer to another and lives in an unstable situation, in which the balance between interactions with the solvent and between neighboring residues does not allow it to find a stable conformation. In reverse Histidine SASA remains stable for most of the process, with a negative peak near 35 ns, for sure referring to an overall change in the backbone. In I228M mutation, Isoleucine is the most hydrophobic residue according to Table in Fig. 7.19 while Methionine in Wild-Type is more neutral; the overall behavior is a stable conformation for Methionine in the Wild-Type, while Isoleucine pass different conformations during the process; another time the explanation could be that Isoleucine find a buried conformation during the equilibration of the protein while Methionine suffer a concerted effects involving interactions with solvent and chain residues. For non-pathogenic V795I the same consideration apply for pathogenic I136V, this also shows that SASA of a single residue is not able to capture different minima, and so the key of pathogenic unfolding is an overall analysis. Interestingly, in the non-pathogenic I1399D, Isoleucine shows a buried conformation at the beginning of the process with a steadily increasing SASA starting at 5 ns, while Aspartic Acid from the Wild-Type, which is a highly hydrophilic residue, shows a buried conformation. Both residues are located in the selectivity filter of the core region. This counterintuitive trend is likely to be due to the interaction of this region with Na^+ cations. This also reflects a probable local unfolding, because Isoleucine, for steric reasons, is constrained to assume conformation more accessible to the solvent. This means that this mutation, although non-pathogenic, has an effect on the backbone structure when solvated in water, as seen from Fig. 7.21.

Total SASA from structures The total SASA of some structures were also taken and the FES constructed vs Kernel Similarity. Since it is a heavy computational task, this parameter was taken only for some key mutations (See Fig 7.21.



(a) Wild-Type (b) V795I (c) R185H

Figure 7.21: Total SASA for Wild-Type, non-pathogenic V795I and pathogenic R185H

In this case the Wild-Type shows an enhanced stability, with a well defined minima. With the Non-pathogenic mutation V795I showing a similar trend albeit with different shape by contrast. R185H pathogenic mutation instead displays a different minimum respect the non-pathogenic mutation, indicating an overall change due to residue-solvent interactions.

Chapter 8

Conclusion

The objectives of this thesis were threefold. Firstly, we aimed at improving a pipeline previously proposed within our group [51] that was devised for addressing a specific problem and hence not easily extendable to other systems. In this case, particular attention has been devoted to implement a well defined input and output protocol, thus avoiding useless time consuming passages from one part to another of the pipeline. The second aim of the thesis was to avoid the use of two web tools that were present in the original pipeline, and replace them with in-house counterparts. This included two major endeavors. In the first one, we have replaced the Fragment-Guided refining tool [56] with a fully internally coded tool on which we have total control. At the present time, this tool uses state-of-the-art algorithms and is 10 times faster than the original one thanks to the possibility to use GPUs instead of CPU, while still providing identical results. The second major improvement stems from the implementation of a in-house RIN generator (RINmaker) that is also replacing the on-line tool RING2.0 web server. In this case too, a brand new coding allows an improvement of a factor 10^5 in terms of computation time required for generating the network of a model (approximately 1 ms as opposed 15 min for a 1650 residues). RINmaker was already implemented at the beginning of this thesis, but an extensive test phase was missing and the contribution has been the development of a suitable set of molecules forming the groundtruth needed to put it in full production. In general, the new pipeline now allows to have an extremely fast flow of data and the modular architecture which is built allows to further interchange steps, implements and insert new script without altering the main code.

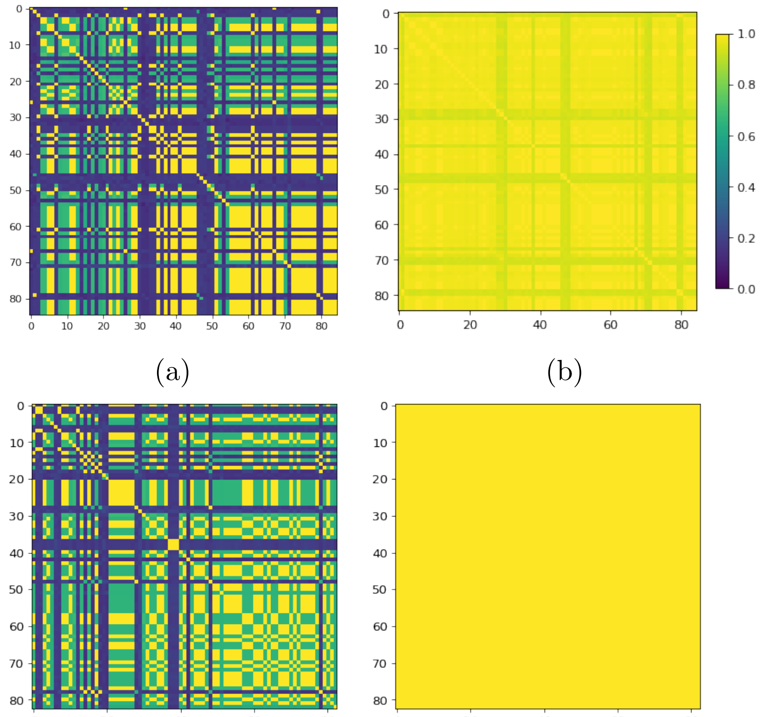
Thanks to this new instrument it was possible a full independent validation of the previous obtained results, that was the third aim of the present thesis. This led us to unveil a subtle shortcoming of the original interpretation, in the following sense. In the original investigation, the coils which are a significant part of the analyzed proteins, when generated by SWISS-MODEL and refined by web-based FGMD, were binary separated in two conformations according to the pathogenicity, giving the well defined clusters on the Gram-matrices obtained by graph comparison. This effect largely dominated the analysis and masked the effective contribution from the rest of the protein. To give a partial solution to this problem, the intra-cellular coils were removed from the mutated structures and a raw filter on the graphs was implemented, with improved clustering and label prediction. On the other hand, the structural effect that a specific mutation induces to the structures shows clear distinct energy landscape, especially when described with the usage of Kernel similarity as collective variable. This results is of prominent interest because it validates Kernel methods on RINs as valid and sensible tool for explore mutated landscape and subsequently select the conformation which the effective structural change took place, as well as it reinforces the results obtained from Dimos et al.[24] supporting the independence of the unfolding pathway from the biological function of the NaV1.7 sodium channel.

8.1 Future perspective

Although there is an effective improvement with respect to the original pipeline in both the methodology and the workflow, several shortcomings still persist. First and foremost, the difficulty in a consistent interpretation of the binary pathogenic/non-pathogenic decision that is masked by irrelevant contributions that are inserted by the only webtool (Swiss-model) still included in the pipeline. The detailed analysis of the free energy landscape (FEL) presented in this thesis was indeed devoted to the understanding of this point. The main question to address is how to manage data scaling in number with the size of the system and what kind of collective variables are suitable for extracting only the "essential" information, thus minimizing the presence of noise and unnecessary information in the graphs, especially when obtained from Molecular Dynamics trajectories. As shown by FEL analysis, this problem is mitigated by the use of Kernel similarity as collective variable, that is by the comparison of trajectory graphs with the graph at $t = 0$. The solution adopted in this thesis is to try to isolate only the subset of edges and nodes (as subgraphs) whose interactions are important. An appropriate score function is however needed, where all of the data encoded in the RIN (physical and abstract) might be rearranged in a linear combination able to identify the biological "importance" of edge and nodes. Once this score function is defined, the coefficient of the combination could be computed, for instance, via machine learning techniques and used for extract important subgraphs, which can then be contrasted by Kernel functions. In general, what is needed is to force the correspondence between the physical and topological representation, in order to automatize the identification of difference between pathogenic and non-pathogenic mutation. Another future perspective for fully validate the set of mutations, is to perform all-atoms Molecular Dynamics on the ”real” environment, with membrane lipid layer and an external electric field. The solution of this class of problems needs tools from different disciplines as computer science and statistical mechanics, tracing a clear path toward an interdisciplinary approach.

Appendix A

Images



(a)

(b)

(c)

(d)

Figure A.1: TM-score matrix of SWISS-MODEL models. MOESM3: **(a)** intra-cellular coil, **(b)** core+extracellular coil, 6A90: **(c)** intra-cellular coil **(b)** core+extracellular coil

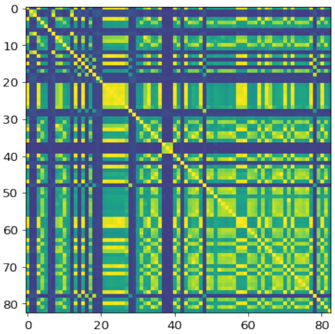


Figure A.2: TM-Score matrix of 6a90 full-models after the in-house FG-MD refining

Appendix B

Methods

B.1 Angles difference distribution

Given a Wild-Type structure and a set of point-mutated structures:

1. Retrieve values relative to angles ϕ,ψ,X_1,X_2,X_3,X_4 for every residue in the Wild-Type
2. Apply the same procedure in step 1 for every point-mutated structure
3. Subtract residue by residue the values obtained from a mutated structure to the respective values obtained for the Wild-Type and store the values obtained in a L_{diff} list.
4. Repeat the step 3 for each mutation and store the value in the same list L.
5. Obtain the histogram of the L_{diff}.

B.2 Filter stage

Let $\mathcal{G}_{\mathrm{NET}}$ and $\mathcal{G}_{\mathrm{PAT}}$ be the set of graphs related respectively to non-pathogenic and pathogenic mutations and $f_{rel}^{\mathcal{G}_k}(i)$ the relative frequency of appearance of an observed edge i in a given set of graph $\mathcal{G}_k$:

1. Retrieve the whole set of edges: $\mathcal{E}_{common} = \{E(G_1) \cap E(G_2) \cap E(G_3) \cap ....E(G_{85})\}$
2. Construct the set as: $\mathcal{E}_{dif} = \{i \in \mathcal{E}_{common} :\mid f_{rel}^{\mathcal{G}_{NET}}(i) - f_{rel}^{\mathcal{G}_{PAT}}(i) \mid \geq T_{dif}\}$. The threshold T_{dif} in this work is manually set.
3. For every $G \in \mathcal{G}_{\mathrm{NET}}$,$G \in \mathcal{G}_{\mathrm{PAT}}$ remove edges which not belongs to $\mathcal{E}_{dif}$
4. For every $G \in \mathcal{G}_{\mathrm{NET}}$,$G \in \mathcal{G}_{\mathrm{PAT}}$ remove edges which the $f_{rel}^{\mathcal{G}_{\mathrm{NET}} \cup \mathcal{G}_{\mathrm{PAT}}}(i) < T_{noise}$ where T_{noise} is manually set usually to 0.2.
5. For each graph $G \in \mathcal{G}_{\mathrm{NET}}$,$G \in \mathcal{G}_{\mathrm{PAT}}$ remove the isolated vertexes.

B.3 Brief recap of important backbone and sidechains parameters

Taken a set of residues in a given protein, most of the energy conformations can be uniquely binded to the angles defining the backbone and the sidechains. In particular, two sets of these angles are the most important:

- Backbone angles (ϕ, ψ), respectively the phi dihedral angle between C_α and N carbons and psi dihedral angle between C_α and

C carbons.

- Sidechains dihedral angles (X_1,X_2), where X_1 is the dihedral angles between C_α and C_β, where the latter belong to the sidechain carbon chain. The X_2 angle is introduced in the case the sidechain contain a C_γ eventually binded to the C_β. A particular combination of these two angles is called rotamer.

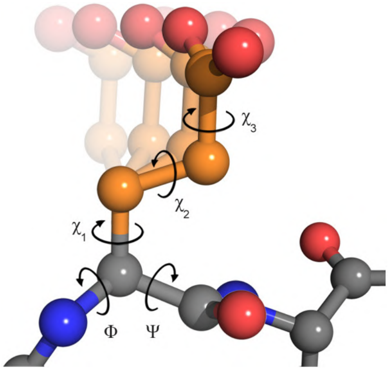


Figure B.1: 3D cartoon depicting $\phi, \psi, X_1, X_2, X_3, X_4$ angles

This set of angles (ϕ,ψ,X_1,X_2) fully describe the backbone and sidechains conformation of a given protein. Phi and theta angles are strongly binded to the quality and conformation of a model, in fact not all of the angles are "allowed" or "favourable"; this aspect can be carried out by a Ramachandran plot.

Appendix C

Tables

Mutation	SW vs FG-inhouse	SW vs FG-web	SW vs no-FG
V194I.pdb	0.984	0.987	0.848
A1746G.pdb	0.985	0.988	0.862
M1627K.pdb	0.983	0.985	0.884
A766V.pdb	0.985	0.989	0.855
T1548S.pdb	0.984	0.985	0.816
A815S.pdb	0.984	0.988	0.803
I1577L.pdb	0.984	0.985	0.848
V795I.pdb	0.985	0.983	0.869
T1398M.pdb	0.985	0.984	0.827
T773S.pdb	0.985	0.988	0.844
D1662A.pdb	0.982	0.986	0.825
M1532I.pdb	0.984	0.986	0.847
D1586E.pdb	0.985	0.986	0.816
V1299F.pdb	0.984	0.985	0.898
H1531Y.pdb	0.985	0.988	0.861
E1534D.pdb	0.985	0.985	0.822
S1419N.pdb	0.985	0.986	0.813

I848T.pdb	0.985	0.983	0.816
T370M.pdb	0.983	0.988	0.805
I1399D.pdb	0.985	0.985	0.806
V1428I.pdb	0.983	0.987	0.829
N395K.pdb	0.985	0.984	0.808
T1210N.pdb	0.983	0.989	0.834
T1596I.pdb	0.984	0.985	0.855
V1298D.pdb	0.984	0.989	0.845
V1298F.pdb	0.981	0.984	0.825
Q1530D.pdb	0.983	0.991	0.834
D890V.pdb	0.984	0.983	0.828
Q1530P.pdb	0.982	0.984	0.846
V1565I.pdb	0.983	0.989	0.819
I228M.pdb	0.984	0.985	0.818
G1674A.pdb	0.984	0.987	0.812
S126A.pdb	0.984	0.985	0.801
T1590R.pdb	0.984	0.983	0.867
E759D.pdb	0.985	0.989	0.833
F1449V.pdb	0.985	0.984	0.836
L823R.pdb	0.984	0.987	0.829
L858F.pdb	0.984	0.986	0.833
T1590K.pdb	0.986	0.985	0.826
K1700A.pdb	0.982	0.989	0.839
V872G.pdb	0.983	0.984	0.828
K1415I.pdb	0.983	0.985	0.817
L1267V.pdb	0.984	0.983	0.834
Y1537N.pdb	0.985	0.984	0.828
S1509T.pdb	0.983	0.983	0.842
A766T.pdb	0.984	0.981	0.834
S241T.pdb	0.985	0.988	0.826
M145L.pdb	0.985	0.987	0.864

L858H.pdb	0.983	0.986	0.828
R1207K.pdb	0.983	0.980	0.835
I1235V.pdb	0.985	0.980	0.814
A1632E.pdb	0.985	0.988	0.817
V400M.pdb	0.984	0.986	0.825
L127A.pdb	0.982	0.988	0.836
G856D.pdb	0.983	0.986	0.859
W1538R.pdb	0.985	0.984	0.838
A1505V.pdb	0.985	0.989	0.831
G1607R.pdb	0.984	0.989	0.804
K1412E.pdb	0.984	0.989	0.808
N146S.pdb	0.984	0.986	0.805
R185H.pdb	0.981	0.987	0.826
I767V.pdb	0.985	0.984	0.808
A863P.pdb	0.982	0.988	0.826
M932L.pdb	0.985	0.986	0.809
I739V.pdb	0.983	0.983	0.828
L201V.pdb	0.985	0.989	0.808
V1613I.pdb	0.986	0.986	0.828
P1308L.pdb	0.985	0.984	0.793
T920N.pdb	0.984	0.986	0.810
S1509A.pdb	0.983	0.989	0.812
K1412I.pdb	0.981	0.984	0.810
D1411N.pdb	0.984	0.982	0.815
H1560C.pdb	0.982	0.983	0.838
F216S.pdb	0.985	0.982	0.804
D890E.pdb	0.984	0.986	0.822
Q1530K.pdb	0.981	0.985	0.804
M1532V.pdb	0.985	0.988	0.828
N1245S.pdb	0.981	0.989	0.837
K1176R.pdb	0.983	0.989	0.839

H1560Y.pdb	0.985	0.985	0.816

Table C.1: Comparison of TM-score between 6a90 obtained from SWISS-MODEL vs FGMD-web and FGMD-inhouse

Mutation	SW vs FG-inhouse	SW vs FG-web	SW vs no-FG
V194I.pdb	0.990	0.987	0.808
A1746G.pdb	0.988	0.988	0.822
M1627K.pdb	0.983	0.988	0.804
A766V.pdb	0.984	0.988	0.815
T1548S.pdb	0.987	0.988	0.836
A815S.pdb	0.985	0.983	0.823
I1577L.pdb	0.981	0.986	0.848
V795I.pdb	0.982	0.983	0.849
T1398M.pdb	0.988	0.984	0.857
T773S.pdb	0.986	0.988	0.864
D1662A.pdb	0.984	0.986	0.875
M1532I.pdb	0.989	0.986	0.847
D1586E.pdb	0.982	0.986	0.886
V1299F.pdb	0.985	0.985	0.858
H1531Y.pdb	0.983	0.988	0.891
E1534D.pdb	0.989	0.985	0.842
S1419N.pdb	0.985	0.986	0.883
I848T.pdb	0.985	0.983	0.856
T370M.pdb	0.983	0.988	0.835
I1399D.pdb	0.988	0.985	0.876
V1428I.pdb	0.980	0.987	0.889
N395K.pdb	0.982	0.984	0.898
T1210N.pdb	0.982	0.989	0.864
T1596I.pdb	0.984	0.985	0.855
V1298D.pdb	0.985	0.989	0.845

V1298F.pdb	0.986	0.987	0.835
Q1530D.pdb	0.980	0.991	0.884
D890V.pdb	0.981	0.983	0.818
Q1530P.pdb	0.982	0.984	0.826
V1565I.pdb	0.985	0.984	0.839
I228M.pdb	0.984	0.985	0.848
G1674A.pdb	0.987	0.989	0.872
S126A.pdb	0.984	0.985	0.861
T1590R.pdb	0.984	0.985	0.887
E759D.pdb	0.985	0.989	0.843
F1449V.pdb	0.985	0.988	0.856
L823R.pdb	0.984	0.987	0.879
L858F.pdb	0.984	0.983	0.843
T1590K.pdb	0.986	0.985	0.886
K1700A.pdb	0.982	0.985	0.859
V872G.pdb	0.983	0.989	0.838
K1415I.pdb	0.988	0.985	0.837
L1267V.pdb	0.983	0.984	0.834
Y1537N.pdb	0.989	0.988	0.858
S1509T.pdb	0.983	0.983	0.842
A766T.pdb	0.984	0.982	0.874
S241T.pdb	0.985	0.988	0.846
M145L.pdb	0.985	0.988	0.894
L858H.pdb	0.983	0.983	0.838
R1207K.pdb	0.983	0.984	0.835
I1235V.pdb	0.985	0.985	0.814
A1632E.pdb	0.985	0.984	0.797
V400M.pdb	0.984	0.984	0.825
L127A.pdb	0.982	0.983	0.816
G856D.pdb	0.983	0.986	0.829
W1538R.pdb	0.985	0.984	0.818

A1505V.pdb	0.985	0.989	0.821
G1607R.pdb	0.984	0.989	0.864
K1412E.pdb	0.984	0.987	0.868
N146S.pdb	0.984	0.986	0.845
R185H.pdb	0.986	0.987	0.886
I767V.pdb	0.985	0.984	0.848
A863P.pdb	0.980	0.988	0.836
M932L.pdb	0.987	0.986	0.889
I739V.pdb	0.983	0.983	0.828
L201V.pdb	0.985	0.987	0.808
V1613I.pdb	0.986	0.982	0.868
P1308L.pdb	0.985	0.982	0.733
T920N.pdb	0.984	0.981	0.860
S1509A.pdb	0.983	0.989	0.812
K1412I.pdb	0.981	0.984	0.820
D1411N.pdb	0.984	0.985	0.825
H1560C.pdb	0.982	0.986	0.858
F216S.pdb	0.985	0.982	0.804
D890E.pdb	0.987	0.986	0.862
Q1530K.pdb	0.988	0.985	0.814
M1532V.pdb	0.989	0.988	0.828
N1245S.pdb	0.983	0.993	0.877
K1176R.pdb	0.985	0.982	0.839
H1560Y.pdb	0.982	0.981	0.816

Table C.2: Comparison of TM-score between MOESM3 based model obtained from SWISS-MODEL vs FGMD-web and FGMD-inhouse

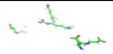
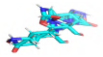
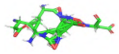
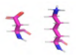
Test N°	Image	Type	Description	Param	Expected bonds
1		NEG	Test on an ensemble of 4 residues. The C.o.M rules is fulfilled only for residues with the same charge	Default	-
2		POS	Two residues fulfilling ion-ion bond rules	Default	One IONIC edge
3		POS	Three residues fulfilling ion-ion bond rules	Default	3 IONIC edges
4		POS	Two ionic groups (NZ, OE1) fulfilling ion-ion bond rules	Default	One IONIC edge
5		POS	Three ionic groups (NH2,OE,NZ) fulfilling ion-ion bond rules	Default	Two IONIC edges
6		NEG	Two ionic groups (NZ,OE1) not fulfilling C.of.M rule	Default	-
7		NEG	Three ionic groups (NH2,OE,NZ) not fulfillinf ion-ion bond rules	Default	-
8		POS	Two residues (LYS-ASP) fulfilling ion-ion bond rules	Default	One IONIC edges
9		NEG	Two residues (LYS-ASP) not fulfilling C.o.M rule	Default	-

Table C.3: Caption

Bibliography

[1] Patrick A. Alexander et al. "The design and characterization of two proteins with 88% sequence identity but different structure and function". In: Proceedings of the National Academy of Sciences 104.29 (2007), pp. 11963–11968. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.0700922104. URL: https://www.pnas.org/doi/abs/10.1073/pnas.0700922104.

[2] Gil Amitai et al. "Network analysis of protein structures identifies functional residues". In: Journal of Molecular Biology 344 (4 Dec. 2004), pp. 1135–1146. ISSN: 00222836.

[3] "An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein-protein complexes". In: Journal of Molecular Biology 326 (4 Feb. 2003), pp. 1239–1259. ISSN: 00222836.

[4] Bender and Williamson. Lists, Decisions and Graphs. 2010, pp. 194–195.

[5] M. Biasini et al. "iOpenStructure/i: an integrated software framework for computational structural biology". In: Acta Crystallographica Section D Biological Crystallography 69.5 (Apr. 2013), pp. 701–709. DOI: 10.1107/s0907444913007051. URL: https://doi.org/10.1107/s0907444913007051.

[6] Bela Bollobas. Modern Graph Theory. 2002, pp. 53–54.

[7] Philip E. Bourne and Helge. Weissig. Structural bioinformatics. Wiley-Liss, 2003, p. 649. ISBN: 0471201995.

[8] Jair Cervantes et al. "A comprehensive survey on support vector machine classification: Applications, challenges and trends". In: Neurocomputing 408 (2020), pp. 189–215. ISSN: 0925-2312.

[9] Wikimedia Commons. 2007. URL: https://upload.wikimedia.org/wikipedia/commons/thumb/c/c9/Multi-pseudograph.svg/1024px-Multi-pseudograph.svg.png,%20note%20=.

[10] Y. Dehouck, D. Gilis, and M. Rooman. "A New Generation of Statistical Potentials for Proteins". In: Biophysical Journal 90.11 (2006), pp. 4010–4017. ISSN: 0006-3495.

[11] Weinan E and Dong Li. "The Andersen thermostat in molecular dynamics". In: Communications on Pure and Applied Mathematics 61.1 (2008), pp. 96–136. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cpa.20198.

[12] Peter Eastman et al. "OpenMM 7: Rapid development of high performance algorithms for molecular dynamics". In: PLOS Computational Biology 13.7 (July 2017), pp. 1–17.

[13] Edward C Emery, Ana Paula Luiz, and John N Wood. "Nasubv/sub1.7 and other voltage-gated sodium channels as drug targets for pain relief". In: Expert Opinion on Thera
20.8 (Apr. 2016), pp. 975–983. DOI: 10.1517/14728222.2016.1162295. URL: https://doi.org/10.1517/14728222.2016.1162295.

[14] Roland Faller and Juan J. De Pablo. "Constant pressure hybrid molecular dynamics-Monte Carlo simulations". In: Journal of Chemical Physics 116 (1 Jan. 2002), pp. 55–59. ISSN: 00219606.

[15] FG-MD: High-resolution proteins structure refinement by Fragment-Guided MD simulation —
https://zhanggroup.org/FG-MD/. [Accessed 23-Jun-2022].

[16] Understanding molecular simulations: from algorithm to applications, pp. 309–312.

[17] Dmitrij Frishman and Patrick Argos. "Knowledge-based protein secondary structure assignment". In: Proteins: Structure, Function, and Genetics 23.4 (Dec. 1995), pp. 566–579. DOI: 10.1002/prot.340230412. URL: https://doi.org/10.1002/prot.340230412.

[18] Achille Giacometti and DONGMO FOUMTHUIM Cedrix Jurgal. Noncovalent interactions : Gen

[19] Kristen Grauman and Trevor Darrell. The Pyramid Match Kernel: Efficient Learning with Sets o
2007, pp. 725–760.

[20] Kalina Hristova and William C. Wimley. A look at arginine in membranes. Jan. 2011. DOI: 10.1007/s00232-010-9323-9.

[21] Jing Huang and Alexander D MacKerell Jr. "CHARMM36 all-atom additive protein force field: validation based on comparison to NMR data". en. In: J. Comput. Chem. 34.25 (Sept. 2013), pp. 2135–2145.

[22] P.G. Jambrina and J. Aldegunde. “Chapter 20 - Computational Tools for the Study of Biomolecules”. In: Tools For Chemical Product Design. Ed. by Mariano Martín, Mario R. Eden, and Nishanth G. Chemmangattuvalappil. Vol. 39. Computer Aided Chemical Engineering. Elsevier, 2016, pp. 583–648.

[23] 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. ISSN: 00219606.

[24] Dimos Kapetis et al. “Network topology of NaV1.7 mutations in sodium channel-related painful disorders”. en. In: BMC Syst. Biol. 11.1 (Dec. 2017).

[25] Georgii G. Krivov, Maxim V. Shapovalov, and Roland L. Dunbrack. “Improved prediction of protein side-chain conformations with SCWRL4”. In: Proteins: Structure, Function and Bioinformatics 77 (4 2009), pp. 778–795. ISSN: 08873585.

[26] J Kyte and R F Doolittle. “A simple method for displaying the hydropathic character of a protein”. en. In: J. Mol. Biol. 157.1 (May 1982), pp. 105–132.

[27] In Ho Lee and Seung Yeon Kim. “Dynamic folding pathway models of the trp-cage protein”. In: BioMed Research International 2013 (2013). ISSN: 23146133.

[28] Richard H. Lee. “Protein model building using structural homology”. In: Nature 356.6369 (Apr. 1992), pp. 543–544. ISSN: 1476-4687. DOI: 10.1038/356543a0. URL: https://doi.org/10.1038/356543a0.

[29] Toby Lewis-Atwell, Piers A. Townsend, and Matthew N. Grayson. “Comparisons of different force fields in conformational analysis and searching of organic molecules: A review”. In: Tetrahedron 79 (2021), p. 131865. ISSN: 0040-4020. DOI: https://doi.org/10.1016/j.tet.2020.131865. URL: https://www.sciencedirect.com/science/article/pii/S0040402020311236.

[30] When literature et al. “AMBER 14, University of California, San Francisco”. In: (Mar. 2014). DOI: 10.13140/RG.2.2.17892.37766.

[31] Ian K. McDonald and Janet M. Thornton. “Satisfying Hydrogen Bonding Potential in Proteins”. In: Journal of Molecular Biology 238.5 (May 1994), pp. 777–793. DOI: 10.1006/jmbi.1994.1334. URL: https://doi.org/10.1006/jmbi.1994.1334.

[32] 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. DOI: 10.1016/j.bpj.2015.08.015.

[33] "Molecular dynamics characterisations of the Trp-cage folding mechanisms: In the absence and presence of water solvents". In: Molecular Simulation 38 (2 Feb. 2012), pp. 161–171.

[34] Giannis Nikolentzos, Giannis Siglidis, and Michalis Vazirgiannis. "Graph Kernels: A Survey". In: (Apr. 2019).

[35] Yang Ning et al. "Chapter 11 - Pore-Scale Modeling and Simulation in Shale Gas Formations". In: Petrophysical Characterization and Fluids Transport in Unconventional Re Ed. by Jianchao Cai and Xiangyun Hu. Elsevier, 2019, pp. 217–246. ISBN: 978-0-12-816698-7.

[36] Dietmar Paschek, Sascha Hempel, and Angel E. García. "Computing the stability diagram of the Trp-cage miniprotein". In: Proceedings of the National Academy of Sciences 105.46 (2008), pp. 17754–17759. DOI: 10.1073/pnas.0804775105. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.0804775105. URL: https://www.pnas.org/doi/abs/10.1073/pnas.0804775105.

[37] Massimiliano Pavan and Marcello Pelillo. Dominant Sets and Hierarchical Clustering. 2003.

[38] Damiano Piovesan, Giovanni Minervini, and Silvio C.E. Tosatto. "The RING 2.0 web server for high quality residue interaction networks". In: Nucleic Acids Research 44.W1 (May 2016), W367–W374. ISSN: 0305-1048. DOI: 10.1093/nar/gkw315. eprint: https://academic.oup.com/nar/article-pdf/44/W1/W367/7632571/gkw315.pdf. URL: https://doi.org/10.1093/nar/gkw315.

[39] Jay W Ponder and David A Case. FORCE FIELDS FOR PROTEIN SIMULATIONS. 2003.

[40] Jay W. Ponder and David A. Case. "Force Fields for Protein Simulations". In: Protein Simulations. Elsevier, 2003, pp. 27–85. DOI: 10.1016/s0065-3233(03)66002-x. URL: https://doi.org/10.1016/s0065-3233(03)66002-x.

[41] A R Priatama, https://journal.ipb.ac.id/index.php/jmht/article/view/33983/23160, and Y Setiawan. "Regression models for estimating aboveground biomass and stand volume using Landsat-based indices in post-mining area". In: J. Manaj. Hutan Trop. (J. Tr 28.1 (Apr. 2022), pp. 1–14.

[42] Judemir Ribeiro et al. "Calculation of accurate interatomic contact surface areas for the quantitative analysis of non-bonded molecular interactions". en. In: Bioinformatics 35.18 (Sept. 2019), pp. 3499–3501.

[43] Manuel De Lera Ruiz and Richard L. Kraus. Voltage-Gated Sodium Channels: Structure, Functio Sept. 2015. DOI: 10.1021/jm501981g.

[44] Giannis Siglidis et al. GraKeL: A Graph Kernel Library in Python. 2018. DOI: 10.48550/ARXIV.1806.02193. URL: https://arxiv.org/abs/1806.02193.

[45] "Simulation Studies of Ion Permeation and Selectivity in Voltage-Gated Sodium Channels". In: Current Topics in Membranes 78 (2016), pp. 215–260. ISSN: 10635823. DOI: 10.1016/bs.ctm.2016.07.005.

[46] Relatore Dott Alvise Spanó and Laureando Davide Pizzolato. Residue interaction network maker 2021.

[47] Maarten van. Steen. Graph theory and complex networks : an introduction. Maarten van Steen, 2010, p. 285. ISBN: 9789081540612.

[48] Gabriel Studer, Marco Biasini, and Torsten Schwede. "Assessing the local structural quality of transmembrane protein models using statistical potentials (QMEANBrane)". In: Bioinformatics 30.17 (Aug. 2014), pp. i505–i511. DOI: 10.1093/bioinformatics/btu457. URL: https://doi.org/10.1093/bioinformatics/btu457.

[49] Gabriel Studer et al. "ProMod3—A versatile homology modelling toolbox". In: PLOS Computational Biology 17 (Jan. 2021), pp. 1–18. URL: https://doi.org/10.1371/journal.pcbi.1008667.

[50] A. P. Thompson et al. "LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales". In: Comp. Phys. Comm. 271 (2022), p. 108171. DOI: 10.1016/j.cpc.2021.108171.

[51] Alberto Toffano et al. "Computational Pipeline to probe NaV1.7 gain-of-functions variants in neuropathic painful syndromes". In: (Oct. 2020).

[52] SV N Vishwanathan, Nicol N Schraudolph JMLR, and Risi Kondor RISI. Graph Kernels Karsten M. Borgwardt. 2010, pp. 1201–1242.

[53] Guoli Wang and Jr Dunbrack Roland L. "PISCES: a protein sequence culling server". In: Bioinformatics 19.12 (Aug. 2003), pp. 1589–1591. DOI: 10.1093/bioinformatics/btg224.

[54] Andrew Waterhouse et al. "SWISS-MODEL: homology modelling of protein structures and complexes". In: Nucleic Acids Research 46.W1 (May 2018), W296–W303. ISSN: 0305-1048. eprint: https://academic.oup.com/nar/article-pdf/46/W1/W296/25110428/gky427.pdf.

[55] Adam Zemla et al. Processing and Analysis of CASP3 Protein Structure Predictions. URL: http://PredictionCenter.llnl.gov.

[56] Jian Zhang, Yu Liang, and Yang Zhang. "Atomic-level protein structure refinement using fragment-guided molecular dynamics conformation sampling". en. In: Structure 19.12 (Dec. 2011), pp. 1784–1795.

[57] Y. Zhang. "TM-align: a protein structure alignment algorithm based on the TM-score". In: Nucleic Acids Research 33.7 (Apr. 2005), pp. 2302–2309. DOI: 10.1093/nar/gki524. URL: https://doi.org/10.1093/nar/gki524.

[58] Yang Zhang and Jeffrey Skolnick. "Scoring function for automated assessment of protein structure template quality". In: Proteins: Structure, Function and Genetics 57 (4 Dec. 2004), pp. 702–710. ISSN: 08873585. DOI: 10.1002/prot.20264.

[59] Ruhong Zhou. Trp-cage: Folding free energy landscape in explicit water. 2003.