eman ta zabal zazu
Universidad del País Vasco
Euskal Herriko Unibertsitatea
ZIENTZIA ETA TEKNOLOGIA FAKULTATEA
FACULTAD DE CIENCIA Y TECNOLOGÍA

Bachelor Thesis
Degree in physics

Computational analysis of 12 mutations in the potassium channel KCNQ2

Author:
Amaia Razquin Lizarraga
Directors:
Dr. Aitor Bergara
Dr. Aritz Leonardo

Leioa, 2021eko ekainaren 17a /Leioa, 17 de junio de 2021

Contents

1 Introduction

Benign familial neonatal seizures (BFNS) is a rare disease with an incidence of 1 in 100,000 people that affects newborn children in their first year of life. It is characterised by an early onset of seizures (starting within the first three days after birth) that gradually disappear. Studies show that it is very often linked to mutations in the Kv7.2 potassium channel located in the membrane of neuronal cells, but it is yet unknown what are the specific processes that cause the disease [32].

In this work we used a computational approach to compute the stability of a series of 12 mutations in the Kv7.2 channel sent by neurology research doctors in the Sant Joan de Déu hospital in Barcelona. The stability was computed considering the energetic penalization of introducing the mutation. To avoid biases we did not know if all of these mutations were found in patients with the mentioned phenotype or if there were any control subjects, neither any other information about the patients. We did our analysis using Rosetta software [20], a macromolecular modeling suite that evaluates the physical plausibility of biological macromolecules, such as the protein we are analysing. Within Rosetta we chose the MPddG package because it is designed to calculate the change in stability upon mutation in the specific case of proteins that are located in the membrane, like Kv7.2. With the aim of gaining accuracy we tried to enhance its prediction power by slightly modifying the algorithm, and finally we compared the results with the ones given by the Flex_ddG package. This package performs the same calculation with a more computationally expensive protocol to get more accurate results, but it is not able to consider the effects of the membrane.

Therefore, after introducing the biophysics involved in this study, and the physical approach taken, we will dive into the Rosetta software, how does it work and how could it be applied to the subject we worked on. We also used this chance to evaluate Rosetta's performance in the analysis of stability of mutations where no experimental analysis had been done before, so future users know which are its strong and lacking points. Finally, we analysed the results given by the programs we run, trying to give a physical explanation why certain mutations were considered stabilizing and others destabilizing.

Due to the complexity of biological processes it is a fairly difficult task to experimentally prove theoretical hypothesis. Computational modeling provides a way to systematically analyse biophysical systems both to predict results and interpret experimental data, extending our knowledge of the laws that govern the macromolecular world. Even if in recent years technological advances have made work in laboratories easier, computational biophysics is being established as a partner to experiments and a widely used tool. For the development of efficient algorithms it is crucial to test them in the most varied problems so that they can be improved. Thus, in the following pages we try to understand, use and get results of the biophysical software Rosetta for mutations in a potassium channel that result in BFNS.

2 Brief introduction to membrane proteins

The basic architecture of any amino acid is a central tetrahedral carbon atom (C_α) that is attached to a hydrogen atom, an amino group (NH_2) and a carboxyl group (COOH) (Fig. 1). Each amino acid is distinguished from the others by the side-chain attached to the remaining valence of the C_α. The DNA encodes 20 different side chains from which the canonical 20 amino acids are created [1].

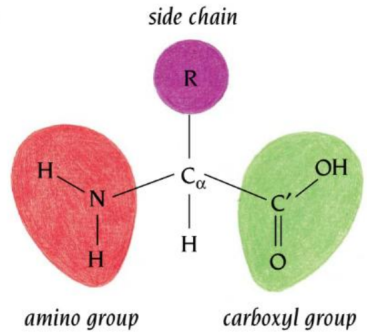

Figure 1: Basic structure of an amino acid [7].

Amino acids are commonly classified in three different classes considering the chemical nature of their side chains: hydrophobic, charged and polar. Hydrophobic amino acids are non-polar and in consequence do not interact with water, while charged and polar tend to be hydrophilic. However, the electrical properties vary in solution due to the possible different pH values.

A protein is a polypeptide chain, meaning that it is formed by amino acids joined by the formation of *peptide bonds* (Fig. 2). This is done when the carboxyl group of one amino acid covalently bonds to the amino group of another amino acid, losing in the process a water molecule and forming a peptide bond. The chain is elongated repeating the same process until the whole protein is formed. As a consequence, the first amino acid keeps its amino group intact and the last amino acid keeps its carboxyl group, the chain then is said to extend from its amino terminus (N-terminus) to its carboxyl terminus (C-terminus) [7].

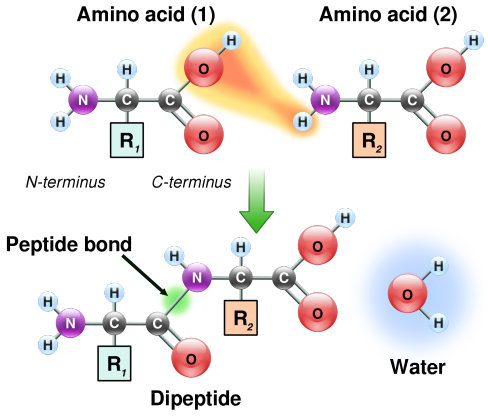

Figure 2: Formation of a peptide bond between to amino acids releasing a water molecule. Source: en.wikipedia.org/wiki/Peptide_bond.

Thus, a protein has a "main chain" or "backbone" consisting of the succession of the sequence NH-C_αH-C′=O, where C′ is the carbon of the carboxyl group, of each amino acid and the various side chains are projected from there. The peptide bonds are effectively rigid groups, so the degrees of freedom of the backbone are the rotations around the C_α-C′ and the N-C_α bonds. The angles these rotations make are called ϕ and ψ for N-C_α bonds and C_α-C′ bonds, respectively. The remaining degrees of freedom of the protein structure come from the conformations that side-chains can acquire by rotations of the bonds between carbon atoms, these are called torsion angles and referred as χ. Since some conformations are considerably more energetically favorable than others, most side-chains have few conformations that occur more frequently, called *rotamers* [33].

For the polypeptide chain to become a biologically functional protein it must fold into a specific three-dimensional conformation. The protein synthesis and folding is char-

acterised in four orders of protein structure: primary, secondary, tertiary and quaternary structure:

Primary structure refers to the sequence of amino acids in a polypeptide chain, while **secondary structure** considers the folding of short contiguous segments into geometrically ordered units. This happens when series of residues[1] adopt similar ϕ and ψ angles, the most known secondary structures are the *α-helix* and *β-sheet*. The first one is characterised by a twisted backbone containing an average of 3.6 amino acids per turn with the side-chains facing outward; this structure has additional stability due to the hydrogen bonds created between the oxygen of the carbonyl and the hydrogen of the amide group of the peptide bonds of the fourth amino acid down the chain (Fig. 3). β-sheets are characterised by a zigzag pattern where the side-chains of adjacent residues face in opposite directions; this structure is also more stable due to the hydrogen bonds between the carbonyl oxygens and amide hydrogens in the peptide bonds. Other secondary structure are loops, turns and bends (see Fig. 3).

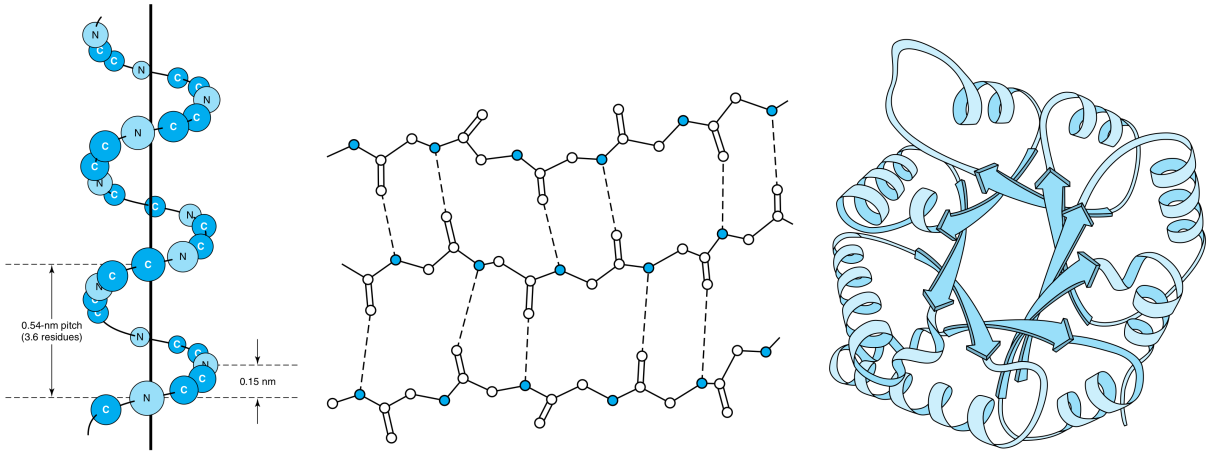

Figure 3: Left: orientation of the backbone atoms in a α-helix. Center: three strands of polypeptide chains linked by hydrogen bonds (dotted lines) to form a β-sheet. Right: tertiary structure of and enzyme (triose phosphate isomerase); arrows represent β-sheets and coils α-helices [28].

Tertiary structure refers to how the secondary structure features assemble to form *domains*, a region of a protein structure able to perform chemical or physical tasks (such as binding a ligand), and how different domains relate spatially to one another. Finally, **Quaternary structure** considers the polypeptide composition of a protein, taking into account if it is formed by a single polypeptide chain or many and if these are different or identical [28].

In humans approximately one-third of proteins are located in the cell membrane. The cell membrane is composed of two phospholipid layers with their hydrophobic tails facing each other creating a lipid bilayer. Phospholipids are mainly constructed from fatty acids and glycerol. Fatty acids are characterised by a long hydrocarbon chain (hydrophobic) and a carboxyl group that is ionized in solution. In phospholipids glycerol is joined to two

[1]In biochemistry amino acids are often also referred as residues as they loose both their amino and acid groups in the peptide bond.

fatty acid chains and a hydrophilic phosphate group that it is linked to some hydrophilic compound. Thus, the membrane bilayer is hydrophobic on the inner side and hydrophilic on the outer side (see Fig. 8).

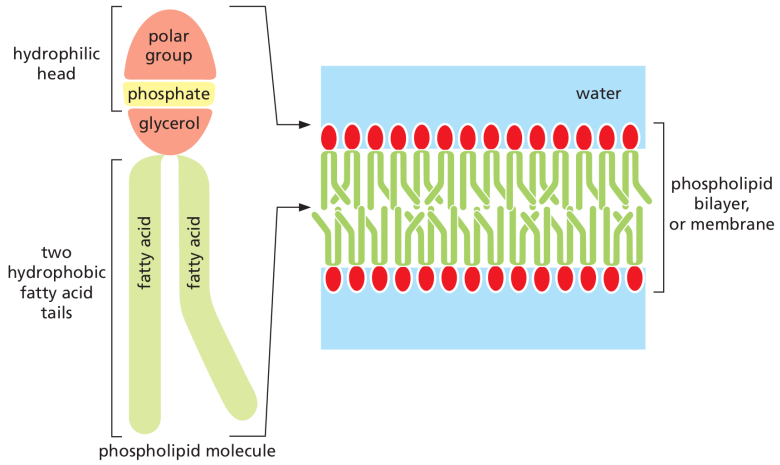

Figure 4: Composition of phospholipids and how they are positioned to form the membrane bilayer [1].

Given the properties of the membrane environment, membrane proteins have unique features. These proteins can be classified in two main groups: integral and peripheral. Integral proteins have one or more segments embedded in the phospholipid bilayer, usually containing hydrophobic groups that interact with the fatty groups of the phospholipids. On the other hand, peripheral proteins are bound to the membrane indirectly by integral proteins or directly by interactions with the polar heads of the phospholipids. Transmembrane (integral) proteins tend to have α-helix domains that span the membrane and interact hydrophobicly with the lipids of the membrane as well as by ionic interactions with the polar heads. β-sheets also appear in membrane proteins, often creating barrel like structures [22].

The functional ability of membrane proteins is very diverse. Furthermore, since they are located in the membrane, there are certain functions only them can fulfil. For example, these proteins allow transportation through the membrane bilayer and into or out of the cell; in addition, they are used for intracellular joining, connecting different cells, as well as attaching filaments and fibers that create the cytoeskeleton. They also accomplish cell recognition which is crucial in the immune system [10].

Throughout this work, all of the visualizations of proteins are done with VMD [14] and for simplicity we will only use the three visual representation methods in Fig. 5. The first two will be useful for understanding the spatial configuration of the protein, while the third one shows the secondary structure (arrows indicate β-sheets while coils are α-helices). In addition, each amino acid has a unique name and a one-letter and a three-letter code for identification. For simplicity, each relevant amino acid and its name will be described when considered necessary and only the one-letter code will be used.

Moreover, to refer to the mutations we will use the AxxxB nomenclature where xxx

[1]Color-code for amino acid types in VMD is white: nonpolar, green: polar, blue: basic, red: acidic.

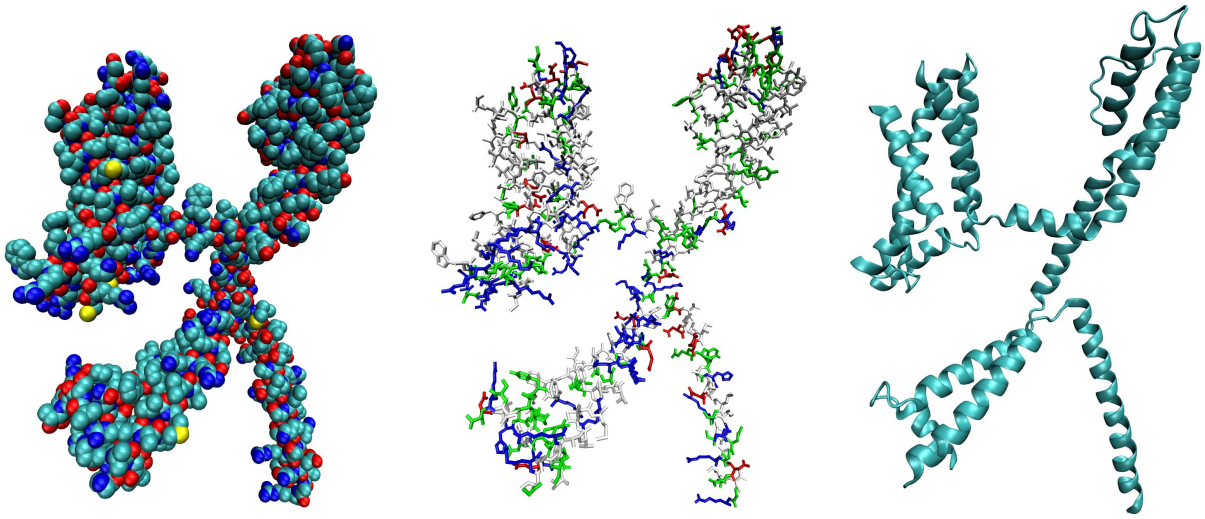

Figure 5: Visual representation of a single monomer of Kv7.2 using a representation where each atom's Van der Waals radius is shown and each element has a different colour (left); a representation where the bonds between atoms are shown as cylinders and each type of amino acid has a different colour (center); and a representation where the secondary structure is shown (right). All three created with VMD.

is the position in the sequence of the mutation, A is the original amino acid and B is the new amino acid after the mutation (both using the one-letter code). As an example, the change of the arginine (R) in position 213 to a tryptophan (W) will be called R213W. All the mutations we will analyse here are substitutions of a single amino acid by another.

2.1 Potassium channel Kv7.2

Potassium channels are proteins located in the cell membrane and regulate the transfer of potassium ions in and out of the cell. In consequence, they control cell volume, proliferation, differentiation and survival, with a primary roll in intrinsic electrical properties in excitable cells. This very diverse functional ability comes from the remarkably heterogeneous genetic and structure of the K^+ channel family.

Kv7.2 is a voltage-gated (Kv) K^+ channel mainly present in neurons and skeletal muscle cells of the Kv7 family. Within this same family (encoded by the KCNQ genes), there are other four potassium channels (Kv7.1-5). Kv7.1 is mainly expressed in cardiac cells, while Kv7.3-5 are distributed in neuronal and primary sensory cells.

Unlike voltage-gated Na^+ or Ca^+ channels, K^+ channels are not translated from a single gene, but they are made up of four identical or compatible monomers that come together in the membrane to form the pore (see Fig. 6) [29]. This enhances the diversity of the channels, since not only can they be built by four identical subunits (say, Kv7.2) but they can also mix different subunits (i.e. two Kv7.2 and two Kv7.3 monomers) to form the desired tetramer. This combinations do not happen between all of the family subunits, and can only be arranged between compatible monomers.

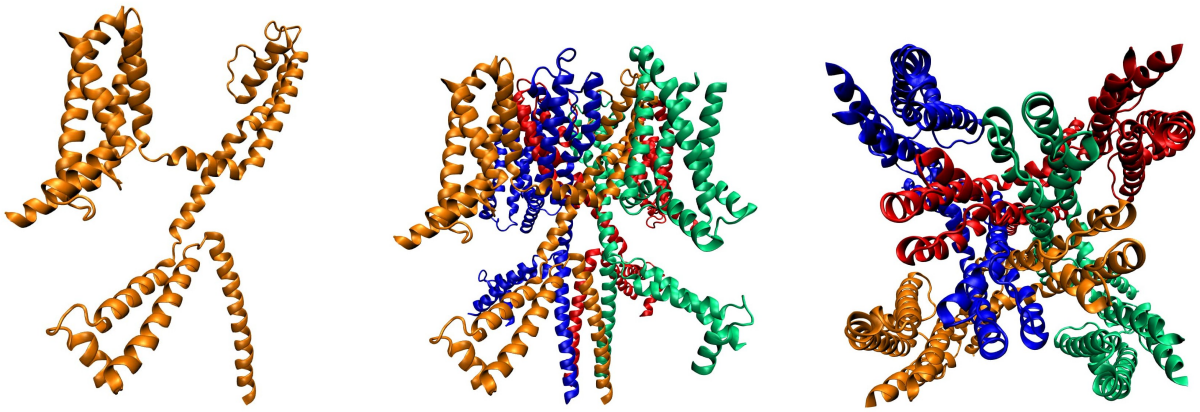

Figure 6: Left: monomer of the potassium channel KCNQ2; center: full Kv7.2 channel built with four monomers; right: upper view of the channel showing the pore in the middle through which ions are transported.

Each monomer in the Kv7 channels is topologically arranged with six transmembrane segments (S1-S6) with a pore loop between the last two segments. The region from S1 to S4 forms the voltage-sensing domain (VSD), whereas the ion-selective pore is between the last two segments and form the outer half of the K^+ channel pore. The S4 segment is thought to have a major role in voltage sensing since it is built with a distribution of four (for KCNQ1) to six positively charged arginines separated by two to three uncharged residues [35]. The channel pore contains the selectivity signature T/SxxTxGYG amino-acid sequence (Fig. 7).

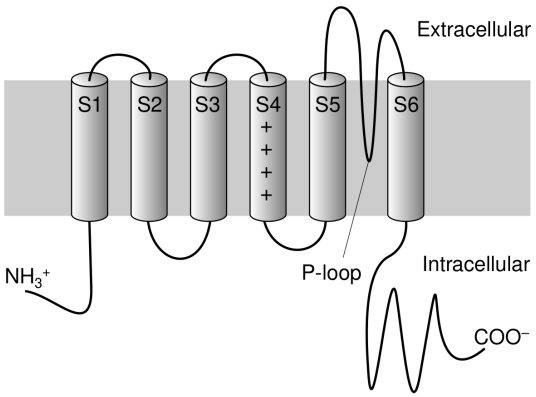

Figure 7: KCNQ channel structure: the transmembrane segments of a monomer are shown, the gray rectangle represents the membrane bilayer. For KCNQ2 the P-loop contains the selectivity sequence TxxTxGYGxxY. [30].

To properly comprehend the relevancy of the KCNQ family, we first need to understand how potassium channels work. In the particular case of neurons, if we take a cross section of the axion, the inner part of the membrane bilayer is negatively charged while the outer part is positively charged, resulting in a resting potential of -70mV. This potential is sustained by electrochemical gradients of different ions that come in and out through leak channels and bumps in the membrane.

When the membrane gradient rises to -55mV the sodium voltage-gated channels open up, letting a noticeably big amount of Na^+ ions into the cell. The potential in consequence goes even higher (depolarization) and when it reaches 30mV, potassium channels open up in order to get K^+ ions out of the cell to restore the electrochemical balance (repolarization). At this stage there is an undershoot of the membrane potential, which goes lower than -70mV, but the resting potential is recovered by the aforementioned membrane permeability. This change in potential is called action potential (AP) and it is carried along the axion of the neuron by local changes, i.e. a potassium channel that has been activated by the input of sodium ions by the previous channel will heighten the voltage in the vicinity of the next sodium channel provoking a new input of sodium, and so on.

The four members of the Kv7 family that are expressed in the nervous system form subunits of the originally termed "M-channel". This channels activate at subthreshold potentials, around -60mV, and induces outwardly rectifying currents with little or no inactivation. Since the activation is relatively slow (tens of milliseconds) they do not contribute materially to the repolarization of individual action potentials, but they have significant dampening effects in neuron excitability. Thus, they assist in stabilizing the membrane potential when depolarizing currents are present and contribute to the resting potential [8].

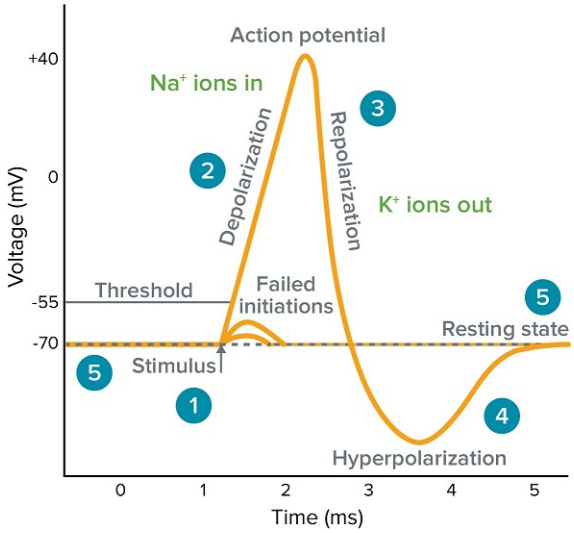

Figure 8: Change in voltage of an action potential. Source: www.moleculardevices.com.

The KCNQ gene family is one of the first K^+ channel families where mutations have been directly linked to human diseases and most of the expressed family of channel genes may have a clear physiological correlate. Benign familial neonatal seizures (BFNS) is a rare disease with an incidence of 1 in 100,000 people that has been found to be caused by mutations mostly in KCNQ2 (Kv7.2) and more rarely in KCNQ3 (Kv7.3) genes. It is characterised by an early onset of generalised seizures (starting a few days after birth) that disappear after the first months of life, being mostly non-appearing after a year. Some patients (around 16%), however, display seizures later in life. The mutations in these channels have mostly a very subtle effect on the functionality of the protein. Studies show that there is no apparent alteration in the ionic selectivity of the gating, and the total current is reduced by only 20-30%. This reduction in the postnatal brain development is critical but it gets less crucial later in life [35].

Furthermore, mutations in Kv7.2 have also been found in patients with a more severe disease called KCNQ2 enthephalopathy (EOEE). This patients have a similar onset with seizures early in life, being more frequent in BFNS, and the main discrepancy in diagnoses lies in the suppression-burst pattern on the electroencephalography (EEG). Poor outcomes in EOEE include severe development delay [17].

3 Physical approach

In physics, the natural procedure to solve a problem is through identifying the components of the system and deriving the Hamiltonian by analysing the interactions between the "particles". For that purpose the potential energy of the interaction of a particle with every other particle needs to be considered. By summing this contributions we get the total potential of the system as:

$$V(\boldsymbol{r}_1, ..., \boldsymbol{r}_N) = \sum_{i}^{N} \sum_{j>i}^{N} \phi(|\boldsymbol{r}_1 - \boldsymbol{r}_j|) \tag{1}$$

where $\boldsymbol{r}_i$ are the coordinates of particle i, and N is the total number of particles.

Proteins can have from a couple hundred atoms to hundred thousands, which makes that approach unfeasible. With the development of faster and more powerful computers and algorithms the dream of modeling biomolecules in detail has come true. However, nowadays it is still very time consuming and not viable for most computers and biological systems. Particularly, for protein simulations to be realistic they should be surrounded by water molecules, which would exponentially increase the number of particles. Thus, many modeling programs use different approximations to simulate as accurately as possible the biological systems.

Rosetta uses a "score function" to approximate the energy of the system. This is based on the hypothesis that native conformations[2] are low energy thermodynamically stable conformations. These conformations are called folded states and they can be energetically compared to the unfolded states, where the tertiary structure is not formed and the components of the protein are spread. Therefore, the folded states correspond to minima in the energy gradient and they have a net favorable change in Gibbs free energy with respect to the unfolded states: $\Delta G = \Delta H + T\Delta S$.

The score function is a set of different energy terms (explained in section 4.2.1.) weighted and summed to fit experimental results representing the change in Gibbs free energy [3]. In the particular case of membrane proteins this change of Gibbs free energy is the energy contribution of each amino acid isolated in solution (water) minus the energy contribution when they are embedded in the membrane with the rest of the conformation.

Since we are working with the stability of mutations, our focus is in the change of the energy when we produce the mutation. Consequently, what we are going to calculate is the variation of the change in Gibbs free energy, namely $\Delta\Delta G$. The lower the energy is the more stable the system is and since $\Delta\Delta G$ is defined as

$$\Delta\Delta G = \Delta G_{mutant} - \Delta G_{native} \tag{2}$$

a mutation is considered to be destabilizing when $\Delta\Delta G$ is positive, because the native conformation has a lower score than the mutant conformation ($\Delta G_{mutant} > \Delta G_{native}$) and stabilizing when it is negative, because $\Delta G_{mutant} < \Delta G_{native}$.

[2]Configurations that are found in nature.

4 Computational tool: Rosetta

4.1 Overview and architecture

Rosetta is a software suite for macromolecular modeling mainly written in C++ where the evaluation of the physical plausibility of biological macromolecules such as proteins, nucleic acids and ligands, is performed. For this purpose the software contains different protocols that are appropriate for evaluation of specific environments. It uses an object-oriented software design, encapsulating the individual concepts for biomolecule modeling into software objects. This objects contain specific data and methods to represent the concepts needed.

The main object where the biomolecule is stored is called Pose. Withing the Pose, the class Conformation contains three objects that altogether define the protein's physical and chemical properties. Firstly, the Residue object stores the coordinate information for each individual residue, and it points to the ResidueType, which contains the information on the chemical connectivity of a single abstract residue under one specific name. Lastly, the AtomTree stores the information on how internal coordinate changes propagate through the system. To achieve this the general way of doing it is by generating a tree[3] whose nodes represents the atoms of the molecule and whose edges represents kinematic connections. To keep track of the changes in the tree, residues are "colored": two residues that have not moved with respect to each other will have the same color.

As previously mentioned, Rosetta relays on the score function to compute the energy of the system. To manage this information the Energy object was designed, where the most recent score function evaluation is stored in an EnergyGraph (using Graph Theory technology). Each vertex in the EnergyGraph represent a residue in the Pose and each edge represents an interaction between two residues. The scoring procedure starts by the score function iteration across all edges of the EnergyGraph and it updates them considering the color of the nodes. After that, it detects residue neighbours and iterates across them creating new edges for any pair of residue with non-matching colors (see AtomTree). The energetic calculus in the Rosetta environment is done in Rosetta Energy Units, which has no direct interpretation as real physical energy but it is useful for modeling changes in energy. Recently, it has been said that it fairly approximates units of kcal/mol [3].

Finally, Movers are used to interact with a pose either to change it, reference it or analyze it. Thus, Movers are used to build the protocols in Rosetta, which can be seen as processes where an input Pose is taken, treated by the Movers and an output Pose is given. For example, the Monte Carlo technique is a mover of its own. Other objects, classes and protocols are also designed in Rosetta.

For analyzing proteins in the transmembrane region of cells the RosettaMP (Rosetta Membrane Protein) framework was developed, containing the specific physical and chemical characteristics of the membrane environment. It uses the same object-oriented design as Rosetta but expanding it by creating new objects to represent the membrane environment and adding scoring and sampling routines that account for the lipid bilayer.

[3] "Tree" as in graph theory.

The information about the membrane is stored in a classed called MembraneInfo that is part of the Conformation object inside the Pose. It attaches a "virtual" residue to the Pose that represents the membrane bilayer chemistry and geometry. For that it uses three virtual atoms that define the membrane center, normal vector and thickness. It also stores the information about the transmembrane spans of the protein (normally α-helices or β-sheets) so that the biomolecule can be correctly placed in the membrane. By default, the membrane thickness is chosen to be 30Å, including both the hydrophobic membrane core and the membrane-water interface. As any other residue, the membrane can be fixed or movable during modeling [4].

Lastly, Rosetta contains several servers and interfaces for different types of users. The most well known ones are PyRosetta [9], written in Python, and RosettaScripts [12], using XML format. We mainly used PyRosetta since the MPddG package was developed in Python, but Flex_ddG is scripted in RosettaScripts so it was also used.

4.2 MPddG Package

Since the main goal of this work is to analyze mutations in order to know which ones destabilize the system, the Rosetta package that we used is the one that revolves around the calculation of $\Delta\Delta G$: MPddG. As previously explained, $\Delta\Delta G$ is the change of the variation of Gibbs free energy of the molecule upon mutation, thus it is a measure of the thermodynamic cost of the substitution. In the particular case of membrane proteins, the ΔG is calculated as the energy change between the separated residues in solution (water) and the whole protein in the membrane. The lower the energy, G, the more stable the amino acid is in that environment.

Furthermore, since we need information about the change of energy after mutations, the package will give the difference of the change in ΔG between the native molecule and the mutant molecule, $\Delta\Delta G$. If this value is negative the mutation applied stabilizes the molecule since it would mean that the new amino acid generates an environment with lower free energy than the native configuration and in consequence it is more stable. On the other hand if $\Delta\Delta G$ is positive the mutation destabilizes the molecule. Since experiments have a certain error, we modified the range of stabilizing and the destabilizing $\Delta\Delta G$. For that purpose we used Khatun *et al.* work [18] as reference, the maximum experimental error for $\Delta\Delta G$ that they derived was of 0.48kcal/mol and since Rosetta Energy Units are said to be compared to kcal/mol, we considered a mutation to be destabilizing when $\Delta\Delta G > 0.5$, stabilizing when $\Delta\Delta G < 0.5$ an neutral when $0.5 < \Delta\Delta G < 0.5$.

4.2.1 Score Function

The score function used in the original mpddG package is called *mp_framework_smooth_fa_2012* [37], a score function dedicated to membrane proteins with several terms where the membrane is explicitly considered. These terms are mainly knowledge based and take into account the position of a residue with respect to the membrane and other residues to benefit or penalize a certain structure.

This score function uses the default Rosetta score function for modeling soluble proteins *ref2015* [3] as basis where the membrane terms are added. However, since the membrane terms date back to 2012 we searched for a newer version that could also take into consideration the distinct effects of a molecule in a lipidic bilayer. We found the recently designed score function *franklin2019* by Alford *et al.* [2] and chose it as our to-go energy function. This score function is also based in *ref2015* with one added term. We are intending to describe the package and not our own program, thus we will explain here the default score function *ref2015* and in section 4.3.1 we will go deeper into the new term.

As any other score function, the *ref2015* is computed as a linear combination of independent energy terms, E_i, that are a function of geometric degrees of freedom (Θ) and chemical identities (aa), scaled by individual weights (w_i) chosen to best approximate real energies.

$$\Delta E_{TOT} = \sum_i w_i E_i(\Theta_i, aa_i) \tag{3}$$

All the score terms and its weights are shown in Table 1 and can be found published in Alford *et al.* work [3]. In this section we will focus on the non-zero scores that were most relevant in our later analysis.

Firstly, we will explain the energy terms that consider atom-pair interactions within the molecule. **Van der Waals** interactions are short-range forces that vary with the distance between atoms. These can be attractive (due to cross-correlated motion of electrons in neighboring non-bonded atoms) or repulsive (caused by the Pauli exclusion principle). Rosetta models this interaction using the Lennard-Jones 6-12 potential [16], which calculates said interaction between atoms i and j using the atom radii, $\sigma_{i,j}$, distance between atoms, $d_{i,j}$, and the geometric mean of well depths, $\epsilon_{i,j}$, as:

$$E_{VdW}(i,j) = \epsilon_{i,j}\left[\left(\frac{\sigma_{i,j}}{d_{i,j}}\right)^{12} - 2\left(\frac{\sigma_{i,j}}{d_{i,j}}\right)^{6}\right] \tag{4}$$

Rosetta uses two different terms, `fa_atr` and `fa_rep`, to consider the Van der Waals atractive and repulsive forces, respectively. For that, the potential is split at its minimum, when $d_{i,j} = \sigma_{i,j}$, and both terms are weighted separately. However, this terms are designed to consider atoms in different residues, *ref2015* also contains a repulsive Van der Waals term for interactions between atoms in the same residue, `fa_intra_rep`.

For the **electrostatic** interactions that arise between non-bonded atoms that are partially or fully charged Coulomb's Law is used. This term is a function of the distance between both atoms, $d_{i,j}$, partial atomic charges, q_i, dielectric constant, ϵ, and Coulomb's constant, $C_0 = 322$ Åkcal/mole^{-2}. To better adapt the law to biomolecules, the dielectric constant is substituted by a more complex function that accounts for the difference between the protein core and solvent-exposed surface, $\epsilon(d_{i,j})$.

$$E_{Coulomb}(i,j) = \frac{C_0 q_i q_j}{\epsilon}\frac{1}{d_{i,j}} \tag{5}$$

It is known that protein conformations minimize the exposure of hydrophobic side chains to the surrounding polar solvent. To accurately consider this it would be needed to model the interactions between solvent and protein atoms, but as it is computationally expensive, Rosetta uses the Lazaridis-Karplus implicit Gaussian exclusion model to represent the solvent as bulk water [19]. The core of this model resides in the energy required to desolvate (remove contacting water) atom i when atom j approaches it.

Rosetta considers three different terms regarding **solvation**. The first of them is `fa_sol`, that assumes that bulk water is uniformly distributed around atoms and its computed as a weighted sum that includes atom i desolvating atom j and vice-versa. The second of them is the intra residue version of this one, `fa_intra_sol`, and the third one is `lk_ball_wtd` that accounts for specific waters nearby polar atoms that form the solvation shell. This term essentially increases the desolvation penalty when ideal water sites where hydrogen bonds may have formed are occluded to polar atoms. When this three terms are considered together and there is no occlusion of polar atoms the penalty comes solely from the `fa_sol` term.

To consider the contribution of **hydrogen bonds** to the conformational energy *ref2015* considers five different terms, from which one of them is the previously explained `fa_elec`, Coulomb's electrostatic term, and the remaining four evaluate the energies based on the orientation preferences of hydrogen bonds found in crystal structures. This energy is then separated in four terms: `hbond_ls_bb`, long range backbone hydrogen bonds; `hbond_sr_bb`, short range backbone hydrogen bonds; `hbond_bb_sc`, hydrogen bonds between backbone and side chain atoms; `hbond_sc`, hydrogen bonds between side chain atoms.

For protein backbone and side chain torsions, the energy function has three score terms. Rosetta uses knowledge-based terms for the angles to more accurately model the preferred conformations. For the backbone ϕ and ψ angles the term `rama_prepro`, based on **Ramachandran** maps[4], computes the energies of each configuration by inverting the probabilities with the inverted Boltzmann relation. It also considers the difference in the probability in the specific case of an amino acid when it is located before a proline.

Furthermore, *ref2015* takes into account the likelihood of an amino acid side chain being placed given a certain ϕ, ψ **backbone conformation**. The term `p_aa_pp` shows the propensity of observing one amino acid relative to the other 19 canonical amino acids. For that, the propensity, $P(\phi, \psi|aa)$, was derived using adaptive kernel density estimates and Bayes' rule:

$$E_{p\_aa\_pp} = \sum_r -\ln \frac{P(aa_r|\phi_r, \psi_r)}{P(aa_r)} \tag{6}$$

Protein **side chains** mostly occupy discrete conformations, called rotamers, that are separated by large energy barriers. To evaluate the contribution of the rotamers, Rosetta derives the probabilities from the backbone-dependent rotamer library [34]. There are three components of the probability to take into account: (1) given the backbone dihedral angles, observing a specific rotamer; (2) observing specific angles between the bonds of the carbon atoms, χ, given the rotamer, and (3) observing the terminal χ angle distribution.

[4]Maps where all the combinations that the dihedral angles can have in proteins are shown.

The logarithmic sum of this probabilities for each residues yields then the energy term `fa_dun`.

Energy Term	Weight	Function
fa_atr	1.0	Van der Waals atractive interaction from LJ potential
fa_rep	0.55	Van der Waals repulsive interaction from LJ potential
fa_intra_rep	0.005	Van der Waals repulsive term for atoms in the same residue
fa_elec	1.0	Coulomb electrostatic interaction
fa_sol	1.0	Considerss ability of an atom being solvated by the surrounding atoms
fa_intra_sol_xover4	1.0	Intra residue solvation term
lk_ball_wtd	1.0	Considers specific water around polar atoms forming the solvation shell
hbond_sr_bb	1.0	Long range backbone hydrogen bonds
hbond_lr_bb	1.0	Short range backbone hydrogen bonds
hbond_bb_sc	1.0	Hydrogen bonds between backbone and side chain atoms
hbond_sc	1.0	Hydrogen bonds between side chain atoms
dslf_fa13	1.25	Covalent bonds between sulfur atoms in cysteine
rama_prepro	0.45	Backbone torsions considering Ramachandran maps
fa_dun	0.7	Backbone dependent probability of finding a given amino-acid
omega	0.4	Peptide bond dihedral angle penalty
p_aa_pp	0.6	Propensity of locating one amino acid considering the surrounding amino acids
pro_close	1.25	Consideres proline's special case torsion
yhh_planarity	0.625	Consideres tyrosine's χ_3 angle
ref	1.0	Design reference term considering energy gap between folded and unfolded states

Table 1: Summary of all the score terms in *ref2015*.

Lastly, *ref2015* contains a **design reference term** (`ref`) dedicated to protein design to compare relative stability of different amino acid sequences given a desired structure to identify models that exhibit a large free energy gap between folded and unfolded states. Rosetta calculates the free energy of the unfolded state as a sum of individual constant unfolded state reference energies, ΔG_i^{ref}, that are empirically optimized to maximize native sequence recovery during design simulations.

4.2.2 Minimization

The original MPddG package contains a fixed backbone protocol when the mutation is introduced, being the only energy minimization process a side chain conformation optimization. This optimization is performed by taking the residues within 8Å of the mutated residue and trying different rotamers in a Monte Carlo process until the best scoring configuration is chosen by a Metropolis algorithm. The process is called "packing".

This is carried by the Packer mover that first evaluates the TaskOperations that controls which residues are packable, designable, or held fixed. By default, everything is able to pack and design. Then, the packer makes a list of all possible rotamers at each position and performs a precomputation where all possible pairs of interacting rotamers are listed and their pairwise interaction energies are calculated. These conformations are chosen from a library of possible rotamers that was developed by doing a Bayesian statistical analysis [11]. After that, simulated annealing is performed by choosing a random position and replacing the current rotamer with a randomly chosen one from the allowed ones at that position.

The Metropolis criterion is used to accept the new rotamers where if the new conformation has a lower energy than the previous one it is always accepted and if it has a higher energy it is accepted with a probability of $e^{-\frac{\Delta E}{k_B T}}$. The temperature factor $k_B T$ determines the likelihood of large increases of energy being accepted, meaning that the larger the factor the more frequently conformations with large increases in energy are accepted. Simulated annealing starts with high temperature factors to avoid getting stuck in local minima and explores the space more accurately and ramps it down throughout the simulation to get to the bottom of the lowest-energy well that it has found [31].

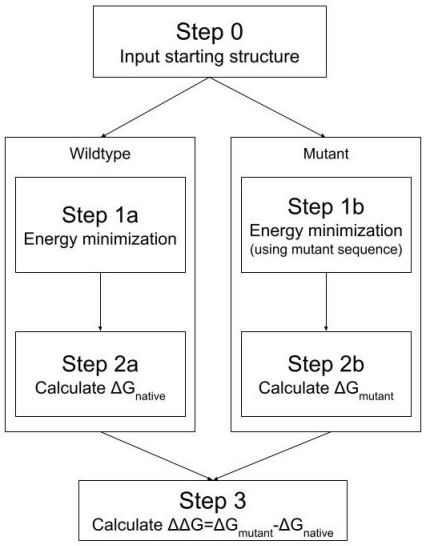

Figure 9: Process of the improved MPddG program.

4.2.3 Input preparation: PDB and span file

The MPddG program requires for two input files: a PDB and a span file. The PDB (or Protein Data Bank) file is a text file where the three dimensional structure of the protein is stored. In it the information is stored in lines which can take up to thousands of them. In the first lines information about the researchers that defined the structure and other facts about the molecule are written; they can also contain information on how to compute the coordinates and the list and number of amino acids in the molecule. Following, the "ATOM" lines fully describe in different columns the position of each atom of the molecule in three dimensions as well as other information such as in what amino acid they are, the occupancy, temperature factor and the element name. Finally, the "HETATM" lines add information of the atoms that are not part of the molecule itself but are useful to fully understand the system. The only essential lines in the PDB are the ATOM lines, but in our case the HETATM are also important since they store the positioning of the membrane.

The coordinates for the potassium channel Kv7.2 can be found in the PDB called 7CR3 [21] in RCSB (Research Collaboratory for Structural Bioinformatics), the US data center for the global PDB archive [6]. This PDB contains eight chains from which four are the monomers that form the channel, while the remaining four are the calmodulines

(a calcium binding messenger protein). To prepare the PDB for the package, PyRosetta contains the `clean_pdb` program that takes one chain of the protein and renumbers the residues as well as erases any atom that is not essential.

The span file is also a text file that contains information on how a molecule spans a membrane. The first line is left as a comment while the second one contains the number of predicted transmembrane helices and the total number of residues of the molecule. The third line is used to explain the topology of the helices as parallel or antiparallel and the rest of the file is used to describe the identification number of the first and last residue of each of the helices, writing the two numbers of each helix in a different line.

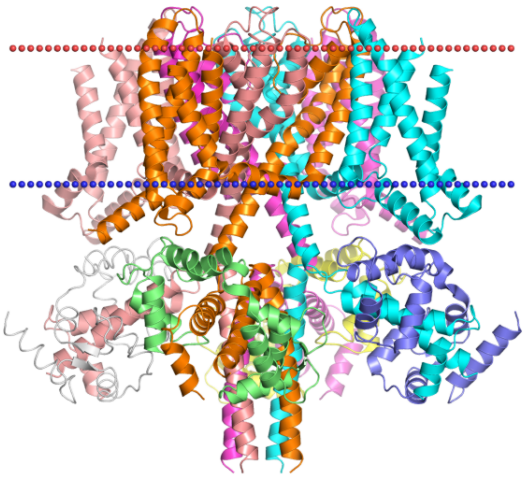

Figure 10: Potassium channel Kv7.2 in the membrane (red line: outer layer of the membrane, blue line: inner layer). In the intracellular part calmodulines are attached to each of the monomers [23].

For Kv7.2 we found this information in OPM, the database for Orientation of Proteins in the Membrane [23]. Simply put, these transmembrane spans are found computationally by considering the protein as a rigid body floating in a hydrophobic slab of adjustable thickness and calculating the transfer energies $\Delta G_{transfer}$ until the optimal positioning is found. Once we have the information, the span file can be manually created.

4.3 Improvements

As previously mentioned, the original MPddG program had two things that we wanted to improve. On the one hand, the membrane terms of the score function dated back to 2012 so we searched for a more updated one, *franklin2019*. On the other hand, when the mutation was introduced, the repacking step was the only energy minimization process and we found it insufficient. Thus, we tried to build an appropriate protocol that could adapt the structure better.

4.3.1 Score Function

The new term that the *franklin2019* score function added to *ref2015* is called `fa_water_to_bilayer` and is designed to take into account the membrane's effect in the conformation [2]. For that purpose, a set of water-to-bilayer transfer energies was derived for each atom type. This was done by taking the Moon and Fleming hydrophobicity scale [27] that provides the transfer energies for the 20 canonical amino acids and using regression (least-squares fitting) to get the corresponding energies to atom types, $\Delta G_{w,l}^{atom}$. After, all-atom molecular dynamics simulations were performed to get properties of membranes with different phospholipid compositions. With all the data captured, a water-density

analytic profile was designed as:

$$f_{thk} = \frac{1}{1 + \tau exp(-\kappa z)} \tag{7}$$

where z is the membrane depth and κ and τ are the steepness and width, respectively, and are derived for all simulated lipid compositions.

Additionally, a pore was introduced for proteins with more than three transmembrane segments. The transition between the water-filled pore and lipid phase is defined by the radius, g_{radius}, of the ellipse that bounds the coordinates of pore-facing atoms (previously computed) given the transition steepness, n.

$$f_{pore} = 1 - \frac{g^n_{radius}}{1 + g^n_{radius}} \tag{8}$$

To summarize, Alford *et al.* [2] modeled a biologically realistic implicit membrane model as a continuum of (1) an isotropic phase representing bulk lipids, (2) isotropic phase representing bulk water and (3) an anisotropic phase representing the interfacial region. The result is an energy term dependant on $\Delta G^{atom}_{w,l}$ and the fractional hydration, defined as $f_{hyd} = f_{thk} + f_{pore} - f_{thk} f_{pore}$. This last term is null when the atomic group is exposed to the lipid phase and 1 when it is exposed to the water phase.

$$\Delta G_{memb} = \sum_{r=1}^{N_{res}} \sum_{a=1}^{N_{atom}(r)} (1 - f_{hyd})(\Delta G^{atom}_{w,l}(a)) \tag{9}$$

This term is summed to the rest of the *ref2015* energy terms with a weight of 0.5.

4.3.2 Minimization

To improve the energy minimization of the structure we added a minimization process that is performed by the Minimizer in Rosetta, which calls for the minimization method of our choice. It is recommended to use the BFGS, the acronym stands for Broyden-Fletcher-Goldfarb-Shanno algorithm, an iterative method for solving unconstrained nonlinear optimization problems. It is a Quasi-Newton method where firstly a vector is chosen as descent direction and, after determining an appropriate step along that vector, it moves through the gradient. Then a new vector and step size are selected and the process is repeated. The second-derivative (Hessian) matrix is approximated and used to modify the descent step direction so that is no longer straight down the gradient, but results in a faster convergence.

The minimization used by default in Rosetta is the limited memory variant, L-BFGS. Additionally, the `Armijo` variant is used, which is an inexact line search version where the step along the search direction only needs to improve the energy by a certain amount flatter (but not necessarily reach the minimum). This is more efficient than the exact line search. Furthermore, the recommended version is the `nonmonotone`, an even less exact line search along the descent direction so that the step need only be better than one of the last few points visited. This allows temporary increases in energy so that the search

may escape shallow local minima and flat basins. Convergence is checked by the norm of the gradient, $|| \nabla f(\vec{x}_k) ||< tolerance$, and we set the tolerance to 10^{-6}. The maximum iterations of the minimization can also be set in case the tolerance threshold is not reached sufficiently soon, it is by default set to 200 iterations.

The energy minimization is carried out through the MinMover which essentially performs the minimization of the backbone and side chains. For this purpose the mover requires a MoveMap that defines which degrees of freedom are available to be minimized. The default setting of MoveMap depends on the protocol, but for the MinMover all backbone, χ, and jump degrees of freedom are allowed to change. However, this can be changed, we chose to set all χ and backbone angles movable for the residues within 8Å of the mutated residue by using the same calculation the original program had. The rest of angles and jumps were set not movable.

Lastly, we also added a structural change that was more suitable to the minimizer. The original program performed one repacking step in the input structure and that same structure was mutated suffering another repacking step. Since the packer is not very strong, the double repacking did not give any advantage to the mutated conformation. However, the minimizer is more exhaustive so the result was affected by the fact that the structure was minimized twice. Thus, we decided to perform the minimizations independently, one for each of the structures and then compare them, which yielded more logical results (see Fig. 9).

4.3.3 Benchmarking

To test if our changes were beneficial for the calculation we tried the modified program in two membrane proteins: palmitoyl transferase (PagP) and phospholipase A1 (OmpLA), shown in Fig. 11. We chose these two proteins because experimental values are

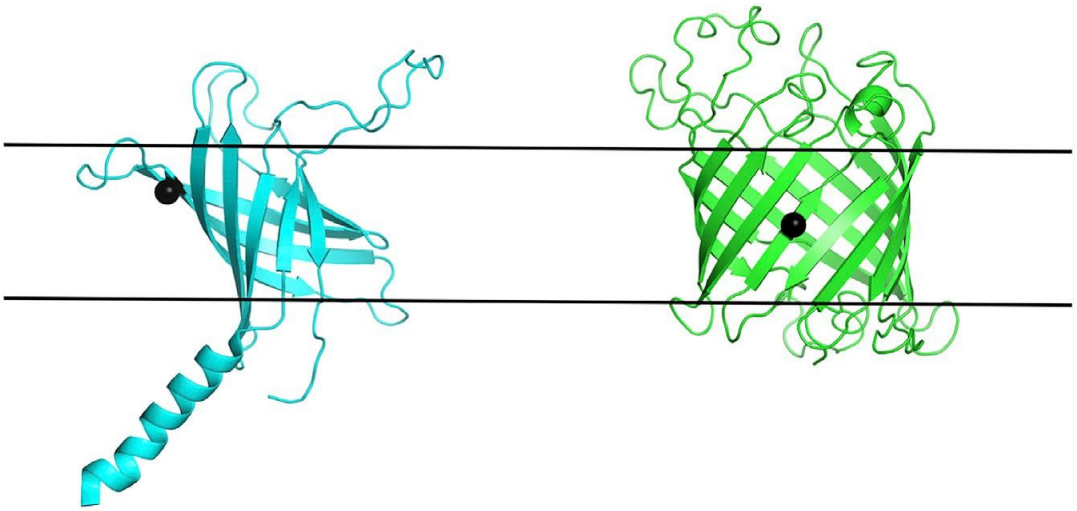

Figure 11: Proteins PagP (left) and OmpLA (right) embedded in the membrane (represented by two stright lines). Mutation sites are shown with a black sphere [24].

known [24][27]. We performed mutations in position 111 and 210 for PagP and OmpLA,

respectively, substituting the original amino acid for every other canonical amino acid.

The IDs of the needed PDBs are 1QD6 (OmpLA) and 3GP6 (PagP). The treatment for these files and the rest of input files and commands is the one described in section 4.2.3. We first run the original program, with the only energy minimization process being the repacking; and later we run our modified program containing the minimizer. The results and correlation with experimental values are shown in figure 12. For PagP results could not be taken straightforwardly because experimental results were described respect to the change of ΔG of the mutation to alanine (A) [24], considering this the value we used to make the comparison was $\Delta\Delta G$ subtracting the $\Delta\Delta G$ of the result for alanine.

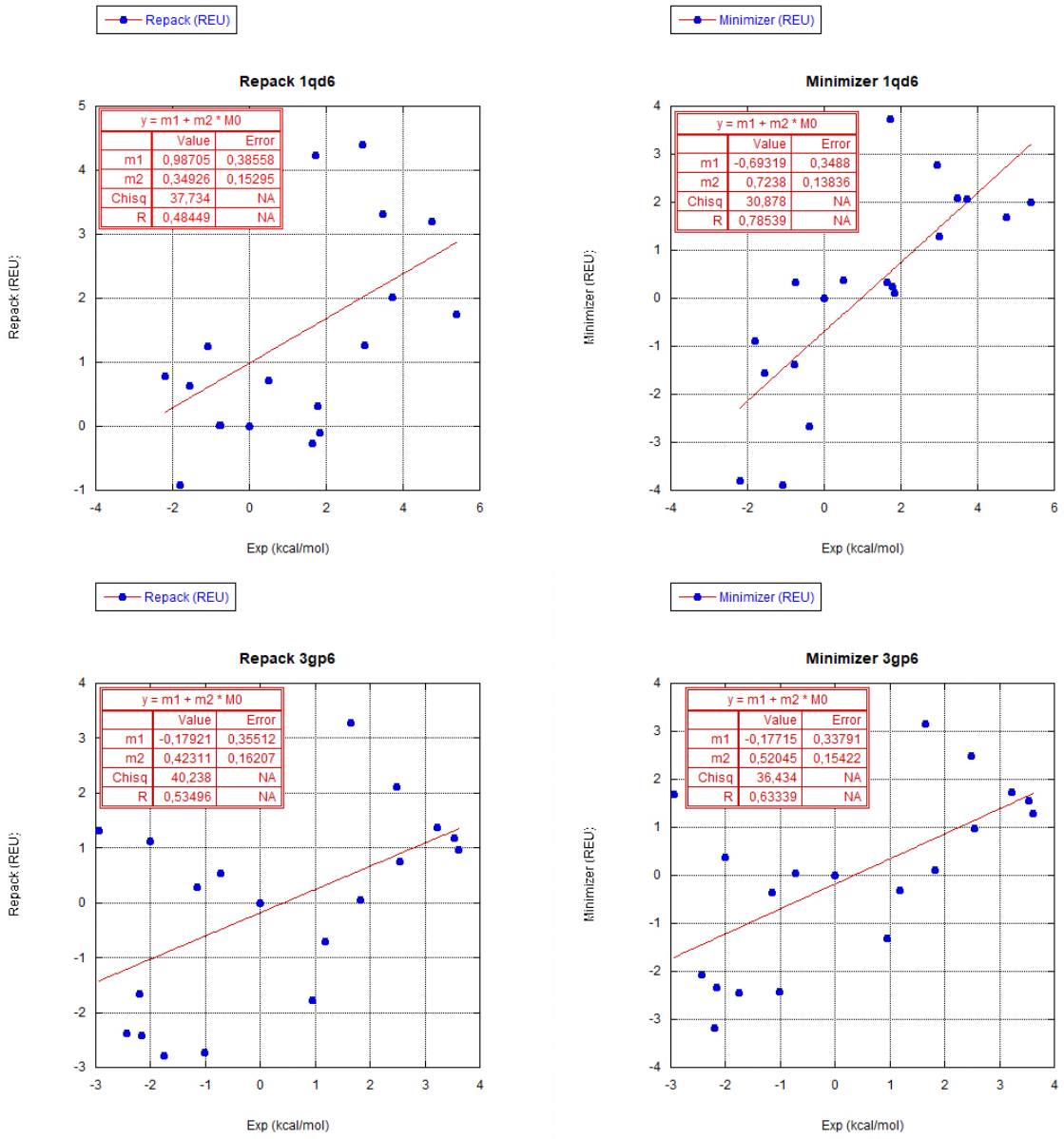

Figure 12: Correlation plots between Rosetta results and experimental values. Each dot represents a mutation and the red line is the fit.

The Pearson correlation coefficient (R) is considerably higher when using the mod-

ified program, which encouraged us to consider positively its results with the potassium channel.

5 Result analysis

5.1 Repack within 8Å from mutation

5.1.1 Monomer

Firstly, we will analyze the results we got from running the original program for a single monomer of the channel. Since the energy minimization method was only a repacking step in the vicinity of the mutations, most of the results showed an overestimation of the Van der Waals repulsive term. It can be seen in Table. 4 that `fa_rep` is one to two orders of magnitude bigger than the rest of the scores. This happens because the best conformations the packing can get are not realistic and probably have clashes between residues. However, we could get four mutations that can be analysed: E130K, R198Q, R213W and Y284D.

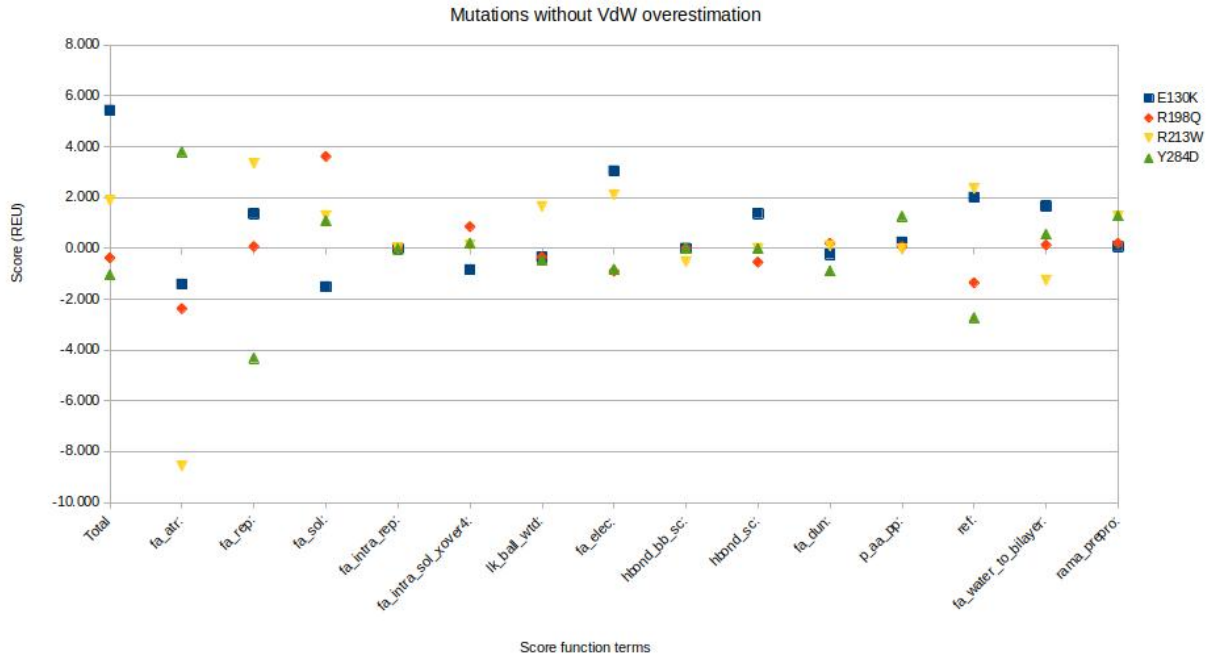

Figure 13: Non-zero scores given by the origial MPddG program for the mutations without Van der Waals energy term overestimation.

Therefore, we will begin our analysis by analyzing those mutations. Firstly, we will study E130K mutation, located in the S2 segment, which is a mutation from a negatively charged side chain, glutamic acid, to a positively charged one, lysine; thus it is a change from an acidic amino acid to a basic one.

The most relevant scores to the final results are `fa_elec`, `ref`, `fa_water_to_ bilayer` and `hbond_sc`. In the case of the Coulomb electrostatic interaction term (`fa_sol`) most probably the destabilization comes from the need for the lysine to bend away from the arginine in the segment S4. Because of this an its large nature it gets very close to the non-polar residues around, destabilizing the electrostatic equilibrium between them. Furthermore, that same arginine and the wildtype[5] (WT) glutamic acid generate an attractive Coulombian force that does not exist any more after the mutation. The reference term also plays a big role in determining that the mutation is destabilizing, this might be due to lysine having two more CH_2 groups distancing the C_α atom from the polar end; as the CH_2 groups are hydrophobic the reference term is more positive for the lysine. Additionally, the membrane term is positive, which is also related to the more hydrophobic nature of lysine compared to glutamine, because Rosetta models the four segments as a water filled pore.

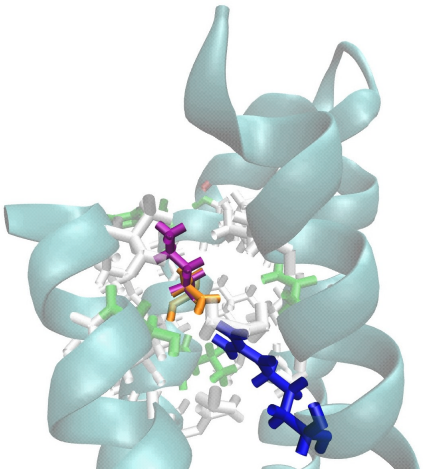

Figure 14: Visualization of mutation E130K with the original amino acid, E, in orange and the mutant, K, in purple. The rest of the surrounding residues are color coded by charge. The opaque blue residue is the arginine in S4.

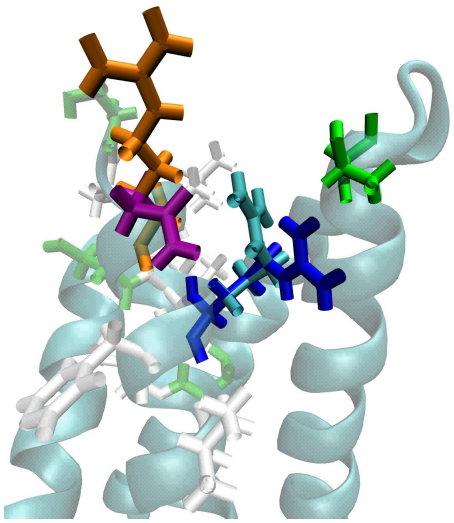

Figure 15: Visualization of mutation R198Q with the original amino acid, R, in orange and the mutant, Q, in purple. The rest of the surrounding residues are color coded by charge. The opaque blue residue is the arginine 201, the cyan colored residue is the same arginine but in the mutant configuration. Opaque green is threonine 114.

Secondly, R198Q is located in the S3 segment and it is a mutation from an arginine to a glutamine, thus from a positively charged residue to a nonpolar one. The program considers that the mutation is not destabilizing since the total score is -0.375, but it is not as low as -0.5 to be considered stabilizing. The most relevant scores in this case are `fa_atr`, `fa_sol` and `ref`.

Starting from the Van der Waals terms, it is clear that it is favorable to glutamine, since the repulsive term is almost zero (`fa_rep`) while the attractive term is negative. This might be because glutamine's lack of charge lets the arginine in position 201 to get

[5]Native protein.

closer and more packed and it can have attractive forces between both glutamine's and threonine's (position 114) dipole. The solvation term (`fa_sol`), on the other hand, is more favorable to the original amino acid. This might be due to the large shape of arginine in comparison with glutamine, as well as the closeness of the previously mentioned arginine. Since this term favours the closeness of atoms with the same nature, the presence of another arginine makes the solvation term go lower.

In the same transmembrane segment, S4, mutation R213W is located. In this case the replaced residue (arginine) is a p-charged residue and it is mutated to tryptophan, an aromatic residue. While arginine is positively charged, tryptophan is in principle non-polar, but due to an electron deficit in the ring hydrogen atoms it has significant potential for electrostatic interactions an electron transfer. It is mild hydrophobic because its aromacity is juxtaposed with polar properties.

Mutation	Total score	fa_rep
S122L	279.810	286.657
E130K	5.434	1.365
A178V	17.220	20.893
R198Q	-0.375	0.072
R213W	1.883	3.340
C242F	94.971	101.930
F261Y	35.669	33.559
T263I	16.762	17.115
L263P	303.197	294.329
W270K	-43.341	-55.019
Y284D	-1.043	-4.317
F305L	-106.455	-102.088

Table 2: Scores for each mutation given by the unmodified MPddG package (inputting a single monomer). Yellow-colored rows show mutations without VdW overestimation.

As it can be seen in Fig. 13 the most relevant scores are `fa_atr`, `fa_elec`, `fa_rep` and `ref`. For the Van der Waals interaction terms it is clear that the attractive term, `fa_atr`, has a bigger absolute value than the repulsive term, `fa_rep`, thus implying that the Van der Waals interaction favours the mutation. This might be due to the next arginine on the 214 position being able to get closer to the tryptophan because it does not have the positive charge that the arginine has. This way, that arginine lies closer to a glutamic acid, which has a negative charge and stabilizes the attractive term. Coulomb electrostatic potential interaction, however, is less stable upon mutation. Arginine might be favored because, being itself a basic amino acid with positive charge is closely surrounded with two acidic amino acids, aspartic acid and glutamic acid. This can potentially form salt bridges which help stabilizing proteins.

The membrane energy term favours the mutation, being the only one among the four here explained that is more negative upon mutation. This result is probably because in the empirically determined hydrophobicity scale tryptophan is more stabilizing than arginine. Overall, the program applied with only repacking close to the mutation considers that R213W is a destabilizing mutation.

Lastly, in the S5 segment we find mutation Y242D, which is a change in amino acid from tyrosine (aromatic) to aspartic acid (negatively charged). While tyrosine is mildy hydrophobic because the aromacity is juxtaposed with the hydroxyl group's negative charge, aspartic acid is hydrophilic because of its acidic nature. The only difference between the two is the aromatic ring between the CH_2 group and the hydroxyl group in

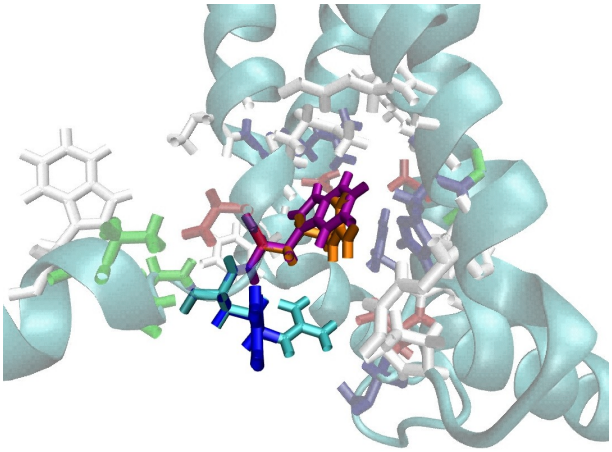

Figure 16: Visualization of mutation R213W with the original amino acid, R, in orange and the mutant, W, in purple. The rest of the surrounding residues are color coded by charge. The opaque blue residue is the arginine 214, the opaque cyan residue is the same arginine but in the mutant conformation.

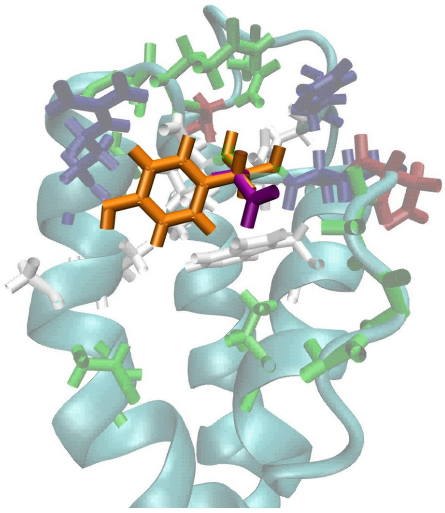

Figure 17: Visualization of mutation Y242D with the original amino acid, Y, in orange and the mutant, D, in purple. The rest of the surrounding residues are color coded by charge.

tyrosine, this makes it larger and more hydrophobic. Rosetta considers that this mutation is stabilizing and the most relevant scores are `ref`, `rama_prepro`, `fa_sol` and `p_aa_pp`. Even if the Van der Waals atractive and repulsive score terms are bigger in absolute value than any other, the whole VdW interaction term is less relevant than the previously mentioned ones.

The Ramachandran map term (`rama_prepro`) is lower upon mutation. This happens because the dihedral angles for both of them are $\phi = -125.203°$ and $\psi = 142.114°$ and while it is a very populated region of the Ramachandran map for most of the residues, aspartic acid has a smaller probability of having those angles than tyrosine because its Ramachandran map is more scattered than tyrosine's. Also related to the dihedral angles, `p_aa_pp` accounts for the superior probability of tyrosine to have said angles.

The solvation term is also positive, meaning that it favours Y over D. This might be due to the fact that position 242 is surrounded by polar uncharged and nonpolar amino acids, and tyrosine, being itself a uncharged amino acid with potential for polarity, has a higher capability of solvation compared to a charged polar amino acid as aspartic acid. Tyrosine its also considerably bigger making it more favorable to solvation.

5.1.2 Full protein

With the goal of having more accurate results, we modified the program so that it could account for the whole protein. For this purpose we mutated each of the monomers and run a repacking in the surroundings of each mutation as in the previous part. Results for the mutations with overestimation of Van der Waals interaction where still overestimated except for W270K, which yielded a logical result this time.

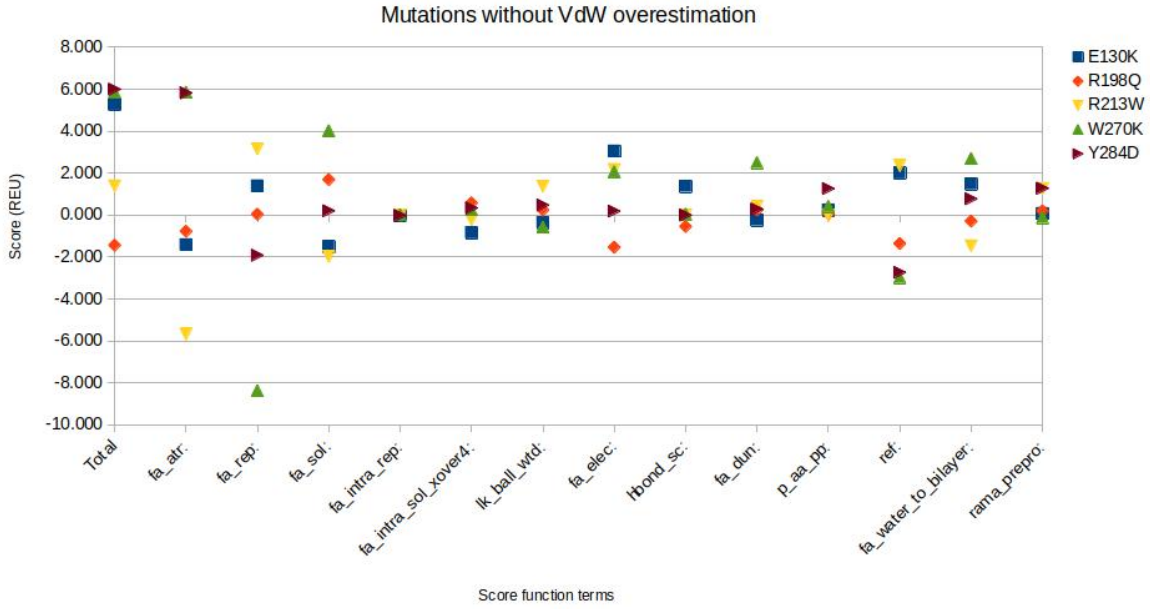

Figure 18: Non-zero scores given by the origial MPddG program (modified to consider the four monomers) of the protein for the mutations without Van der Waals energy term overestimation.

This mutation is in the pore region of the protein and it is a change between an aromatic amino acid (tryptophan) and a positively charged one (lysine). As explained in the previous section, tryptophan is mildly hydrophobic because of the combination of the indole-ring nitrogen polarity and the aromacity. On the other hand, lysine is one of the longest amino acids and it is charged. Rosetta considers this change as destabilizing and the most relevant scores are `fa_sol`, `fa_water_to_bilayer`, `fa_dun`, `ref` and the Van der Waals energy terms.

In the solvation term tryptophan is favored, this is because it fills the space in between residues better and it is surrounded with a lot of non-polar residues so tryptophan's mild hydrophobicity is helpful in the stabilization of the area.

The rotamer energy term (`fa_dun`) is also less stable upon mutation. This term accounts for the probability of the the side-chain to have the specific rotamer that the repacking has chosen. Due to the largeness of lysine and the packed conformation of tryptophan, there are more possible rotamers for lysine than tryptophan, making the probability of the chosen rotamer for the mutation smaller than the original and thus being less stable.

For the Van der Waals energy terms, it is interesting to see that the overestimation did not come in the first place by the repulsive forces of the mutant, but by the repulsive forces of the wildtype. In the case of the full protein the difference is more balanced, but still tryptophan is considered less stable due to the repulsive VdW forces. This might be due to the negative polarity that the indole-ring nitrogen brings that is repelled by the rest of negative charges around, like glutamic acid (254).

Between the previously analyzed mutations only Y284D changes the qualitative result

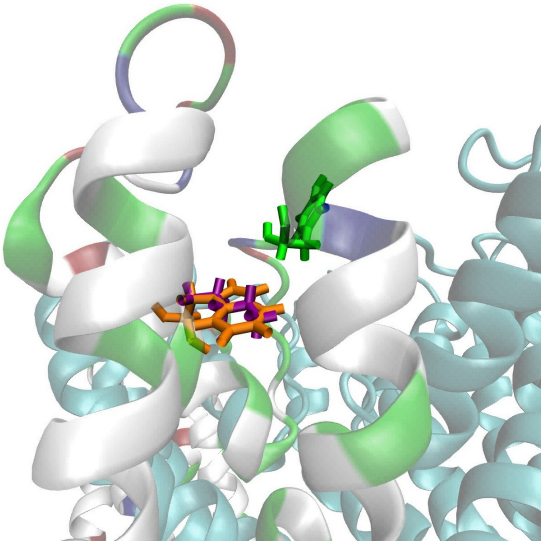

Figure 19: Visualization of mutation W270K with the original amino acid, W, in orange and the mutant, K, in purple. The secondary structure of the monomer is color-coded showing the type of residues, it can be seen that it mostly white, that is non-polar. Other monomers are shown in cyan.

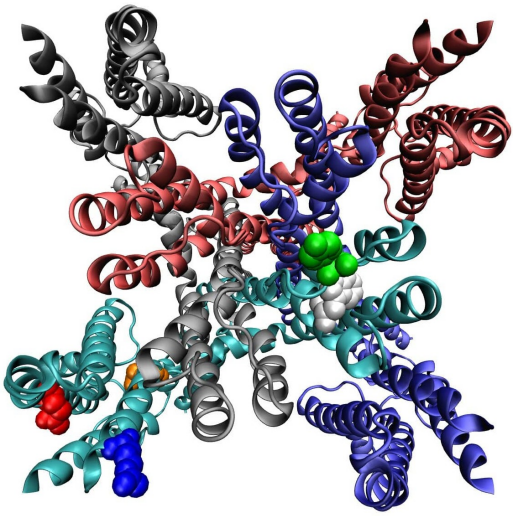

Figure 20: Visualisation to show where are the non-overestimated mutations distributed. Each monomer has a different colour and the mutation sites are red: E130, blue: R198, orange: R213, white: W270 and green Y284.

after the whole protein being considered, when only one monomer was inputted Rosetta considered the mutation stabilizing, but as a full protein it is fairly destabilizing. The change resides essentially in the Van der Waals interaction terms, while most energy terms stay unchanged or even closer to 0, both `fa_atr` and the repulsive term are almost two units larger than in the monomer version. This happens because when considering the full protein the S6 segment of the monomer in front approaches the position enough for the Van der Waals interactions to apply. The residues completely surround the mutation position so that the forces cancel each other out better. Furthermore, the close presence of a lysine, an arginine and a histidine create a more atractive force to the tyrosine than to the aspartic acid.

Since E130K is not near the pore region it is reasonable that the result did not change for the full protein case. However, for R198Q and R213W it did change slightly because some of the proteins in S5 and S6 segments of the previous chain lay within the influence of the mutation position. They do not influence the result very drastically since they are only three new residues for each of the mutations and they are on the limit of the effect of the energy terms.

5.2 Repacking and minimizing within 8Å from mutation

When running our improved program for a single monomer and the whole protein the minimization was apparently successful, only L268P showed overestimation in the Van der Waals repulsive energy term, due to proline's special nature. For consistency purposes

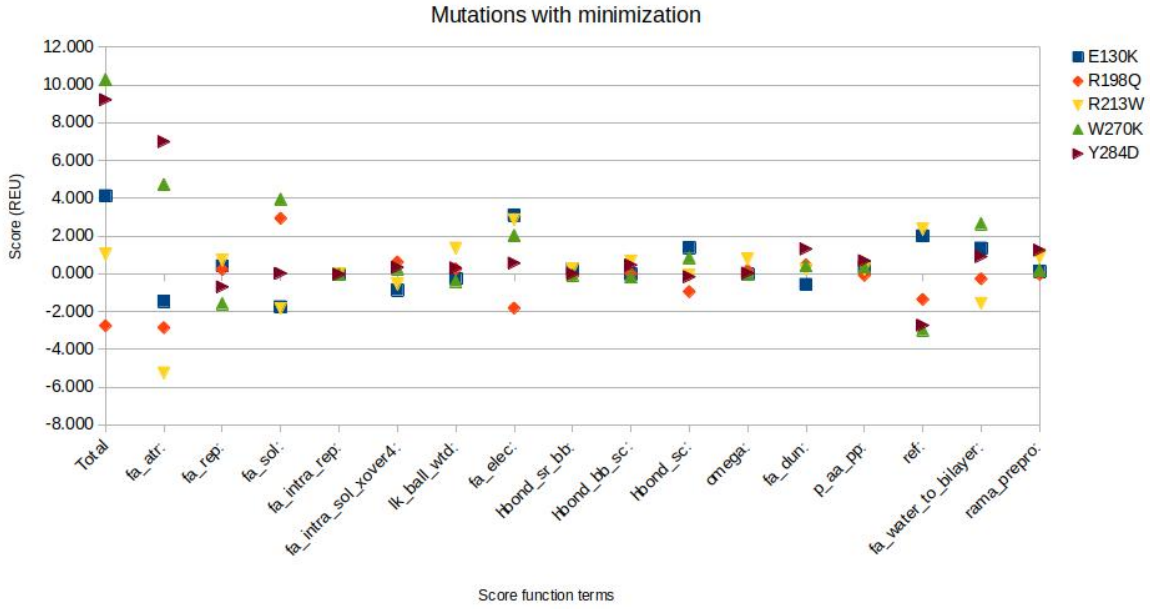

Figure 21: Non-zero scores given by the improved program for the mutations that did not show energy overestimation in the original program. Program run with the whole protein.

we will first analyze the change in the results of the mutations that gave a readable solution in the only-repacking program. Both in the monomer and full protein versions for E130K and R213W the results are mostly similar to the repack version, the tendency in the score terms is the same and they only change slightly in value. This lack of noticeable difference makes sense since E130K and R213W are close to the endings of segments S2 and S4, respectively, and facing the interior of the barrel that segments S1-4 form, leaving not much space to change (see Fig. 20).

For R198Q in the monomer version the total score goes high enough to make the qualitative result uncertain. This is probably because it is at the top and facing outwards of the S4 segment, so the surrounding residues and itself have more freedom of movement. In the full protein version, however, the difference between repack and minimizer is not so big and it is considered a stabilizing mutation with even a lower score. This is probably because the residues in the S6 segment of the previous chain do not let the rest of the residues move that much.

In the monomer version of Y284D the result changes quite drastically from stabilizing to destabilizing, which approaches more the result with the whole protein (both with minimizer and without it). This mutation is in the pore region, which is more flexible than the transmembrane helices which explains why the minimizer is able to have a bigger effect. In the whole protein versions the change is mostly visible in the electrostatic terms, particularly in the VdW terms, while in the monomer versions a noticeable difference is present in the solvation term as well. Overall, the minimizer is able to minimize the wildype conformation more than the mutated one, so the total score is always higher after minimizing.

Finally, for W270K the Van der Waals repulsive term was overestimated for the

Mutation	MPddG monomer	MPddG full	New monomer	New full
S122L	279.810	279.736	-4.876	-3.056
E130K	5.434	5.279	4.324	4.134
A178V	17.220	17.299	-0.74	-0.754
R198Q	-0.375	-1.439	-1.659	-2.752
R213W	1.883	1.391	1.946	1.033
C242F	94.971	95.052	1.304	1.149
F261Y	35.669	36.399	-0.252	3.751
T263I	16.762	36.024	2.195	2.560
L268P	303.197	305.129	16.757	19.449
W270K	-43.341	5.863	8.757	10.286
Y284D	-1.043	5.996	4.691	9.224
F305L	-106.455	-110.359	-0.223	2.401

Table 3: Results for all the mutations for the original program (MPddG) and for the improved program (New) for a single monomer and the full protein. Values for whole protein are divided by four to consider the effect of one mutation.

wildtype so the total score was overly negative. This was fixed when running the program with the full protein and also the minimizer fixed that problem. As the previous mutation, this one is also in the pore region, which is more flexible for the minimization.

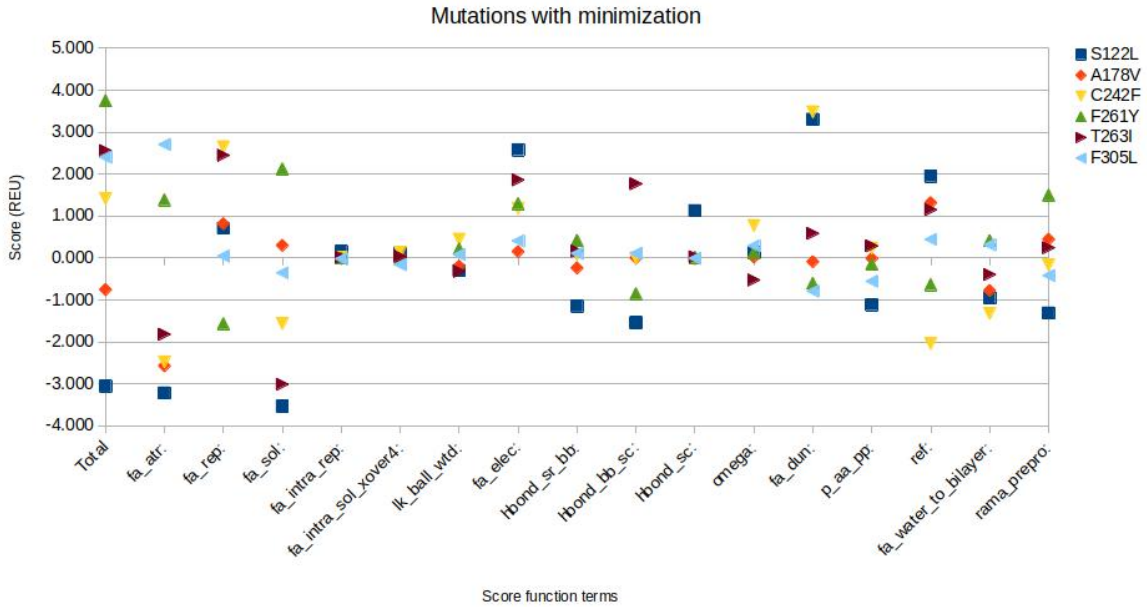

Figure 22: Non-zero scores given by the improved program for the mutations that showed energy overestimation in the original program. Program run with the whole protein. Results for L268P are excluded.

The mutants for which we could not get any acceptable result in the repack program gave good results with the minimzer. Results were quite similar when running a single

monomer and the whole protein, except for F261Y and F305L. We will focus on the full protein results since a more accurate conformation is considered.

Starting with the mutation in the first segment, S122L, it is a mutation between two non-polar amino acids, serine to leucine. While serine has an hydroxyl group, leucine has two CH_3 groups. Thus, both are aliphatic but serine is able to form hydrogen bonds. Rosetta considers that this mutation is stabilizing with $\Delta\Delta G = -3.056$ and the most relevant scores are `fa_atr`, `fa_sol`, `fa_elec` and `fa_dun`. The attractive Van der Waals term is more favorable to leucine because it is bigger and it can get closer to the surrounding non-polar amino acids. As for the Coulomb term, a big part of it might come due to the tyrosine in position 118 being able to get close to the tryptophan in S6 in the native conformation and thus interacting favorably. The bigger size is also responsible for the negative solvation term, since it gets closer to the surrounding residues interacting with them instead of water.

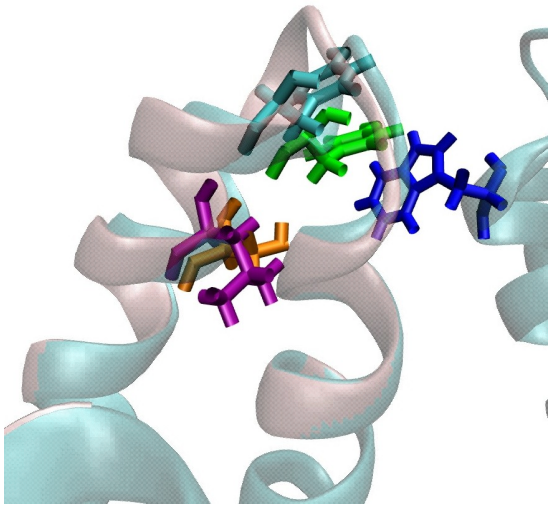

Figure 23: Visualization of mutation S122L with the original amino acid, S, in orange and the mutant, L, in purple. The green residue is tyrosine 118 and the cyan residue the same amino acid but in the mutant conformation. The blue residue is the tryptophan in S6 and the rose color shows how the secondary structure is move after the mutation.

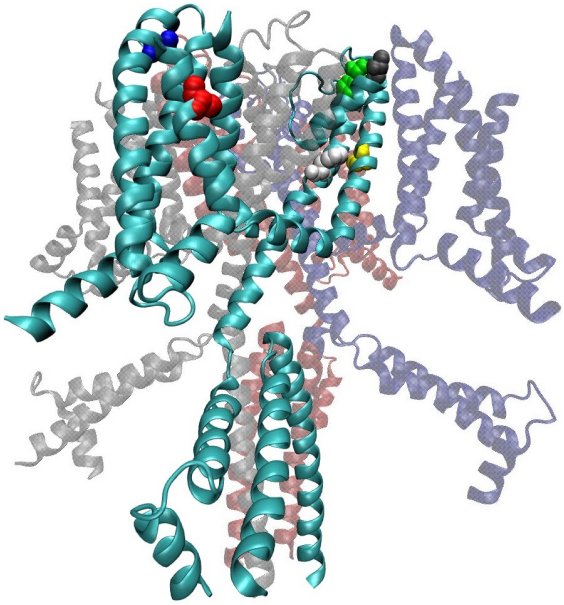

Figure 24: Visualization of the mutations that showed overestimation. Blue: S122, red: A178, yellow: C242F, green: F261Y, gray: T263 and white: F305L. L268 is excluded.

Another non-polar to non-polar mutation is placed in the S3 segment, A178V, which is a change from alanine to valine. Both are aliphatic and valine is like an alanine but with an extension of two CH_3 branches. It is not surprising then that Rosetta considers this mutation as stabilizing with $\Delta\Delta G = -0.754$. The most relevant scores are `fa_atr`, `fa_rep` and `fa_water_to_bilayer`, but overall scores are mostly close to zero. As in S122L, valine is longer and gets closer to the non-polar residues in the surrounding, provoking the Van der Waals interaction to be negative.

Mutation C242F is located in the S5 segment where a sulfur containing amino acid (cysteine) is replaced by an aromatic amino acid (phenylalanine). While cysteine is hy-

drophobic due to the sulfur but is very reactive and can form disulfide bonds with other cysteine's, phenylalanine is mildly hydrophobic with significant potential for electrostatic interactions. The most relevant terms in this case are `fa_sol`, `fa_elec`, `fa_dun` and `fa_water_to_bilayer` resulting in a positive total score ($\Delta\Delta G = 1.419$), implying that the mutation is destabilizing. Since phenylalanine is bigger, it is closer to the nonpolar residues in S6 and can help solvate the protein better. However, cysteine is not very flexible while F can take other conformations, which results in a less probable conformation, worsening the `fa_dun` term.

T263I is a polar to non-polar mutation situated over the membrane bilayer in the pore region. Threonine has an hydroxyl group and is capable of forming hydrogen bonds while I is also aliphatic with only CH_2 groups. The most relevant scores for a final positive score ($\Delta\Delta G = 2.560$) are `fa_sol`, `fa_elec` and `hbond_bb_sc`. Due to threonine having potential for polarity it might interact with water which worsens the polarity in comparison to isoleucine. This same reason is why the Coulombian interaction is worse after mutation, since the surroundings are mainly polar and charged. The hydrogen bonding term is favorable to threonine because it stabilizes the conformation by creating H-bonds with the aspartic acid in position 266.

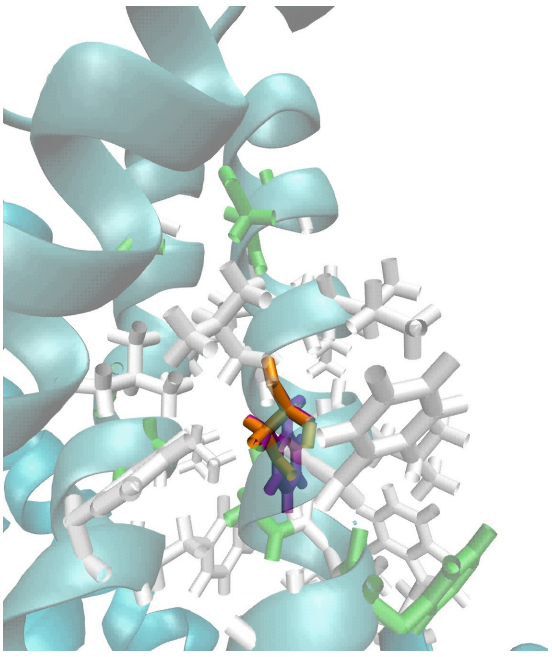

Figure 25: Visualization of mutation C242F with the original amino acid, F, in orange and the mutant, Y, in purple. The residues color-coded showing the type of amino acids, showing in the vicinity they are mainly non-polar.

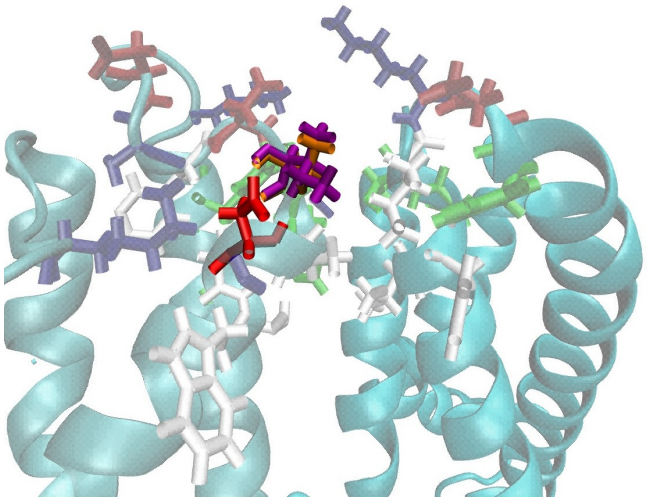

Figure 26: Visualization of mutation T263I with the original amino acid, F, in orange and the mutant, L, in purple. The residues color-coded showing the type of amino acids. The bright red residue is aspartic acid 266. Different shades of cyan indicate different monomers.

Finally, considering the mutations that showed the biggest difference in score between monomer and full protein, the difference lied mainly in the Van der Waals terms and both are destabilizing when the whole protein is considered. This is reasonable since F305L is in the S6 segment closely surrounded by the residues in the S6 segment of the next chain.

Similarly, F261Y is located in the pore and outside of the membrane region.

The latter mutation is a change from phenylalanine to tyrosine, thus aromatic to aromatic and the only difference between them is the hydroxyl group that tyrosine has in the aromatic group, which makes it mildly hydrophobic instead of highly hydrophobic as phenylalanine (it is sometimes considered uncharged polar). `Fa_sol`, `rama_prepro` and `fa_elec` are the score terms with the biggest effect in the total score. The solvation term is higher for Y because F is more hydrophobic and does not want to interact with water, interacting with the rest protein instead. The hydroxyl group is also responsible of having less favorable electrical interactions with the surrounding non-polar residues and adopting a worse conformation so the `rama_prepro` is higher after the mutation.

F305L is also a mutation from phenylalanine but in this case to leucine, which is non-polar and aliphatic. The final positive score ($\Delta\Delta G = 2.401$) is dominated almost entirely by the Van der Waals atractive term. Phenynalalinen is bigger so it can have more favorable interactions. In addition, also due to its bulkiness and because it is in a very populated region it does not have space to have the most probable conformation and thus `fa_dun` and `pa_aa_pp` are more favorable for leucine.

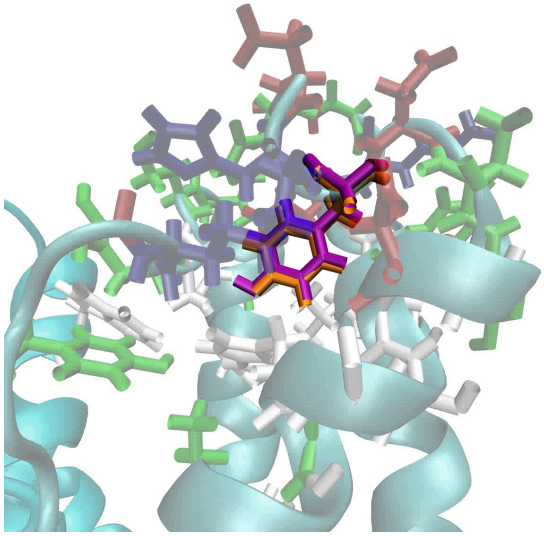

Figure 27: Visualization of mutation F261Y with the original amino acid, F, in orange and the mutant, Y, in purple. The residues color-coded showing the type of amino acids, showing in the vicinity they are mainly polar. Different shades of cyan indicate different monomers.

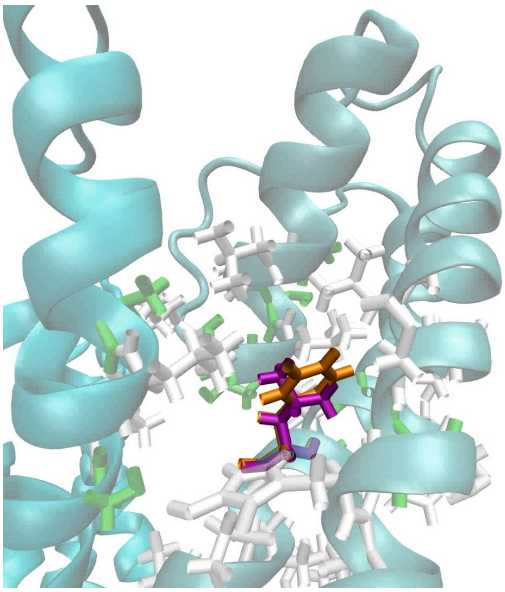

Figure 28: Visualization of mutation F305L with the original amino acid, F, in orange and the mutant, L, in purple. The residues color-coded showing the type of amino acids, showing in the vicinity they are mainly non-polar. Different shades of cyan indicate different monomers.

5.3 Membrane's contribution

In order to understand the relevance that the membrane has in the results, we run the previously mentioned two versions (for a single monomer and the whole protein), but

ignoring the membrane and as a consequence using only the *ref2015* score function without the `fa_water_to_bilayer` term. Since the repacking is not a very sensitive method for energy minimization, in most cases the conformation taken in the membrane version and the non-membrane version were the same.
As a consequence all the score terms had the same value and the only difference in the overall score was given by the addition of the membrane term. The only exceptions to this appeared for S122L and C242F in the monomer version and S122L and W270K in the full protein version. However, only with the repacking the membrane score term is not capable of changing the result qualitatively, the mutations that were destabilizing with *ref2015* are still destabilizing with franklin2019.

Mutation	With	Without	Membrane term
S122L	-3.056	-1.155	-0.952
E130K	4.134	2.710	1.361
A178V	-0.754	0.040	-0.774
R198Q	-2.752	-2.614	-0.266
R213W	1.033	2.565	-1.567
C242F	1.419	3.032	-1.320
F261Y	3.751	3.300	0.419
T263I	2.560	3.101	-0.392
L263P	19.449	19.531	-0.364
W270K	10.086	7.540	2.656
Y284D	9.224	8.362	0.915
F305L	2.401	2.074	0.314

Table 4: Scores for each mutation by the improved program with and without membrane (with the whole protein). Fourth column is the value of the membrane term in the program with it.

On the other hand, when running the improved program without membrane, the minimization showed significant differences in final score; nonetheless, the qualitative result (stabilizing or destabilizing) did not change in most of the mutations. Only A178V (from stabilizing to uncertain when not considering the membrane for both the monomer and the full protein versions), and F305L (from uncertain to stabilizing in the monomer version) showed a qualitative shift in the results. As expected, the improved program showed a more noticeable effect in the minimization caused by the membrane term, resulting in a bigger gap than the membrane energy term solely.

5.4 Flex_ddG

Flex_ddG is a protocol that focuses in the changes in binding free energy after mutation, $\Delta\Delta G_{bind}$, defined as the change in Gibbs free energy of the complex compared to the partners of the complex separated.

$$\Delta G_{bind} = \Delta G_{complex} - \Delta G_{partnerA} - \Delta G_{partnerB} \tag{10}$$

To evaluate the effects of the mutations, the protocol calculates the Gibbs free energy for both the wildtype and the mutant and compares them. Following Barlow *et al.* [5] work, the criteria we will consider is the following: a mutation will be stabilizing when $\Delta\Delta G < 1.0$ destabilizing when $\Delta\Delta G > 1.0$ and neutral otherwise.

$$\Delta\Delta G_{bind} = \Delta G_{bind}^{MUT} - \Delta G_{bind}^{NAT} \tag{11}$$

The key for the success for this protocol is the implementation of several rounds of energy minimization. Initially, it performs a L-BFGS minimization with Armijo inexact line search conditions in the input structure with all the binding partners. After that a backrub protocol is performed, this is started by choosing random protein segments consisting of three to twelve neighbouring residues in the neighbourhood of the mutated position, which is defined by all the residues with the C_β within 8Åof the mutation position. Once the segment is chosen, up to 50,000 Monte Carlo steps are run with a temperature of 1.2kT by rotating each segment locally around the vector between endpoint C_α atoms. 50 output models are generated and from this point on two sets (of 50 models each) are generated: one with the wildtype configuration, and the other one with the mutant amino acid. Both sets go through the following protocol independently.

For each model the packer previously explained is applied and again the minimizer is run as in the first step. Next up, the complex as a whole and the partners individually are scored to get ΔG. For the latter, the scores are computed by separating the partners from each other and calculating the score then. Finally, the $\Delta\Delta G$ is calculated by subtracting the result for the native models to the mutated ones. This is done by averaging over all models. The score function used is *ref2015* by default and it is yet no prepared to consider the membrane. We applied this program considering each of the chains as partners, thus, calculating the binding affinity of binding the four monomers together:

$$\Delta G_{bind} = \Delta G_{complex} - \sum_i \Delta G_i \tag{12}$$

where i is the chain ID.

5.4.1 Results

Flex_ddG yielded the results shown in table 5. To approach the results for the protein in the membrane, we chose for every mutation reference wildtype and mutated structures to calculate the energy of the membrane term averaging over them.

As it can be seen in the summation, four mutations yield similar result between Flex_ddG and the improved program: S122L, A178V, W270K and Y284D; however, for A178V the result comes purely from the membrane term which implies that it would probably have a more noticeable difference in the Flex_ddG result. Similarly, E130K, R213W and C242F are uncertain if we only consider the result from the flex program, and the membrane term is what yields a qualitative result. On the other hand, T263I and F305L are stabilizing and destabilizing, respectively, and the membrane term enhances it, while R198Q and F261Y are uncertain any way. Finally, it is important to realize that using Flex_ddG is the only way we could find a logical result for L268P. It is very common to have problems computationally with proline, due to its cyclic nature, because it can not get a proper conformation and it clashes with the surrounding residues. The algorithm for Flex_ddG is able to minimize the clashes to get proper results and Rosetta considers that L268P is destabilizing.

Ultimately, Flex_ddG could only give a qualitative certain result for five mutations S122L, T263I (stabilizing), L268P, W270K and Y284D (destabilizing). These results are

Mutation	Flex_ddG	Membrane term	Sum	New full
S122L	-3.32	-1.678	-4.999	-4.876
E130K	-0.003	1.589	1.586	4.324
A178V	0.118	-0.690	-0.571	-0.740
R198Q	0.172	0.054	0.226	-1.659
R213W	0.066	-1.190	-1.125	1.946
C242F	-0.447	-1.239	-1.685	1.304
F261Y	-0.008	0.106	0.097	-0.252
T263I	-1.395	-0.337	-1.732	2.195
L268P	4.556	-0.077	4.479	16.757
W270K	2.716	1.778	4.494	8.757
Y284D	6.607	0.255	6.323	4.691
F305L	0.885	0.315	1.199	-0.223

Table 5: Results for all the mutations got by Flex_ddG. The third column shows the score of the membrane term from *franklin2019* for output structures of the program and the fourth column is the sum between the second and third columns. Fifth column are the results of our program for reference.

in agreement with the ones we got using the improved program for the whole protein, except for T263I, that was considered destabilizing in our program.

5.5 Self-consistency of the results

Assessing the accuracy of the tools for the calculation of $\Delta\Delta G$ upon mutation is difficult given the limitations and inconsistencies of experimental data. However, Thiltgen *et al.* [36] evaluated different programs based on their ability to generate consistent results for forward and backward mutations. They based their analysis on the fact that a mutation in a given location of X to Y should have an opposite effect to the reverse mutation from Y to X: $\Delta\Delta G_{YX} = -\Delta\Delta G_{XY}$ Unknown changes in stability can cause fluctuations in the results, so the computationally predicted values are $\Delta\Delta G^P_{XY}$ and their errors $\delta_{XY} = \Delta\Delta G^P_{XY} - \Delta\Delta G_{XY}$.

Since the real values are unknown we consider the value $\Delta\Delta G^*_{XY}$ that would minimize the error:

$$\Delta\Delta G^*_{XY} = \frac{\Delta\Delta G^P_{XY} - \Delta\Delta G^P_{YX}}{2} \tag{13}$$

With the following error:

$$\delta^* = \Delta\Delta G^P_{XY} - \Delta\Delta G^*_{XY} = \frac{\Delta\Delta G^P_{XY} + \Delta\Delta G^P_{YX}}{2} \tag{14}$$

Using the same self-consistency method we tried to assess the quality of our results. We performed the following protocol for all the programs exposed previously: firstly, we run the program with the input PDB being 7CR3 (wildtype) and performing the X to Y mutation; secondly, we used the outputted mutated PDBs of this program to use as input

structures in an independent run where we produce the mutation Y to X. As expected, the programs with only repacking shows little to no error when comparing the normal and the reverse results, thus our main interest lied in the minimizer version. The consistency results we got are shown in Table 6.

Mutation	Normal score	Reverse score	$\Delta\Delta G^*$	δ^*
S122L	-3.056	4.635	-3.845	0.790
E130K	4.134	-3.940	4.037	0.097
A178V	-0.754	0.710	-0.732	0.022
R198Q	-2.752	0.426	-1.589	1.163
R213W	1.033	0.225	0.404	0.629
C242F	1.149	-1.001	1.210	0.209
F261Y	3.751	-2.106	2.929	0.822
T263I	2.560	-2.339	2.450	0.110
L268P	19.449	-12.568	16.008	3.441
W270K	10.286	5.421	2.433	7.853
Y284D	9.224	-7.858	8.541	0.683
F305L	2.401	-2.622	5.11	0.111

Table 6: Self-consistency results using equations [16] and [17].

As we can see, results are consistent for most of the mutations, except for R198Q, R213W and W270K which would suggest that those results need to be considered more carefully.

5.6 Experimental data of mutations

Seven out of the twelve mutations that we were given to analyze had already appeared in previous articles associated with epilepsy. Two of them were discovered in patients diagnosed with BFNS: S122L [15] and R213W [25]. Our program considered the first one as stabilizing with $\Delta\Delta G = -3.056$ and the second one destabilizing with a score of $\Delta\Delta G = 1.033$. Functional analysis were performed for both of the mutations and S122L showed a current reduction in the subthreshold range of an action potential of 75%. R213W also had a decrease in current density of 98,75% and it generated functional voltage-dependent currents with maximal densities identical to those of WT but required more depolarized potentials to become activated. This mutation was also found in a patient with EOEE [38].

The other five mutations are E130K, L268P, Y284D [13], R198Q [26] and F305L [38], and they were also found in patients diagnosed with EOEE. Our program identified E130K, Y284D and F305L as destabilizing with scores of $\Delta\Delta G = 4.134$ and $\Delta\Delta G = 9.224$ and $\Delta\Delta G = 2.401$, respectively. On the other hand, R198Q was considered stabilizing with $\Delta\Delta G = -2.752$. Functional analysis had already been performed in this mutation, the channels with the mutation were activated at less-depolarizing potentials, showing a gain-of-function (GOF) effect [26]. This mutation is located in the S4 segment that is thought to act as voltage-sensor, so further consideration is required for a proper analysis.

Lastly, we could not get results for L268P with our own program, but Flex_ddG yielded that it should also be destabilizing.

These results may suggest that Rosetta is not able to identify BFNS mutations as destabilizing, but might be able to properly identify the mutations that cause EOEE, which is a more deteriorating type of epilepsy. However, the thermodynamic stability is only one of the components of the global stability and functionality of a protein, so a broader set of sample mutations are needed to make a statement as such.

6 Conclusion

In this work we have analysed 12 mutations in the Kv7.2 potassium channel that could be linked to BFNS, a rare disease occurring in newborn children. For that purpose we have used Rosetta, a macromolecular modeling software, and particularly the packages MPddG and Flex_ddG included in it. In a first approach to the problem, using the MPddG as it was designed for (with a single chain as input), the results indicated that mutations E130K and R213W were destabilizing, while R198Q was neutral and Y284D was stabilizing. The rest of the results were not suitable for analysis for the lack of energy minimization power.

Extending the program to work for the four chains that the protein contains, results stated that mutations E130K, R213W, W270K and Y284D were destabilizing, while R198Q was stabilizing. The rest of the results also contained a large overestimation of Van der Waals interaction energies and were in consequence not suitable for analysis. Finally, in a last attempt to improve the program, we used a more sophisticated energy minimization process. This resulted in comprehensible results for all of the mutations except for L268P due to proline's ring. In this case, the mutations that were destabilizing were E130K, R213W, C242F, F261Y, T263I, W270K, Y284D and F305L. On the contrary, mutations S122L, A178V and R198Q were considered stabilizing. These were the best and more reasonable results that we could get considering the membrane.

Moreover, we also tried to understand the effect that the membrane had on the results, both in the final result and as a parameter in the minimization process. In the case of the program containing solely the repacking protocol, the membrane effect was only visible in the final summation, since the membrane term was not present in the nonmembrane program. Nonetheless, it did not suppose any difference in the repacking process for almost any mutation. On the other hand, when the minimizer was applied, the membrane term change considerably the minimized structure and thus the results were different not only because of the summed term but also because of the rest of the terms that changed. These results supposed a particular interest because we were under the knowledge that Rosetta contained a more exhaustive protocol (Flex_ddG) that could also inform us of the stability of mutations, but it could not consider the membrane. Therefore it was important to notice that even if we added the energy term for the membrane using the output structures in Flex_ddG *ad hoc*, it would not be enough to consider the whole scope of effects of the membrane since it was not considered during minimization.

Nevertheless, we believe that this results are also interesting to consider to comple-

ment our previous results. Three mutations L268P, W270K and Y284D, were considered destabilizing, and other two mutations, S122L and T263I, stabilizing. Adding the membrane term after the computation, E130K and F305L were considered destabilizing and R213W and C242F were considered stabilizing. This analysis shows the importance of the membrane term for $\Delta\Delta G$ calculations in membrane protein, since it can change considerably the results. Creating a similar protocol to Flex_ddG that could account for the membrane would be interesting in the future of Rosetta.

We also used this work to evaluate Rosetta. Firstly, it is important to note that the packages that we used perform only thermodynamic calculations, thus they do not consider the functionality of the protein, nor know how to differentiate essential domains from less important or variable ones. This is the reason why results can only be properly analysed by understanding where the mutations are located and the importance of those positions; and it shows a clear limitation of this kind of calculations.

In addition, we encountered some problems when it came to the minimization process. We tried several protocols, such as FastRelax and MPRelax, with no success due to the large error that they created. As we have seen, results are at most of an order of magnitude of ten, which needs for minimization processes that are deterministic or with a very small error. We also tried a statistical approximation of the results when using the mentioned protocols, but it felt insufficient to yield proper results. It seems clear that in the specific case of $\Delta\Delta G$ calculations Rosetta users have a very limited capacity to improve the program, and protocols with statistical approaches, such as Flex_ddG, need to be built within the software to suitably select the results.

Furthermore, one of the main difficulties we faced during this process was the lack of clear documentation. Rosetta is completely open when it comes to the code, being fully accessible in the bundle, but is scripted in several languages. The core of the software is written in C++ but it is enhanced in python (PyRosetta) and some other very important protocols exist only in XML (RosettaScripts). In consequence, in some cases one needs to understand three languages to fully comprehend what the program is doing. For the most part this is an advantage since it suits a broader spectrum of users when they attempt to get the results without going deeper, but it makes the work more complicated for users who need to properly understand how the algorithm works. In spite of that, the main advantage that we found in MPddG was that it could be run in ordinary computers giving results in a few seconds, which makes it a very accessible protocol and worth using to get preliminary results. Flex_ddG, on the other hand, needed to be run for hours in a high performance computer.

Lastly, it is important to understand that Rosetta is a fairly young software (2006) and that its improvements depend on the comparisons with experimental results that researchers give; the more it is used the better it will get. The potential that this kind of softwares have is undeniable and they will surely predominate the future of theoretical research. At the moment, Rosetta needs to be considered within its limitations. When it comes to mutations in sites as sensitive as membrane channels, it seems clear that the results can only be considered superficially, as thermodynamic approximations, and not as indisputable statements.

References

[1] B. Alberts, D. Bray, K. Hopkin, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Essential Cell Biology*. Garland Science/Taylor Francis Group, 2 edition, 2004.

[2] Rebecca F. Alford, Patrick J. Fleming, Karen G. Fleming, and Jeffrey J. Gray. "Protein structure prediction and design in a biologically realistic implicit membrane". *Biophysical journal*, 118(8):2042–2055, 2020.

[3] Rebecca F. Alford, Andrew Leaver-Fay, Jeliazko R. Jeliazkov, Matthew J. O'Meara, et al. "The Rosetta all-atom energy function for macromolecular modeling and design". *Journal of chemical theory and computation*, 13(6):3031–3048, 2017.

[4] Rebecca F. Alford, Julia Koehler Leman, Brian D. Weitzner, Amanda M. Duran, Drew C. Tilley, et al. "An integrated framework advancing membrane protein modeling and design". *PLoS Computational Biology*, 11(9):e1004398, 2015.

[5] Kyle A. Barlow, Shane O Conchuir, Samuel Thompson, Pooja Suresh, James E. Lucas, Markus Heinonen, and Tanja Kortemme. "Flex ddG: Rosetta ensemble-based estimation of changes in protein–protein binding affinity upon mutation. *The Journal of Physical Chemistry B*, 122(21):5389–5399, 2018.

[6] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, et al. "The Protein Data Bank". *Nucleic Acids Research*, 28:235–242, 2000.

[7] C. Branden and J. Tooze. *Introduction to protein structure*. Garland Science, 2 edition, 1999.

[8] David A. Brown and Gayle M. Passmore. "Neural KCNQ (Kv7) channels". *British Journal of Pharmacology*, 156(8):1185–1195, 2009.

[9] Sidhartha Chaudhury, Sergey Lyskov, and Jeffrey J. Gray. "PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta". *Bioinformatics*, 26(5):689–691, 2010.

[10] Zoe Cournia, Toby W. Allen, Ioan Andricioaei, Bruno Antonny, Daniel Baum, Grace Brannigan, Nicolae-Viorel Buchete, Jason T. Deckman, Lucie Delemotte, et al. "Membrane protein structure, function, and dynamics: a perspective from experiments and theory". *The Journal of membrane biology*, 248(4):611–640, 2015.

[11] Roland L. Dunbrack Jr. and Fred E. Cohen. "Bayesian statistical analysis of protein side-chain rotamer preferences". *Protein Science*, 6(8):1661–1681, 1997.

[12] Sarel J. Fleishman, Andrew Leaver-Fay, Jacob E. Corn, Eva-Maria Strauch, Sagar D. Khare, et al. "RosettaScripts: a scripting language interface to the Rosetta macromolecular modeling suite". *PloS one*, 6(6):e20161, 2011.

[13] Montesclaros Hortigüela, Ana Fernández-Marmiesse, Verónica Cantarín, Sofía Gouveia, Juan J. García-Peñas, et al. Clinical and genetic features of 13 Spanish patients with KCNQ2 mutations". *Journal of human genetics*, 62(2):185–189, 2017.

[14] William Humphrey, Andrew Dalke, and Klaus Schulten. "VMD – Visual Molecular Dynamics". *Journal of Molecular Graphics*, 14:33–38, 1996.

[15] Jessica Hunter, Snezana Maljevic, Anupama Shankar, Anne Siegel, Barbara Weissman, et al. "Subthreshold changes of voltage-dependent activation of the Kv7.2 channel in neonatal epilepsy". *Neurobiology of disease*, 24(1):194–201, 2006.

[16] John Edward Jones. "On the determination of molecular fields. I. from the variation of the viscosity of a gas with temperature". *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 106(738):441–462, 1924.

[17] Mitsuhiro Kato, Takanori Yamagata, Masaya Kubota, Hiroshi Arai, Sumimasa Yamashita, et al. "Clinical spectrum of early onset epileptic encephalopathies caused by KCNQ2 mutation". *Epilepsia*, 54(7):1282–1287, 2013.

[18] Jainab Khatun, Sagar D. Khare, and Nikolay V. Dokholyan. "Can contact potentials reliably predict stability of proteins?". *Journal of molecular biology*, 336(5):1223–1238, 2004.

[19] Themis Lazaridis and Martin Karplus. "Effective energy function for proteins in solution". *Proteins: Structure, Function, and Bioinformatics*, 35(2):133–152, 1999.

[20] Julia Koehler Leman, Brian D. Weitzner, Steven M. Lewis, Jared Adolf-Bryfogle, Nawsad Alam, et al. "Macromolecular modeling and design in Rosetta: recent methods and frameworks". *Nature methods*, 17(7):665–680, 2020.

[21] X. Li, Zhang, Q., P. Guo, J. Fu, L. Mei, D. Lv, J. Wang, D. Lai, S. Ye, H. Yang, and J. Guo. "Molecular mechanisms and structural basis of retigabine analogues in regulating KCNQ2 channel". *Journal of membrane biology*, 253:167–181, 2020.

[22] H. Lodish, A. Berk, S.L. Zipursky, et al. *Molecular Cell Biology*. W. H. Freeman, 4th edition, 2000.

[23] Mikhail A Lomize, Irina D Pogozheva, Hyeon Joo, Henry I Mosberg, and Andrei L Lomize. "OPM database and PPM web server: resources for positioning of proteins in membranes". *Nucleic acids research*, 40(D1):D370–D376, 2012.

[24] Dagen C. Marx and Karen G. Fleming. "Influence of protein scaffold on side-chain transfer free energies". *Biophysical journal*, 113(3):597–604, 2017.

[25] Francesco Miceli, Maria Virginia Soldovieri, Paolo Ambrosino, Vincenzo Barrese, Michele Migliore, et al. "Genotype–phenotype correlations in neonatal epilepsies caused by mutations in the voltage sensor of Kv7. 2 potassium channel subunits". *Proceedings of the National Academy of Sciences*, 110(11):4386–4391, 2013.

[26] John J Millichap, Francesco Miceli, Michela De Maria, Cynthia Keator, Nishtha Joshi, Baouyen Tran, et al. "Infantile spasms and encephalopathy without preceding neonatal seizures caused by KCNQ2 R198Q, a gain-of-function variant". *Epilepsia*, 58(1):e10–e15, 2017.

[27] C. Preston Moon and Karen G. Fleming. "Side-chain hydrophobicity scale derived from transmembrane protein folding into lipid bilayers". *Proceedings of the National Academy of Sciences*, 108(25):10174–10177, 2011.

[28] Robert K. Murray, Darryl K. Granner, Peter A. Mayes, and Victor W. Rodwell. *Harper's Illustrated Biochemistry*. LANGE Basic Science. McGraw-Hill Medical, 26 edition, 2003.

[29] Jon Robbins. "KCNQ potassium channels: physiology, pathophysiology, and pharmacology". *Pharmacology Therapeutics*, 90(1):1–19, 2001.

[30] Michael A. Rogawski. "KCNQ2/KCNQ3 K+ channels and the molecular pathogenesis of epilepsy: implications for therapy". *Trends in neurosciences*, 23(9):393–398, 2000.

[31] Carol A. Rohl, Charlie E.M. Strauss, Kira M.S. Misura, and David Baker. "Protein Structure Prediction Using Rosetta". In *Numerical Computer Methods, Part D*, volume 383 of *Methods in Enzymology*, pages 66–93. Academic Press, 2004.

[32] G.M. Ronen, T.O. Rosales, M. Connolly, V.E. Anderson, and M. Leppert. "Seizure characteristics in chromosome 20 benign familial neonatal convulsions". *Neurology*, 43(7):1355–1355, 1993.

[33] Tamar Schlick. *Molecular Modeling and Simulation: An Interdisciplinary Guide: An Interdisciplinary Guide*. Interdisciplinary Applied Mathematics 21. Springer-Verlag New York, 2 edition, 2010.

[34] Maxim V. Shapovalov and Roland L. Dunbrack Jr. "A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions". *Structure*, 19(6):844–858, 2011.

[35] Maria Virginia Soldovieri, Francesco Miceli, and Maurizio Taglialatela. "Driving With No Brakes: Molecular Pathophysiology of Kv7 Potassium Channels". *Physiology*, 26(5):365–376, 2011.

[36] Grant Thiltgen and Richard A. Goldstein. "Assessing predictors of changes in protein stability upon mutation using self-consistency". *PloS one*, 7(10):e46084, 2012.

[37] Vladimir Yarov-Yarovoy, Paul G. DeCaen, Ruth E. Westenbroek, Chien-Yuan Pan, et al. "Structural basis for gating charge movement in the voltage sensor of a sodium channel". *Proceedings of the National Academy of Sciences*, 109(2):E93–E102, 2012.

[38] Q Zhang, J Li, Y Zhao, X Bao, L Wei, and J Wang. "Gene mutation analysis of 175 Chinese patients with early-onset epileptic encephalopathy". *Clinical genetics*, 91(5):717–724, 2017.